Τεχνολογίες Παγκόσμιου Ιστού και Ηλεκτρονικού Εμπορίου

Χρήστος Κ. Γεωργιάδης

Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα
www.kallipos.gr
HEALLINK
Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών
Ευρωπαϊκή Ένωση
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης
ΕΣΠΑ 2007-2013

ΧΡΗΣΤΟΣ ΓΕΩΡΓΙΑΔΗΣ
Καθηγητής Πανεπιστημίου Μακεδονίας

Τεχνολογίες Παγκόσμιου Ιστού και Ηλεκτρονικού Εμπορίου

Σύγχρονες τάσεις και προκλήσεις

Τεχνολογίες Παγκόσμιου Ιστού και Ηλεκτρονικού Εμπορίου

Συγγραφή

Χρήστος Γεωργιάδης

Κριτικός αναγνώστης

Γιώργος Γιαγλής

Συντελεστές έκδοσης

Τεχνική Επεξεργασία: Νικόλαος Πολατίδης

ISBN: 978-960-603-125-0

ΣΥΝΔΕΣΜΟΣ ΕΛΛΗΝΙΚΩΝ ΑΚΑΔΗΜΑΪΚΩΝ ΒΙΒΛΙΟΘΗΚΩΝ

Εθνικό Μετσόβιο Πολυτεχνείο

Ηρώων Πολυτεχνείου 9, 15780 Ζωγράφου

www.kallipos.gr

Στη Μαριάννα, τον Κωνσταντίνο και τη Νάσια.

Πίνακας περιεχομένων

Πίνακας συντομεύσεων-ακρωνύμια

ΗΕ	Ηλεκτρονικό Εμπόριο
ΠΙ	Παγκόσμιος Ιστός
ΥΔΚ	Υποδομή Δημόσιου Κλειδιού
ΥΙ	Υπηρεσίες Ιστού
AAV	Accountholder Authentication Value
ACL	Access Control List
AES	Advanced Encryption Standard
AHP	Analytical Hierarchy Process
API	Application Programming Interface
ASP	Active Server Pages
B2B	Business-to-Business
BI	Business Intelligence
BPEL	Business Process Execution Language
BYOD	Bring Your Own Device
C2B	Consumer-to-Business
C2C	Consumer-to-Consumer
CAVV	Cardholder Authentication Verification Value
CF	Collaborative Filtering
CSS	Cascading Style Sheets
DAC	Discretionary Access Control
DES	Data Encryption Standard
DOS	Denial of Service
DSA	Digital Signature Algorithm
ESB	Enterprise Service Bus
GPS	Global Positioning System
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTPS	HyperText Transfer Protocol Secure

IDS	Intrusion Detection Systems
IETF	Internet Engineering Task Force
MAC	Mandatory Access Control
MCDA	MultiCriteria Decision Analysis
MITM	Man-In-The-Middle
NDEF	NFC Data Exchange Format
NFC	Near Field Communication
P2P	Peer-to-Peer
PKI	Public Key Infrastructure
PoS	Point of Sale
QoS	Quality of Service
RBAC	Role-Based Access Control
REST	REpresentational State Transfer
RFID	Radio Frequency Identification
RPC	Remote Procedure Calls
RS	Recommender Systems
SaaS	Software as a Service
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SSL	Secure Sockets Layer
TLS	Transport Layer Security
UCAF	Universal Cardholder Authentication Field
UX	User eXperience
WA	Web Analytics
WoT	Web of Things
WS	Web Services
WSDL	Web Services Description Language
XML	eXtensible Markup Language
XSRF	Cross-site Request Forgery
XSS	Cross-site Scripting

Ευρετήριο Αντιστοίχισης
Ελληνόγλωσσων – Ξενόγλωσσων Όρων

Αδυναμία Απάρνησης	Non-repudiation
Αίτηση Χρήστη	User Request
Ακεραιότητα	Integrity
Ακρίβεια (μετρική)	Precision
Αλγόριθμος Ψηφιακών Υπογραφών	Digital Signature Algorithm
Ανάκληση (μετρική)	Recall
Ανάλυση Δεδομένων Ιστού	Web Analytics
Αναπαραστατική Μεταφορά Κατάστασης	REpresentational State Transfer
Ανίχνευση Ανωμαλίας	Anomaly-based Detection
Ανίχνευση Βασισμένη στη Συμπεριφορά	Behavior-based Detection
Ανίχνευση Κακής Χρήσης	Misuse-based Detection
Ανίχνευση Υπογραφής	Signature-based Detection
Ανοικτά Δεδομένα	Open Data
Αξιοπιστία	Reliability
Απάντηση Συστήματος	System Response
Απλό Πρωτόκολλο Προσπέλασης Αντικειμένων	SOAP
Άρνηση Υπηρεσίας	Denial of Service
Αρχιτεκτονική Βασισμένη-σε-Υπηρεσίες	Service Oriented Architecture
Ασύρματες Επιθέσεις	Wireless Attacks
Ασφαλές Πρωτόκολλο Μεταφοράς Υπερκειμένου	Hypertext Transfer Protocol Secure
Αυτόματη Ανακατεύθυνση	Pharming
Αφθονία	Richness
Βαθμός Εξουσιοδότησης	Clearance
Βασισμένη-σε-Όψεις Προστασία	View-based Protection
Βασισμένος-σε-Ρόλους Έλεγχος Προσπέλασης	Role-Based Access Control
Γλώσσα Σήμανσης Υπερκειμένου	HyperText Markup Language
Διαβάθμιση	Classification

Διαδίκτυο	Internet
Διαδίκτυο των Αντικειμένων	Internet of Things
Διαδοχικά Φύλλα Στυλ	Cascading Style Sheet
Διαδραστικότητα	Interactivity
Διαθεσιμότητα	Availability
Διαλειτουργικότητα	Interoperability
Διατοποθεσιακή Δημιουργία Δέσμης Ενεργειών	Cross-site Scripting
Διατοποθεσιακή Επίθεση Πλαστογραφίας Αιτήματος	Cross-site Request Forgery
Διεπαφή Προγραμματισμού Εφαρμογών	Application Programming Interface
Δούρειος Ίππος	Trojan Horse
Δραστηριότητα Επεξεργαστή	CPU Activity
Εγγενείς Κινητές Εφαρμογές	Native Mobile Applications
Εκτεταμένη Γλώσσα Σήμανσης	eXtensible Markup Language
Έλεγχος Προσπέλασης	Access Control
Εμπειρία Χρήστη	User eXperience
Εμπιστευτικότητα	Confidentiality
Εμπιστοσύνη	Trust
Εξ αποστάσεως Κινητή Πληρωμή	Remote Mobile Payment
Εξατομίκευση	Personalization
Εξόρυξη Ιστού	Web Mining
Εξουσιοδότηση	Authorization
Επιβολή Πολιτικής κατά τον Χρόνο Εκτέλεσης	Run-time Policy Enforcement
Επίγνωση Θέσης	Location Awareness
Επίγνωση Πλαισίου	Context Awareness
Επιθέσεις Διάρρηξης	Break-in Attacks
Επιθέσεις μέσω Υποδομής	Infrastructure-based Attacks
Επικοινωνία Κοντινού Πεδίου	Near Field Communication
Επιχειρηματική Ευφυΐα	Business Intelligence
Ευπάθεια	Vulnerability
Ευχρηστία	Usability

Ηλεκτρονικό Εμπόριο	Electronic Commerce
Ηλεκτρονικό Ψάρεμα	Phising
Ιδιωτικότητα	Privacy
Ιεραρχική Αναλυτική Διαδικασία	Analytical Hierarchy Process
Ιός	Virus
Ιστοφάρος	Web Beacon
Καθολικό Αναγνωριστικό Πόρου	Universal Resource Identifier
Κακόβουλο Λογισμικό	Malware
Κατά-απαίτηση Έλεγχος Προσπέλασης	Mandatory Access Control
Κατά-διάκριση Έλεγχος Προσπέλασης	Discretionary Access Control
Κινητή Διασκέδαση	Mobile Entertainment
Κινητή Πληρωμή	Mobile Payment
Κινητή Πληρωμή Εγγύτητας	Proximity Mobile Payment
Κινητή Τραπεζική	Mobile Banking
Κινητός Κώδικας	Mobile Code
Κινητικότητα Χρήστη	User Mobility
Κινητό Εμπόριο	Mobile Commerce
Κινητός Ιστός	Mobile Web
Κοινωνική Μηχανική	Social Engineering
Κοινωνικός Ιστός	Social Web
Λογισμικό ως Υπηρεσία	Software as a Service
Λογισμικό Ανοικτού Κώδικα	Open Source Software
Ομότιμη Κινητή Πληρωμή	Peer-to-peer Mobile Payment
Παγκόσμια Εμβέλεια	Global Reach
Παγκόσμια Πρότυπα	Universal Standards
Παγκόσμιος Ιστός	World Wide Web
Πανταχού Παρουσία	Ubiquity
Περιεχόμενο Παραγόμενο από τον Χρήστη	User Generated Content
Πλαίσιο (ή Περιβάλλουσα Κατάσταση)	Context
Πληρωμή Κλειστού Βρόχου	Closed Loop Payment

Ποιότητα Υπηρεσίας	Quality of Service
Πολυκριτηριακή Ανάλυση Αποφάσεων	MultiCriteria Decision Analysis
Προγράμματα Υποκλοπής	Sniffers
Προηγμένο Πρότυπο Κρυπτογράφησης	Advanced Encryption Standard
Προσποίηση	Spoofing
Πρότυπο Κρυπτογράφησης Δεδομένων	Data Encryption Standard
Πυκνότητα Πληροφοριών	Information Density
Σημασιολογικός Ιστός	Semantic Web
Σημείο Πώλησης	Point of Sale
Σκουλήκι	Worm
Συλλογική Συνείδηση	Collective Awareness
Συνδεδεμένα Ανοικτά Δεδομένα	Linked Open Data
Συνδεδεμένα Δεδομένα	Linked Data
Συνεργατικό Φιλτράρισμα	Collaborative Filtering
Σύνοψη Μηνύματος	Message Digest
Συστήματα Ανίχνευσης Εισβολών	Intrusion Detection Systems
Συστήματα Συστάσεων	Recommender Systems
Σχεδίαση Διάδρασης	Interaction Design
Σχεδιασμός Προσαρμοστικού Ιστού	Responsive Web Design
Τεχνικές Μηχανικής Μάθησης	Machine Learning Techniques
Τυχαιοποίηση	Randomization
Υβριδικές Κινητές Εφαρμογές	Hybrid Mobile Applications
Υπηρεσίες Ιστού	Web Services
Υποδομή Δημόσιου Κλειδιού	Public Key Infrastructure
Υποκλοπή	Eavesdropping
Υποκλοπή της Επικοινωνίας	Communication Interception
Υπολογιστικό Νέφος	Cloud Computing
Χρόνος Απόκρισης	Response Time
Χωρίς Επαφή Πληρωμή	Contactless Payment

Πρόλογος

Ο στόχος του παρόντος συγγράμματος είναι να παρουσιάσει με κατανοητό τρόπο σύγχρονες τάσεις και προκλήσεις στο χώρο των τεχνολογιών Παγκόσμιου Ιστού (ΠΙ) και Ηλεκτρονικού Εμπορίου (ΗΕ). Αξίζει να σημειωθεί ότι η οργάνωση του υλικού στα διάφορα κεφάλαια, εκτός ίσως του αρχικού εισαγωγικού κεφαλαίου, έγινε με τρόπο ώστε να μπορεί (ως ένα βαθμό) ο αναγνώστης να επιλέγει το κεφάλαιο που τον ενδιαφέρει, εστιάζοντας στις αντίστοιχες έννοιες, χωρίς να δεσμεύεται να ακολουθήσει κάποια συγκεκριμένη προκαθορισμένη σειρά.

Στην αρχή (Κεφάλαιο 1), θα αναφερθούμε σε βασικές έννοιες του Διαδικτύου και του ΠΙ, εστιάζοντας στον ρόλο τους ως τεχνολογική υποδομή του ΗΕ. Κατόπιν, στα επόμενα δύο κεφάλαια, θα εστιάσουμε σε ζητήματα ασφαλείας. Στο Κεφάλαιο 2, παρουσιάζονται οι βασικές έννοιες γύρω από τα ζητήματα ασφάλειας, και εξετάζονται οι βασικοί χειρισμοί, οι απαιτήσεις αλλά και σημαντικές τεχνολογίες ασφάλειας στον ΠΙ, όπως η κρυπτογράφηση, οι ψηφιακές υπογραφές, και οι υποδομές δημόσιου κλειδιού. Ακολουθεί, στο Κεφάλαιο 3, περιγραφή των απειλών που υπάρχουν σε σύγχρονα περιβάλλοντα ΗΕ, καθώς και παρουσίαση θεμελιωδών μοντέλων και πολιτικών ελέγχου προσπέλασης, με ιδιαίτερο βάρος στην απόδοση προνομίων βάσει ρόλων (RBAC), καθώς και στην αξιοποίηση του πλαισίου (context). Συζητούνται επίσης τα συστήματα ηλεκτρονικών πληρωμών με τα ειδικότερα ζητήματα ασφάλειας που αυτά θέτουν.

Το Κεφάλαιο 4 παρέχει μια επισκόπηση εννοιών σχετικά με την εξατομίκευση και τα συστήματα συστάσεων, παρουσιάζοντας τους αλγόριθμους που χρησιμοποιούν τα συστήματα αυτά, καθώς και τα ζητήματα ιδιωτικότητας που προκύπτουν. Ο χαρακτήρας του επόμενου κεφαλαίου (Κεφάλαιο 5) είναι διαφορετικός: παρουσιάζει ένα σύνολο εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών), με σκοπό την κατανόηση τεχνικών προγραμματισμού από την πλευρά του διακομιστή.

Στη συνέχεια εστιάζουμε σε ζητήματα κινητού ηλεκτρονικού εμπορίου. Το Κεφάλαιο 6 παρέχει μια εισαγωγή σε σημαντικές έννοιες (όπως κινητό περιβάλλον, κινητικότητα χρήστη, αξιοποίηση της τοποθεσίας και του περιβάλλοντος πλαισίου) και τεχνολογίες (όπως κινητός Ιστός, συστήματα πληρωμών μέσω κινητών συσκευών. Το Κεφάλαιο 7 που ακολουθεί, ασχολείται αρχικά με τις απειλές που καλείται να αντιμετωπίσει ο χρήστης μίας έξυπνης συσκευής σε περιβάλλοντα κινητού εμπορίου. Στη συνέχεια, διερευνώνται ζητήματα ιδιωτικότητας και εμπιστοσύνης σε περιβάλλοντα κινητού εμπορίου και ειδικότερα σε κινητά συστήματα συστάσεων. Η ενότητα τεχνολογιών κινητού εμπορίου κλείνει με την παρουσίαση (Κεφάλαιο 8) ενός συνόλου εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών), με σκοπό την κατανόηση τεχνικών προγραμματισμού ανάπτυξης περιεχομένου για κινητές συσκευές, κυρίως από την πλευρά του πελάτη.

Τέλος, υπάρχουν δυο κεφάλαια αφιερωμένα στην τεχνολογία των Υπηρεσιών Ιστού (ΥΙ). Στο Κεφάλαιο 9 περιγράφονται τα βασικά συστατικά και οι προδιαγραφές που υποστηρίζουν μια λειτουργικότητα βασισμένη-σε-υπηρεσίες, ικανή να επιτρέπει την απρόσκοπτη επικοινωνία και τη διαλειτουργικότητα ανάμεσα σε ετερογενή συστήματα ΗΕ. Στο Κεφάλαιο 10 στη συνέχεια, παρουσιάζεται ένα σύνολο εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών), με σκοπό την κατανόηση τεχνικών ανάπτυξης, επιλογής και σύνθεσης ΥΙ. Ενώ επιπλέον, παρουσιάζεται μια προσέγγιση επιλογής ΥΙ, η οποία αξιοποιεί μια δημοφιλή μέθοδο πολυκριτήριας ανάλυσης αποφάσεων, την AHP.

Πιστεύω ειλικρινά ότι το σύγγραμμα αυτό θα φανεί αρκετά χρήσιμο κυρίως σε φοιτητές, σπουδαστές, αλλά και σε τεχνικούς επαγγελματίες και σε τεχνικά προσανατολισμένα στελέχη επιχειρήσεων. Ελπίζω ότι θα αποτελέσει ένα αξιόπιστο σημείο αναφοράς.

Στο σημείο αυτό θα ήθελα να ευχαριστήσω όλους τους συντελεστές της έκδοσης αυτής για τη συνεισφορά τους. Ιδιαίτερες ευχαριστίες αξίζουν στον Νίκο Βεσυρόπουλο, ο οποίος εργάστηκε ακούραστα βοηθώντας να ολοκληρωθεί με τον καλύτερο δυνατό τρόπο η προσπάθεια αυτή.

Χρήστος Γεωργιάδης
Οκτώβριος 2015

Κεφάλαιο 1: Ο Παγκόσμιος Ιστός ως Τεχνολογική Υποδομή του Ηλεκτρονικού Εμπορίου

Σύνοψη
Στο κεφάλαιο αυτό θα αναφερθούμε σε βασικές έννοιες του Διαδικτύου και του Παγκόσμιου Ιστού (ΠΙ), εστιάζοντας στο ρόλο τους ως τεχνολογική υποδομή του Ηλεκτρονικού Εμπορίου. Αρχικά θα περιγράψουμε τα βασικά χαρακτηριστικά της τεχνολογίας Ηλεκτρονικού Εμπορίου, και στη συνέχεια θα εξετάσουμε την εξέλιξη του ΠΙ (τάσεις, χαρακτηριστικά, σύγχρονα θέματα) και πώς αυτή επηρεάζει τη σχεδίαση ηλεκτρονικών συναλλαγών. Τέλος, θα αναφερθούμε στις δυνατότητες που προσφέρει η Ανάλυση Δεδομένων Ιστού (Web Analytics) για καταγραφή της συμπεριφοράς των χρηστών σε περιβάλλοντα ηλεκτρονικών συναλλαγών με σκοπό τη συνεχή βελτίωση της εμπειρίας χρήστη μέσω των κατάλληλων μετρικών που προβλέπει.

Προαπαιτούμενη γνώση
Δεν προβλέπεται προαπαιτούμενη γνώση καθώς το κεφάλαιο είναι εισαγωγικό.

1. Εισαγωγή

Ο όρος Ηλεκτρονικό Εμπόριο (ΗΕ), «καλύπτει οποιαδήποτε μορφή επιχειρηματικής ή διοικητικής συναλλαγής ή ανταλλαγής πληροφοριών, η οποία εκτελείται με τη χρησιμοποίηση οποιασδήποτε τεχνολογίας πληροφορικής και τηλεπικοινωνιών». Στις μέρες μας, οι κυρίαρχες τάσεις είναι η αξιοποίηση των σύγχρονων τεχνολογιών του παγκόσμιου Ιστού και των δυνατοτήτων που προσφέρουν οι έξυπνες κινητές συσκευές.

Η επιστημονική περιοχή της "Τεχνολογίας Ηλεκτρονικού Εμπορίου" διερευνά τις δυνατότητες των προηγμένων τεχνολογιών των βασισμένων-στον-Ιστό (Web-based) πληροφοριακών συστημάτων και την εφαρμογή αυτών προκειμένου να υποστηρίξουν την υλοποίηση διαδικασιών, συναλλαγών και υπηρεσιών ΗΕ. Στο επίκεντρο βρίσκεται όλη η απαραίτητη τεχνολογία υποδομής του παγκόσμιου Ιστού και των προγραμματιστικών τεχνικών για ανάπτυξη εφαρμογών Ιστού - ιστοτόπων ΗΕ και ηλεκτρονικού επιχειρείν, ενώ σημαντικές είναι οι ιδιαιτερότητες στο χώρο του κινητού εμπορίου.

Η δημιουργία ασφαλούς περιβάλλοντος ΗΕ αποτελεί έναν κρίσιμο παράγοντα για την ευρεία διάδοση και αποδοχή των υπηρεσιών ΗΕ, για αυτό και οι τεχνολογίες ασφάλειας στον Ιστό αποτελούν μια σημαντική πλευρά της τεχνολογίας ΗΕ. Άλλες ενδιαφέρουσες τεχνολογίες που χρησιμοποιούνται κατά το σχεδιασμό και την ανάπτυξη συστημάτων ΗΕ, είναι η προσέγγιση εννοιών τεχνολογίας λογισμικού σε web/mobile περιβάλλοντα με έμφαση στη διαλειτουργικότητα μέσω της αξιοποίησης της αρχιτεκτονικής της βασισμένης-σε-υπηρεσίες Ιστού, αλλά και την ευχρηστία και αποδοτικότητα μέσω της προσαρμογής των προσφερόμενων εφαρμογών ΗΕ.

Η "Τεχνολογία Ηλεκτρονικού Εμπορίου" είναι ένα πράγματι συνθετικό αντικείμενο, με ζητούμενο την εις βάθος μελέτη των εμπλεκόμενων τεχνολογιών σε περιβάλλον παγκόσμιου Ιστού, ενσύρματου ή ασύρματου, ώστε με την ολοκλήρωση-ενοποίηση αυτών να υποστηρίζονται οι αυξημένες απαιτήσεις ασφάλειας, διαλειτουργικότητας, και ευχρηστίας των λειτουργιών των ηλεκτρονικών υπηρεσιών ενός ιστοτόπου ΗΕ.

Sound 1.1.mp3	Ηχητικό απόσπασμα (audio)
Τα 8 μοναδικά χαρακτηριστικά των τεχνολογιών ΗΕ	

Για να μπορούμε να κατανοήσουμε ευκολότερα το ΗΕ και τις τεχνολογικές δυνατότητες που προσφέρει μπορούμε να δούμε αναλυτικότερα τα 8 μοναδικά χαρακτηριστικά που χαρακτηρίζουν τις τεχνολογίες ΗΕ (Laudon & Traver, 2014):

1. **Πανταχού παρουσία** (Ubiquity): η διαθεσιμότητα του ΗΕ είναι σχεδόν παντού και μπορεί ο χρήστης οποιαδήποτε στιγμή να έχει πρόσβαση και να χρησιμοποιήσει τις υπηρεσίες του. Πιο

συγκεκριμένα, για να κατανοήσουμε τη διαφορά μεταξύ του ΗΕ και του παραδοσιακού εμπορίου ας δούμε τα ακόλουθα σημεία:

- Στην περίπτωση του παραδοσιακού εμπορίου για να πραγματοποιήσει ο καταναλωτής τις συναλλαγές που επιθυμεί πρέπει να μεταβεί σε έναν φυσικό χώρο, αυτός ο χώρος αποτελεί το σημείο αγοράς, και μέσω της φυσικής του παρουσίας μπορούν να εκτελεστούν οι συναλλαγές.
- Στην περίπτωση του ηλεκτρονικού εμπορίου το σημείο αγοράς δεν απαιτεί τόσο τον όρο της φυσικής παρουσίας του καταναλωτή, όσο και του όρου του φυσικού χώρου ως σημείου πραγματοποίησης μιας συναλλαγής. Τα χαρακτηριστικά του ΗΕ είναι ότι η διαθεσιμότητά του είναι από οποιαδήποτε σημείο και ότι πάντα μπορούν να πραγματοποιηθούν συναλλαγές. Ο καταναλωτής μπορεί να διεκπεραιώσει τις συναλλαγές του από οποιοδήποτε χώρο βρίσκεται (δουλειά, σπίτι κλπ., αλλά και αυτοκίνητο, ουρά τράπεζας κλπ., μέσω της κινητής του συσκευής και χρήσης διαδικασιών κινητού εμπορίου). Ως αποτέλεσμα της "αφαίρεσης" της φυσικής υπόστασης, ο ορισμός της αγοράς ξεπερνά την παραδοσιακή έννοια του όρου και επεκτείνεται σε πιο ευρεία έννοια όπου ο χρόνος και η γεωγραφική τοποθεσία δεν είναι παράγοντες που εμποδίζουν την πραγματοποίηση των συναλλαγών όποια ώρα επιθυμεί ο καταναλωτής.

2. **Παγκόσμια εμβέλεια** (Global reach): μέσα από τις συναλλαγές που πραγματοποιούνται στο ΗΕ, πλέον δεν υπάρχει ο γεωγραφικός περιορισμός. Η εμβέλεια του εμπορίου γίνεται παγκόσμια για τους εμπόρους χωρίς να υπάρχει ο φραγμός των συνόρων. Πιο συγκεκριμένα, το χαρακτηριστικό της εμβέλειας στο ΗΕ αναφέρεται στον συνολικό αριθμό των πελατών / χρηστών που μπορεί να αποκτήσει μια εταιρία μέσω της παρουσίας της στο ΗΕ. Για να κατανοήσουμε καλύτερα το χαρακτηριστικό της εμβέλειας αξίζει να αναφέρουμε ότι θεωρητικά στο ΗΕ το πλήθος των καταναλωτών ισούται με το μέγεθος του πληθυσμού που υπάρχει στο Διαδίκτυο (για το 2014 ανέρχεται σε περίπου 2,92 δισεκ. χρήστες, σύμφωνα με τα επίσημα στατιστικά χρήσης).
3. **Παγκόσμια πρότυπα** (Universal standards): για να μπορέσει να υπάρχει στο Διαδίκτυο μια κοινή υποδομή υπάρχουν συγκεκριμένα πρότυπα τα οποία αποτελούν ένα σύνολο τεχνολογιών. Τα πρότυπα τεχνολογίας είναι κοινά για όλες τις χώρες του κόσμου. στις συναλλαγές ΗΕ έχουμε την πλευρά του καταναλωτή και την πλευρά του εμπόρου, όπου και οι δύο πλευρές μπορούν να αποκομίσουν οφέλη από την εφαρμογή των παγκόσμιων πρότυπων. Από την πλευρά των καταναλωτών μπορούμε να δούμε ότι οι καταναλωτές εξοικονομούν σημαντικό πολύτιμο χρόνο στη διαδικασία της αναζήτησής τους για νέα προϊόντα και αντίστοιχα μειώνεται και το κόστος αναζήτησής τους. Από την πλευρά τους οι έμποροι μπορούν να έχουν ευκολότερη είσοδο στην αγορά και το κόστος τους να είναι αισθητά μειωμένη από ότι θα συνέβαινε αν δεν υπήρχε η εφαρμογή των παγκόσμιων προτύπων.
4. **Αφθονία** (Richness): Όπως στο παραδοσιακό πλέον και στο ΗΕ είναι διαθέσιμη η άμεση επικοινωνία ενός χρήστη / αγοραστή με κάποιον πωλητή μέσω ζωντανής συνομιλίας. Ο καταναλωτής μπορεί να ενημερώνεται από τον πωλητή μέσω προσωπικής επικοινωνίας σχετικά με ζητήματα ή προβλήματα που αντιμετωπίζει κατά τη διάρκεια της πώλησης του προϊόντος. Η αφθονία σχετίζεται με το περιεχόμενο και την πολυπλοκότητα της επικοινωνίας.
5. **Διαδραστικότητα** (Interactivity): Σε πιο παραδοσιακές μορφές εμπορίου η επικοινωνία που υπήρχε μεταξύ του αγοραστή / καταναλωτή του προϊόντος και του εμπόρου ήταν μονόδρομη. Ο καταναλωτής πραγματοποιούσε την αγορά του και η αλληλεπίδραση με τον έμπορο τερμάτιζε και ως αποτέλεσμα ο έμπορος δεν μπορούσε να έχει κάποια πληροφόρηση από τον πελάτη σχετικά με το πόσο ικανοποιημένος έμεινε από το προϊόν. Με την εξέλιξη των τεχνολογιών ΗΕ πλέον η επικοινωνία μεταξύ των δυο πλευρών έχει καταστεί αμφίδρομη και μέσα από τη διαδραστικότητα που υπάρχει αναπτύσσεται μια δυναμική συζήτηση. Αποτέλεσμα της ανατροφοδότησης της αμφίδρομης επικοινωνίας είναι ότι ο έμπορος μπορεί να ενημερώνεται άμεσα από τον ίδιο τον χρήστη / καταναλωτή σχετικά με τη χρηστική

εμπειρία που είχε σε όλα τη διάρκεια της συναλλαγής. Η νέα αυτή μορφή επικοινωνίας παρέχει στον έμπορο τη δυνατότητα να ανακαλύπτει τα δυνατά και αδύναμά του σημεία και να τα βελτιώνει ώστε να μπορεί να προσελκύει περισσότερους πελάτες και να διεκδικεί μεγαλύτερο μερίδιο στην αγορά.

6. **Πυκνότητα πληροφοριών** (Information density): με τη ραγδαία εξέλιξη του Διαδικτύου και των συναλλαγών που πραγματοποιούνται παρατηρούμε ότι έχει αυξηθεί εκθετικά ο συνολικός όγκος ποσότητας πληροφοριών που υπάρχει και είναι διαθέσιμος στους εμπόρους και στους καταναλωτές. Μέσα από τη χρήση της τεχνολογίας επιτυγχάνεται και η ποιότητα των πληροφοριών που προσφέρονται. Πιο συγκεκριμένα, μέσα από τη χρήση των τεχνολογιών ΗΕ έχουμε μειωμένο κόστος και αύξηση της ποιότητας των πληροφοριών.
   Οι καταναλωτές πραγματοποιούν καθημερινά έναν μεγάλο όγκο αγορών. Επιπλέον, ένας ορθολογικός καταναλωτής θέλει να πραγματοποιήσει την καλύτερη αγορά με το μικρότερο δυνατό κόστος. Μέσα από την πυκνότητα των πληροφοριών μπορεί να ενημερώνεται για τις τιμές των προϊόντων που αγοράζει και να υπάρχει διαφάνεια των τιμών με άμεση και γρήγορη πρόσβαση. Ένα ακόμα βήμα, πέρα από τη διαφάνεια των τιμών που ανακαλύπτει εύκολα ο καταναλωτής, είναι ότι μπορεί να ανακαλύψει (έως ένα βαθμό) και το πραγματικό κόστος που έχει μια επιχείρηση που εμπορεύεται τα προϊόντα που αγοράζει (επιτυγχάνεται λοιπόν και διαφάνεια κόστους).
   Η αξιοποίηση της πυκνότητας της πληροφορίας (με την εφαρμογή ειδικών τεχνικών και αλγορίθμων σχετικών με την εξόρυξη ποιοτικών πληροφοριών και σε συνδυασμό με την ομαδοποίηση των χρηστών και των προτιμήσεων τους) μπορεί να προσφέρει στους εμπόρους ανταγωνιστικό πλεονέκτημα. Οι έμποροι μπορούν να ομαδοποιήσουν τους αγοραστές τους, να αντιστοιχήσουν ποιες ομάδες αγοραστών είναι διατεθειμένες να δαπανήσουν συγκεκριμένο χρηματικό ποσό για ένα προϊόν και να τα διαθέτουν ανάλογα.
7. **Εξατομίκευση** / Προσαρμογή (Personalization / Customization): όπως έχουμε αναφέρει και σε προηγούμενο χαρακτηριστικό ο χρήστης όταν πραγματοποιούσε μια συναλλαγή στο παραδοσιακό εμπόριο η διαδικασία ήταν γενική, μπορεί να μην ήταν καθόλου κοντά στις προτιμήσεις του, και αρκετά συχνά σε καταστήματα μεγάλου πλήθους επιλογών ήταν μια απρόσωπη διαδικασία με περιορισμένη δυνατότητα βοήθειας για υποστήριξη της καλύτερης δυνατής επιλογής. Στο ΗΕ ο χρήστης / καταναλωτής πλέον έρχεται στο επίκεντρο και του παρέχονται πληροφορίες με κεντρικό άξονα τις μεμονωμένες προτιμήσεις του. Ένα απλό παράδειγμα είναι αυτό της ηλεκτρονικής εφημερίδας: πολλοί χρήστες έχουν πρόσβαση αλλά ο καθένας τους έχει να επιλέξει ανάλογα με τα ενδιαφέροντα και τις προτιμήσεις του τι θέλει να εμφανίζεται όταν μπαίνει στη σελίδα της εφημερίδας (πχ. κάποιος χρήστης προτιμά να βλέπει νέα σχετικά με τον αθλητισμό, ενώ ένας άλλος σχετικά με τις οικονομικές εξελίξεις). Αποτέλεσμα της στοχευμένης παροχής και διαμόρφωσης του περιεχομένου είναι η εξατομίκευση.
   Μέσα από την εξατομίκευση οι έμποροι μπορούν να χρησιμοποιούν εξατομικευμένα μηνύματα που αναφέρονται συγκεκριμένα σε κάθε χρήστη ξεχωριστά. Αυτά τα μηνύματα μπορεί να σχετίζονται με συστάσεις για προϊόντα που αναμένεται να ενδιαφέρουν τον χρήστη και μπορεί να σχετίζονται και με το ιστορικό παλαιότερων αγορών που είχε αυτός πραγματοποιήσει. Η έννοια της εξατομίκευσης είναι άμεσα συνδεδεμένη με τη δυνατότητα προσαρμογής προϊόντων ή υπηρεσιών. Η ανάλυση χαρακτηριστικών συμπεριφοράς των χρηστών μέσα στις εφαρμογές/τόπους του παγκόσμιου Ιστού, είναι αυτή που κυρίως τροφοδοτεί την εξατομίκευση και προσαρμογή των μηνυμάτων, προϊόντων και υπηρεσιών. Τα χαρακτηριστικά αυτά αφορούν τις προτιμήσεις κάθε αγοραστή ξεχωριστά, το ιστορικό των περιηγήσεων και συναλλαγών του και γενικά άλλα ατομικά μοναδικά χαρακτηριστικά κάθε καταναλωτή τα οποία όμως συνδυάζονται με τις προτιμήσεις αγοραστών που φαίνεται να συμπεριφέρονται παρόμοια.
8. **Κοινωνική τεχνολογία** (social technology): στους χρήστες πλέον είναι εύκολη η διαμοίραση πολυμεσικού περιεχομένου (κείμενο, ήχος, εικόνες, βίντεο κλπ.) μέσω των τεχνολογιών του Διαδικτύου. Αποτέλεσμα είναι η τεχνολογία να έρχεται κοντά στους χρήστες και τις ανάγκες τους, και να αποκτά μάλιστα και κοινωνικό χαρακτήρα με την ενδυνάμωση της κοινωνικότητας (κυρίως μέσα από την ύπαρξη των κοινωνικών δικτύων, social networks). Οι χρήστες δεν είναι απλώς παθητικοί δέκτες πληροφοριών στον Παγκόσμιο Ιστό, αλλά είναι

και δημιουργοί του περιεχομένου του. Στα πρώτα χρόνια του ΗΕ η επικοινωνία ήταν «ένας προς πολλούς» ενώ στο σύγχρονο ΗΕ είναι «πολλοί προς πολλούς». Η μονόδρομη επικοινωνία («ένας προς πολλούς») σημαίνει ότι υπάρχει μια συγκεκριμένη ομάδα που δημιουργεί, διανέμει και διαχειρίζεται το περιεχόμενο. Το επίπεδο της αλληλεπίδρασης με τους χρήστες που λαμβάνουν το περιεχόμενο είναι χαμηλό. Το μοντέλο της αμφίδρομης επικοινωνίας («πολλοί προς πολλούς») που υπάρχει στο σύγχρονο, βασισμένο στην κοινωνική τεχνολογία ΗΕ, αναφέρεται στη δημιουργία και διαμοίραση μαζικού περιεχομένου από τους ίδιους τους χρήστες. Το επίπεδο αλληλεπίδρασης που υπάρχει μεταξύ των χρηστών (είτε είναι οι δημιουργοί του περιεχομένου είτε αυτοί που το παραλαμβάνουν και το αξιοποιούν) είναι υψηλό. Οι χρήστες οργανώνονται σε δομές κοινωνικών δικτύων και μέσω αυτών μοιράζονται με το δίκτυο των «φίλων» τους το περιεχόμενο ενδιαφέροντός τους.

Στην παρακάτω εικόνα μπορούμε να δούμε σε μια σχηματική αναπαράσταση τα οκτώ χαρακτηριστικά που περιγράψαμε προηγουμένως με κεντρικό κοινό σημείο τις τεχνολογίες ΗΕ.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 1.1.gif	Κινούμενη Εικόνα (interactive)
Χαρακτηριστικά Τεχνολογίας Ηλεκτρονικού Εμπορίου	

Εικόνα 1. 1 *Χαρακτηριστικά Τεχνολογίας Ηλεκτρονικού Εμπορίου*

2. Το Διαδίκτυο και η δομή του

Sound 1.2.mp3	Ηχητικό απόσπασμα (audio)
Ορισμός για το Διαδίκτυο	

Το Διαδίκτυο (Internet) ως όρος δεν είναι πρόσφατος. Η έναρξη της λειτουργίας του Διαδικτύου έχει τοποθετηθεί σχεδόν περισσότερες από πέντε δεκαετίες πριν, όταν και είχε δημιουργηθεί η πρώτη μορφή σε επίπεδο επικοινωνίας πανεπιστημίων. Η βασική δομή του Διαδικτύου στηρίζεται στην ύπαρξη εκατομμυρίων υπολογιστών οι οποίοι ομαδοποιούνται και συνδέονται μεταξύ τους σε χιλιάδες δίκτυα. Το Διαδίκτυο αποτελεί ένα διασυνδεδεμένο δίκτυο πολλών δικτύων που ακολουθούν κοινά πρότυπα.

Στην παρακάτω εικόνα μπορούμε να δούμε συνοπτικά τα στάδια της εξέλιξης του Διαδικτύου. Η αρχική δομή του Διαδικτύου ξεκίνησε από τη σύνδεση μεγάλων υπολογιστών που υποστήριζαν την επικοινωνία μεταξύ διαφορετικών πανεπιστημιουπόλεων (πρώτη περίοδος: 1961-1974). Στην πρώτη περίοδο της ανάπτυξης του Διαδικτύου, η περίοδος αυτή ονομάζεται «Καινοτομία» γιατί μόλις πρωτοξεκίνησε το Διαδίκτυο, παγιώθηκαν τα βασικά στοιχεία (μεταγωγή πακέτων, πρωτόκολλο ελέγχου TCP/IP και μοντέλο πελάτη-διακομιστή) και αναπτύχθηκαν ψηφιακές συσκευές και συστήματα λογισμικού για να μπορεί να υποστηριχθεί η επικοινωνία των διαφορετικών μερών. Το δεύτερο στάδιο όπου καθιερώθηκε και ευρέως το Διαδίκτυο (1975-1994), υποστηρίχθηκε και χρηματοδοτήθηκε από την κυβέρνηση και το υπουργείο Άμυνας της Αμερικής. Μέσα από τη χρηματοδότηση που δόθηκε στόχος ήταν να διερευνηθούν περεταίρω τα ήδη υπάρχοντα βασικά στοιχεία του Διαδικτύου και να γίνουν επίσημα αποδεκτά. Τελικός στόχος ήταν η ανάπτυξη ενός συστήματος που θα προσέφερε πλεονέκτημα στην επικοινωνία του στρατού σε περίπτωση πυρηνικού πολέμου. Αποτέλεσμα σε αυτό το στάδιο είναι η κατασκευή του γνωστού υπερ-υπολογιστικού συστήματος ARPANET (Advanced Research Projects Agency Network). Από το 1995 έως και σήμερα το Διαδίκτυο περνάει στην επιχειρηματική του διάσταση και γίνεται διαθέσιμο σε ευρεία χρήση σε όλους τους πολίτες μέσω ιδιωτικών παρόχων που χρηματοδοτούνται, ιδιαίτερα τα πρώτα χρόνια από τις κυβερνήσεις.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 1.2.gif	Κινούμενη Εικόνα (interactive)
Τα τρία στάδια εξέλιξης του Διαδικτύου	

Εικόνα 1. 2 *Τα τρία στάδια εξέλιξης του Διαδικτύου*

Το Διαδίκτυο έχει τη δική του δομή που όσα χρόνια και αν περάσουν παραμένει πάντα ο αμετάβλητος πυρήνας του (Tanenbaum & Wetherall, 2010). Αυτή η δομή βασίζεται σε τρεις πυλώνες, τους οποίους θα περιγράψουμε εκτενέστερα στην παρούσα ενότητα, και είναι:

1. Μεταγωγή πακέτων
2. Πρωτόκολλο ελέγχου μετάδοσης/πρωτόκολλο Διαδικτύου (TCP/IP)
3. Μοντέλο πελάτη-διακομιστή

Αν θέταμε το ερώτημα «Πρέπει κάποιος να γνωρίζει τη δομή του Διαδικτύου;» η απάντηση είναι «Ναι», ιδιαίτερα αν δραστηριοποιείται στον σημερινό επιχειρηματικό κόσμο. Ανάλογα βέβαια με το πόστο στο οποίο βρίσκεται κάποιος επαγγελματίας μέσα σε έναν οργανισμό ορίζεται και η οπτική ή το ποσοστό

εμβάθυνσης που θα πρέπει να έχει στη δομή του Διαδικτύου. Ένα χαρακτηριστικό παράδειγμα είναι ότι οι γνώσεις της δομής που απαιτούνται να έχει ο υπεύθυνος μάρκετινγκ σε σύγκριση με έναν προγραμματιστή είναι πολύ διαφορετικές. Ο πρώτος αρκεί να έχει γενικές γνώσεις της δομής του Διαδικτύου, προσπαθώντας απλώς να κατανοήσει ότι υπάρχουν σελίδες που συνδέονται με κάποιο τρόπο μεταξύ τους και οι οποίες δημιουργούνται (κυρίως) σε έναν διακομιστή. Ο προγραμματιστής πρέπει να είναι άρτια καταρτισμένος και να έχει πλήρως ξεκαθαρισμένες τις έννοιες που αποτελούν τη δομή του Διαδικτύου για να μπορεί να παράγει και να συντηρεί λογισμικό για τις υπάρχουσες υποδομές.

2.1 Μεταγωγή πακέτων

Κατά τη μεταφορά της από έναν πομπό σε έναν δέκτη, η πληροφορία είναι κατακερματισμένη σε πακέτα δεδομένων. Ο υπολογιστής-πομπός στέλνει / εκπέμπει την πληροφορία, η πληροφορία κατακερματίζεται σε πακέτα για να μεταφερθεί μέσα από το δίκτυο, και στον υπολογιστή-δέκτη που παραλαμβάνει τα πακέτα της πληροφορίας γίνεται η επανασύνθεσή της. Πιο συγκεκριμένα, στη μεταγωγή πακέτων τα ψηφιακά μηνύματα "τεμαχίζονται" σε διακριτές μονάδες που ονομάζονται πακέτα. Τα πακέτα αυτά στέλνονται προς διάφορες διαθέσιμες διαδρομές επικοινωνίας και τα μηνύματα επανασυντίθενται όταν φτάσουν στον τελικό προορισμό τους. (Tanenbaum & Wetherall, 2010)

Sound 1.3.mp3	Ηχητικό απόσπασμα (audio)
Διαδικασία μεταγωγής πακέτων	

Στην διαδικασία της μεταγωγής πακέτων από το σημείο αποστολής του μηνύματος μέχρι τον προορισμό, όπου το μήνυμα ανασυγκροτείται, υπάρχουν κάποια στάδια τα οποία ακολουθούνται:

1. Αρχικά ο αποστολέας του μηνύματος (που είναι και ο δημιουργός του) συντάσσει το μήνυμα που θέλει να στείλει (με χρήση της φυσικής γλώσσας που είναι κατανοητή από τον άνθρωπο). Όταν ολοκληρώσει τη σύνθεση του μηνύματός του στην πλευρά του ο αποστολέας, μπαίνει στη διαδικασία της αποστολής του.
2. Ξεκινάει μια διαδικασία μετατροπής του μηνύματος σε μια μορφή που είναι κατανοητή από τους υπολογιστές και το δίκτυο που είναι υπεύθυνο για τη μεταφορά του μηνύματος. Το αρχικό μήνυμα ψηφιοποιείται σε bits και χωρίζεται σε πακέτα καθορισμένου μεγέθους. Αυτά τα πακέτα μεταφέρονται μέσα στο δίκτυο.
3. Σε κάθε ένα από τα πακέτα προστίθενται πληροφορίες κεφαλίδας που δείχνουν διάφορες λεπτομέρειες για την ορθή μεταγωγή του πακέτου όπως είναι η διεύθυνση προέλευσης και η διεύθυνση προορισμού του πακέτου. Άλλες πληροφορίες που υπάρχουν στις κεφαλίδες που προσαρτώνται στα πακέτα είναι πληροφορίες για έλεγχο σφαλμάτων που μπορεί να συμβούν στη διάρκεια της μεταγωγής.
4. Κατά τη διάρκεια της μεταγωγής τα πακέτα δεν ακολουθούν πάντοτε κάποια προκαθορισμένη διαδρομή μέσα στο δίκτυο, αλλά δρομολογούνται από υπολογιστή σε υπολογιστή μέχρι να φτάσουν στον τελικό τους προορισμό. Οι υπολογιστές (και οι υλικές διατάξεις) που δρομολογούν τα πακέτα ονομάζονται δρομολογητές και είναι ουσιαστικά υπολογιστές που έχουν αναλάβει την αποστολή τον πακέτων στον δέκτη μέσω των διασυνδέσεων των δικτύων υπολογιστών που αποτελούν το Διαδίκτυο. Για να υπάρχει διασφάλιση στο δίκτυο ότι η διαδρομή που ακολουθούν τα πακέτα στο δίκτυο είναι η καλύτερη δυνατή που υπάρχει (δεδομένων και των δυναμικών συνθηκών που υπάρχουν στα δίκτυα) μέχρι να φτάσουν στον τελικό προορισμό τους, χρησιμοποιούνται προγράμματα που ονομάζονται αλγόριθμοι δρομολόγησης.
5. Το τελικό στάδιο είναι όταν όλα τα πακέτα φτάσουν επιτυχώς στον τελικό προορισμό (όχι απαραιτήτως με τη σειρά με την οποία ξεκίνησαν) και ανάλογα με τις πληροφορίες κεφαλίδας που διαθέτουν ξεκινάει η διαδικασία της ανασυγκρότησης του μηνύματος (με την τοποθέτηση στη σωστή σειρά των πακέτων) για να το διαβάσει ο παραλήπτης.

2.2 Πρωτόκολλο ελέγχου μετάδοσης/πρωτόκολλο Διαδικτύου (TCP/IP)

Η μεταφορά των πακέτων που περιγράψαμε προηγουμένως αποτέλεσε για το Διαδίκτυο το πρώτο μεγάλο επίτευγμα για τη δυνατότητα της επικοινωνίας, ωστόσο δεν ήταν αρκετό. Στη μεταφορά πακέτων δεν υπήρχε μια κοινά αποδεκτή μέθοδος η οποία θα ήταν ικανή να διασπά σε πακέτα τα ψηφιακά μηνύματα, να τα στέλνει όλα στην κατάλληλη διεύθυνση και να τα ανασυγκροτεί όλα πάλι σε ένα μήνυμα. Λύση για αυτό το πρόβλημα δόθηκε με την εφαρμογή της χρήσης ενός κοινώς αποδεκτού συνόλου κριτηρίων και κανόνων που εφαρμόζονται για τη μεταφορά των δεδομένων και είναι ουσιαστικά το πρωτόκολλο.

Η εφαρμογή του πρωτοκόλλου μας βοηθάει να ανιχνεύσουμε σφάλματα στα μηνύματα και στην ταχύτητα μετάδοσης τους στο δίκτυο, και είναι υπεύθυνο για τη διάταξη, μορφοποίηση και συμπίεση των μηνυμάτων. Επίσης ένα πρωτόκολλο καθορίζει τα μέσα με τα οποία οι συσκευές που είναι υπεύθυνες για τη διακίνηση των μηνυμάτων στο δίκτυο δείχνουν ότι έχουν σταματήσει να λαμβάνουν ή να στέλνουν μηνύματα. Βασικό πρωτόκολλο επικοινωνίας στο Διαδίκτυο έχει αποτελέσει το **Πρωτόκολλο ελέγχου μετάδοσης/Πρωτόκολλο Internet (TCP/IP)** (Tanenbaum & Wetherall, 2010). Το πρωτόκολλο που είναι υπεύθυνο για τη σύνδεση ανάμεσα στους υπολογιστές του πομπού και του δέκτη του ψηφιακού μηνύματος στο Διαδίκτυο, είναι υπεύθυνο για τη συγκέντρωση των πακέτων στον δέκτη και χειρίζεται τη συλλογή αυτών των πακέτων στα ενδιάμεσα στάδια, είναι το πρωτόκολλο TCP. Για την πραγματική παράδοση των πακέτων το πρωτόκολλο που είναι υπεύθυνο είναι το IP που παρέχει και το σύστημα διευθυνσιοδότησης του Διαδικτύου.

Το πρωτόκολλο TCP/IP είναι ένα πρωτόκολλο για δίκτυα μεγάλης κλίμακας και η διαστρωμάτωση της αρχιτεκτονικής του βασίζεται σε 4 επίπεδα (Comer, 2014):

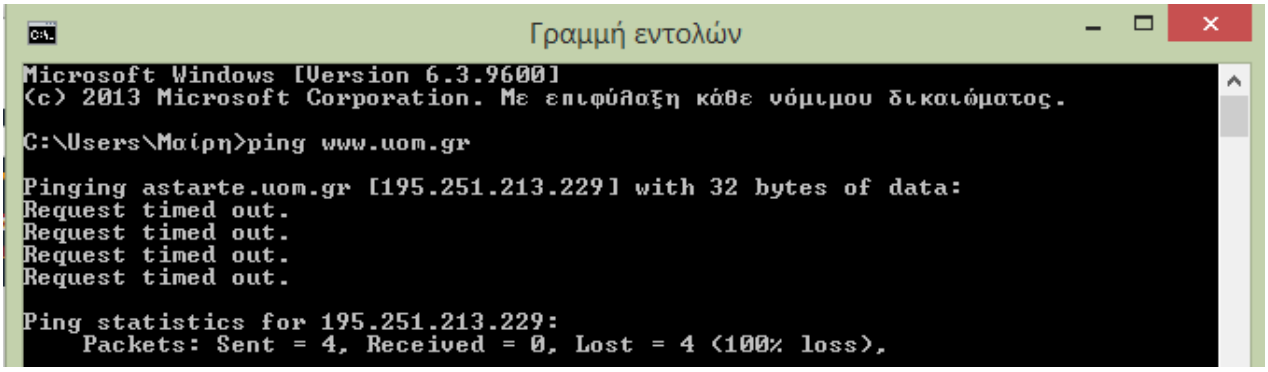
- **Επίπεδο εφαρμογής**: παρέχει ένα σύνολο από μια πληθώρα πολλών διαφορετικών εφαρμογών οι οποίες έχουν πρόσβαση στις υπηρεσίες που προσφέρονται από τα κατώτερα επίπεδα της αρχιτεκτονικής του πρωτοκόλλου.
- **Επίπεδο μεταφοράς μεταξύ κεντρικών υπολογιστών (TCP)**: σε αυτό το επίπεδο της αρχιτεκτονικής παρέχεται η επικοινωνία με την εφαρμογή. Αυτό το επίπεδο θέτει σε ακολουθία και ενημερώνει τα πακέτα που εισέρχονται και εξέρχονται από την εφαρμογή.
- **Επίπεδο Internet (IP)**: αναλαμβάνει τη διευθυνσιοδότηση των πακέτων, τη συγκέντρωσή τους και την αποστολή τους στον τελικό παραλήπτη / δέκτη μέσω του Διαδικτύου.
- **Επίπεδο διεπαφής δικτύου**: αυτό το επίπεδο αναλαμβάνει την ευθύνη για την τοποθέτηση και τη λήψη των πακέτων μέσω του δικτύου.

Το σύστημα που έχει αναπτυχθεί στο Διαδίκτυο για την IP διευθυνσιοδότηση παρέχει τη δυνατότητα σε εκατομμύρια υπολογιστές που είναι συνδεδεμένοι στο Διαδίκτυο να έχουν επικοινωνία μεταξύ τους. Κάθε ένας υπολογιστής που συνδέεται στο Διαδίκτυο για να μπορεί τόσο να λάβει όσο και να στείλει πακέτα TCP πρέπει να έχει μια διεύθυνση IP. Σήμερα υπάρχουν 2 εκδόσεις διευθύνσεων IP. Η μια (τρέχουσα) έκδοση είναι η IPv4: είναι ένας αριθμός που εκφράζεται με έναν αριθμό της τάξης των 32 bit, χωρισμένος σε τέσσερις ομάδες με χρήση τελείας (πχ. 52.21.430.12). Δίνει δυνατότητα για 2^32 (περίπου 4,3 δισεκατ.) διαφορετικές διευθύνσεις. Η άλλη (νεότερη) έκδοση, που στοχεύει να πάρει τη θέση της τρέχουσας, είναι η διεύθυνση IPv6: ένας αριθμός αυτή τη φορά της τάξης των 128bit. Δίνει δυνατότητα για 2^128 (περίπου 3,4×10^38) διαφορετικές διευθύνσεις. Ο λόγος για τον οποίο δημιουργήθηκαν οι διευθύνσεις IPv6 είναι η ανάγκη για υποστήριξη περισσότερων διευθύνσεων. Οι διευθύνσεις IPv6 είναι αριθμοί χωρισμένοι σε οκτώ ομάδες, με χρήση άνω-κάτω τελείας ως διαχωριστή, όπου κάθε ομάδα έχει 4 δεκαεξαδικά ψηφία (πχ. 2001:0db8:85a3:0042:1000:8a2e:0370:7334).

Όπως προαναφέραμε μια διεύθυνση IPv4 αποτελείται από 32 bit (4 bytes) και είναι μοναδική για κάθε υπολογιστή και κατά ευρεία κλίμακα και για κάθε χρήστη που εισέρχεται στο Διαδίκτυο. Επειδή είναι δύσκολο για ένα χρήστη να μπορεί να θυμάται τις αριθμητικές τιμές, η IP εκφράζεται σε φυσική γλώσσα μέσα από το όνομα τομέα. Ένα σύστημα ονομάτων τομέα (DNS) εκφράζει σε φυσική γλώσσα τις αριθμητικές τιμές από τις οποίες αποτελείται μια διεύθυνση IP (παράδειγμα η IP διεύθυνση 195.251.213.229 είναι η αντίστοιχη στο όνομα τομέα uom.gr). Η διεύθυνση που χρησιμοποιεί ένα πρόγραμμα περιηγητή για να εντοπίσει μια θέση περιεχομένου στον ΠΙ ονομάζεται Ενιαίος Εντοπιστής Πόρων (URL). Πχ. αν βρισκόμαστε στην κεντρική σελίδα του Πανεπιστημίου Μακεδονίας και θέλουμε να πάμε στις πληροφορίες που αφορούν το τμήμα της Εφαρμοσμένης Πληροφορικής, το URL που θα ακολουθήσουμε είναι

http://www.uom.gr/index.php?tmima=6&categorymenu=2. Παρατηρούμε ότι η χρήση διεύθυνσης σε φυσική γλώσσα (για να μεταβεί σε μια ιστοσελίδα και για να περιηγηθεί μέσα σε αυτήν) είναι παρόμοια με του καταλόγου/φακέλου αρχείων σε έναν υπολογιστή.

Αν θέλουμε να δούμε αν μια ιστοσελίδα είναι ενεργή και γνωρίζουμε το URL της σελίδας, μπορούμε να χρησιμοποιήσουμε την εντολή ping και αμέσως μετά με έναν κενό χαρακτήρα να δώσουμε το URL αυτής. Έτσι στέλνουμε και λαμβάνουμε κάποια πακέτα δοκιμής σύνδεσης, για να δούμε αν είναι ενεργός ο τομέας (μπορούμε να πληροφορηθούμε και την IP διεύθυνση της σελίδας). Για λόγους ασφαλείας αρκετές διευθύνσεις μπορεί να μην ανταποκρίνονται στην εντολή ping (δε στέλνουν πίσω πακέτα), για να προστατευθούν από πιθανές επιθέσεις.



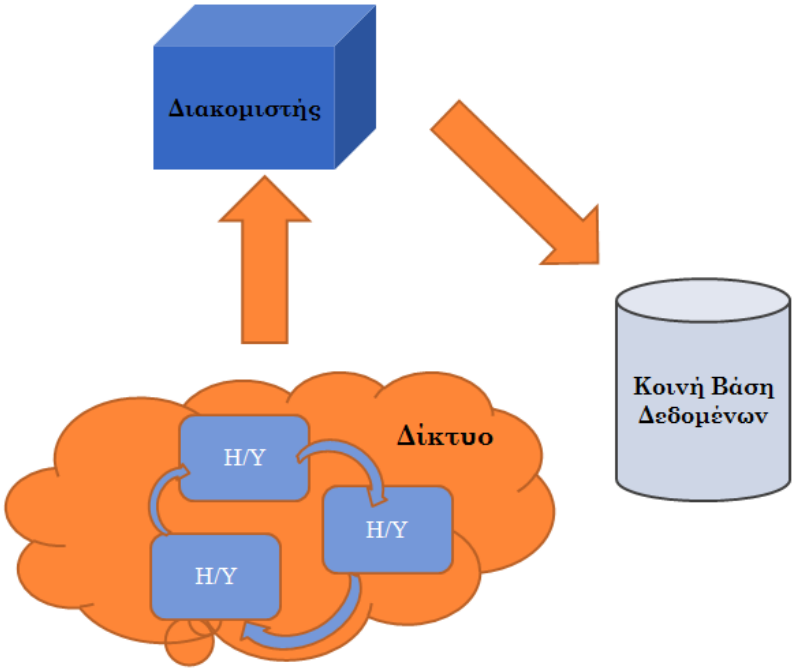
Εικόνα 1. 3 *Παράδειγμα φυσικής γλώσσας και IP διεύθυνσης*

2.3 Μοντέλο πελάτη-διακομιστή

Η μεταγωγή πακέτων και το πρωτόκολλο TCP / IP κατάφεραν να παρέχουν τους θεμέλιους λίθους για τους νόμους και τους κανόνες που διέπουν την επικοινωνία στο Διαδίκτυο. Ωστόσο για να φτάσουμε στο επίπεδο που γνωρίζουμε το Διαδίκτυο σήμερα, χρειάστηκε η αξιοποίηση του μοντέλου πελάτη-διακομιστή. Στο μοντέλο αυτό, αρχικά σχετικά μικρής ισχύος υπολογιστές, αλλά πλέον και αρκετά ισχυροί υπολογιστές (ως πελάτες), συνδέονται σε ένα δίκτυο μέσω ενός ή περισσότερων διακομιστών. Όπως βλέπουμε και στην παρακάτω εικόνα έχουμε στην πλευρά του δικτύου ένα σύνολο από υπολογιστές που είναι οι πελάτες και μέσω ενός διακομιστή εισέρχονται για την πλοήγησή τους στο Διαδίκτυο. Οι διακομιστές είναι δικτυωμένοι υπολογιστές που η χρήση τους βασίζεται στη συνεχή εκτέλεση απαραίτητων κοινών λειτουργιών που πραγματοποιούνται από τους υπολογιστές – πελάτες, λειτουργίες όπως εφαρμογές λογισμικού, αποθήκευση αρχείων κλπ.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 1.4.gif	Κινούμενη Εικόνα (interactive)
Μοντέλο πελάτη/διακομιστή	



Εικόνα 1. 4 *Μοντέλο πελάτη/διακομιστή*

3. Ο Παγκόσμιος Ιστός

Sound 1.4.mp3	Ηχητικό απόσπασμα (audio)
Σημαντικότητα του Παγκόσμιου Ιστού	

Ο Παγκόσμιος Ιστός (ΠΙ) είναι μια από τις πιο διαδεδομένες και ευρέως χρησιμοποιούμενες υπηρεσίες που προσφέρει το Διαδίκτυο. Ο ΠΙ στηρίζεται στην υποδομή του Διαδικτύου, είναι κατά μία έννοια ένα τμήμα του όπως παρουσιάζουμε στην ακόλουθη εικόνα. Για να κατανοήσουμε καλύτερα τη σύνδεση που υπάρχει μεταξύ του Διαδικτύου και του ΠΙ παρέχουμε παρακάτω τους ορισμούς τους:

Διαδίκτυο (Internet): αποτελεί τον γενικευμένο όρο για τη φυσική διασύνδεση σε επίπεδο υλικού (hardware) μεταξύ ενός δικτύου υπολογιστών. Μέσα στο κοινό δίκτυο των Η/Υ που δημιουργείται υπάρχουν μηχανισμοί όπως το TCP/IP πρωτόκολλο, peer-to-peer δίκτυα, δρομολογητές κλπ. που όλα μαζί συμβάλλουν στη δημιουργία του Διαδικτύου.

Παγκόσμιος Ιστός (ΠΙ, Web): το Διαδίκτυο λειτουργεί ως το φυσικό μέσο αποθήκευσης της πληροφορίας που δημιουργείται στον ΠΙ μέσω των χρηστών του. Ο ΠΙ αφορά στην πλευρά του λογισμικού (software) και προσφέρει τη δυνατότητα διαμοίρασης πληροφοριών και υπηρεσιών (π.χ. κοινωνικά δίκτυα, ηλεκτρονικές συναλλαγές).

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 1.5.gif	Κινούμενη Εικόνα (interactive)
Το Διαδίκτυο και ο Παγκόσμιος Ιστός	

Εικόνα 1. 5: *Το Διαδίκτυο και ο Παγκόσμιος Ιστός*

Ο ΠΙ αποτελεί ένα κατανεμημένο σύστημα μεγάλης κλίμακας με δισεκατομμύρια ιστοσελίδες που δημιουργούνται από μη-συντονισμένες ενέργειες εκατομμυρίων χρηστών. Ο ΠΙ είναι ένα μοναδικό μέσο παγκόσμιας εμβέλειας για αποθήκευση, μεταφορά και διαμοίραση περιεχομένου / δεδομένων μεταξύ χρηστών σε όλων τον κόσμο. Έχει συμβάλει σημαντικά στη βελτίωση πολλών διαφορετικών τομέων όπως είναι οι συναλλαγές, η εκπαίδευση, οι επιχειρήσεις κλπ.

Όπως συμβαίνει στα περισσότερα ζητήματα, έτσι και στον ΠΙ υπάρχουν εκτός από καθαρά θετικές και λιγότερο θετικές πτυχές που δημιουργούν προκλήσεις για την αντιμετώπισή τους. Πιο συγκεκριμένα, τα πλεονεκτήματα που προσφέρει ο ΠΙ είναι:

- εύκολη, γρήγορη, άμεση πρόσβαση και χρήση
- ανοικτή/ελεύθερη πρόσβαση σε μεγάλο όγκο περιεχομένου (γνώσεις, πληροφορίες)
- εφαρμογές και υπηρεσίες που συμβάλλουν στη συλλογική συνείδηση (collective awareness)
- Από την άλλη όψη του νομίσματος του ΠΙ υπάρχουν και οι ακόλουθες προκλήσεις:
- Υπάρχει (και συνεχώς αυξάνει) μεγάλος όγκος δεδομένων από τις διαφορετικές ιστοσελίδες, εφαρμογές και υπηρεσίες που υπάρχουν στον ΠΙ Λόγω του μεγάλου όγκου πληροφορίας, απαιτείται συνεχής προσπάθεια για δημιουργία ειδικών εργαλείων υποστήριξης της σύνδεσης και οργάνωσης πληροφοριών, για παροχή ποιοτικών αποτελεσμάτων αναζήτησης.
- Ανάγκη για συνεχή εξέλιξη στην υποστήριξη ποιοτικής εμπειρίας πλοήγησης: η σχεδίαση μιας ιστοστελίδας (διεπαφής και λειτουργικότητάς της) που στοχεύει στη βελτίωση της εμπειρίας χρήστη (user experience) είναι μια σημαντικά ισχυρή παράμετρος διατήρησης του ενδιαφέροντος του χρήστη
- Οι πληροφορίες βρίσκονται αρκετές φορές διάσπαρτες και ασύνδετες (ανάγκη για πληρέστερες μορφές διασύνδεσης των δεδομένων, linked data).

Η τρέχουσα κατάσταση που επικρατεί στον ΠΙ είναι η συνεχής εξέλιξη και ανανέωση τεχνολογιών, η ολοένα και μεγαλύτερη συμμετοχή των χρηστών ως δημιουργοί περιεχομένου, καθώς και η συνεχής αύξηση της χρήσης των κινητών συσκευών ως μέσο πρόσβασης στις υπηρεσίες και εφαρμογές του. Προσπαθώντας να συνοψίσουμε την υπάρχουσα κατάσταση του ΠΙ μπορούμε να αναφέρουμε τα εξής:

- Σε επίπεδο υποδομής έχουμε φθηνότερες και περισσότερες μηχανές με μεγαλύτερη ταχύτητα
- Οι χρήστες είναι διαχειριστές και ρυθμιστές της πληροφορίας.
- Υπάρχουν συνεχείς αλλαγές και γρήγορες εξελίξεις που αφορούν τόσο τις τεχνολογίες (π.χ. εξέλιξη σε κινητές συσκευές και χρήση κατάλληλων τεχνολογιών, όπως jQuery mobile) όσο και τις νέες ανάγκες που εμφανίζονται

- Αλλαγή του μεγέθους της κλίμακας σύμφωνα με τον όγκο των δεδομένων (αλγόριθμοι και τεχνικές που θεωρούνται αποδοτικές σε μια κλίμακα 100X δεν είναι το ίδιο με μια 1000X).

Ο ΠΙ μπορεί να αναπαρασταθεί και με τη μορφή γράφου, δηλαδή ενός συνόλου κόμβων (nodes) και ακμών (arcs). Στο επίπεδο του Διαδικτύου οι κόμβοι είναι οι υπολογιστές και οι δρομολογητές ενώ στο επίπεδο του ΠΙ κόμβοι είναι οι ιστοσελίδες και ακμές οι υπερσύνδεσμοι (Broder et al., 2000). Σημεία προσοχής:

- Ο ΠΙ αποτελεί έναν μεγάλης κλίμακας γράφο που είναι δυναμικά εξελισσόμενος και γεωγραφικά κατανεμημένος.
- Η δομή του γράφου δεν είναι μια τυχαία 'κατάσταση': τη δομή του γράφου τη συναντάμε σε αρκετές επιστήμες, όπως είναι η βιολογία ή τα μαθηματικά.
- Στόχος της μελέτης του ΠΙ ως γράφο είναι η ανακάλυψη νέων τεχνικών και αλγορίθμων που συμβάλλουν στην καλύτερη ευρετηριοποίηση και κατανομή του περιεχομένου.
- Η μελέτη του ΠΙ ως γράφο βοηθά και τη διάσταση της αναπαράστασης ενός κοινωνιολογικού δικτύου αφού οι χρήστες (στα επιμέρους κοινωνικά δίκτυα τα οποία συμμετέχουν) μοιράζονται τις απόψεις και τα συναισθήματά και υπάρχουν αναφορές σε τρέχοντα γεγονότα.
- Η δομή του ΠΙ ως γράφου μπορεί να συμβάλει στην κατανόηση της συμπεριφοράς των χρηστών κατά τη διάρκεια των συναλλαγών τους.

3.1 Ιστοσελίδες και υπερκείμενο

Οι ιστοσελίδες που επισκέπτονται οι χρήστες κατά την πλοήγησή τους στο ΠΙ χωρίζονται σε στατικές και δυναμικές ιστοσελίδες. Η πλειονότητα πλέον των ιστοσελίδων είναι δυναμικού περιεχομένου. Τα χαρακτηριστικά των στατικών σελίδων είναι:

- Είναι γραμμένες με πηγαίο κώδικα html.
- Το περιεχόμενο προϋπάρχει.
- Το περιεχόμενό τους παραμένει το ίδιο μέχρι που κάποιος να το αλλάξει στον πηγαίο κώδικα (source code).
- Η ροή της πληροφορίας είναι μονόδρομη και δεν παράγεται σε πραγματικό χρόνο.
- Είναι η μορφή σελίδων που χρησιμοποιήθηκαν στην αρχική φάση του ΠΙ (Web 1.0), ενώ σήμερα είναι περιορισμένες οι καταστάσεις στις οποίες θεωρούνται ως κατάλληλή λύση.

Τα χαρακτηριστικά για τις δυναμικές σελίδες που χρησιμοποιούνται πλέον στον ΠΙ είναι:

- Η πληροφορία παράγεται σε πραγματικό χρόνο.
- Το περιεχόμενο της σελίδας δημιουργείται δυναμικά (δηλαδή συντίθεται τη στιγμή που ένας χρήστης επισκέπτεται την ιστοσελίδα μέσω ενός περιηγητή).
- Οι δυναμικές ιστοσελίδες συμβάλλουν στη γρήγορη ανταπόκριση του περιεχομένου που ζητούν οι χρήστες και αυτό ενημερώνεται/μεταβάλλεται με γρήγορους ρυθμούς.
- Συνδέονται με ένα σύνολο τεχνολογιών (π.χ. front-end και back-end) και τεχνικών προγραμματισμού είτε στην πλευρά του πελάτη, είτε στην πλευρά του διακομιστή.

Ο λόγος για τον οποίο οι χρήστες μπορούν να προσπελάσουν τις ιστοσελίδες είναι γιατί το λογισμικό του περιηγητή που χρησιμοποιούν μπορεί να ζητά τις ιστοσελίδες που είναι αποθηκευμένες σε έναν κεντρικό διακομιστή χρησιμοποιώντας το πρωτόκολλο HTTP (Hypertext Transfer Protocol). Ως υπερκείμενο (hypertext) λογίζεται ένα κείμενο το οποίο περιέχει λέξεις-φράσεις οι οποίες είναι συνδεδεμένες με άλλες λέξεις, φράσεις ή σελίδες με τρόπο λογικό και μη σειριακό-γραμμικό. Όταν ο χρήστης επιλέξει τη λέξη – σύνδεσμο οδηγείται στο συνδεδεμένο νέο σημείο. Ο τρόπος μορφοποίησης των ιστοσελίδων ο οποίος συνδέει κείμενα (ή και πολυμεσικά στοιχεία) ή άλλες σελίδες μεταξύ τους είναι η προαναφερθείσα λογική του υπερκειμένου (υπερμέσα, όταν εμπλέκονται και πολυμεσικά στοιχεία).

Το πρωτόκολλο του Διαδικτύου που αναλαμβάνει να μεταφέρει ιστοσελίδες είναι το Πρωτόκολλο Μεταφοράς Υπερκειμένου (HTTP). Το πρωτόκολλο HTTP σχεδιάστηκε από την ομάδα σχεδίασης του Διαδικτύου (IETF) και από τον παγκόσμιο οργανισμό (W3C). Όταν ένας πελάτης ζητάει να δει ένα ψηφιακό περιεχόμενο, όπως είναι μια ιστοσελίδα, τότε ξεκινάει το πρωτόκολλο HTTP. Η συνδιάλεξη HTTP τερματίζεται αν ο διακομιστής ανταποκριθεί στο αίτημα και αφού στείλει πίσω το περιεχόμενο που ζητήθηκε.

3.2 Γλώσσες σήμανσης στον ΠΙ

Στον ΠΙ χρησιμοποιούνται ευρέως δύο γλώσσες σήμανσης (markup) κειμένου:

1. HTML (Hyper Text Markup Language): είναι γλώσσα που χρησιμοποιείται για την ανάπτυξη ιστοσελίδων στην πλευρά του πελάτη (client-side, front end) και είναι μια σχετικά εύκολη γλώσσα. Για τη μορφοποίηση των ιστοσελίδων παρέχεται στους σχεδιαστές ιστοσελίδων ένα σταθερό σύνολο «ετικετών» το οποίο τους βοηθάει να μορφοποιούν τις σελίδες που σχεδιάζουν κάτω από ένα κοινά αναγνωρισμένο και ευρέως χρησιμοποιούμενο πλαίσιο. Η HTML βασίζεται σε ετικέτες (tags ή elements):
    - Μια ετικέτα (tag) έχει αρχή και κλείσιμο π.χ. <h1> </h1>
    - Ένα στοιχείο (element) είναι και το tag και το κείμενο που περιέχεται μεταξύ της αρχής και του κλεισίματος του tag (element content). Παράδειγμα: <h1> Τίτλος 1 </h1>
    - Υπάρχει δυνατότητα ένθεσης των ετικετών
2. XML (Extensible Markup Language): αφορά την ηλεκτρονική κωδικοποίηση κειμένου μέσα από ένα σύνολο κανόνων που προσδιορίζει την εγκυρότητα μιας ετικέτας σήμανσης σε ένα συγκεκριμένο πλαίσιο λειτουργίας. Περιλαμβάνει δηλαδή στο λεξιλόγιό της, όχι μόνο κάποια σύνολα προκαθορισμένων ετικετών, αλλά και τη δυνατότητα καθορισμού νέων ετικετών για την υποστήριξη επιμέρους αναγκών. Κατά ένα τρόπο μπορεί να θεωρηθεί υπερσύνολο της σύγχρονης HTML, αφού οι σύγχρονοι περιηγητές όταν αποδίδουν το περιεχόμενο σελίδων HTML μπορούν πέρα από τις προκαθορισμένες HTML ετικέτες, να ‘κατανοήσουν’ και οποιαδήποτε ‘νόμιμη’ ως προς τους XML κανόνες ετικέτα έχει δημιουργηθεί και έχει γίνει αποδεκτή (μέσω χρήσης XML σχημάτων).

4. Σύγχρονα χαρακτηριστικά στον παγκόσμιο Ιστό

Για περισσότερες από δυο δεκαετίες έχουμε γίνει μάρτυρες της ραγδαίας και δυναμικής εξέλιξης του ΠΙ και της υιοθέτησης νέων χαρακτηριστικών που προσαρμόζονται στις απαιτήσεις των καιρών. Το Web 2.0 μπορεί να χαρακτηριστεί ως η καινοτομία στην ιστορία του Διαδικτύου μέχρι σήμερα, καθώς η έλευσή του αποτέλεσε ορόσημο για μια “τεχνολογική έκρηξη” σε σύγκριση με το Web 1.0, που πρόσφερε έναν απλό τρόπο πλοήγησης η οποία καθιστούσε τη σχέση με τον χρήστη μονόδρομη. Στην επιστημονική και επιχειρηματική κοινότητα έχει αξιοποιηθεί στο έπακρο το Web 2.0 τόσο από την οπτική των τεχνολογικών υποδομών όσο και από τη διαχείριση του περιεχομένου. Θεμέλιος λίθος του Web 2.0 είναι η αλληλεπίδραση με τους τελικούς χρήστες οι οποίοι παρέχουν δεδομένα στον ΠΙ και δεν πλοηγούνται απλώς στις σελίδες του. Στην παρούσα υπό-ενότητα θα αναφερθούμε σε σύγχρονες τάσεις, αναδυόμενες τεχνολογίες και χαρακτηριστικά που παρατηρούνται στον ΠΙ

4.1 Η εξέλιξη του Διαδικτύου και του παγκόσμιου Ιστού

Καθημερινές πρακτικές που υπάρχουν σε πολλαπλούς τομείς της ζωής μας, όπως είναι η εκπαίδευση ή η υγεία ή ο χρηματοοικονομικός τομέας ή ακόμα και η ίδια η προσωπική ζωή ενός ανθρώπου, έχουν επηρεαστεί σε υψηλό βαθμό από τη ραγδαία εξέλιξη του Διαδικτύου καθώς και από τις υπηρεσίες και εφαρμογές που προσφέρει. Η αστείρευτη συμμετοχή των χρηστών στον ΠΙ μέσα από τις εφαρμογές και τις υπηρεσίες που παρέχονται, όπως είναι τα κοινωνικά δίκτυα ή τα ηλεκτρονικά καταστήματα, αποτελεί μια από τις κύριες πηγές τροφοδότησης περιεχομένου στον Ιστό, η οποία συνέβαλε στην άνθιση του Web 2.0. Η

εξέλιξη του Διαδικτύου και η ενεργή συμμετοχή των χρηστών υποστηρίζεται τόσο από το υλικό όσο και από το λογισμικό:

- υλικό (hardware): η πρόσβαση των χρηστών στο Διαδίκτυο και η αξιοποίηση των χαρακτηριστικών του συνδέεται άμεσα με την οικουμενική πρόσβαση που προσφέρεται. Οι χρήστες μπορούν να έχουν σε πραγματικό χρόνο από οποιαδήποτε σημείο βρίσκονται πρόσβαση στο Διαδίκτυο, με αποτέλεσμα να μπορούν να αλληλεπιδρούν με άλλους χρήστες, να πραγματοποιούν αγορές ή γενικά να χρησιμοποιούν τις υπηρεσίες του Διαδικτύου. Σε αυτήν τη δυνατότητα έχει συμβάλλει η δημιουργία κατάλληλων τηλεπικοινωνιακών υποδομών οι οποίες έχουν προσφέρει ασύρματη πρόσβαση και η ευρεία χρήση των κινητών συσκευών.
- λογισμικό (software): από την πλευρά του λογισμικού αναφερόμαστε τόσο στο λογισμικό που χρησιμοποιείται για να υποστηρίξει τη λειτουργία των συσκευών (π.χ. Windows, Android κ.α.) όσο και σε τεχνολογίες Διαδικτύου (π.χ. PHP, HTML, XML κ.α.) οι οποίες υποστηρίζουν τη δημιουργία υπηρεσιών και εφαρμογών του ΠΙ. Οι χρήστες χρησιμοποιώντας τις συσκευές και το λογισμικό, εκτελούν διαδικτυακές δραστηριότητες από οποιαδήποτε σημείο βρίσκονται οποιαδήποτε χρονική στιγμή το θελήσουν.

Η εποχή του Web 2.0, η οποία χαρακτηρίζεται από τον Κοινωνικό Ιστό και την πρόσβαση πληροφοριών σε πραγματικό χρόνο, έχει δύο όψεις που αναπτύσσονται ταυτόχρονα και επηρεάζουν η μία την εξέλιξη της άλλης. Η πρώτη όψη αφορά την ύπαρξη εφαρμογών και υπηρεσιών που προσφέρονται στους χρήστες του ΠΙ, με στόχο τη διευκόλυνσή τους για ανάρτηση υλικού, αξιολογήσεων, κρίσεων, σχολίων και γενικότερα για δημιουργία κλίματος 'κοινοτήτων' χρηστών (όπως είναι πχ. τα ιστολόγια ή τα κοινωνικά δίκτυα). Η δεύτερη όψη είναι οι απαιτούμενες τεχνολογίες. Για να λειτουργήσουν αυτές οι εφαρμογές αναπτύσσονται αντίστοιχα τεχνολογίες για να υποστηρίξουν τις απαιτήσεις ανάπτυξης και λειτουργίας τους (όπως είναι οι οντολογίες ή οι τεχνολογίες ανάπτυξης διαδικτυακών εφαρμογών). Πλέον με την εξάπλωση του Σημασιολογικού Ιστού (Semantic Web) βρισκόμαστε στην αρχή της εποχής του Web 3.0 και κύριο χαρακτηριστικό αυτού είναι ότι τα δεδομένα που υπάρχουν στον ΠΙ αναλύονται και έτσι μπορούν να προσφέρονται εξατομικευμένες απαντήσεις/υπηρεσίες στους τελικούς χρήστες.

Οδεύοντας πλέον προς έναν «ευφυή» Ιστό (Intelligent Web), που όπως προμηνύεται σε μερικά χρόνια θα αποτελεί το Web 4.0 (Aghaei et al., 2012). Μπορούμε να παρατηρήσουμε ότι η τωρινή κατάσταση του Διαδικτύου τροφοδοτεί την εξέλιξή του. Πιο συγκεκριμένα έχουμε την άνθηση του Διαδικτύου των Αντικειμένων (Internet of Things, IoT), όπου «αντικείμενα» (όπως είναι οι φυσικοί αισθητήρες) και άνθρωποι, μέσα από τις εικονικές προσωπικότητες που διαμορφώνουν στον ΠΙ, παράγουν δεδομένα τα οποία μέσα από την αξιοποίησή τους παρέχουν υπηρεσίες και εξατομικευμένες λύσεις για την εξυπηρέτηση διαφόρων τομέων της καθημερινής ζωής. Στα πλαίσια του IoT μια μεγάλη πρόκληση είναι τα μεγάλα δεδομένα και η διαχείρισή τους για την εξαγωγή ποιοτικών πληροφοριών. Στην ίδια εν μέρει λογική, εμφανίζεται και μια νέα μορφή, το «Διαδίκτυο από Εσένα» (Internet of You, IoY) όπου κύριο χαρακτηριστικό του είναι τα μικρά δεδομένα που υπάρχουν ξεχωριστά από κάθε έναν χρήστη. Μέσα από την ανάλυση των μικρών δεδομένων στόχος είναι να παρατηρείται η δραστηριότητα κάθε χρήστη στο Διαδίκτυο κατά τη διάρκεια της ημέρας, να συλλέγονται και να αναλύονται τα «προσωπικά ίχνη» τους.

Στον ακόλουθο πίνακα μπορούμε να δούμε τις τρεις εξελίξεις του ΠΙ και την αντίστοιχη εξέλιξη κάποιων χαρακτηριστικών που υπάρχουν στην πρώτη κάθετη στήλη.

	Web 1.0 (Σύρσιμο, crawl)	**Web 2.0 (Βάδισμα, walk)**	**Web 3.0 (Τρέξιμο, run)**
Πληροφορίες	Περισσότερο μόνο για ανάγνωση	Ευρεία χρήση και για ανάγνωση και για εγγραφή	Φορητές και Προσωπικές
Επικοινωνία	Αναμετάδοση	Αλληλεπίδραση	Δέσμευση / Επένδυση
Εστίαση	Εστίαση στην Εταιρία	Εστίαση στην κοινότητα / στο σύνολο	Εστίαση στο άτομο
Προσωπικά	Σελίδες Χρηστών (Home pages)	Ιστολόγια και Wikis	Κοινωνικά ρεύματα (lifestreams)

Περιεχόμενο	Κάτοχος του περιεχομένου	Διαμοίραση περιεχομένου	Ενοποίηση περιεχομένου
Αλληλεπίδραση	Φόρμες Ιστού	Εφαρμογές Ιστού	Έξυπνες Εφαρμογές
Αναζήτηση	Κατάλογοι	Ετικέτες	Συμπεριφορά χρήστη
Μετρικές	Προβολή Σελίδας	Κόστος ανά click	Συμμετοχή του χρήστη
Διαφήμιση	Διαφήμιση με Banners	Διαδραστική Διαφήμιση	Διαφήμιση βασισμένη στη συμπεριφορά του χρήστη
Έρευνα	Online διαθέσιμη η εγκυκλοπαίδεια Britannica	Wikipedia	DBpedia
Τεχνολογίες	HTML / Portals	XML / RSS	RDF / RDFS / OWL

Πίνακας 1. 1 *Σύγκριση χαρακτηριστικών Web 1.0, Web 2.0 και Web 3.0*

4.2 Κοινωνική τεχνολογία / κοινωνικός Ιστός

Το Web 2.0 έχει αναπτυχθεί ως πλατφόρμα πάνω στην αρχική υποδομή του Διαδικτύου, που βασιζόταν περισσότερο στο υλικό κομμάτι, και στη συνέχεια προσέφερε εφαρμογές και υπηρεσίες. Ιδιαίτερα συμβολική ήταν η ταχύτητα εξέλιξη των τεχνολογιών που σχετίζονται με την κοινωνική δικτύωση και αποτέλεσαν αρωγό για τη ραγδαία εξάπλωση του Διαδικτύου και των χαρακτηριστικών του. Παρατηρούμε στο Web 2.0 ότι η υπερπληθώρα διαθεσίμων εφαρμογών και υπηρεσιών προς ευρεία χρήση βρίσκει υψηλά επίπεδα απήχησης στους χρήστες και μέχρι και σήμερα ακολουθεί αυξητική τάση. Οι χρήστες υιοθετούν στην καθημερινή (σταθερή και κινητή) πλοήγησή τους στον κυβερνοχώρο τις νέες προτάσεις της τεχνολογικής εξέλιξης και συμμετέχουν ενεργά στη δημιουργία, δημοσίευση, ανταλλαγή και αξιολόγηση περιεχομένου στον ΠΙ (Georgiadis, 2012).

Οι εφαρμογές προσφέρουν τη δυνατότητα στους χρήστες να αλληλεπιδρούν με το κοινωνικό εικονικό τους δίκτυο και με άλλους χρήστες και να δημιουργούν περιεχόμενο. Ως αποτέλεσμα είναι να παρέχεται η δυνατότητα οι χρήστες να έχουν τη δική τους "φωνή" για να εκφράζουν την άποψή τους για τρέχοντα γεγονότα καθώς και να επηρεάζουν τις εξελίξεις. Έτσι έχουμε την υποστήριξη έκφρασης της σοφίας του πλήθους (wisdom of crowds, WOC). Η αλληλεπίδραση των χρηστών οδηγεί στη δημιουργία δικτύων κοινότητας μεταξύ τους. Αυτά χαρακτηρίζονται από ένα σύνολο χαρακτηριστικών (όπως είναι το ενδιαφέρον για μια κοινή θεματολογία ή η κοινωνική σύνδεση που έχουν μεταξύ τους στον πραγματικό κόσμο, όπως είναι οι φίλοι ή οι συνάδελφοι, και μεταφέρονται και στον εικονικό). Για να μπορούν οι χρήστες να διαμοιράζονται μεταξύ τους περιεχόμενο, να παρακολουθούν και να επηρεάζουν τις εξελίξεις και γενικά να έχουν κοινωνική δραστηριότητα στον ΠΙ διαμορφώθηκαν κάποιες τάσεις - υπηρεσίες στο Web 2.0:

- **Wikis**: αποτελούν ένα θεμέλιο χαρακτηριστικό, που είναι και ένα από τα πρώτα χαρακτηριστικά του Web 2.0, το οποίο δίνει με τον πιο εύκολο και γρήγορο τρόπο τη δυνατότητα στους χρήστες να δημιουργήσουν το δικό τους περιεχόμενο. Σε ένα wiki πολλαπλοί χρήστες έχουν τη δυνατότητα να δημιουργήσουν και να επεξεργαστούν περιεχόμενο χωρίς να υπάρχει κάποια ειδική απαίτηση για εγγραφή ή χρήση εξειδικευμένης τεχνολογίας. Οι χρήστες χρησιμοποιώντας έναν απλό περιηγητή και την απλή γλώσσα σήμανσης των wikis μπορούν να δημιουργήσουν το περιεχόμενό τους. Μέσα από τη συνεργατικότητα που παρέχεται στους χρήστες, μπορεί να φτιαχτεί περιεχόμενο το οποίο μπορεί να εξυπηρετεί από τον πιο απλό σκοπό ενημέρωσης μέχρι και την παροχή εξειδικευμένης ανοικτής γνώσης. Ένα πολύ γνωστό wiki είναι η Wikipedia, που περιέχει πάνω από 31,000,000 άρθρα σε 285 γλώσσες[1]. Χαρακτηριστικό συμβάν που αποτυπώθηκε στη χρήση της Wikipedia ήταν το 2004: όταν είχε συμβεί το καταστροφικό τσουνάμι στον Ινδικό Ωκεανό, κατά τις πρώτες στιγμές που συνέβη η φυσική καταστροφή κανένα ΜΜΕ δεν είχε έγκυρες πληροφορίες αλλά οι χρήστες της Wikipedia κατάφεραν να δημιουργήσουν

[1] http://en.wikipedia.org/wiki/Wikipedia:About

μέσα σε λίγες ώρες περιεχόμενο το οποίο ήταν αξιόπιστο και περιέγραφε την εξέλιξη της καταστροφής.

- **Ιστολόγια** (Blogs): τα ιστολόγια είναι σελίδες οι οποίες δημιουργούνται από έναν συγκεκριμένο χρήστη, ο οποίος είναι και ο διαχειριστής της σελίδας. Ο τύπος της διαδραστικότητας ενός ιστολογίου χαρακτηρίζεται αμφίδρομος με τους υπόλοιπους χρήστες γιατί αυτοί μπορούν να παρακολουθούν και να σχολιάζουν το περιεχόμενο που αναρτάται. Στο σύνολό τους τα ιστολόγια έχουν ανοικτή πρόσβαση και οι διαχειριστές αναρτούν περιεχόμενο που μπορεί να περιέχει προσωπικές απόψεις για τρέχοντα ή μη θέματα, ή ακόμη με στόχο απλώς την πληροφόρηση. Το περιεχόμενο που δημοσιεύεται στα ιστολόγια είναι απλό κείμενο ή πολυμεσικό υλικό ή περιέχει συνδέσμους σε άλλες σελίδες στον ΠΙ. Αρχικά τα ιστολόγια είχαν ξεκινήσει και με την έννοια του προσωπικού ημερολογίου. Τα ιστολόγια όμως πλέον έχουν εξελιχθεί και δίνουν τη δυνατότητα στους δημιουργούς τους να έχουν δυναμικό το περιεχόμενό τους. Βασίζονται κυρίως σε χρήση λογισμικού τύπου 'Σύστημα Διαχείρισης Περιεχομένου' (CMS, Content Management System), όπως πχ. WordPress, Joomla, κ.ά. Ενδεικτικά το 2014 στον ΠΙ καταγράφονται 75.8 εκατομμύρια ιστολόγια που δημιουργήθηκαν με το WordPress και 172 εκατομμύρια στο Tumblr[2].
- **Κοινωνικά Συστήματα Ετικετοποίησης και Κοινωνικής Σελιδοσήμανσης** (Social Tagging Systems, STS και Social Bookmarking, SB): οι χρήστες στον Κοινωνικό Ιστό ανεβάζουν ή βλέπουν το υπάρχον περιεχόμενο. Για να μπορεί να υπάρχει ένα κοινό κανάλι 'συνεννόησης' και να βρίσκεται και διαμοιράζεται εύκολα το περιεχόμενο, οι χρήστες επισημαίνουν το περιεχόμενο με κάποιες ετικέτες (tags). Η ετικετοποίηση είναι ελεύθερη και συνδέεται με τη χρήση συγκεκριμένων λέξεων-κλειδιών που δίνουν την περιληπτική εικόνα σχετικά με το περιεχόμενο που υπάρχει σε μια ιστοσελίδα. Οι χρήστες χρησιμοποιούν τις ετικέτες ως κοινό αναγνωριστικό που είναι κοινά αποδεκτό από την κοινότητα για την κατηγοριοποίηση του περιεχομένου. Γνωστό STS είναι η κοινότητα του Stackoverflow[3], όπου οι χρήστες διατυπώνουν προβλήματα που αντιμετωπίζουν σχετικά με συγκεκριμένες τεχνολογίες. Για να μπορούν να έχουν μια κοινή βάση ομαδοποίησης στο περιεχόμενό τους, εισάγουν ετικέτες που συνδέονται με την τεχνολογία που χρησιμοποιούν ώστε να μπορούν οι αντίστοιχοι χρήστες που γνωρίζουν την εκάστοτε τεχνολογία να τους απαντήσουν. Ένα ακόμα παράδειγμα διαδεδομένου STS είναι το Flickr[4] όπου οι χρήστες διαμοιράζονται φωτογραφίες και τις κατηγοριοποιούν με τη χρήση ετικετών. Το Delicious[5], είναι στην παρεμφερή κατηγορία της Κοινωνικής Σελιδοσήμανσης: μέσα από τη χρήση ετικετών και της σήμανσης ιστοσελίδων επιτρέπει στους χρήστες να αποθηκεύσουν και να μοιραστούν τις αγαπημένες τους σελίδες στον ΠΙ και να έχουν πρόσβαση από όπου και αν βρίσκονται αρκεί να συνδεθούν στην υπηρεσία.
- **Υπηρεσίες Κοινωνικής Δικτύωσης** (Social Networking Services): τα συστήματα SNS άρχισαν να βρίσκουν πρόσφορο έδαφος από τη συνεχώς παρεχόμενη απευθείας σύνδεση στο δίκτυο. Τα κοινωνικά δίκτυα βασίζονται σε δεσμούς που δημιουργούν οι χρήστες μεταξύ τους όταν φτιάχνουν το δίκτυό τους. Οι δεσμοί αυτοί μπορεί να είναι κοινωνικοί δεσμοί που υπάρχουν στον πραγματικό κόσμο, όπως είναι ο κοινωνικός περίγυρος ενός ατόμου (οικογένεια, φίλοι ή συνάδελφοι), οι οποίοι και μεταφέρονται στον εικονικό κόσμο των κοινωνικών δικτύων. Ωστόσο μπορεί να είναι και δεσμοί που δημιουργούνται στον εικονικό κόσμο, όπως είναι η δημιουργία κοινοτήτων που κοινό τους χαρακτηριστικό είναι το κοινό ενδιαφέρον για ένα συγκεκριμένο θέμα, όπως θα μπορούσαν να είναι οι θαυμαστές ενός μουσικού συγκροτήματος. Οι κοινότητες είτε προέρχονται από τον πραγματικό κόσμο είτε δημιουργήθηκαν στον εικονικό, έχουν ως κοινό χαρακτηριστικό τους ότι στοχεύουν να

---

2 http://en.wikipedia.org/wiki/Blog

3 http://stackoverflow.com/

4 https://www.flickr.com/

5 https://delicious.com/

συμβάλλουν στην επικοινωνία μεταξύ των μελών τους και στη δημιουργία περιεχομένου που εμπίπτει στο ενδιαφέρον της κάθε μίας από αυτές. Το πιο ευρέως γνωστό μέσο κοινωνικής δικτύωσης είναι το Facebook[6]. Άλλο διαδεδομένο κοινωνικό δίκτυο είναι το LinkedIn[7]. Ένα ακόμη αρκετά ευρέως χρησιμοποιούμενο κοινωνικό δίκτυο, που βασίζεται στη λογική των ιστολογίων (blogs), και αποτελεί μια μορφή εξέλιξής τους είναι το Twitter[8] (χαρακτηρίζεται ως υπηρεσία microblogging).

- **RSS**: σύνοψη της πληροφορίας και προσφορά των αρχικών πηγών στον χρήστη. Πρέπει να τηρεί το XML πρότυπο. Συχνά παραφράζεται ως Really Simple Syndication (Πολύ Απλή Διανομή).
- **Περιεχόμενο παραγόμενο από τους χρήστη** (User Generated Content, UGC): οι χρήστες του ΠΙ μοιράζονται το περιεχόμενο τους ή ενημερώνουν το κοινωνικό προφίλ τους με πληροφορίες. Ο ρόλος ενός χρήστη στο Web 2.0 δεν είναι μονόδρομος, αλλά αμφίδρομος/διαδραστικός. Οι χρήστες δε χρησιμοποιούν τον ΠΙ μόνο για να ενημερωθούν, αλλά μπορούν επίσης να δημοσιεύουν το δικό τους περιεχόμενο ή να διαμοιράζονται με άλλους χρήστες οτιδήποτε θεωρούν ενδιαφέρον. Η φύση της ανταλλαγής περιεχομένου μπορεί να είναι προσωπική, ενημερωτική ή επαγγελματική.
- **Mashups**: προσφέρει ενοποιημένο περιεχόμενο από πολλαπλές πηγές / ιστοσελίδες.

4.3 Τεχνολογίες σημασιολογικού Ιστού και οντολογίες

- OWL (Web Ontology Language): είναι μια γλώσσα σημασιολογικού Ιστού η οποία σχεδιάστηκε για την αναπαράσταση πολύπλοκης γνώσης μεταξύ ομάδων πραγμάτων καθώς και των σχέσεων μεταξύ αυτών των πραγμάτων.
- RDF (Resource Description Framework): είναι μια οικογένεια προδιαγραφών του W3C που σχεδιάσθηκε αρχικά ως ένα μοντέλο περιγραφής μεταδεδομένων. Χρησιμοποιείται ως μια γενική μέθοδος εννοιολογικής περιγραφής ή μοντελοποίησης πληροφοριών που περιέχονται/παρέχονται μέσω πόρων του ΠΙ. Η XML παρέχει τη σύνταξη για την RDF αλλά δεν είναι συστατικό της RDF (Antoniou et al., 2012).

4.4 Διασυνδεδεμένα ανοικτά δεδομένα

Για να μπορέσουμε να κατανοήσουμε την περιοχή των ανοικτών διασυνδεδεμένων δεδομένων θα πρέπει να δούμε ποια είναι τα συστατικά στοιχεία που συνθέτουν το πλέγμα:

- Ανοικτά δεδομένα (Open Data): είναι πληροφορίες (δημόσιες ή άλλες) στις οποίες ο καθένας έχει ελεύθερη πρόσβαση και μπορεί να τις χρησιμοποιεί για οποιονδήποτε σκοπό αφορά στη συλλογή συνόλων δεδομένων και δημοσίευσή τους.
- Διασυνδεδεμένα ανοικτά δεδομένα (Linked Open Data): αποτελεί τη σύνδεση δεδομένων από ετερογενείς πηγές. «Είναι ένα σύνολο από τεχνικές και εργαλεία που χρησιμοποιούνται για τη δημοσίευση, ενσωμάτωση και διασύνδεση δεδομένων, δομημένης πληροφορίας και γνώσης στον σημασιολογικό Ιστό με τη χρήση των προτύπων URI και RDF» (Μουνταντωνάκης & Φαφαλιός, 2014).
- URI (Καθολικό Αναγνωριστικό Πόρου, Universal Resource Identifier): κάθε πόρος στο δίκτυο προσδιορίζεται με ένα URI. Οι URL διευθύνσεις είναι ένα υποσύνολο των URIs. Ενδεικτικά παραδείγματα:
    - URL: http://example.org/absolute/URI/with/absolute/path/to/resource.txt
    - URI: http://dbpedia.org/resource/Tim_Berners-Lee

6 https://www.facebook.com/

7 https://www.linkedin.com/

8 https://twitter.com/

5. Στοιχεία ανάλυσης δεδομένων παγκόσμιου Ιστού (web analytics)

Με την Ανάλυση Δεδομένων από τον Παγκόσμιο Ιστό μπορούμε να απαντήσουμε σε πολλαπλά ερωτήματα που προκύπτουν όπως:

- Πώς οι δυνητικοί χρήστες / πελάτες θα καταφέρουν να βρουν το ηλεκτρονικό μας κατάστημα;
- Πώς ξέρουμε και γιατί πρέπει να μας ενδιαφέρει πώς μας βρίσκουν / αναζητούν οι χρήστες (μέσω μηχανών αναζήτησης ή από συνδέσμους από άλλες ιστοσελίδες);
- Τι μπορεί να σημαίνει ο χρόνος παραμονής τους στη σελίδα μας (πχ. έρχονται και φεύγουν αμέσως);
- Τι σημαίνει η παρουσία του χρήστη για αρκετή ώρα στην ιστοσελίδα μας και μετά το ότι φεύγει χωρίς να κάνει κάποια συναλλαγή;
- Εάν κάποιος χρήστης εγκαταλείψει την ιστοσελίδα πριν ολοκληρώσει την παραγγελία του, πώς πρέπει να το λάβουμε υπόψη (πχ. χρειάζεται να επανασχεδιαστεί η ιστοσελίδα ή τα βήματα για την ολοκλήρωση της παραγγελίας);

- Οι δραστηριότητες των χρηστών στην ιστοσελίδα, μας παρέχουν κάποιες χρήσιμες πληροφορίες ή είναι ικανές να μας υποδεικνύουν αλλαγές που πρέπει να γίνουν από την πλευρά μας;

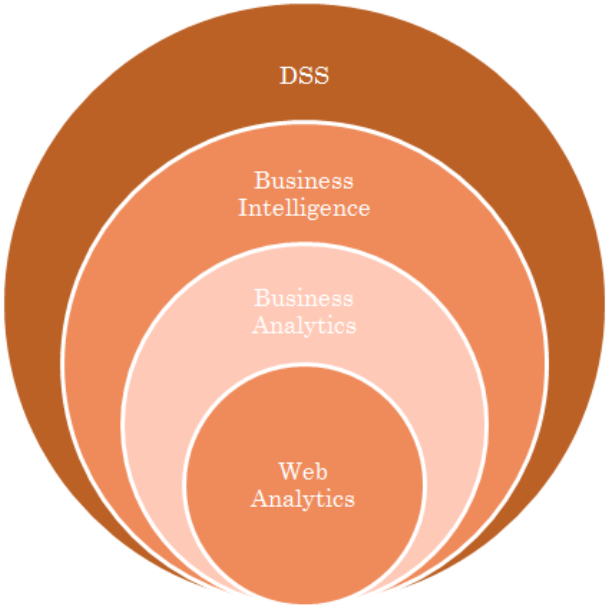
5.1 Εισαγωγικές έννοιες

Μια προσέγγιση κατανόησης της Ανάλυσης Δεδομένων Ιστού, είναι μέσα από το ακόλουθο πλαίσιο:

- **DSS – Συστήματα Υποστήριξης Αποφάσεων** (**Decision Support System**): Ένα εννοιολογικό πλαίσιο το οποία αφορά τη διαδικασία για τη στήριξη διοικητικών αποφάσεων, συνήθως με τη μοντελοποίηση των προβλημάτων που απασχολούν και ειδικότερα την επιλογή ποσοτικών μοντέλων για την εύρεση της καταλληλότερης λύσης.
- **BI - Επιχειρηματική Ευφυΐα** (**Business Intelligence, BI**): αποτελεί υποσύνολο των DSS. Ένας γενικός όρος που συνδέεται με αρκετές έννοιες όπως αρχιτεκτονικές, εργαλεία, βάσεις δεδομένων, εφαρμογές, μεθοδολογίες και επιχειρηματικοί κανόνες, όλες σχετικές με υποστήριξη *επιχειρηματικών* αποφάσεων.
- **BA – Ανάλυση Επιχειρηματικών Δεδομένων** (**Business Analytics, BA**): αποτελούν υποσύνολο της BI. Η εφαρμογή των μοντέλων υποστήριξης αποφάσεων απευθείας στα δεδομένα των επιχειρήσεων.
- **WA – Ανάλυση Δεδομένων Ιστού** (**Web Analytics, WA**): αποτελεί υποσύνολο του BA. Η εφαρμογή των δραστηριοτήτων BA για Web-based διαδικασίες.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 1.6.gif	Κινούμενη Εικόνα (interactive)
Ανάλυση Δεδομένων Ιστού και Επιχειρηματική Ευφυΐα	



Εικόνα 1. 6 *Ανάλυση Δεδομένων Ιστού και Επιχειρηματική Ευφυΐα*

5.2 Μέθοδοι συλλογής δεδομένων για web analytics

Αρχεία καταγραφής (log files): ο κύριος σκοπός για τον οποίο έχουν αναπτυχθεί είναι για αποσφαλμάτωση. Είναι ένα αρχείο που δημιουργείται από τον web server και κάθε φορά που ένας συγκεκριμένος πόρος ζητείται, γράφονται πληροφορίες για αυτόν στο αρχείο καταγραφής.

JavaScript Tags (ετικέτες): Όταν χρησιμοποιείται η ετικετοποίηση JavaScript, κάθε ιστοσελίδα θα πρέπει να έχει και έναν σύντομο κώδικα JavaScript. Όταν ένας επισκέπτης ζητά μια διεύθυνση URL από ένα web server, ο web server στέλνει πίσω τη σελίδα, συμπεριλαμβανομένου του κώδικα JavaScript. Αυτός ο κώδικας εκτελείται ενώ η σελίδα φορτώνει. Συλλαμβάνει διαφορετικά δεδομένα όπως προβολές σελίδων και cookies και τα στέλνει σε ένα διακομιστή συλλογής δεδομένων. Η ποικιλία των δεδομένων που μπορούν να συλλεχθούν είναι τεράστια. Κυμαίνεται από τα clicks και τη θέση του δρομέα του ποντικιού που κινείται μέσα στη σελίδα, μέχρι και την πληκτρολόγηση κειμένου ή την εγκατάσταση πρόσθετων στοιχείων (plug-ins).

Web beacon (Ιστοφάρος): Είναι μια εικόνα 1x1 pixel που αποστέλλεται από τον web server μαζί με την ιστοσελίδα και στη συνέχεια εκτελείται για να στείλει δεδομένα στον διακομιστή συλλογής δεδομένων. Έχουν αναπτυχθεί και χρησιμοποιούνται κυρίως για τη μέτρηση δεδομένων διαφημίσεων, διαφημιστικών banners και e-mails, και για να παρακολουθούν τους χρήστες σε πολλαπλούς διαδικτυακούς τόπους. Υλοποιούνται εύκολα με χρήση ετικετών εικόνων HTML <img src>.

Page sniffing: εφαρμόζεται μεταξύ του χρήστη (browser) και του διακομιστή ιστοσελίδων. Ένας χρήστης ζητά μια σελίδα (αίτημα). Το αίτημα υποβάλλεται στον web server που εκτελεί (μέσω λογισμικού ή μέσω εξειδικευμένου hardware) ένα πακέτο sniffer ικανό να συλλέγει δεδομένα. Στη συνέχεια διαβιβάζεται αυτό μέσω του διακομιστή. Στον δρόμο της επιστροφής προς τον χρήστη το πακέτο sniffer είναι και πάλι μεταξύ του διακομιστή και του πελάτη το υπεύθυνο για τη συλλογή των δεδομένων. Χρησιμοποιώντας αυτή τη μέθοδο συλλογής δεδομένων, η επικοινωνία παρακολουθείται και συλλέγονται ποιοτικά τα δεδομένα.

Δοκιμές χρηστικότητας/ευχρηστίας (Usability testing): Οι δοκιμές ευχρηστίας είναι μια τεχνική που χρησιμοποιείται με επίκεντρο τον χρήστη. Ο σχεδιασμός τους βασίζεται στην αλληλεπίδραση για την αξιολόγηση ενός προϊόντος δοκιμάζοντάς το σε χρήστες. Συνήθως όταν θέλουμε να υλοποιήσουμε αξιολογήσεις ευχρηστίας για την ιστοσελίδα μας συντάσσουμε μια task λίστα με ενέργειες που θέλουμε να πραγματοποιήσει ο χρήστης στην ιστοσελίδα μας. Εκτός από τον χρήστη, υπάρχει και ένας παρατηρητής που παρατηρεί τον χρήστη τόσο στις φυσικές του κινήσεις όσο και στη διαδικασία της περιήγησής του στην ιστοσελίδα. Επίσης υπάρχει και ένα άτομο που εξηγεί στον χρήστη ποια είναι τα βήματα που πρέπει να κάνει. Σε δοκιμές ευχρηστίας μπορεί να καταγράφονται και οι εκφράσεις του προσώπου του χρήστη για να παρατηρηθεί αν υπάρχει δυσκολία σε κάποια tasks που του έχουν ανατεθεί. Με τα αποτελέσματα που

προκύπτουν μπορεί να παρατηρηθεί ότι ο χρήστης δυσκολεύεται να εντοπίσει μια πληροφορία και έτσι ίσως να χρειάζεται να επανασχεδιαστεί κάτι στην ιστοσελίδα.

Ευρετική αξιολόγηση (Heuristic evaluation): είναι μια μέθοδος ελέγχου ευχρηστίας για λογισμικό υπολογιστή, που βοηθά να διαπιστωθούν προβλήματα χρηστικότητας στον σχεδιασμό του περιβάλλοντος εργασίας χρήστη (UI). Σημειώστε: αρχές ευχρηστίας = heuristics.

Επισκέψεις (Site visits): χρησιμοποιείται για να μετρήσει τις επισκέψεις μιας ιστοσελίδας και να αναλύσει τη χρονική διάρκεια της παραμονής τους και τις δραστηριότητες που πραγματοποίησαν στην ιστοσελίδα.

Έρευνα (Survey): συνήθως δίνονται κάποια ερωτηματολόγια στους χρήστες μετά την ολοκλήρωση της περιήγησής τους στην ιστοσελίδα και αφορούν δυσκολίες που αντιμετώπισαν, αν κατάφεραν να βρουν εύκολα μια συγκεκριμένη πληροφορία που αναζητούσαν κλπ.

Μέθοδος	**Τεχνική**
Ποσοτική	**Server – side** - αρχεία καταγραφής (log files)
	Client– side - Page tagging - Web beacon
	Εναλλακτικές μέθοδοι - Packet sniffer
Ποιοτική	Τεχνικές όπως: - Δοκιμές χρηστικότητας / ευχρηστίας (usability testing) - Ευρετική αξιολόγηση (Heuristic evaluation) - Επισκέψεις (Site visits) - Έρευνα (Survey)

Πίνακας 1. 2 *Ενδεικτικές μέθοδοι συλλογής δεδομένων για Web Analytics*

Προκλήσεις στην Ανάλυση Δεδομένων Ιστού

- Εύρεση σχετικών πληροφοριών: παρόλο που από το έτος 2000, οι μηχανές αναζήτησης στο Διαδίκτυο έχουν γίνει πολύ καλύτερες, αρκετές φορές είναι αρκετά δύσκολο να βρει κανείς τις πληροφορίες που ψάχνει. Σημειώστε ότι παρόλο που όλες οι μηχανές αναζήτησης εφαρμόζουν βελτιστοποίηση στην εύρεση των αποτελεσμάτων τους, στον χρήστη δεν εμφανίζονται τα ίδια αποτελέσματα με την ίδια σειρά όταν χρησιμοποιεί διαφορετικές μηχανές αναζήτησης (Miller, 2012).
- Ανάπτυξη νέων γνώσεων με τις διαθέσιμες πληροφορίες: η εύρεση των πληροφοριών είναι η μια πλευρά του νομίσματος, η άλλη πλευρά είναι η καταλληλότητα/εγκυρότητα αυτών. Ιδιαίτερα στον ΠΙ πλέον, όπου ο καθένας μπορεί να αναρτά περιεχόμενο, η διαπίστωση καταλληλότητας/αξιοπιστίας της πληροφορίας είναι σημαντική πρόκληση.
- Εξατομικευμένες πληροφορίες: οι δυνατότητες για την εξατομίκευση δικτυακών τόπων είναι τεράστιες (ποιος θέλει να δει τι, και για τι ενδιαφέρεται). Για να μπορεί να παρέχεται η πληροφορία που θέλει να δει ένας χρήστης σε μια ιστοσελίδα πρέπει πρώτα να μάθουμε/μελετήσουμε τον κάθε χρήστη.
- Μαθαίνοντας για τους χρήστες: η κατανόηση των χρηστών είναι το κλειδί για μια αποτελεσματική εξατομίκευση, αποτελεσματικό σχεδιασμό και διαχείριση ιστοσελίδων, αποτελεσματικό μάρκετινγκ, κλπ.

Αντιμετώπιση των προκλήσεων (βασιζόμενοι στο περιεχόμενο, στη δομή και στη χρήση των διαδικτυακών τόπων) επιτυγχάνεται μέσα από την Εξόρυξη Ιστού (Web Mining).

Εξόρυξη Ιστού

Η διαδικασία της Εξόρυξης Ιστού περιλαμβάνει τα ακόλουθα βήματα:

- Εύρεση των πόρων: σε αυτό το στάδιο εντοπίζουμε που βρίσκονται τα δεδομένα μας.

- Εξαγωγή πληροφορίας (επιλογή και προ-επεξεργασία): οι πληροφορίες από τους πόρους που έχουμε ανακαλύψει θα πρέπει αυτόματα να εξάγονται/αποθηκεύονται (πχ. σε κάποια αρχεία καταγραφής). Επίσης αυτές οι πληροφορίες θα πρέπει να υποστούν μια προ-επεξεργασία/καθαρισμό, για να αφαιρεθεί ο περιττός θόρυβος και να είναι πλέον κατάλληλες για το επόμενο βήμα της ανάλυσης.
- Γενίκευση/ανάλυση προτύπων και αναγνώριση: αυτό το βήμα μπορεί να αποκαλύψει γενικές τάσεις και να βοηθήσει στη σύγκριση με άλλες ιστοσελίδες.
- Ανάλυση/επικύρωση και ερμηνεία: τα πρότυπα που έχουν ανακαλυφθεί πρέπει να αναλυθούν και να επικαιροποιηθούν προκειμένου να έχουμε αξιόλογα συμπεράσματα.

5.2.1 Κατηγοριοποίηση των Δεδομένων

Δομημένα δεδομένα (structured data): δεδομένα τα οποία είναι σε μια σταθερή δομή μέσα σε κάποιο αρχείο και μπορεί να περιέχονται σε σχεσιακές βάσεις δεδομένων ή υπολογιστικά φύλλα.

Ημι-δομημένα δεδομένα (semi-structured data): είναι μια μορφή δομημένων δεδομένων που δεν είναι σύμφωνη με την επίσημη δομή των μοντέλων δεδομένων που σχετίζονται με σχεσιακές βάσεις δεδομένων ή άλλες μορφές πινάκων δεδομένων, αλλά παρ' όλα αυτά περιέχει ετικέτες ή άλλους δείκτες για να διαχωρίσει σημασιολογικά στοιχεία και να επιβάλει ιεραρχίες των εγγραφών και των πεδίων μέσα στα δεδομένα.

Αδόμητα δεδομένα (unstructured data): αναφέρεται σε ελεύθερο κείμενο ή πολυμεσικό περιεχόμενο που μπορεί να βρεθεί σε κοινωνικά δίκτυα ή e-mails κλπ. και χρειάζεται ειδική επεξεργασία/φιλτράρισμα για να εξαχθεί χρήσιμη πληροφορία.

5.2.2 Κατηγοριοποίηση Web Mining

Διακρίνουμε τις ακόλουθες περιπτώσεις στην περιοχή της Εξόρυξης Ιστού:

Εξόρυξη Περιεχομένου (Content Mining): το περιεχόμενο που διατίθεται στο Διαδίκτυο είναι ποικίλο. Στοιχεία όπως κείμενο, ήχος, βίντεο, μεταδεδομένα και υπερσύνδεσμοι είναι τα βασικά χαρακτηριστικά που συνθέτουν το περιεχόμενο των ιστοσελίδων. Αυτά τα στοιχεία είναι στο σύνολό τους σε αδόμητη ή ημι-δομημένη μορφή. Όταν σκεφτόμαστε το περιεχόμενο των ιστοσελίδων δεν εστιάζουμε μόνο στο ίδιο το περιεχόμενο αλλά και στις δυνατότητες που δίνουν οι εταιρίες σε χρήστες να έχουν πρόσβαση σε δεδομένα τους. Με αυτόν τον τρόπο καθίσταται ευκολότερη η προσπάθεια για την ανακάλυψη νέων χρήσιμων πληροφοριών, τη βελτίωση και το φιλτράρισμά τους.

Εξόρυξη Δομής (Structure Mining): αυτός ο τύπος εξόρυξης δεδομένων από τον Ιστό ασχολείται με τη δομή. Η πηγή της πληροφορίας συνήθως είναι η δομή των υπερσυνδέσμων σε μια ιστοσελίδα. Χρησιμοποιείται συνήθως για να κατηγοριοποιηθούν οι ιστοσελίδες προκειμένου να βρεθούν ομοιότητες και σχέσεις μεταξύ τους. Επίσης χρησιμοποιείται για να περιγράψει την κατηγοριοποίηση του περιεχομένου ενός διαδικτυακού τόπου και τη δομή της σελίδας.

Εξόρυξη Χρήσης (Usage Mining): αφορά την ανακάλυψη και κατανόηση της συμπεριφοράς των χρηστών. Συγκρίνεται η πραγματική χρήση με την αναμενόμενη και αν κρίνεται απαραίτητο επανασχεδιάζεται η ιστοσελίδα. Τα δεδομένα που συλλέγονται και αναλύονται μπορεί να είναι δεδομένα από αρχεία καταγραφής (log files) ή ετικέτες που χρησιμοποιούνται για τον χαρακτηρισμό περιεχομένου.

5.3 Λογισμικό web analytics

Έχουν αναπτυχθεί πολλά διαφορετικά πακέτα λογισμικού για την ανάλυση των ακατέργαστων δεδομένων στην περιοχή των Web Analytics. Κάθε λογισμικό εξυπηρετεί και διαφορετικές ανάγκες. Τα πακέτα λογισμικού που έχουν φτιαχτεί για υποστήριξη Web Analytics μπορούν να χωριστούν σε 2 κατηγορίες: i) Εμπορικά (commercial) και ii) Ανοικτού κώδικα (open source). Ωστόσο πρέπει να επισημάνουμε ότι κάποιος μπορεί να φτιάξει από το μηδέν και το δικό του περιβάλλον ή ιστοσελίδα που θα υλοποιεί τη διαχείριση των μετρικών Web Analytics (που θα σχετίζονται με τη σελίδα που συντηρεί), αξιοποιώντας βιβλιοθήκες, πλαίσια και λειτουργίες διασύνδεσης.

Κριτήρια για την επιλογή ενός κατάλληλου λογισμικού είναι:

- Κόστος
- Πλήθος χαρακτηριστικών που προσφέρει: ιστορικά δεδομένα, διαγράμματα, χάρτες, συμβατότητα με κινητά δεδομένα κλπ.
- Πραγματικός χρόνος απόκρισης και καθυστερήσεις
- Φιλοξενία (hosting): που θα αποθηκεύονται τα δεδομένα που συλλέγονται
- Γλώσσα, υποστήριξη και οδηγίες χρήσης (documentation)
- Προσδοκώμενος προϋπολογισμός
- Δυνατότητες διευκολύνσεων για δοκιμή, εφαρμογή και εγκατάσταση

Παραδείγματα λογισμικού Web Analytics			
Εμπορικά		**Ανοικτού κώδικα**	
Ονομασία	**Link**	**Ονομασία**	**Link**
Google Analytics	http://www.google.com/analytics/	Piwik	http://piwik.org/
Coremetrics	http://www-03.ibm.com/software/products/en/category/digital-marketing-optimization	Open Web Analytics (OWA)	http://www.openwebanalytics.com/
Omniture	http://www.adobe.com/solutions/digital-marketing.html	Yahoo! Web Analytics	http://web.analytics.yahoo.com/
Webtrends	http://webtrends.com/	Twitalyzer	http://www.twitalyzer.com/5/index.asp
Alexa	http://www.alexa.com		

Πίνακας 1. 3 *Λύσεις λογισμικού για Web Analytics*

5.4 Προχωρημένες μέθοδοι παρακολούθησης και τεχνικές υλοποίησης ανάλυσης δεδομένων Ιστού

5.4.1 Θεμελιώδεις μετρικές για Ανάλυση Δεδομένων Ιστού

Για την Ανάλυση Δεδομένων Ιστού υπάρχει ένα σύνολο από θεμελιώδεις μετρικές (Beasley, 2013):

Μοναδικοί επισκέπτες (Unique Visitors): Ο αριθμός των μεμονωμένων ατόμων/επισκεπτών (φιλτράρεται για τα προγράμματα τύπου 'αράχνη' και 'ρομπότ') που φτάνει στη σελίδα μας μέσα σε ένα καθορισμένο χρονικό πλαίσιο αναφοράς, με δραστηριότητα που συνίσταται από μία ή περισσότερες επισκέψεις σε μια τοποθεσία. Κάθε άτομο υπολογίζεται μόνο μία φορά – είναι ένα μοναδικό μέτρο επισκέπτη για την περίοδο αναφοράς.

Επισκέψεις/Συνεδρίες (Visits/Sessions): Μια επίσκεψη είναι για τον αναλυτή μια αλληλεπίδραση, από ένα άτομο με μια ιστοσελίδα που αποτελείται από μία ή περισσότερες αιτήσεις, οριζόμενες από τη μονάδα του περιεχομένου (π.χ. "προβολή σελίδας"). Εάν ένα άτομο δεν έχει κάνει άλλη ενέργεια (συνήθως πρόσθετες προβολές σελίδας) στην περιοχή μέσα σε μια καθορισμένη χρονική περίοδο, η σύνοδος/επίσκεψη θα τερματίσει. Δηλαδή τερματίζει αν έχει αφήσει τον browser του ανοικτό για πολύ ώρα χωρίς να έχει κάνει κάποια κίνηση μέσα στη σελίδα (μπορεί να έχει αφήσει απλώς το συγκεκριμένο παράθυρο ανοικτό).

Προβολές σελίδας (Page views): Πόσες φορές μια σελίδα προβλήθηκε.

Συμβάντα (Events): Μία εναλλακτική λύση για το 'Page Views' που χρησιμοποιείται για RIA (Rich Internet Applications). Μια ενδεικτική ενέργεια είναι πχ. το κλικ, ή η αιώρηση από πάνω σε μια 'περιοχή' της σελίδας δυναμικού περιεχομένου (πχ. περιεχόμενο Flash ή πολυμεσικό περιεχόμενο ετικετών HTML5) και η πρόκληση με αυτόν τον τρόπο εμφάνισης ανανεωμένου περιεχομένου.

Επισκέψεις / Χτυπήματα (Hits) - τα χτυπήματα αναφέρονται στον αριθμό των αρχείων στον ιστότοπό μας (θα μπορούσε να περιλαμβάνει φωτογραφίες, γραφικά, κουμπιά, κλπ.). Φανταστείτε πχ. μια ιστοσελίδα που έχει φωτογραφίες (κάθε φωτογραφία είναι ένα αρχείο και αντίστοιχα ένα εν δυνάμει "χτύπημα").

Εικόνα 1. 7 *Θεμελιώδεις μετρικές για Web Analytics*

Video 1.1.mp4	Βίντεο (video)
Μετρικές Ανάλυση Δεδομένων Ιστού	

5.4.2 Μετρικές Χαρακτηρισμού Επίσκεψης

1. **Σελίδα Εισόδου (Entry Page)**: Δείχνει την πρώτη σελίδα της επίσκεψης (το URL). Συνήθως όταν υπάρχουν πολλές σελίδες εισόδου εμφανίζονται σε μια λίστα διευθύνσεων URL με τον αριθμό των επισκέψεων που δέχτηκε η κάθε μία.
2. **Σελίδα Προορισμού (Landing Page)**: είναι μια σελίδα στην οποία οδηγείται ένας δυνητικός πελάτης όταν κάνει click σε μια διαφήμιση (banner). Παράδειγμα: ένας χρήστης που βρίσκεται σε μια ιστοσελίδα εάν πατήσει click στο διαφημιστικό banner τότε οδηγείται σε μια νέα ιστοσελίδα. Αυτή η νέα ιστοσελίδα είναι η σελίδα προορισμού. Υπάρχουν 2 κατηγορίες σελίδων προορισμού:

    - **Σελίδα προορισμού αναφοράς (reference landing page)**: παρουσιάζει πληροφορίες σχετικές με τον επισκέπτη όπως κείμενο, εικόνες, σχετικούς συνδέσμους και άλλα στοιχεία.
    - **Σελίδα προορισμού μέσω μιας συναλλαγής (transactional landing page)**: μια τέτοια σελίδα προσπαθεί να "πείσει" έναν επισκέπτη να ολοκληρώσει μια συναλλαγή, με το να συμπληρώσει μια φόρμα ή να παίξει ένα διαδραστικό παιχνίδι. Συμβαίνει συνήθως σε διαφημιστικά banners, ώστε μετά από κάποια clicks ο χρήστης να οδηγείται στη σελίδα προορισμού. Στόχος είναι η πώληση ενός προϊόντος ή μιας υπηρεσίας.

3. **Σελίδα Εξόδου (Exit Page):** η τελευταία σελίδα σε μια τοποθεσία στην οποία είχε πρόσβαση ο επισκέπτης πριν τερματίσει την επίσκεψη / συνεδρία. Σημαντικός δείκτης είναι η ακόλουθη αναλογία:

    Page Exit Ratio = (Page Exits/Page Visits)

4. **Διάρκεια Επίσκεψης (Visit Duration)**: πόσο διήρκησε η επίσκεψη στη σελίδα.
5. **Click-through**: είναι ο αριθμός των κλικ που έγιναν σε (διαφημιστικούς) web σύνδεσμους για να πραγματοποιηθούν (μέσω αυτών) επισκέψεις στο προορισμό (ενός διαφημιζόμενου).

6. **Αναλογία Click-through** (**Click-through rate, CTR**): συνήθως ορίζεται ως ο αριθμός των clicks σε έναν διαφημιστικό σύνδεσμο και διαιρείται με τον αριθμό των εμφανίσεων διαφημίσεων για μια δεδομένη χρονική περίοδο. Είναι ένας τρόπος μέτρησης της επιτυχίας μιας online διαφημιστικής καμπάνιας. Υπολογίζουμε το CTR (συνήθως επί τοις εκατό), διαιρώντας τον «αριθμό των χρηστών που έκαναν κλικ σε μια διαφήμιση» (clicks) σε μια ιστοσελίδα διά του «αριθμού των φορών που η διαφήμιση εμφανίστηκε» (impressions).

   CTR = (clicks / impressions) × 100

   Παράδειγμα: αν μια διαφήμιση έχει εμφανιστεί 100 φορές, και έχει γίνει μόνο ένα click σε αυτήν, τότε το CTR είναι 1%.
   Σήμερα μια μέση τιμή CTR είναι 0.2-0.3, αλλά υπάρχουν διαφοροποιήσεις ανάλογα με το προϊόν/υπηρεσία που διαφημίζεται.
7. **Παραπομπή - Αναφορά (Referrer):** είναι μια σελίδα σε έναν άλλο ιστότοπο που οδήγησε τον χρήστη στη δικιά μας ιστοσελίδα. Τα URLs διευθύνσεων αναφορών μας λένε από πού έφτασαν οι επισκέπτες στον ιστότοπό μας. Χρησιμοποιώντας αυτή την πληροφορία μπορούμε να αξιολογήσουμε καλύτερα ποιοι εξωτερικοί ιστότοποι οδηγούν συχνότερα στη σελίδα μας και να προσθέσουμε συνδέσμους ή διαφημίσεις που να οδηγούν στον ιστότοπό μας. Υπάρχουν 4 διαφορετικές παραπομπές:
   - Εσωτερικές παραπομπές (Internal referrer): είναι μια διεύθυνση URL της σελίδας που βρίσκεται στο εσωτερικό του δικτυακού τόπου
   - Εξωτερικές παραπομπές (External referrer): είναι μια διεύθυνση URL της σελίδας που βρίσκεται έξω από τον ιστότοπό μας.
   - Παραπομπές αναζήτησης (Search referrer): είναι μια εσωτερική ή εξωτερική παραπομπή για τις οποίες η διεύθυνση URL έχει δημιουργηθεί από μια λειτουργία αναζήτησης.
   - Παραπομπές επίσκεψης (Visit referrer): είναι η πρώτη παραπομπή σε μια συνεδρία, είτε εσωτερική, είτε εξωτερική.
8. **Page Views per Visit:** πόσες περισσότερες σελίδες επισκέφθηκε ο χρήστης στον ιστότοπό μας, «πόσο πιο βαθιά έφτασε».

5.4.3 Μετρικές Χαρακτηρισμού Περιεχομένου

1. **Αναλογία Σελίδας Εξόδου** (**Page Exit Ratio**): Είναι ο αριθμός των εξόδων από μια συγκεκριμένη σελίδα διαιρούμενο με τον συνολικό αριθμό των προβολών σελίδων της σελίδας αυτής.
2. **Επισκέψεις Μιας Σελίδας** (**Single Page Visits**): ο χρήστης δεν προχώρησε πέρα από την πρώτη σελίδα που επισκέφτηκε. Χρησιμοποιείται και ο όρος '**Αναπήδηση**' (**Bounce**). Συμβαίνει όταν ένας επισκέπτης σε μια ιστοσελίδα βλέπει μόνο μία σελίδα του δικτυακού τόπου και την αφήνει χωρίς να επισκεφτεί άλλες σελίδες. Μια αναπήδηση ενός χρήστη μπορεί να συμβεί:
   - Κάνοντας κλικ σε ένα σύνδεσμο από μια σελίδα του ιστότοπού μας σε μια ιστοσελίδα διαφορετικού ιστότοπου
   - Κλείνοντας ένα ανοικτό παράθυρο ή καρτέλα
   - Πληκτρολογώντας μια νέα διεύθυνση URL
   - Κάνοντας κλικ στο κουμπί "Επιστροφή" για να εγκαταλείψει τη σελίδα
   - Λόγω τερματισμού (timeout) της συνόδου
3. **Ρυθμός Αναπήδησης** (**Bounce Rate**): είναι το ποσοστό των 'αναπηδήσεων' (επισκέψεων μιας σελίδας)

Bounce rate = total number of visits that left one page / total number of web visits

5.4.4 Μετρικές Μετατροπής

Ως ‘μετατροπή’ (conversion), ορίζεται μια ενέργεια που σηματοδοτεί την ολοκλήρωση μιας ορισμένης δραστηριότητας. Ο χρήστης της σελίδας ‘μετατρέπεται’ σε πελάτη όταν πχ. αγοράζει ένα προϊόν, εγγράφεται για ένα ενημερωτικό δελτίο ή κατεβάζει ένα αρχείο. Χρησιμοποιούνται συνήθως cookies που εγκαθίστανται στην πλευρά του χρήστη για να γνωρίζουν την επισκεψιμότητά του, να τον αναγνωρίζουν ως μοναδικό χρήστη και βαθμιαία αυτός ‘προάγεται’ σε ‘πελάτη’. Αν ο χρήστης διαγράφει συνεχώς τα cookies, τότε όταν επισκέπτεται το site θα φαίνεται σαν νέος επισκέπτης, και θα παραμένει ΄χρήστης’.

Σημαντική μετρική μετατροπής για την ανάλυση δεδομένων Ιστού είναι (Clifton, 2012) ο ρυθμός μετατροπής (conversion rate) η οποία ορίζεται ως η σχέση μεταξύ των επισκεπτών σε μια ιστοσελίδα και ενεργειών που θεωρούνται ότι είναι ‘μετατροπή’, όπως μια πώληση ή το να ζητήσουν να λάβουν περισσότερες πληροφορίες. Ο ρυθμός μετατροπής είναι το ποσοστό των επισκεπτών που θα μετατραπούν σε πελάτες.

5.4.5 Άλλες Σημαντικές Μετρικές Web Analytics

Σελίδα με τις Περισσότερες Επισκέψεις (Most Viewed Page): η πιο δημοφιλής σελίδα ή η περισσότερο ζητούμενη διεύθυνση URL με τις περισσότερες προβολές σελίδας. Μετράει απλώς ποια σελίδα είχε τον μεγαλύτερο αριθμό επισκέψεων.
Χρόνος στη Σελίδα (Time on Site): από καιρό έχει θεωρηθεί ότι όσο μεγαλύτερο χρονικό διάστημα μένει ο επισκέπτης στη σελίδα τόσο πιο επιτυχημένη είναι η σελίδα.
Πραγματικός Χρόνος Παραμονής: Υπολογισμός:

Πραγματικός χρόνος παραμονής στο site = Average Time on Site / (1 - Bounce Rate)

Παράδειγμα: έστω ότι bounce rate = 40% και average time on site = 1 λεπτό, τότε:

Ο πραγματικός χρόνος παραμονής = 1/(1-0.40) = 1/0.6 = 1 λεπτό και 40 δευτερ.

Έξοδοι (Exits): Η αναχώρηση/έξοδος του επισκέπτη από την ιστοσελίδα, που σηματοδοτεί το τέλος μιας επίσκεψης ή συνεδρίας (υπολογίζεται από την ιστοσελίδα αδράνεια για περισσότερο από 30 λεπτά όπου και μετά από αυτό το όριο τερματίζει η σύνοδος).

Κορυφαίες σελίδες (Top Pages): σελίδες που δέχονται τη μεγαλύτερη επισκεψιμότητα.

Διαδρομή επισκέπτη (Visitor Path): η ακολουθία των συνδέσμων, η διαδρομή, που ο επισκέπτης χρησιμοποιεί για να περιηγηθεί μέσα στον ιστότοπο.

Μήκος Επίσκεψης (Visit Length): το συνολικό ποσό του χρόνου που ο επισκέπτης περνά στην ιστοσελίδα.

Ανάλυση λέξης κλειδιού (Keyword Analysis): οι λέξεις-κλειδιά που χρησιμοποίησαν οι επισκέπτες για να βρουν την ιστοσελίδα στις μηχανές αναζήτησης.

6. Συμπεράσματα

Στο παρόν κεφάλαιο αναφερθήκαμε σε ένα σύνολο βασικών εννοιών που διέπουν το Διαδίκτυο και τον ΠΙ και αποτελούν το βασικό υπόβαθρο για την κατανόηση των τεχνολογιών Ηλεκτρονικού Εμπορίου. Η ραγδαία εξέλιξη που ακολουθεί τα τελευταία χρόνια ο ΠΙ οδηγεί σε παράλληλη εξέλιξη και τις τεχνολογίες ΗΕ. Το ΗΕ αποτελεί την εμπορική διάσταση του ΠΙ και η επίδραση των τεχνολογιών ΠΙ σε αυτό είναι προφανής: κάθε μέρα πραγματοποιούνται εκατομμύρια συναλλαγές και οι χρήστες διαμοιράζονται περιεχόμενο του οποίου ο όγκος χρειάζεται ειδική διαχείριση για να μπορούν να εξαχθούν ποιοτικές πληροφορίες. Όπως μάλιστα ανέδειξε η σύντομη επισκόπησή μας της περιοχής της Ανάλυσης Δεδομένων Ιστού, η εξαγωγή ποιοτικών πληροφοριών μπορεί να συμβάλλει στον εντοπισμό λαθών και στη συνεχή βελτίωση των προσφερόμενων υπηρεσιών.

Βιβλιογραφία / Αναφορές

Aghaei, S., Nematbakhsh, M. A. & Farsani, H. K. (2012). Evolution of the world wide web: from Web 1.0 to Web 4.0. *International Journal of Web & Semantic Technology*, 3(1), 1-10.

Antoniou, G., Groth, P., Harmelen, F. & Hoekstra, R. (2012). *A Semantic Web Primer (Information Systems)*, The MIT Press, Third edition edition.

Beasley, M. (2013). *Practical Web Analytics for User Experience*, Elsevier Inc.

Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A. & Wiener, J. (2000). Graph structure in the web. *Computer networks*, 33(1), 309-320.

Clifton, B. (2012). *Advanced web metrics with Google Analytics*. John Wiley & Sons.

Comer, D. E. (2014). *Computer Networks and internets*. Pearson Higher Ed.

Georgiadis, C. K. (2012). Design and Implementation of Mobile News Services: Supporting Social Networking Features, in *Proc. of the AIS-affiliated conference 11th International Conference on Mobile Business 2012* (ICMB 2012), TUDelft, The Netherlands, June 2012, Association of Information Systems e-Library (AISeL), http://http://aisel.aisnet.org/icmb2012/7/, pp. 101-112.

Laudon, K. & Traver, C. (2014). *E-Commerce, 10/E*, Prentice Hall, ISBN: 978-0133024449.

Miller, S. A. (2012). *Piwik Web Analytics Essentials*. Packt Publishing Ltd.

Tanenbaum, S. A. & Wetherall, J. D. (2010). *Computer Networks*. Prentice Hall.

Μουνταντωνάκης & Φαφαλιός. (2014). Ithaca: Από Ανοιχτά Δεδομένα σε Ανοιχτά Διασυνδεδεμένα Δεδομένα, https://ellak.gr/2014/07/ithaca-apo-anichta-dedomena-se-anichta-diasindedemena-dedomena/

Quiz1.htm	Τεστ αξιολόγησης (interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Η βασική δομή (οι 'πυλώνες') του Διαδικτύου:

Α) Μεταβάλλεται και προσαρμόζεται στις νέες τεχνολογικές απαιτήσεις

Β) Η βασική δομή με όποια εξέλιξη και αν συμβαίνει παραμένει πάντα η ίδια.

Γ) Εξελισσόταν μέχρι τη δεύτερη περίοδο όπου «Καθιερώθηκε το Διαδίκτυο» (1975-1994).

Απάντηση/Λύση

Β) Η βασική δομή με όποια εξέλιξη και αν συμβαίνει παραμένει πάντα η ίδια.

Κριτήριο αξιολόγησης 2

[*] Κατά τη διαδικασία της μεταγωγής πακέτων η μεταφορά πραγματοποιείται:

Α) Με πλήρη μεταφορά όλου του ψηφιακού μηνύματος από τον πομπό στον δέκτη του μηνύματος.

Β) Με τεμαχισμό του ψηφιακού μηνύματος σε διακριτές μονάδες που ονομάζονται πακέτα και επανασύνθεσή τους στον τελικό προορισμό.

Γ) Με αποστολή του συνολικού μηνύματος από την πλευρά του πομπού και «τεμαχισμό» του πακέτου στον παραλήπτη για να λάβει όλο το μήνυμα.

Απάντηση/Λύση

Β) Με τεμαχισμό του ψηφιακού μηνύματος σε διακριτές μονάδες που ονομάζονται πακέτα και επανασύνθεσή τους στον τελικό προορισμό.

Κριτήριο αξιολόγησης 3

[*] Τι σημαίνει το ακρωνύμιο HTML;

A) Hyper Text Markup Language

B) Home Tool Markup Language

Γ) Hyperlinks and Text Markup Language

Απάντηση/Λύση

A) Hyper Text Markup Language

Κριτήριο αξιολόγησης 4

[*] Η εξέλιξη του Κοινωνικού Ιστού αντιστοιχίζεται στο στάδιο της εξέλιξης του ΠΙ:

Α) Web 1.0

Β) Web 2.0

Γ) Web 3.0

Απάντηση/Λύση
Β) Web 2.0

Κριτήριο αξιολόγησης 5
[*] Η Επιχειρηματική Ευφυΐα αποτελεί υποσύνολο του πεδίου 'Ανάλυση Δεδομένων Ιστού':

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 6
[*] Η μετρική της αναλογίας Click-Through (CTR) εκφράζει το ποσοστό των 'αναπηδήσεων' (επισκέψεων μιας σελίδας).

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 7
[**] Στο πρωτόκολλο TCP/IP το επίπεδο Internet αναλαμβάνει:

Α) Τη διευθυνσιοδότηση των πακέτων

Β) Την ακολουθία και την ενημέρωση των πακέτων που εισέρχονται και εξέρχονται από την εφαρμογή

Γ) Την ευθύνη για την τοποθέτηση και τη λήψη των πακέτων μέσω του δικτύου

Δ) Όλα τα παραπάνω

Απάντηση/Λύση
Α) Τη διευθυνσιοδότηση των πακέτων

Κριτήριο αξιολόγησης 8
[**] Η τεχνολογία Mashup:

Α) Προσφέρει ενοποιημένο περιεχόμενο από πολλαπλές πηγές και ιστοσελίδες

Β) Προσφέρει σύνοψη της πληροφορίας και προσφορά των αρχικών πηγών στον χρήστη

Γ) Είναι ένα σύνολο προδιαγραφών σημασιολογικού Ιστού η οποία σχεδιάστηκε για την αναπαράσταση πολύπλοκης γνώσης μεταξύ ομάδων πραγμάτων

Δ) Είναι μια οικογένεια προδιαγραφών του W3C που σχεδιάσθηκε αρχικά ως ένα μοντέλο περιγραφής μεταδεδομένων.

Απάντηση/Λύση
Α) Προσφέρει ενοποιημένο περιεχόμενο από πολλαπλές πηγές και ιστοσελίδες

Κριτήριο αξιολόγησης 9
[**] Ποια από τις ακόλουθες περιπτώσεις δεν αποτελεί περιοχή της Εξόρυξης Ιστού;

Α) Εξόρυξη Περιεχομένου

Β) Εξόρυξη Συμπεριφοράς

Γ) Εξόρυξη Δομής

Δ) Εξόρυξη Χρήσης

Απάντηση/Λύση
Β) Εξόρυξη Συμπεριφοράς

Κριτήριο αξιολόγησης 10
[**] Ποια από τα παρακάτω δεν αποτελεί θεμελιώδη μετρική για την Ανάλυση Δεδομένων Ιστού:

Α) Προβολές σελίδας

Β) Επισκέψεις/Συνεδρίες

Γ) Μοναδικοί επισκέπτες

Δ) Διάρκεια Επίσκεψης

Ε) Επισκέψεις/Χτυπήματα

Απάντηση/Λύση
Δ) Διάρκεια Επίσκεψης

Κεφάλαιο 2: Τεχνολογίες Ασφάλειας στον Παγκόσμιο Ιστό

Σύνοψη
Κύριος στόχος της ασφάλειας γενικότερα στα πληροφοριακά συστήματα, αλλά και ειδικότερα στα βασισμένα στον Παγκόσμιο Ιστό (Web-based) περιβάλλοντα, είναι η διαφύλαξη της εμπιστευτικότητας, της ακεραιότητας και της διαθεσιμότητας όλων των συστατικών τους μερών. Κάθε εξέλιξη της τεχνολογίας μοιάζει να δημιουργεί νέα προβλήματα ασφάλειας, έτσι η μεγαλύτερη πρόκληση στον χώρο της ασφάλειας οφείλεται ακριβώς στο ότι απαιτεί την άμεση εκμετάλλευση τεχνολογιών αιχμής για την αντιμετώπιση των νέων προβλημάτων που συνεχώς αναδύονται. Στο κεφάλαιο αυτό θα παρουσιαστούν αρχικά οι βασικές έννοιες γύρω από τα ζητήματα ασφάλειας. Στη συνέχεια θα εξεταστούν οι απειλές, οι βασικοί χειρισμοί, οι απαιτήσεις και οι λειτουργίες ασφάλειας στο Διαδίκτυο και στον Παγκόσμιο Ιστό. Τέλος, θα παρουσιαστούν σημαντικές τεχνολογίες ασφάλειας στον Παγκόσμιο Ιστό, όπως η κρυπτογράφηση, οι ψηφιακές υπογραφές, και οι υποδομές δημόσιου κλειδιού.

Προαπαιτούμενη γνώση
Το κεφάλαιο 1 του παρόντος συγγράμματος

1. Εισαγωγή

Ο αγγλικός όρος "security", φέρεται να είναι Λατινικής προέλευσης, αφού προέρχεται από τις αντίστοιχες λατινικές λέξεις "se" που σημαίνει "χωρίς" και "cura" που σημαίνει "φροντίδα" (Bullock & Benford, 1999). Δηλαδή η έννοια της ασφάλειας σε ένα σύστημα μπορεί και να ειδωθεί ως μια επιθυμητή ιδιότητα – κατάστασή του, κατά την οποία οι χρήστες του απαλλάσσονται κάθε έγνοιας και φροντίδας ως προς τη σωστή λειτουργία του.

Στις κυριότερες διαθέσιμες τεχνολογίες ασφάλειας στο Διαδίκτυο περιλαμβάνονται η κρυπτογράφηση, οι ψηφιακές υπογραφές, και οι υποδομές δημόσιου κλειδιού. Η κρυπτογραφία είναι στις μέρες μας κοινά αποδεκτή σαν το πλέον απαραίτητο εργαλείο ασφάλειας στο Διαδίκτυο. Δύο σημαντικές εφαρμογές κρυπτογραφίας είναι η κρυπτογράφηση και οι ψηφιακές υπογραφές. Η κρυπτογράφηση μπορεί να εξασφαλίσει ότι οι διακινούμενες πληροφορίες είναι εμπιστευτικές. Οι ψηφιακές υπογραφές βοηθούν στην επικύρωση της προέλευσης δεδομένων και επιβεβαιώνουν αν τα δεδομένα έχουν αλλοιωθεί. Περαιτέρω δυνατότητες προσφέρονται μέσω των υποδομών δημοσίου κλειδιού, οι οποίες με την έκδοση των πιστοποιητικών ταυτότητας, αποδεικνύονται ικανές για την υποστήριξη ενός μεγάλου μέρους λειτουργιών ασφάλειας στο Internet.

2. Ασφάλεια στο Διαδίκτυο

2.1 Εισαγωγικές έννοιες

Χρειάζεται να γίνει περισσότερο σαφής η εικόνα των «επικίνδυνων καταστάσεων» ή «ζημιών». Τι ακριβώς διακυβεύεται; Οι επικρατούσες απόψεις διακρίνουν τις τρεις ακόλουθες βασικές έννοιες σε σχέση με τη διαχείριση ενός ασφαλούς συστήματος (Cherdantseva & Hilton, 2013):

Εμπιστευτικότητα (confidentiality): Είναι έννοια στενά συνδεμένη με την ιδιωτικότητα (privacy) και τη μυστικότητα (secrecy). Αφορά τη μη αποκάλυψη των ευαίσθητων πληροφοριών σε χρήστες που δεν έχουν την κατάλληλη εξουσιοδότηση.

Sound 2.1.mp3	Ηχητικό απόσπασμα (audio)
Η έννοια της εμπιστευτικότητας	

Ακεραιότητα (integrity): Αφορά τη δυνατότητα τροποποιήσεων (προσθήκες, διαγραφές και μεταβολές) των πληροφοριών. Μόνο σε κατάλληλα εξουσιοδοτημένους χρήστες πρέπει το σύστημα να επιτρέπει τέτοιου είδους ενέργειες. Έτσι διαφυλάσσεται η ακρίβεια και η πληρότητα των περιεχομένων ενός πληροφοριακού συστήματος.

Sound 2.2.mp3	Ηχητικό απόσπασμα (audio)
Η έννοια της ακεραιότητας	

Διαθεσιμότητα (availability): Αφορά τη δυνατότητα άμεσης πρόσβασης στις πληροφορίες, στις υπηρεσίες και γενικότερα σε όλους τους πόρους πληροφορικής τεχνολογίας (IT resources) όταν ζητούνται, χωρίς αδικαιολόγητες καθυστερήσεις.

Sound 2.3.mp3	Ηχητικό απόσπασμα (audio)
Η έννοια της διαθεσιμότητας	

Ανάλογα με τη φύση τους, τα διάφορα πληροφοριακά συστήματα είναι περισσότερο ή λιγότερο «ευαίσθητα» στη δυνατότητα να υποστηρίξουν τα προαναφερθέντα χαρακτηριστικά της ασφάλειας. Γι' αυτό και η προσέγγιση της ασφάλειας πληροφοριακών συστημάτων ξεκινάει από την ανάλυση των αναγκών και των σχετικών κινδύνων που παρουσιάζονται σε κάθε περίπτωση. Στη συνέχεια υπολογίζονται οι επιπτώσεις που θα έχει η εφαρμογή των μηχανισμών προστασίας των πληροφοριών στην απόδοση του συστήματος (ταχύτητα, κόστος επεξεργασίας, ευκολία στη διαχείριση, φιλικότητα στο χρήστη κλπ.) (Pangalos, 1992) και τελικά διαμορφώνεται το κατάλληλο επίπεδο ασφάλειας ως η «χρυσή τομή» ανάμεσα στους κινδύνους που αποφεύγονται, στη συνολική απόδοση του συστήματος και στο κόστος ανάπτυξης και εφαρμογής των μηχανισμών ασφάλειας.

Είναι όμως κοινά αποδεκτό ότι δεν υπάρχει πλήρης ασφάλεια, με την έννοια ότι τα μέτρα πρόληψης ποτέ δε θα είναι ικανά να εμποδίσουν όλων των ειδών τις επικίνδυνες ενέργειες. Προνοώντας λοιπόν για κάθε ενδεχόμενο, μια ακόμη έννοια έρχεται να συμπληρώσει τα χαρακτηριστικά της διαχείρισης ασφάλειας: η υπευθυνότητα (accountability) (ISO/IEC 27000:2014, 2014). Πρέπει το σύστημα να είναι ικανό να καταγράφει επιλεκτικά κάποιες ενέργειες των χρηστών, έτσι ώστε να είναι δυνατόν όσες επηρεάζουν την ασφάλειά του να μπορούν να «ερευνηθούν», και να «οδηγήσουν» στο υπεύθυνο μέρος. Οπότε και είναι δυνατή η απόδοση ευθυνών στον κάθε χρήστη ανάλογα με τη δράση του.

Ο όρος αδυναμία-απάρνησης (non-repudiation) ως χαρακτηριστικό ασφάλειας, αποτελεί μια ειδική περίπτωση της έννοιας της υπευθυνότητας και αναφέρεται ακριβώς στο ότι ένας χρήστης δεν μπορεί να αρνηθεί την ανάληψη της ευθύνης για κάποια πράξη που έκανε. Τέλος, υπάρχουν και όροι που έχουν κάποια σχέση-αναλογία με την ασφάλεια συστημάτων όπως η αξιοπιστία (reliability) ή σιγουριά (safety), δηλαδή η ικανότητα των συστημάτων να λειτουργούν σωστά κάτω από αντίξοες συνθήκες, και η εγκυρότητα (dependability) η οποία ενσωματώνει συνήθως τις έννοιες και της ασφάλειας και της αξιοπιστίας.

2.2 Απειλές σε περιβάλλον Internet

Στις τυπικές απειλές ασφάλειας σε ένα περιβάλλον Διαδικτύου, συμπεριλαμβάνονται (Pangalos, 1998):

- Βλάβες συστατικών μερών (component failure): Σχεδιαστικά λάθη ή ελαττωματικά μέρη υλικού-λογισμικού, είναι ικανά να προκαλέσουν δυσλειτουργία σε κάποιο συστατικό του συστήματος και να οδηγήσουν έτσι σε άρνηση εξυπηρέτησης ή άλλες καταστάσεις επικίνδυνες για την ασφάλεια.
- Ξεφύλλισμα πληροφοριών (information browsing): Η αποκάλυψη ευαίσθητων πληροφοριών σε μη-εξουσιοδοτημένους χρήστες, είτε είναι εισβολείς είτε είναι νόμιμοι χρήστες που επιχειρούν παράνομους τρόπους προσπέλασης, οδηγεί στην απώλεια εμπιστευτικότητας και μπορεί να προκληθεί από την εκμετάλλευση διάφορων μηχανισμών.
- Μη-εξουσιοδοτημένη διαγραφή, μεταβολή ή εισαγωγή πληροφοριών (unauthorized deletion, modification or insertion of information): Η εκούσια ή και ακούσια πρόκληση ζημιών στα πληροφοριακά αγαθά (information assets) οδηγεί στην απώλεια ακεραιότητας για τις λειτουργίες - δεδομένα οργανισμών και ανθρώπων.
- Κακή χρήση (misuse): Η χρήση των πληροφοριακών αγαθών αλλά και των υπόλοιπων πόρων για σκοπούς διαφορετικούς από αυτούς που έχουν προκαθορισθεί, προκαλεί άρνηση εξυπηρέτησης, αύξηση κόστους λειτουργίας των συστημάτων ή δυσφήμιση. Ας μην ξεχνάμε ότι το καλό «επιχειρηματικό όνομα» αποτελεί σημαντικότατο περιουσιακό στοιχείο για οποιοδήποτε οργανισμό.

- Διείσδυση (penetration): Οι εισβολές από μη-εξουσιοδοτημένα πρόσωπα ή συστήματα μπορούν να προκαλέσουν άρνηση εξυπηρέτησης ή να απαιτήσουν σοβαρότατα χρηματικά ποσά για την αντιμετώπιση των συνεπειών των παρενοχλήσεων του συστήματος.
- Διαστρέβλωση (misrepresentation): Οι προσπάθειες ενός χρήστη που παρανομεί, να μεταμφιεστεί σαν ένας χρήστης με εξουσιοδοτήσεις τέτοιες ώστε να μπορεί να κλέψει πληροφορίες ή να εκμεταλλευτεί υπηρεσίες ή να εκκινήσει συναλλαγές που προκαλούν οικονομικές απώλειες ή δυσχέρειες σε ένα οργανισμό.

Για όλες τις προαναφερόμενες απειλές, εκτός της βλάβης συστατικών μερών, η προσφιλέστερη μέθοδος εκδήλωσης επίθεσης γίνεται παραβιάζοντας την ακεραιότητα του κώδικα. Οι κάθε λογής ιοί βρίσκουν γόνιμο έδαφος στο Διαδίκτυο, αφού η πλειονότητα των συνδεδεμένων συστημάτων χρησιμοποιούν μη-έμπιστα λειτουργικά συστήματα (WINDOWS και LINUX), χωρίς σοβαρές δυνατότητες αντίστασης στις απειλές αυτές (Pfleeger & Pfleeger, 2007). Επιπλέον σημαντικός παράγοντας αύξησης της συγκεκριμένης επικινδυνότητας αποτελεί ο τρόπος μετάδοσης αρχείων στο Web και κυρίως η δυνατότητα «κατεβάσματος» κώδικα (downloading code) για εκτέλεση μικρών ανεξάρτητων εφαρμογών (applets) των προγραμμάτων περιπλάνησης στο Διαδίκτυο που προσφέρονται από τις σύγχρονες γλώσσες όπως η Java. Όπως εύκολα μπορεί κανείς να συμπεράνει, πιθανές ατέλειες (flaws) των γλωσσών αυτών, αποτελούν αναμφισβήτητες ρωγμές ασφάλειας των συστημάτων.

Κάτι που πρέπει να τονιστεί στο σημείο αυτό, είναι ότι οι πιθανότητες να εκδηλωθούν επιθέσεις και να πραγματοποιηθούν απειλές όπως οι προαναφερθέντες, αυξάνονται όταν προσφέρεται στο Διαδίκτυο μια ευδιάκριτη εικόνα της οργάνωσης της δικτυακής υποδομής ενός συστήματος. Πάρα πολλές επιθέσεις στο Internet είναι ευκαιριακής φύσης (opportunistic), με την έννοια ότι δεν έχουν συγκεκριμένο στόχο παραβίασης. Απλά εκδηλώνονται σε ένα συγκεκριμένο σύστημα γιατί εκείνη τη στιγμή το σύστημα αυτό «φαντάζει» ως ιδανικός στόχος (τελικός ή ενδιάμεσος) για τους επίδοξους εισβολείς.

2.3 Βασικοί χειρισμοί ασφάλειας στο Διαδίκτυο

Σε γενικές γραμμές τα πρωτόκολλα του Internet, δίνουν τη δυνατότητα σε ένα τρίτο μέρος να παρέμβει με τους ακόλουθους τρόπους στην επικοινωνία δυο νόμιμων μερών (Gollmann, 2011):

- **Υποκλοπή (eavesdropping)**: Οι πληροφορίες παραμένουν ανέγγιχτες, αλλά παραβιάζεται η εμπιστευτικότητά τους. Πχ. η καταγραφή μιας ιδιωτικής συζήτησης.
- **Παραποίηση (tampering)**: Οι πληροφορίες κατά τη μεταφορά τους μεταβάλλονται ή τροποποιούνται και στη συνέχεια στέλνονται στον αποδέκτη. Πχ. η αλλαγή μιας αίτησης χρήστη (user request) ή μιας απάντησης συστήματος (system response).
- **Πλαστοπροσωπία (impersonation)**: Οι πληροφορίες πηγαίνουν σε ένα πρόσωπο που παριστάνει τον νόμιμο αποδέκτη. Χρησιμοποιείται και ο όρος προσποίηση (spoofing) για την περιγραφή της κατάστασης όπου κάποιος ή κάτι επιχειρεί να φανεί σαν κάποιος ή κάτι άλλο. Πχ. ένας χρήστης μπορεί να ισχυρίζεται ότι έχει μια συγκεκριμένη διεύθυνση e-mail, ή ένας δικτυακός τόπος μπορεί να αυτό-προσδιορίζεται ως μια συγκεκριμένη URL (Uniform Resource Locator) διεύθυνση, χωρίς τίποτε από αυτά να ισχύει στην πραγματικότητα.

Συνεπώς, οι χειρισμοί ασφάλειας (security controls) στο Διαδίκτυο κινούνται σε τρεις κυρίως κατευθύνσεις (Ahuja, 1997):

- Αρχικά, είναι η προστασία της ιδιωτικότητας των δεδομένων με βασικό όπλο τους μηχανισμούς κρυπτογράφησης.
- Στη συνέχεια είναι η προστασία στα επικοινωνούντα μέρη του ενός από τον άλλο, δηλαδή του αποστολέα από τον παραλήπτη, και αντίστροφα. Αυτό σημαίνει την προστασία της ακεραιότητας των δεδομένων από τότε που έφυγαν από τον αποστολέα, αλλά και την υποστήριξη αδυναμίας απάρνησης ενεργειών για τα δυο μέρη. Μηχανισμοί σχετικοί με ψηφιακές υπογραφές χρησιμοποιούνται ευρύτατα για τέτοιες λειτουργίες.
- Τέλος, είναι ο έλεγχος γνησιότητας της ταυτότητας των χρηστών, των προγραμμάτων ή των μηχανημάτων (μέσω κυρίως συνθηματικών και ψηφιακών πιστοποιητικών) καθώς και των

εξουσιοδοτήσεων που διαθέτουν για την προσπέλαση των προστατευμένων πόρων του συστήματος (μέσω μηχανισμών ελέγχου προσπέλασης).

2.4 Απαιτήσεις και λειτουργίες ασφάλειας στον παγκόσμιο Ιστό

Πιο αναλυτικά (Pfleeger & Pfleeger, 2007; Ahuja, 1997; Πάγκαλος & Μαυρίδης, 2002), η διαχείριση ασφάλειας (security management) οφείλει να υποστηρίξει τις ακόλουθες υπηρεσίες ασφάλειας (security services) γνωστές και ως λειτουργίες ασφάλειας (security functions):

- **Εμπιστευτικότητα δεδομένων (data confidentiality)**: Η προστασία ενάντια σε μη-εξουσιοδοτημένες αποκαλύψεις πληροφοριών. Η τεχνολογία της κρυπτογράφησης (encryption or cryptography) είναι σχεδόν συνώνυμη της λειτουργίας αυτής, λόγω του κυρίαρχου ρόλου της.
- **Ακεραιότητα δεδομένων (data integrity)**: Η δυνατότητα εντοπισμού παραποίησης και ανάκτησης των δεδομένων. Για την προστασία της εγκυρότητας των δεδομένων εκτός της κρυπτογράφησης, χρησιμοποιούνται μηχανισμοί δημιουργίας περιλήψεων μηνυμάτων (message digests) και ψηφιακών υπογραφών (digital signatures).
- **Αδυναμία απάρνησης (non-repudiation)**: Η προστασία από τη μη-ανάληψη ευθύνης ενός αποστολέα ότι αυτός έστειλε συγκεκριμένα δεδομένα (non-repudiation of origin), καθώς και από την άρνηση ενός παραλήπτη ότι παρέλαβε κάποια δεδομένα (non-repudiation of delivery). Χρησιμοποιούνται οι προαναφερθέντες μηχανισμοί προστασίας ακεραιότητας δεδομένων, μαζί με υποδομές υποστήριξης και διακίνησης ψηφιακών πιστοποιητικών (X.509 certificates). Εποπτείες ή Αρχές Πιστοποίησης (Certification Authorities) αναλαμβάνουν την ευθύνη, ως τρίτες έμπιστες συμβολαιογραφικές αρχές (3rd party trusted notaries) για τη δημιουργία κλίματος εμπιστοσύνης στα επικοινωνούντα μέρη.
- **Αναγνώριση και πιστοποίηση (identification and authentication)**: Η απαίτηση πληροφοριών πιστοποίησης, οι οποίες διακινούνται συνήθως κρυπτογραφημένα, και οι οποίες μπορούν να επιβεβαιώνουν την ταυτότητα των μερών που επικοινωνούν. Ο έλεγχος αυθεντικότητας αφορά δυο διακεκριμένες περιπτώσεις:

    - την ταυτότητα των χρηστών (user or entity authentication). Συνήθως συμβαίνει στην αρχή μιας τοπικής σύνδεσης (local logon) και οι μηχανισμοί που χρησιμοποιούνται ονομάζονται πρωτόκολλα αυθεντικότητας (authentication protocols). Παραδείγματα τέτοιων μηχανισμών είναι η χρήση αναγνωριστικού και συνθηματικού (user-ID & password), οι τεχνικές πρόκλησης-απόκρισης (challenge-response techniques) και άλλες μορφές διαπιστευτηρίων (credentials).
    - την ταυτότητα των συστημάτων ως αφετηρίες - πηγές προέλευσης μηνυμάτων (origin authentication). Χρησιμοποιείται και ο όρος πιστοποίηση κατανεμημένων συστημάτων (authentication of distributed systems). Η λειτουργία αυτή έχει συναφές έργο με τη λειτουργία της αδυναμίας απάρνησης αποστολέα (non-repudiation of origin) και συνεπώς στηρίζεται στις μηχανισμούς ψηφιακών υπογραφών - πιστοποιητικών και αξιοποίησης έμπιστων τρίτων μερών (trusted third parties).

- **Έλεγχος προσπέλασης (access control) και εξουσιοδοτήσεις (authorizations)**: Η προστασία ενάντια σε μη-εξουσιοδοτημένη χρήση των πόρων, είτε είναι υλικό (δικτυακό υλικό, μονάδες επεξεργασίας - αποθήκευσης κλπ.), είτε λογισμικό (κώδικας που εκτελείται ή πρόκειται να εκτελεστεί), είτε δεδομένα. Μηχανισμοί όπως οι λίστες ελέγχου προσπέλασης (Access Control Lists-ACLs) και οι ετικέτες ασφάλειας (security labels), χρησιμοποιούνται για τον περιορισμό στην προσπέλαση των πόρων. Γενικότερα, υποστηρίζουν πολιτικές ασφάλειας που παρέχουν μια πολλαπλών επιπέδων και διαφοροποιημένη προσπέλαση πόρων (supporting different levels of resource access) στους χρήστες ανάλογα με το επίπεδο εμπιστοσύνης που μπορούν να τεκμηριώσουν. Τα δικαιώματα προσπέλασης (access rights) είναι οι απαραίτητες πληροφορίες που συσχετίζουν ένα σύστημα πελάτη με ένα σύστημα διακομιστή και καθορίζουν αν ο πελάτης θα αποκτήσει συγκεκριμένου τύπου προσπέλαση σε

ένα συγκεκριμένο πόρο του διακομιστή. Να τονιστεί εδώ, ότι στον κόσμο του σύγχρονου παγκόσμιου Ιστού και ανάλογα με τη χρονική στιγμή, ένας διακομιστής λειτουργεί προσωρινά ως πελάτης και το αντίστροφο. Οπότε η ασφάλεια πρέπει κάθε φορά να «βλέπει» και προς τις δυο κατευθύνσεις ροής των πληροφοριών.

Επιπλέον σημαντικές παράμετροι για τη διαχείριση ασφάλειας στο Διαδίκτυο, αποτελούν οι μηχανισμοί:

- **Επίβλεψης (auditing) και υπευθυνότητας (accountability)**: Καταγράφουν τις δηλώσεις ταυτότητας και τις ενέργειες των χρηστών (αλλά και των συστημάτων) που αποκτούν πρόσβαση σε προστατευμένους πόρους.
- **Ελέγχου αποδοτικότητας δικτύου (efficiency controls)**: Πρόκειται για μηχανισμούς που καταγράφουν και παρακολουθούν τη συνολική απόδοση του συστήματος και την κίνηση του δικτύου, με σκοπό την αποτροπή καταστάσεων άρνησης εξυπηρέτησης (prevention of Denial of Service).
- **Υποστήριξης συνεργασίας των υπηρεσιών ασφάλειας που προσφέρονται από εφαρμογές (callable security services from applications)**: Οι εφαρμογές παγκόσμιου Ιστού, διαθέτουν ενδεχομένως χαρακτηριστικά ασφάλειας που πρέπει να μπορούν να κληθούν και να λειτουργούν με ενιαίους τρόπους. Η βασική έννοια της υποστήριξης ενός βασικού πλαισίου συνεργασίας ασφαλών εφαρμογών (Security Application Program Interface) προωθείται μέσω του προτύπου Generic Security Service API (GSSAPI).

2.5 Σχέσεις υπηρεσιών - μηχανισμών

Στηριζόμενοι στις επισημάνσεις των προηγουμένων παραγράφων και στην ευρέως αποδεκτή σύσταση της ITU για υιοθέτηση του προτύπου ISO 7498-2 για την αρχιτεκτονική υπηρεσιών ασφάλειας (ITU-T X.800 & ISO/IEC 7498-2, 1991), παραθέτουμε τον ακόλουθο απλοποιημένο πίνακα που χωρίς να είναι δεσμευτικός, επισημαίνει παραστατικά ποιοι μηχανισμοί ασφάλειας μπορούν να χρησιμοποιηθούν για να υποστηρίξουν αντίστοιχες υπηρεσίες ασφάλειας.

Υπηρεσία	Κρυπτογράφηση	Ψηφιακ Υπογρ.	Έλεγχοι Προσπέλ	Μηχαν. Ακερ.	Πρωτοκ. Αυθεντ.	Παρεμβ. Κίνησης	Έλεγχοι Δρομολ.	Έμπιστα Τρίτα Μέρη
Εμπιστευτικότητα δεδομένων	X							
Εμπιστευτικότητα ροής δεδομένων	X					X	X	
Ακεραιότητα δεδομένων	X	X		X				
Αδυναμία απάρνησης αποστολέα	X	X		X				X
Αδυναμία απάρνησης παραλήπτη	X	X		X				X
Πιστοποίηση χρήστη	X	X		X	X			
Πιστοποίηση προέλευσης-συστημάτων	X	X		X				X
Έλεγχος προσπέλασης - εξουσιοδοτήσεις	X		X					X

Πίνακας 2.1 *Σχέσεις υπηρεσιών - μηχανισμών*

3. Κρυπτογράφηση στο Διαδίκτυο

3.1 Εισαγωγή

Με τη ραγδαία ανάπτυξη της χρήσης του Διαδικτύου παρουσιάστηκαν διάφορα προβλήματα ασφάλειας που αφορούν την εξασφάλιση της μυστικότητας και ακεραιότητας των αποθηκευμένων και διακινουμένων δεδομένων. Κάθε μεταφορά πληροφορίας, θα πρέπει βεβαίως να είναι ασφαλής και αξιόπιστη. Εάν οι χρήστες του Internet δεν έχουν την πεποίθηση ότι η επικοινωνία τους και τα δεδομένα που ανταλλάσσουν είναι ασφαλή από μη εξουσιοδοτημένη πρόσβαση ή παραποίηση, αυτό αποτελεί ανασταλτικό παράγοντα στην επιλογή τους να το χρησιμοποιήσουν ευρύτερα και ως μέσο διακίνησης των πιο κρίσιμων πληροφοριών τους (όπως τα ιατρικά τους δεδομένα, οικονομικής φύσεως στοιχεία κλπ.).

Σε ένα δίκτυο ανοικτό, όπως αυτό του παγκόσμιου Ιστού, τα μηνύματα είναι δυνατόν να υποκλαπούν και να μεταβληθούν, η εγκυρότητα των πληροφοριών είναι διαβλητή και τα προσωπικά δεδομένα μπορεί να καταχωρηθούν παράνομα. Ένα αποδεκτό επίπεδο ασφάλειας σε περιβάλλοντα ηλεκτρονικού εμπορίου (Laudon & Traver, 2014) μπορεί να προσφέρει η συνδυασμένη χρήση πολιτικών, διαδικασιών, και τεχνολογιών διασφάλισης συναλλαγών (transaction security). Είναι σημαντικό να διασφαλισθούν κατά τη μεταφορά τους οι πληροφορίες. Έτσι, η χρησιμότητα της κρυπτογράφησης έχει εύστοχα παρομοιαστεί με αυτή των τεθωρακισμένων φορτηγών που φροντίζουν την ασφαλή μεταφορά χρημάτων από τράπεζα σε τράπεζα. Με την ίδια λογική βέβαια, είναι φανερό ότι τα μέτρα προστασίας που λαμβάνονται κατά τη διακίνηση πολύτιμων αγαθών αποδεικνύονται άχρηστα αν στη συνέχεια αυτά τα αγαθά μετά την παράδοσή τους εγκαταλειφθούν σε χώρους ελεύθερης πρόσβασης.

3.2 Κρυπτογραφία

Η κρυπτογραφία (cryptography) ως επιστήμη της ασφαλούς επικοινωνίας προσφέρει διάφορες τεχνικές για την κρυπτογράφηση και την αποκρυπτογράφηση μηνυμάτων. Ο ρόλος της κρυπτογραφίας αρχικά, με τις συμβατικές τεχνικές υποστήριξης της εμπιστευτικότητας των διακινούμενων πληροφοριών, είχε σημαντικές αλλά περιορισμένες εφαρμογές στη επιστήμη της Πληροφορικής. Όμως από το 1976, με τη δημοσίευση της θεμελιώδους εργασίας των Diffie και Hellman άρχισε μια νέα εποχή για την κρυπτογραφία, η εποχή της κρυπτογράφησης του δημόσιου κλειδιού. Έτσι, σήμερα η κρυπτογραφία μπορεί να ανταποκριθεί στις ανάγκες των ευρύτερα ανοικτών δικτύων με πλήθος εφαρμογών στη βιομηχανία, στις τράπεζες, στα νοσοκομεία κλπ. Στις επόμενες παραγράφους γίνεται μια σύντομη παρουσίαση των κρυπτογραφικών εννοιών και μηχανισμών που εφαρμόζονται για τη διασφάλιση των επικοινωνιών.

Αντικείμενο - σημασία της κρυπτογραφίας

Η κρυπτογραφία είναι γνωστή από αρχαιοτάτων χρόνων και δείγματα χρήσης της μπορεί κανείς να συναντήσει σε πολλά αρχαία κείμενα. Χαρακτηριστικό παράδειγμα το μήνυμα του Ιουλίου Καίσαρα προς τον Κικέρωνα, στο οποίο χρησιμοποιήθηκε μία μέθοδος σύμφωνα με την οποία κάθε γράμμα του αρχικού μηνύματος αντικαθίσταται από το τρίτο κατά σειρά επόμενο γράμμα του λατινικού αλφαβήτου (Spillman, 2005). Έτσι, το όνομα του JULIUS CAESAR κωδικοποιείται στο MXOLXV FDHVOU. Η μέθοδος αυτή εκφράζεται στις μέρες μας από τη σχέση:

$$c = m + k \ (mod\ 26)$$

όπου κάθε γράμμα του αρχικού κειμένου που κατέχει τη θέση m, αντικαθίσταται με ένα άλλο γράμμα που κατέχει τη θέση c (θέτοντας k=3 είναι ο αλγόριθμος του Καίσαρα).

Σε ένα λοιπόν σύστημα κρυπτογράφησης, το αρχικό μήνυμα (plaintext) μετασχηματίζεται σε κρυπτογράφημα (ciphertext), δηλαδή σε ένα μήνυμα το οποίο είναι μη-αναγνώσιμο από τρίτους (εκτός από τον επιδιωκόμενο παραλήπτη). Οι μετατροπές κρυπτογράφησης - αποκρυπτογράφησης (encryption-decryption) γίνονται με τη βοήθεια ενός αλγορίθμου κρυπτογράφησης (cipher) και ενός αρκετά μεγάλου αριθμού, του κλειδιού κρυπτογράφησης (key).

Αρχικά Στάδια της Κρυπτογραφίας

Κατ' ουσία, η κρυπτογραφία είναι ένα αντικείμενο της μαθηματικής επιστήμης και ειδικότερα της περιοχής της θεωρίας αριθμών. Ένας από τους επιδιωκόμενους στόχους κατά την κρυπτογράφηση είναι να κρατηθεί σχετικά απλή η διαδικασία μετατροπής του αρχικού κειμένου σε κρυπτογράφημα. Από την άλλη μεριά, η

αντίστροφη διαδικασία μετατροπής του κρυπτογραφήματος στο αρχικό κείμενο, θα πρέπει να καθίσταται ουσιαστικά αδύνατη. Μια τέτοια διαδικασία, είναι γνωστή στα Μαθηματικά ως ‘μονόδρομη συνάρτηση’ (one-way function) και στηρίζεται στη δυσκολία επίλυσης συγκεκριμένων, επακριβώς διατυπωμένων μαθηματικών προβλημάτων. Οι ειδικοί στην κρυπτογραφία έχουν καταλήξει στο ότι προβλήματα από το πεδίο της θεωρίας ομάδων (group theory) συχνά εξυπηρετούν με τον καλύτερο τρόπο τις ανάγκες τους.

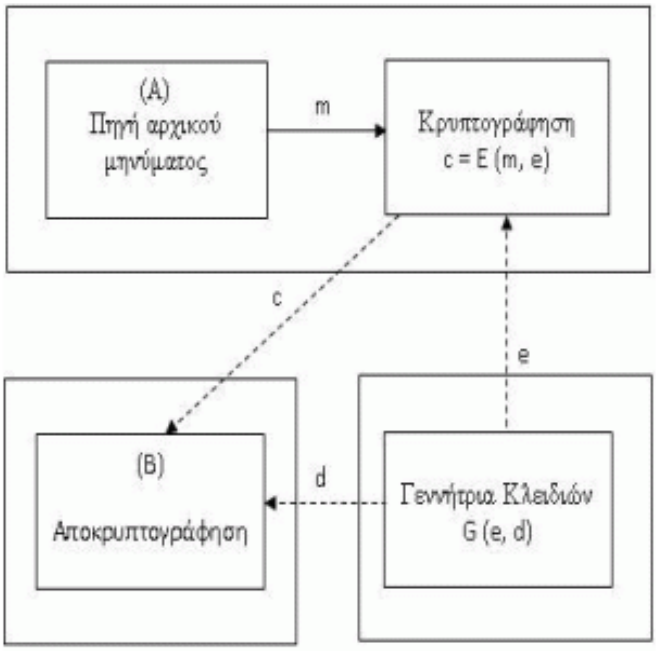
Η Κρυπτογραφία Σήμερα

Στην εποχή της τηλεπληροφορικής που τεράστιες ποσότητες ευαίσθητων πληροφοριών ανταλλάσσονται μεταξύ διαφόρων οργανισμών μέσω μη ασφαλών δημόσιων δικτύων, ο ρόλος της κρυπτογραφίας έχει διευρυνθεί για να ανταποκριθεί στη νέα πραγματικότητα. Ένας ολόκληρος επιστημονικός κλάδος, αυτός της κρυπτολογίας, έχει δημιουργηθεί με αντικείμενο τέτοιου είδους ζητήματα. Η κρυπτολογία περιλαμβάνει δύο επιμέρους κλάδους: την κρυπτογραφία και την κρυπτανάλυση. Στην κρυπτογραφία μελετώνται μέθοδοι εξασφάλισης της μυστικότητας και γνησιότητας των μηνυμάτων που ανταλλάσσονται μεταξύ δύο ή περισσοτέρων οντοτήτων. Αντίθετα στην κρυπτανάλυση μελετώνται μέθοδοι για την αποκάλυψη ή πλαστογράφηση των μηνυμάτων.

Το σχήμα 2.1 αποτυπώνει τη γενική μορφή ενός σύγχρονου κρυπτογραφικού συστήματος. Υποθέτουμε ότι ένας χρήστης Α και ένας χρήστης Β θέλουν να έχουν μια ασφαλή επικοινωνία. Αρχικά, πρέπει να διαλέξουν ή να ανταλλάξουν ένα ζεύγος κλειδιών (e,d). Στη συνέχεια, όταν ο Α (αποστολέας) θελήσει να στείλει μυστικά δεδομένα m στον Β (παραλήπτη), εφαρμόζεται μια μαθηματική συνάρτηση Ε, η οποία χρησιμοποιεί ως παράμετρο το κλειδί e, με σκοπό τον υπολογισμό του κρυπτογραφήματος c = E (m, e). Το κρυπτογράφημα c αποστέλλεται στον Β. Μόλις αυτός το λάβει, εφαρμόζεται μια αντίστροφη μαθηματική συνάρτηση D, η οποία χρησιμοποιεί το άλλο κλειδί d, με σκοπό τον υπολογισμό του m = D (c, d). Οπότε ανακτώνται τα αρχικά δεδομένα m.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 2.1.gif	Κινούμενη εικόνα (interactive)
Εικόνα 2.1 Γενική μορφή κρυπτογραφικού συστήματος	



Εικόνα 2.1 *Γενική μορφή κρυπτογραφικού συστήματος*

Ένα θεμελιώδες ερώτημα που προκύπτει είναι γιατί είναι απαραίτητο να εμπλέκονται στην όλη διαδικασία τα κλειδιά. Έχοντας εισάγει στις συναρτήσεις τα κλειδιά ως παραμέτρους, επιτυγχάνονται τα εξής: Αν κάποια στιγμή αποκαλυφθούν οι αλγόριθμοι, δεν είναι ανάγκη αυτοί να αντικατασταθούν, αλλάζοντας εκ

βάθρων το κρυπτογραφικό σύστημα. Πρέπει απλά να αλλάξουν τα κλειδιά. Ακόμη, δεν απαιτούνται διαφορετικοί αλγόριθμοι για την παραγωγή διαφορετικών κρυπτογραφημάτων από το ίδιο αρχικό κείμενο. Αρκεί η χρήση διαφορετικών κλειδιών. Πραγματικά, τα κλειδιά έχουν το πιο κρίσιμο ρόλο στη λειτουργικότητα των κρυπτογραφικών αλγορίθμων, για αυτόν τον λόγο και αποτελεί ορθή και σώφρονα πρακτική κρυπτογραφίας η συχνή αλλαγή τους.

Η σύγχρονη κρυπτογραφία δε στηρίζεται στη μυστικότητα των αλγορίθμων της. Το αντίθετο μάλιστα, επιδιώκει τη δημοσιοποίηση τους βασιζόμενη σε ανοικτές προς όλους διαδικασίες αποτίμησης της ανθεκτικότητάς τους. Η χρήση κοινόχρηστων - δημόσιων αλγορίθμων (public algorithms), αποτελεί μια απαίτηση των πληροφοριακών συστημάτων της σημερινής πραγματικότητας, τα οποία καλούνται να υποστηρίξουν μια μεγάλη κοινότητα χρηστών με ενδιαφέροντα ανταγωνιστικά μεταξύ τους τις περισσότερες φορές (Gollmann, 2011). Οι δημόσιοι αλγόριθμοι εξελίσσονται με ανοικτές διαδικασίες αξιολόγησης (open evaluation) και συχνά βάσει εκ των πραγμάτων τυποποιήσεων προτύπων (de-facto standardization). Τα χαρακτηριστικά αυτά εξυπηρετούν μια βασική ανάγκη των σύγχρονων δικτυακών περιβαλλόντων: δίνουν τη δυνατότητα στο κάθε μέρος να κάνει με τα δικά του κριτήρια τις δικές του εκτιμήσεις περί της ανθεκτικότητας των αλγορίθμων κρυπτογράφησης και άρα του παρεχόμενου επιπέδου ασφάλειας. Έτσι διευκολύνονται όσα νέα μέρη θέλουν να συμμετάσχουν σε υπάρχουσες δικτυακές υποδομές, εφόσον μειώνονται οι δισταγμοί τους ως προς την υποστηριζόμενη ασφάλεια επικοινωνιών. Συνοψίζοντας, το κλειδί που χρησιμοποιείται σε μια κρυπτογραφική μετατροπή πρέπει ουσιαστικά να αποτελεί το μοναδικό αντικείμενο προστασίας, ενώ ο αλγόριθμος πρέπει να θεωρείται γνωστός. Αυτή η προσέγγιση είναι γνωστή στην κρυπτογραφία ως «Αρχή του Kerckhoffs».

Βεβαίως μια σειρά από ζητήματα που αφορούν τα κλειδιά τίθενται: Που και πώς παράγονται; Που αποθηκεύονται; Πώς αποστέλλονται; Πώς και πότε ανακαλούνται ή αντικαθιστώνται; Όλα αυτά αποτελούν αντικείμενο της περιοχής που καλείται Διαχείριση Κλειδιών (Key Management). Στην ουσία, η κρυπτογραφία μπορεί να ειδωθεί ως ένας μηχανισμός μετατροπής των προβλημάτων ασφάλειας επικοινωνιών σε προβλήματα διαχείρισης κλειδιών, δηλαδή διαχείρισης με συγκεκριμένο τρόπο ορισμένων πολύ ευαίσθητων δεδομένων. Το πλεονέκτημα της κρυπτογραφίας είναι ότι το αρχικό πρόβλημα της εμπιστευτικότητας ενός μεγάλου μεγέθους κειμένου καταλήγει σε ένα πρόβλημα διακίνησης μικρών σε μέγεθος σχετικά κλειδιών, το οποίο και είναι ευκολότερο να αντιμετωπιστεί.

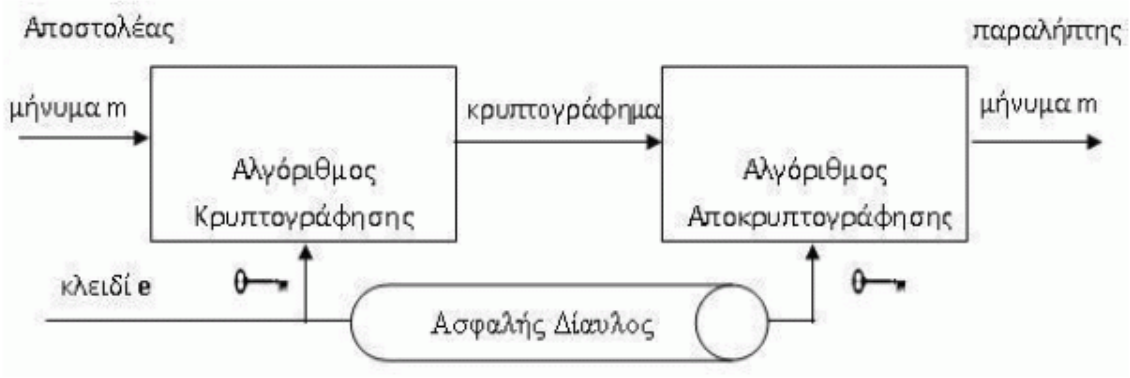
Για τους λόγους που ήδη αναφέρθηκαν, η ασφάλεια ενός κρυπτογραφικού συστήματος βασίζεται αποκλειστικά στην εξασφάλιση της μυστικότητας του κλειδιού. Ανάλογα με τον τρόπο που διασφαλίζεται αυτή η μυστικότητα του κλειδιού, τα κρυπτογραφικά συστήματα διακρίνονται σε δυο κατηγορίες. Στα συμμετρικά ή μυστικού κλειδιού συστήματα (secret key systems) και στα ασύμμετρα ή δημοσίου κλειδιού συστήματα (public key systems).

Κρυπτογραφία μυστικού κλειδιού

Η συμμετρική κρυπτογραφία, η οποία συγκεντρώνει τους παραδοσιακούς αλγόριθμους κρυπτογράφησης, βασίζεται στην ύπαρξη ενός μοναδικού κλειδιού, το οποίο χρησιμοποιείται τόσο για την κρυπτογράφηση όσο και για την αποκρυπτογράφηση των μηνυμάτων. Αναφερόμενοι στο σχήμα 2.1, σε ένα κρυπτογραφικό σύστημα μυστικού κλειδιού ισχύει d = e. Λόγω του ότι το κλειδί αυτό θα πρέπει να είναι γνωστό μόνο στον παραλήπτη και στον αποστολέα των μηνυμάτων, καλείται και κρυπτογραφία διαμοιραζόμενου μυστικού.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 2.2.gif	Κινούμενη εικόνα (interactive)
Εικόνα 2.2 Συμμετρική κρυπτογραφία	



Εικόνα 2.2 *Συμμετρική κρυπτογραφία*

Η συμμετρική κρυπτογραφία έχει ως μοναδικό σκοπό της τη διαφύλαξη της εμπιστευτικότητας των πληροφοριών και είναι κατάλληλη για μετατροπές μεγάλου όγκου δεδομένων επειδή οι υπολογισμοί που απαιτεί εκτελούνται πολύ γρήγορα. Το κύριο χαρακτηριστικό της είναι η ύπαρξη ενός ασφαλούς διαύλου επικοινωνίας για την αποστολή του μυστικού κλειδιού στον παραλήπτη (Σχήμα 2.2). Το Kerberos, είναι ένα πρωτόκολλο αυθεντικοποίησης μέσω δικτύου (network authentication protocol) το οποίο αναπτύχθηκε στο MIT και αποτελεί το πιο διαδεδομένο σύστημα αυθεντικοποίησης για εφαρμογές πελάτη-διακομιστή, που χρησιμοποιεί συμμετρική κρυπτογραφία, υποστηρίζοντας ασφαλή μεταφορά των μυστικών κλειδιών μέσω δημοσίων δικτύων.

Από τα πιο γνωστά συστήματα μυστικού κλειδιού είναι το DES (Data Encryption Standard). Ο αλγόριθμος DES αναπτύχθηκε από την IBM στις αρχές της δεκαετίας του 1970 και από το 1977, το NIST (National Institute for Standards and Technology) των ΗΠΑ το υιοθέτησε ως το επίσημο πρότυπο κρυπτογράφησης ευαίσθητων πληροφοριών. Χρησιμοποιεί ένα κλειδί μεγέθους 56-bit. Το προς κρυπτογράφηση κείμενο εισάγεται σε τμήματα μεγέθους 64-bit και παράγεται ένα 64-bit κρυπτογράφημα. Παρά την πολυπλοκότητα της αρχιτεκτονικής του DES, ο αλγόριθμος αυτός αποτελεί ουσιαστικά έναν κώδικα μόνο-αλφαβητικής αντικατάστασης. Ο DES θεωρείται πλέον ανασφαλής για πολλές εφαρμογές, και αυτό οφείλεται κυρίως στο μικρό μέγεθος του κλειδιού του (Mohapatra, 2000).

Εξέλιξη του DES αποτελεί ο Triple-DES, ο οποίος εκτελεί τρεις φορές την κρυπτογράφηση ή την αποκρυπτογράφηση του DES με αποτέλεσμα να κάνει καλύτερη κρυπτογράφηση από τον κλασικό DES. Με αυτόν τον τρόπο διασκεδάζονται οι ανησυχίες οι σχετικές με την ανθεκτικότητα του αλγορίθμου, λόγω του μικρού μεγέθους του κλειδιού. Το αρχικό κείμενο κρυπτογραφείται με τη χρήση ενός κλειδιού, στη συνέχεια αποκρυπτογραφείται με τη χρήση ενός δεύτερου και μετά κρυπτογραφείται με ένα τρίτο στον αποστολέα. Το αποτέλεσμα στέλνεται στον παραλήπτη που ακολουθεί την αντίστροφη διαδικασία. Πρώτα, αποκρυπτογραφεί με χρήση του τρίτου κλειδιού, στη συνέχεια κρυπτογραφεί με το δεύτερο και τέλος αποκρυπτογραφεί με το τρίτο. Ο Triple-DES αλγόριθμος μπορεί να υλοποιηθεί χρησιμοποιώντας δύο ή τρία κλειδιά μεγέθους 56-bit. Υπό τη μορφή αυτή του τριπλού DES, ο αλγόριθμος θεωρείται πρακτικά ασφαλής.

Από το 1997 είχαν ξεκινήσει από το NIST οι διαδικασίες προσδιορισμού του απογόνου του συστήματος DES, το οποίο καλείται AES (Advanced Encryption Standard), οι οποίες ολοκληρώθηκαν το 2001. Υιοθετήθηκε ο αλγόριθμος Rijnadel με δυνατότητες επιλογής κλειδιού μεγέθους (key size) 128, 192, ή 256 bits, ενώ το προς κρυπτογράφηση τμήμα κειμένου (block size) εισάγεται σε τμήματα μεγέθους 128 bits. Ένας άλλος γνωστός αλγόριθμος συμμετρικής κρυπτογραφίας είναι ο Twofish, με ίδια χαρακτηριστικά μεγέθους κλειδιού και τμήματος κειμένου. Υστερεί ελαφρά σε ταχύτητα από τον AES, όταν το κλειδί είναι 128 bits, αλλά υπερτερεί από αυτόν όταν το κλειδί είναι 256 bits. Είναι αλγόριθμος που διατίθεται ελεύθερος προς χρήση σε όλους, και συμπεριλαμβάνεται στο πρότυπο OpenPGP (Ferguson et al., 2011). Αν και είναι εξέλιξη του διαδεδομένου Blowfish αλγορίθμου, δεν έχει ως τώρα καταφέρει να τον ξεπεράσει σε δημοφιλία.

Περιορισμοί Συμμετρικής Κρυπτογραφίας

Η συμμετρική κρυπτογραφία προσφέρει πολύ γρήγορους σε εκτέλεση αλγορίθμους. Έτσι είναι σε θέση να εγγυηθεί την εμπιστευτικότητα των επικοινωνιών, χωρίς να επιβαρύνει τη διαθεσιμότητα των συστημάτων. Όμως τα πλεονεκτήματά της σταματούν σε αυτό το σημείο. Δύο είναι τα βασικά της μειονεκτήματα:

1. Το βασικό πρόβλημα στην παραδοσιακή κρυπτογραφία είναι αυτό της διανομής και της διαχείρισης γενικότερα των απαιτούμενων κλειδιών (key distribution-management). Σε μια επικοινωνία δυο μερών τα συναλλασσόμενα μέρη πρέπει, πριν αρχίσουν τις διαδικασίες αποστολής και λήψης μηνυμάτων, να χρησιμοποιήσουν ένα ασφαλές κανάλι για τον προσδιορισμό του κλειδιού που θα χρησιμοποιήσουν. Στα μεγάλα δίκτυα, ο αριθμός των διακινούμενων κλειδιών αυξάνεται γεωμετρικά λόγω του πλήθους των χρηστών (σε δίκτυο t πλήθους χρηστών, t(t-1)/2 ζεύγη χρηστών σχηματίζονται), αλλά και επειδή τα κλειδιά πρέπει να αλλάζουν συχνά προκειμένου να διατηρηθεί ένα υψηλό επίπεδο ασφάλειας. Πολλές φορές η διάρκεια ισχύος των κλειδιών περιορίζεται στο διάστημα μιας συνεδρίας επικοινωνίας (communication session). Τα συστήματα ασφαλούς ανταλλαγής κλειδιών, όπως το προαναφερθέν Kerberos, δεν είναι εύκολο να επεκταθούν για την εξυπηρέτηση μεγάλης κλίμακας χρηστών και απαιτούν επίσης πρόσθετες διαδικασίες ασφάλειας, όπως είναι η αποθήκευση των κλειδιών σε ένα κεντρικό ασφαλή διανομέα-εξυπηρετητή.
2. Εκτός από την εμπιστευτικότητα των μηνυμάτων, υπάρχουν και άλλες απαιτήσεις ασφάλειας στα ανοικτά και μεγάλης κλίμακας δίκτυα, όπως το Internet, για τις οποίες η συμμετρική κρυπτογραφία δεν προσφέρει λύσεις. Στο προηγούμενο κεφάλαιο παρουσιάστηκαν αναλυτικά οι έννοιες της ακεραιότητας (αποτροπή μη-εξουσιοδοτημένων μεταβολών), της αυθεντικότητας (έλεγχος γνησιότητας της ταυτότητας ενός χρήστη) και της μη-απάρνησης (αποτροπή των αρνήσεων ενός μέρους να αναλάβει την ευθύνη των ενεργειών που διέπραξε). Στα ζητήματα αυτά, η σχετικά πρόσφατη κρυπτογραφία δημόσιου κλειδιού προσφέρει ικανοποιητικές διεξόδους.

Κρυπτογραφία Δημοσίου Κλειδιού

Τα συστήματα δημοσίου κλειδιού ή ασύμμετρης κρυπτογραφίας χρησιμοποιούν δύο ξεχωριστά αλλά συμπληρωματικά κλειδιά για τις διαδικασίες κρυπτογράφησης και αποκρυπτογράφησης. Το ένα από αυτά διατηρείται απόρρητο και καλείται ιδιωτικό (private) κλειδί, ενώ το άλλο γίνεται γνωστό στον κάθε ενδιαφερόμενο και καλείται δημόσιο (public). Παρόλο που τα δύο κλειδιά έχουν μια σύνθετη μαθηματική σχέση μεταξύ τους, η γνώση του δημόσιου κλειδιού δεν καθιστά εφικτό τον υπολογισμό του μυστικού ιδιωτικού κλειδιού. Η προστασία της εμπιστευτικότητας των μηνυμάτων επιτυγχάνεται ως εξής: Το αρχικό μήνυμα κρυπτογραφείται από τον αποστολέα με το δημόσιο κλειδί του παραλήπτη και μόνο ο κάτοχος του ιδιωτικού κλειδιού, δηλαδή ο ίδιος ο παραλήπτης, μπορεί να το αποκρυπτογραφήσει.

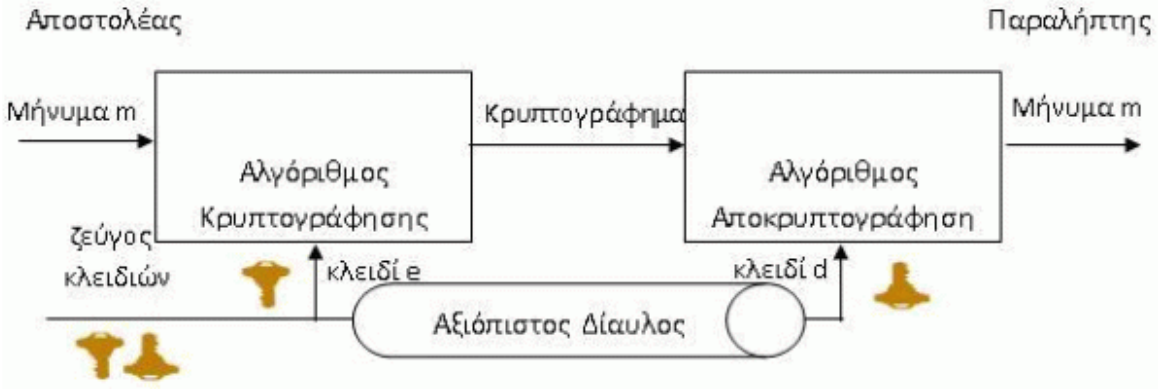
Θεωρητικό υπόβαθρο

Η βασική θεωρητική ιδέα πίσω από την ασύμμετρη κρυπτογραφία είναι η έννοια των "μονόδρομων συναρτήσεων με καταπακτή" (trapdoor one-way functions). Οι συναρτήσεις αυτές είναι ειδικές περιπτώσεις των μονόδρομων συναρτήσεων που ήδη αναφέρθηκαν. Η επιπλέον ιδιότητα που τις κάνει διακριτές, είναι ότι όταν χρησιμοποιηθεί μια επιπλέον πληροφορία (η αποκαλούμενη καταπακτή), γίνεται εφικτός ο υπολογισμός της αντιστρόφου της. Τα ακόλουθα δυο προβλήματα θεωρούνται οι πιο πιθανοί υποψήφιοι για τη δημιουργία τέτοιων συναρτήσεων:

- Πρόβλημα παραγοντοποίησης (factorization problem): Η εύρεση δυο μεγάλων πρώτων αριθμών p και q οι οποίοι δίνουν ως γινόμενο έναν αριθμό n. Μια προσεκτική επιλογή των πρώτων αριθμών καθιστά υπολογιστικά αδύνατη τη λύση του προβλήματος. Όμως, εάν είναι γνωστή η συνάρτηση phi(n) του Euler (δηλαδή το πλήθος των ακεραίων που είναι μικρότεροι από το n και οι οποίοι είναι «πρώτοι μεταξύ τους» με το n), τότε τα p και q μπορούν εύκολα να προσδιορισθούν. Η συνάρτηση phi(n) είναι η επιπρόσθετη πληροφορία, η καταπακτή.
- Πρόβλημα διακριτών λογαρίθμων (discrete logarithm problem): Η εύρεση του μοναδικού αριθμού i για τον οποίο ισχύει a = g^i (mod p), όπου p πολύ μεγάλος πρώτος αριθμός, 0<= i <= p-1 και g ειδικά επιλεγμένος αριθμός.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 2.3.gif	Κινούμενη εικόνα (interactive)
Εικόνα 2.3 Ασύμμετρη κρυπτογραφία	



Εικόνα 2.3 *Ασύμμετρη κρυπτογραφία*

Υλοποιήσεις δημοσίου κλειδιού

Η λογική της κρυπτογραφίας δημοσίου κλειδιού επινοήθηκε και διατυπώθηκε το 1976, σχεδόν ταυτόχρονα από τους Diffie και Hellman και από τον Merkle. Ένα χρόνο μετά, οι Rivest, Shamir και Addleman βασιζόμενοι σε αρχές της θεωρίας των πεπερασμένων πεδίων δημιούργησαν το κρυπτοσύστημα RSA (το όνομα προέκυψε από τα αρχικά γράμματα των ονομάτων τους), το οποίο αποτελεί την πρώτη υλοποίηση κρυπτογράφησης δημοσίου κλειδιού (Spillman, 2005). Η πρακτικότητα του συστήματος RSA βασίζεται στην αριθμητική υπολοίπων ακέραιας διαίρεσης (modulo) και η ανθεκτικότητά του οφείλεται στο προαναφερθέν πρόβλημα παραγοντοποίησης. Συγκεκριμένα, ενώ θεωρείται σχετικά απλό να βρεθούν δυο μεγάλοι πρώτοι αριθμοί (πχ. με 150 δεκαδικά ψηφία) και να υπολογιστεί στη συνέχεια το γινόμενό τους, η παραγοντοποίηση αυτού του γινομένου (ειδικά όταν οι μεγάλοι πρώτοι αριθμοί είναι και ισχυροί ως προς κάποιες μαθηματικές ιδιότητες), δεν είναι εφικτή με τις δυνατότητες των υφιστάμενων αλγορίθμων και υπολογιστικών συστημάτων.

Άλλα γνωστά σχήματα δημόσιου κλειδιού είναι οι αλγόριθμοι των ελλειπτικών καμπυλών (Elliptic Curve Digital Signature Algorithms, ECDSA) και τα κρυπτογραφικά συστήματα El Gamal, των οποίων η ισχύς βασίζεται στα υψηλής δυσκολίας προβλήματα διακριτού λογαρίθμου.

Λειτουργικότητα ασύμμετρης κρυπτογραφίας

Κάθε αλγόριθμος δημοσίου κλειδιού έχει τις δικές του ιδιαιτερότητες, όλοι όμως χρησιμοποιούν ζεύγος κλειδιών και βασίζονται στο ότι όποιο από τα κλειδιά δημοσιευθεί, δε διακυβεύει τη μυστικότητα του άλλου κλειδιού. Οι χρήστες λοιπόν του Διαδικτύου μπορούν ελεύθερα να συμπεριλαμβάνουν στις ιστοσελίδες τους ή σε ειδικούς καταλόγους-ευρετήρια (directories) τα δημόσια κλειδιά τους, οπότε και παύει να υφίσταται το βασικό πρόβλημα διαχείρισης των κλειδιών της κρυπτογραφίας μυστικού κλειδιού. Τα δημόσια κλειδιά δε χρειάζονται για τη διανομή τους έναν ασφαλή δίαυλο. Δεν τίθεται ζήτημα εμπιστευτικότητας στα κανάλια διανομής των δημοσίων κλειδιών, αφού αυτά είναι προσπελάσιμα και ανοικτά προς όλους τους ενδιαφερόμενους. Όμως, χρειάζονται για τη διανομή τους έναν αξιόπιστο δίαυλο (σχήμα 2.3), δηλαδή ένα μέσο που θα υποστηρίζει την ακεραιότητά τους. Οι μικρότερες απαιτήσεις ασφάλειας για τη διανομή των κλειδιών της, κάνουν την κρυπτογραφία δημοσίου κλειδιού ιδανική για ένα εκ φύσεως δημόσιο δίκτυο, το Internet, στο οποίο πολλές φορές χρειάζεται να αποκαθίσταται η εμπιστοσύνη ανάμεσα σε δυο απομακρυσμένους χρήστες χωρίς αυτοί να συναντηθούν ή χωρίς να μεσολαβήσει κάποιο έμπιστο τρίτο μέρος.

Οι αρχές της κρυπτογράφησης δημοσίου κλειδιού αποσαφηνίζονται από τις ακόλουθες αναλογίες: Η πρώτη, παρομοιάζει την ασύμμετρη κρυπτογραφία ως ένα προσωπικό γραμματοκιβώτιο, στο οποίο όλοι βέβαια μπορούν να εναποθέσουν γράμματα αλλά μόνο ο ιδιοκτήτης του γραμματοκιβώτιου μπορεί να τα παραλάβει, εφόσον μόνον αυτός κατέχει το κλειδί του. Το σημαντικό στην περίπτωση αυτή είναι ότι ο αποστολέας πρέπει να είναι σίγουρος για το όνομα του ιδιοκτήτη του γραμματοκιβώτιου. Η δεύτερη αναλογία στηρίζεται στα λεξικά και δίνει έμφαση στην ισοδυναμία δημόσιων και ιδιωτικών κλειδιών, καθώς επίσης και στη δυσκολία υπολογισμού του ιδιωτικού κλειδιού από το δημόσιο. Θεωρούμε το αρχικό κείμενο ως ένα κείμενο γραμμένο στα Ελληνικά και το κρυπτογράφημα ως τη μετάφρασή του στην Ιταλική γλώσσα. Η κρυπτογράφηση απαιτεί ένα δημόσιο κλειδί, δηλαδή ένα λεξικό Ελληνό-Ιταλικό. Η αποκρυπτογράφηση χρειάζεται το ιδιωτικό κλειδί, δηλαδή αναλογικά ένα Ιταλό-Ελληνικό λεξικό. Έχοντας κάποιος (μη-γνωρίζων

Ιταλικά) το Ελληνό-Ιταλικό λεξικό, είναι πάρα πολύ δύσκολο να μεταφράσει ένα κείμενο της Ιταλικής. Και η αναλογία προχωράει ακόμη περισσότερο στο ότι ενώ θεωρητικά υπάρχει τρόπος εύρεσης του ιδιωτικού κλειδιού από το δημόσιο, η απαιτούμενη προσπάθεια κάνει το εγχείρημα πάρα πολύ δύσκολο και ασύμφορο.

Εφαρμογές Κρυπτογραφίας Δημοσίου Κλειδιού
Η ασύμμετρη κρυπτογραφία αποτελεί τεχνολογικά τον ακρογωνιαίο λίθο σε πολλές εφαρμογές και μηχανισμούς ασφάλειας στο Διαδίκτυο, όπως:

- Υποδομές Πιστοποίησης, οι οποίες διαχειρίζονται ψηφιακά πιστοποιητικά έμπιστων τρίτων φορέων, με σκοπό την αναγνώριση και πιστοποίηση της ταυτότητας των χρηστών (πιστοποιητικά ταυτότητας), αλλά και τον έλεγχο των εξουσιοδοτήσεών τους (πιστοποιητικά χαρακτηριστικών).
- Ασφαλής παρουσίαση ιστοσελίδων αλλά και δικτυακών αγορών, βάσει του πρωτοκόλλου SSL (Secure Sockets Layer) αλλά και του πρωτοκόλλου TLS (Transport Layer Security) της IETF (Internet Engineering Task Force).
- Ασφαλείς συναλλαγές μέσω πιστωτικών καρτών, βάσει του πρωτοκόλλου 3-D Secure των Visa και Mastercard.
- Ασφαλής ηλεκτρονική αλληλογραφία, βάσει του πρωτοκόλλου S/MIME (Secure / Multipurpose Internet Mail Extensions) της IETF.

Όλα τα προηγούμενα στηρίζονται στους ακόλουθους τρόπους εκμετάλλευσης της κρυπτογραφίας δημοσίου κλειδιού.

Διανομή Μυστικού Κλειδιού - Ψηφιακοί Φάκελοι
Η ανταλλαγή κλειδιών (key exchange) ήταν η πρώτη χρονολογικά εφαρμογή της κρυπτογραφίας δημοσίου κλειδιού. Για αυτόν τον λόγο άλλωστε αναπτύχθηκε, για την αντιμετώπιση του μεγαλύτερου προβλήματος της κρυπτογραφίας μυστικού κλειδιού, δηλαδή του τρόπου που θα γίνει η μετάδοση του κοινού μυστικού κλειδιού με το οποίο θα γίνει η επικοινωνία. Χρησιμοποιείται η ακόλουθη διαδικασία:

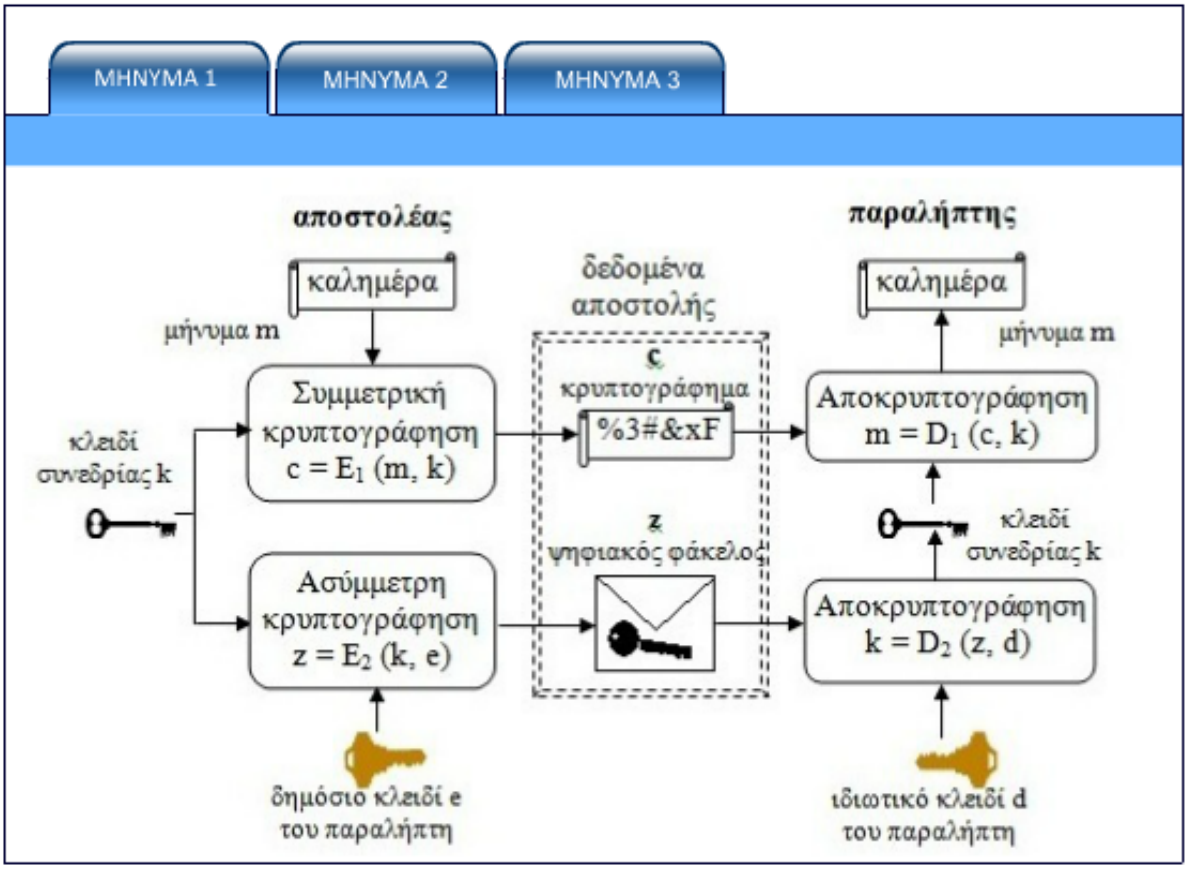
1. Ο αποστολέας καθορίζει το μυστικό κλειδί k που θα χρησιμοποιηθεί κατά την κρυπτογράφηση μυστικού κλειδιού. Για λόγους επιπρόσθετης ασφάλειας το κλειδί k είναι ένας τυχαία παραγόμενος αριθμός, ο οποίος χρησιμοποιείται μόνο για μια φορά και μόλις η επικοινωνία τελειώσει καταστρέφεται ολοσχερώς. Το κλειδί k σε αυτήν την περίπτωση καλείται κλειδί μηνύματος (message key) ή κλειδί συνεδρίας (session key).
2. Ο αποστολέας κρυπτογραφεί το κλειδί συνεδρίας k, χρησιμοποιώντας το δημόσιο κλειδί e του παραλήπτη και του το αποστέλλει ως πληροφορία z.
3. Ο παραλήπτης μόλις παραλάβει το z, το αποκρυπτογραφεί μέσω του ιδιωτικού του κλειδιού d και έτσι ανακτά το κλειδί k.

Με αυτόν τον τρόπο διαφυλάσσεται η εμπιστευτικότητα του κλειδιού μιας συμμετρικής κρυπτογράφησης. Το ερώτημα που αβίαστα προκύπτει είναι γιατί να εμπλέκεται στην όλη διαδικασία μια κρυπτογράφηση μυστικού κλειδιού. Δηλαδή, γιατί να είναι αναγκαίο το πρώτο βήμα της προηγούμενης διαδικασίας και να μην αρκεί στα υπόλοιπα βήματα να κρυπτογραφείται - αποκρυπτογραφείται όχι κάποιο κλειδί αλλά ολόκληρο το υπό προστασία μήνυμα. Ο λόγος βρίσκεται στα εξής μειονεκτήματα της κρυπτογραφίας δημοσίου κλειδιού:

- Απαιτούνται μεγαλύτερου μεγέθους κλειδιά: Σήμερα κλειδιά μήκους από 2048 bits και πάνω, θεωρούνται ασφαλή. Χρειάζονται αιώνες για να σπάσουν με τη σύγχρονη τεχνολογία. Ένα όμως συμμετρικό σύστημα όπως το AES ή το Twofish χρειάζεται κλειδιά πολύ μικρότερου μεγέθους (256 bits) για να παρέχει το ίδιο επίπεδο ασφάλειας.
- Παρουσιάζει πολύ μεγαλύτερη υπολογιστική πολυπλοκότητα από την αντίστοιχη του μυστικού κλειδιού, για αυτό και είναι πιο αργή. Τα συστήματα μυστικού κλειδιού είναι 100-200 φορές γρηγορότερα από κάθε σύστημα δημόσιου κλειδιού. Ακόμη και μέσω βελτιστοποιήσεων, οι ασύμμετροι αλγόριθμοι είναι δυο με τρεις τάξεις μεγέθους πιο αργοί από τους συμμετρικούς.

Η χαμηλή λοιπόν γενικά απόδοση τους οδηγεί σε υβριδικά κρυπτογραφικά συστήματα. Το σχήμα κρυπτογράφησης δημόσιου κλειδιού εξυπηρετεί τη διανομή μυστικών κλειδιών, τα οποία στη συνέχεια χρησιμοποιούνται από γρήγορους συμμετρικούς αλγόριθμους. Έτσι φτάνουμε στην έννοια του ψηφιακού φακέλου (digital envelope), ο οποίος περιέχει ένα μυστικό κλειδί k συμμετρικής κρυπτογραφίας, κρυπτογραφημένο ασύμμετρα βάσει του δημοσίου κλειδιού e του παραλήπτη. Ο ψηφιακός φάκελος συνοδεύει το μήνυμα m που κρυπτογραφήθηκε συμμετρικά βάσει του συγκεκριμένου κλειδιού k. Το ακόλουθο σχήμα 2.4, παρουσιάζει την όλη διαδικασία.

Flash 2.1.swf	Αρχείο flash (interactive)
Εικόνα 2.4 Ψηφιακός φάκελος	



Εικόνα 2.4 *Ψηφιακός φάκελος*

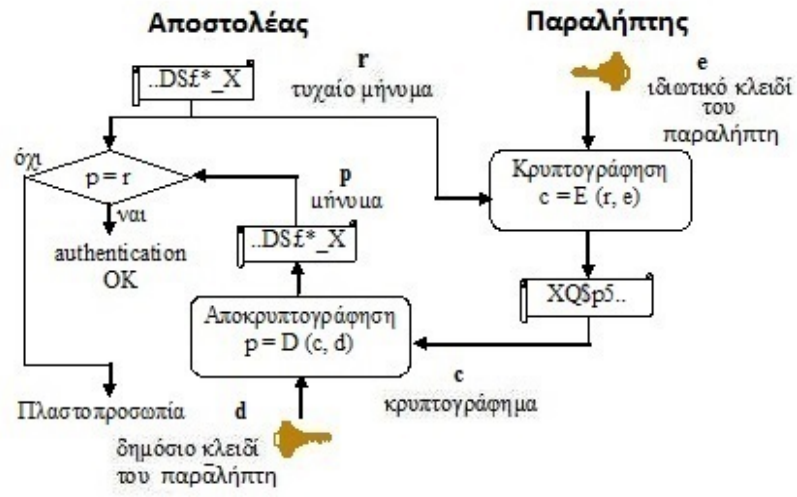
Τέτοιας μορφής διευθέτηση προτείνεται από το πρωτόκολλο SSL. Ένα άλλο παράδειγμα συνδυασμού συμμετρικών και ασύμμετρων αλγορίθμων αποτελεί το πρότυπο OpenPGP.

Έλεγχος Αυθεντικότητας

Η κρυπτογραφία δημοσίου κλειδιού μπορεί να χρησιμοποιηθεί για την επιβεβαίωση της ταυτότητας μιας οντότητας (authentication), αρκεί να χρησιμοποιηθούν τα κλειδιά με αντίστροφο τρόπο. Δηλαδή το ιδιωτικό κλειδί να κάνει σε κάποια φάση της επικοινωνίας κρυπτογράφηση και το δημόσιο να αναλάβει την αποκρυπτογράφηση. Ως παράδειγμα, το πρωτόκολλο SSL χρησιμοποιεί τεχνικές δημοσίου κλειδιού για τον έλεγχο αυθεντικότητας, με τον ακόλουθο τρόπο. Υποθέτουμε ότι το συναλλασσόμενο μέρος Α θέλει να επιβεβαιώσει την ταυτότητα του μέρους Β. Τότε, σύμφωνα και με το σχήμα 2.5:

Κάντε κλικ στα εικονίδια του σχήματος για επεξήγηση

Dynamic 2.1.zip	Διαδραστική εικόνα (interactive)
Εικόνα 2.5 Πιστοποίηση ταυτότητας στην ασύμμετρη κρυπτογραφία	



Εικόνα 2.5 *Πιστοποίηση ταυτότητας στην ασύμμετρη κρυπτογραφία*

1. Το μέρος Α (αποστολέας) στέλνει ένα τυχαίο μήνυμα r, το οποίο καλείται πρόκληση (challenge) στο μέρος Β (παραλήπτης).

2. Το Β κρυπτογραφεί το r, βάσει του ιδιωτικού του κλειδιού e και το κρυπτογράφημα c αποστέλλεται πίσω στο Α.

3. Το Α είναι σε θέση να αποκρυπτογραφήσει το c, βάσει του δημόσιου κλειδιού d του μέρους Β. Συγκρίνοντας το προϊόν της αποκρυπτογράφησης p με το αρχικό μήνυμα r, αποφαίνεται για την ταυτότητα του Β.

Επειδή το ιδιωτικό κλειδί της υπό εξακρίβωση οντότητας (άτομο, μηχάνημα, διεργασία κλπ.) το γνωρίζει μόνον η ίδια, μόνον αυτή θα μπορούσε να κρυπτογραφήσει ένα μήνυμα που αποκρυπτογραφείται με το δημόσιο κλειδί της. Η διαδικασία αυτή δε διασφαλίζει την εμπιστευτικότητα των δεδομένων, αφού οποιοσδήποτε θα μπορούσε να αποκρυπτογραφήσει το μήνυμα μέσω του κοινώς γνωστού δημόσιου κλειδιού. Για αυτό και το μήνυμα που χρησιμοποιείται είναι κάτι τυχαίο και η συγκεκριμένη χρήση των τεχνικών δημοσίου κλειδιού αποτελεί απλά μια φάση η οποία συνδυάζεται με επιπλέον διαδικασίες (όπως αυτές που αναφέρθηκαν σε προηγούμενες παραγράφους ή που θα αναφερθούν στη συνέχεια).

Ψηφιακές Υπογραφές

Για την απόδειξη της γνησιότητας ενός εγγράφου, χρησιμοποιούνται οι συμβατικές υπογραφές. Ειδικότερα, η υπογραφή αποτελεί μαρτυρία της εγκυρότητας του υπογεγραμμένου εγγράφου έτσι ώστε ο υπογράφων να μη μπορεί να το απαρνηθεί. Στο ανοικτό περιβάλλον του Διαδικτύου, καθίσταται αναγκαία η χρησιμοποίηση ενός ηλεκτρονικού ισοδύναμου της συμβατικής υπογραφής, δηλαδή μιας ηλεκτρονικής υπογραφής. Ο μηχανισμός της ηλεκτρονικής υπογραφής θα πρέπει να παρέχει απόδειξη της προέλευσης, της γνησιότητας και της ακεραιότητας των ανταλλασσομένων μηνυμάτων. Επιπλέον, η ηλεκτρονική υπογραφή πρέπει να δημιουργείται και να αναγνωρίζεται εύκολα, ενώ δύσκολα να πλαστογραφείται.

Οι ηλεκτρονικές υπογραφές που βασίζονται στην κρυπτογραφία δημοσίου κλειδιού, ονομάζονται ψηφιακές υπογραφές (digital signatures). Υπάρχουν ψηφιακές υπογραφές που χρησιμοποιούν τον αλγόριθμο κρυπτογράφησης RSA, αλλά ως πρότυπο για τις ψηφιακές υπογραφές από το NIST έχει υιοθετηθεί ο DSA (Digital Signature Algorithm), ο οποίος βασίζεται στο σύστημα ψηφιακής υπογραφής El Gamal. Βασικό του χαρακτηριστικό είναι ότι το μήκος της υπογραφής ενός μηνύματος είναι διπλάσιο του μήκους του μηνύματος, σε αντίθεση με τον αλγόριθμο RSA όπου τα μήκη είναι ίσα.

Ακεραιότητα δεδομένων

Η ψηφιακή υπογραφή, ως ένα είδος ηλεκτρονικής υπογραφής, είναι μία συμβολοσειρά από bits και εξαρτάται πάντοτε από το μήνυμα που συνοδεύει. Ο λόγος αυτής της εξάρτησης είναι η αποτροπή απόσπασης και αντιγραφής τους από τα μηνύματα που συνοδεύουν. Διότι, όσο δύσκολη και αν είναι η διαδικασία δημιουργίας μιας ψηφιακής υπογραφής, εάν δε 'δένεται' κατάλληλα με το μήνυμα, είναι εύκολη η προσθήκη της σε οποιοδήποτε άλλο μήνυμα. Η 'σύνδεση' της ψηφιακής υπογραφής με το περιεχόμενο του μηνύματος που υπογράφει, είναι υπεύθυνη για τη δυνατότητα υποστήριξης της ακεραιότητας των δεδομένων (data integrity), δηλαδή της διασφάλισης ότι δεν έχουν παραποιηθεί τα δεδομένα μέχρι να φτάσουν από τον αποστολέα στον παραλήπτη.

Υπάρχουν διάφοροι αλγόριθμοι που ελέγχουν την ακεραιότητα των δεδομένων που αποστέλλονται, οι οποίοι καλούνται αλγόριθμοι σύνοψης μηνύματος (message digest). Οι συναρτήσεις σύνοψης χρησιμοποιούνται για τον υπολογισμό της 'περίληψης' των δεδομένων, ως πρώτο βήμα για τη δημιουργία ψηφιακής υπογραφής. Εφαρμόζουν μια μονόδρομη συνάρτηση κατατεμαχισμού (one way hash function) σε οποιουδήποτε μεγέθους μήνυμα και παράγουν ένα σταθερό αριθμό από bits που αποτελεί τη 'περίληψη' του μηνύματος, η οποία είναι και μοναδική για κάθε μήνυμα. Η ιδέα βασίζεται στο γεγονός ότι πρέπει μία σύνοψη μηνύματος να αναπαριστά "συνοπτικά" τα αρχικά δεδομένα από τα οποία παράχθηκε, καθιστώντας την έτσι ψηφιακό αποτύπωμα των μεγαλύτερου μεγέθους δεδομένων. Προφανώς η ταχύτητα υπολογισμού των ψηφιακών υπογραφών είναι πολύ μεγαλύτερη αν χρησιμοποιούνται συνόψεις, παρά αν κρυπτογραφείται το ίδιο το κείμενο.

Για την επιλογή τους ως συναρτήσεων σύνοψης, αυτές πρέπει να διαθέτουν συγκεκριμένες ιδιότητες. Συνοπτικά οι σπουδαιότερες ιδιότητες για μία συνάρτηση σύνοψης H είναι (Γκρίτζαλης & Γεωργιάδης, 1997):

1. Η συνάρτηση H πρέπει εύκολα και γρήγορα, με υλοποιήσιμες τεχνικές λογισμικού και υλικού, να μπορεί να εφαρμόζεται σε δεδομένα οποιουδήποτε μεγέθους και να μπορεί να παράγει πάντοτε αποτέλεσμα συγκεκριμένου μεγέθους.
2. Preimage resistance: Για οποιοδήποτε μήνυμα (σύνοψη) m, πρέπει να είναι υπολογιστικά ανέφικτο (computationally infeasible) να αποκαλυφθεί x για το οποίο ισχύει H(x) = m.
3. 2nd preimage resistance: Για κάθε μήνυμα x πρέπει να είναι υπολογιστικά ανέφικτο να βρεθεί y, έτσι ώστε να ισχύει: αν y != x => H(y) = H(x)
4. Collision resistance: Πρέπει να είναι υπολογιστικά ανέφικτο να βρεθεί ζεύγος (x, y) για το οποίο να ισχύει: H(x) = H(y)

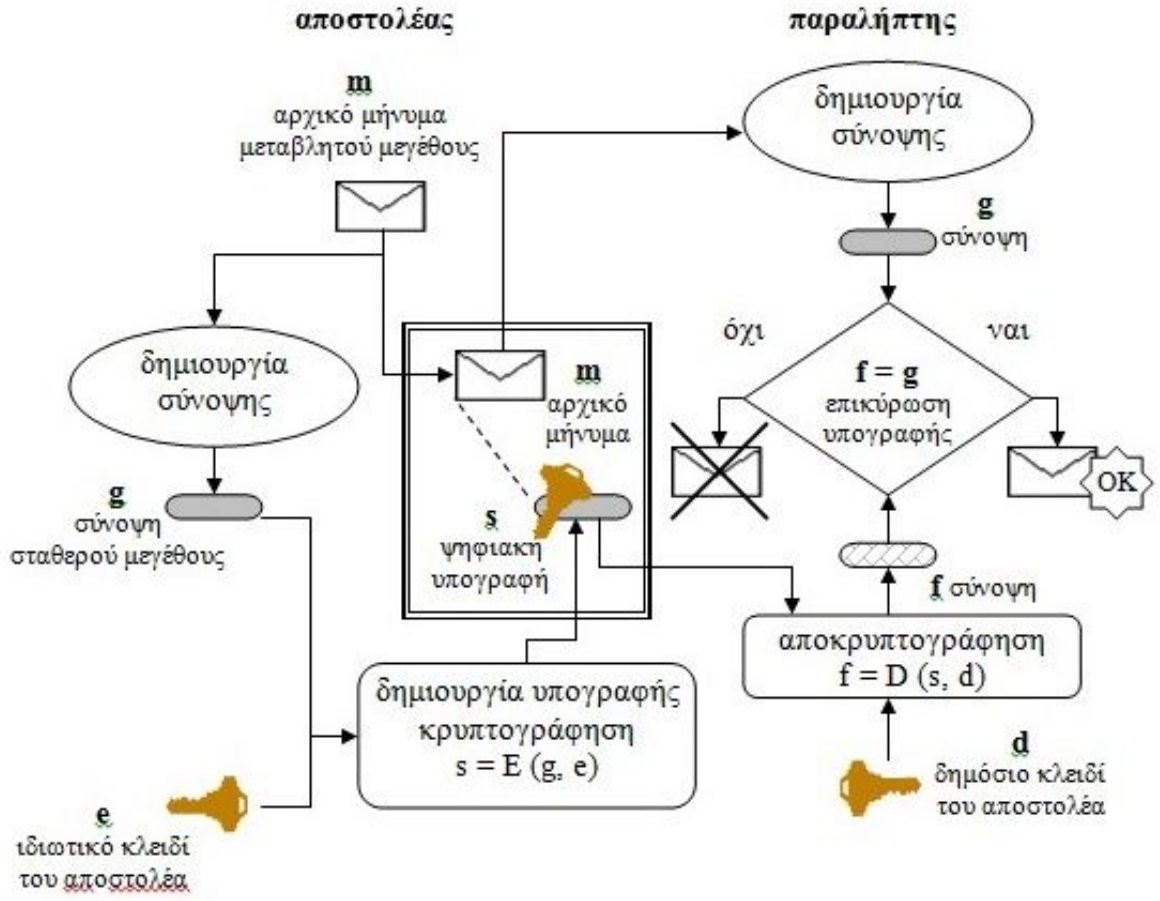
Η πρώτη ιδιότητα αποτελεί τη βασική προϋπόθεση για την πρακτική αξιοποίηση μιας συνάρτησης σύνοψης. Η δεύτερη ιδιότητα εξασφαλίζει μονόδρομη συνάρτηση σύνοψης. Η τρίτη ιδιότητα εξασφαλίζει ότι δεν μπορεί να βρεθεί άλλο μήνυμα. το οποίο να παράγει ίδια σύνοψη με αυτή που παρήγαγε ένα δοσμένο μήνυμα. Συναρτήσεις που ικανοποιούν τις τρεις πρώτες ιδιότητες χαρακτηρίζονται ως ασθενείς (weak) συναρτήσεις σύνοψης. Αν ικανοποιείται και η τέταρτη ιδιότητα της μη-ύπαρξης συγκρούσεων, τότε οι συναρτήσεις χαρακτηρίζονται ισχυρές (strong), αφού αντιμετωπίζουν και μια έξυπνη κατηγορία επιθέσεων, γνωστή στη βιβλιογραφία (Spillman, 2005) ως επίθεση τύπου γενεθλίων (birthday attack).

Οι πιο διαδεδομένοι κρυπτογραφικοί αλγόριθμοι κατατεμαχισμού, παράγουν συνόψεις μεγέθους τουλάχιστον 128 bits, ενώ η πιθανότητα διαφορετικών δεδομένων να δημιουργήσουν την ίδια σύνοψη είναι μόλις 1/ (2^128). Τα σημαντικότερα παραδείγματα μονόδρομων συναρτήσεων σύνοψης αποτελούν: ο MD5 που σχεδιάστηκε από τον Rivest (128-bits), ο οποίος όμως θεωρείται πλέον ανασφαλής λόγω επιτυχημένης επίθεσης το 2008, και οι Secure Hash Algorithm 1 (SHA-1) από το NIST (160 bits). Και ο SHA-1 θεωρείται ανασφαλής γιατί ερευνητές έδειξαν ότι είναι ζήτημα λίγων ετών να 'σπάσει', για αυτό χρησιμοποιούνται παραλλαγές του με 224/256 bits (SHA-2), ενώ το 2014 αναμενόταν από το NIST η υιοθέτηση ως πρότυπο ενός αντικαταστάτη του SHA-2: το επερχόμενο SHA-3 στηρίζεται σε διαφορετικό αλγόριθμο, ο οποίος ήδη έχει επικρατήσει σε σχετικό πρόσφατο διαγωνισμό.

Δημιουργία Ψηφιακών Υπογραφών

Ένα ασφαλές σύστημα παροχής ψηφιακών υπογραφών αποτελείται από δύο μέρη. Στον μεν αποστολέα πραγματοποιείται η μέθοδος υπογραφής ενός κειμένου με "ορθό" τρόπο, στον δε παραλήπτη πραγματοποιείται η μέθοδος επαλήθευσης αν η ψηφιακή υπογραφή παράχθηκε από αυτόν που πραγματικά αντιπροσωπεύει. Η ψηφιακή υπογραφή δημιουργείται από ένα ιδιωτικό κλειδί. Ένα δημόσιο κλειδί χρησιμοποιείται για να επαληθευθεί ότι η υπογραφή δημιουργήθηκε χρησιμοποιώντας το αντίστοιχο ιδιωτικό κλειδί. Η ψηφιακή υπογραφή δημιουργείται κατά τέτοιο τρόπο ώστε να είναι αδύνατο να παραχθεί και πάλι η ίδια ψηφιακή υπογραφή χωρίς τη γνώση του ιδιωτικού κλειδιού.

Video 2.1.mp4	Βίντεο (video)
Δημιουργία Ψηφιακών Υπογραφών	



Εικόνα 2.6 *Ψηφιακή υπογραφή*

Για να δημιουργηθεί μία ψηφιακή υπογραφή (σχήμα 2.6), απαιτούνται δύο βήματα:

- Ο αποστολέας υπολογίζει με χρήση ειδικού λογισμικού μία σύνοψη g του μηνύματος m.
- Κρυπτογραφείται η σύνοψη που προέκυψε (ενδεχομένως μαζί με στοιχεία του αποστολέα, τον τόπο και τον χρόνο υπογραφής), χρησιμοποιώντας το ιδιωτικό κλειδί του αποστολέα e και όχι το δημόσιο κλειδί του παραλήπτη. Η ασύμμετρα κρυπτογραφημένη σύνοψη μαζί με την πληροφορία προσδιορισμού του αλγόριθμου σύνοψης, αποτελεί την ψηφιακή υπογραφή s του μηνύματος. Στη συνέχεια, αυτή αποστέλλεται μαζί με το αρχικό μήνυμα στον παραλήπτη.

Έχουμε ήδη αναφέρει σε προηγούμενη παράγραφο, ότι ο έλεγχος αυθεντικότητας χρησιμοποιεί το ζεύγος των κλειδιών κατ' αυτόν τον τρόπο. Όμως εδώ, στις ψηφιακές υπογραφές, το ιδιωτικό κλειδί δε

χρησιμοποιείται για την κρυπτογράφηση του ίδιου του κειμένου, αλλά μόνο για τη δημιουργία της ψηφιακής υπογραφής, δηλαδή την κρυπτογράφηση της σύνοψης, η οποία επισυνάπτεται στα δεδομένα που αποστέλλονται. Η κρυπτογράφηση λοιπόν με σκοπό τη δημιουργία ψηφιακής υπογραφής αφήνει τα μεταδιδόμενα δεδομένα άθικτα.

Ας σημειωθεί ότι τα δεδομένα αυτά μπορεί να είναι είτε κρυπτογραφημένα, είτε μη κρυπτογραφημένα, ανάλογα με το επίπεδο εμπιστευτικότητας που είναι επιθυμητό, και το οποίο βέβαια δεν έχει σχέση με τους καθ' εαυτού μηχανισμούς των ψηφιακών υπογραφών. Ανεξαρτήτως πάντως της κρυπτογράφησης ή μη των δεδομένων, ο παραλήπτης μπορεί να συμπεράνει αν αυτά έχουν τροποποιηθεί και από πού αυτά προέρχονται, με τη βοήθεια του δημόσιου κλειδιού του αποστολέα. Συνολικά, η επικύρωση της υπογραφής (σχήμα 2.6), χρειάζεται τρία βήματα:

- Το δημόσιο κλειδί d του αποστολέα χρησιμοποιείται για την αποκρυπτογράφηση της ψηφιακής υπογραφής s και κατά συνέπεια της ανάκτησης της σύνοψης (έστω f) του αρχικού κειμένου.
- Χρησιμοποιώντας το κείμενο m που έφθασε στον παραλήπτη, με χρήση του σχετικού λογισμικού που υλοποιεί την ίδια συνάρτηση κατατεμαχισμού, παράγεται η σύνοψή του g.
- Συγκρίνονται οι δύο συνόψεις g και f, δηλαδή αυτή που δημιουργήθηκε από το αφιχθέν κείμενο στο δεύτερο βήμα, με αυτή που αποκρυπτογραφήθηκε στο πρώτο βήμα.

Οποιαδήποτε μεταβολή συνέβη στα δεδομένα, θα έχει ως αποτέλεσμα τη διαφοροποίηση των συνόψεων. Με τον τρόπο αυτόν ο παραλήπτης μπορεί να επιβεβαιώσει:

- ότι τα δεδομένα δεν έχουν μεταβληθεί κατά τη διάρκεια της επικοινωνίας (integrity).
- ότι το δημόσιο και το ιδιωτικό κλειδί του αποστολέα είναι πράγματι ορθό ζεύγος.

Ο παραλήπτης της ψηφιακής υπογραφής πρέπει επίσης λογικά να είναι βέβαιος για την ταυτότητα του αποστολέα (authentication). Επιπλέον, δεν κινδυνεύει να αντιμετωπίσει άρνηση ανάληψης ευθύνης (non-repudiation) από τον αποστολέα, αφού η ψηφιακή υπογραφή είναι κρυπτογραφημένη με το ιδιωτικό κλειδί του. Ή μήπως όχι; Η αλήθεια είναι ότι η επαλήθευση της οντότητας αποστολής και η ακεραιότητα των δεδομένων, αν και πολύ σημαντικά στοιχεία, δεν αποδεικνύουν υποχρεωτικά την ταυτότητα του ιδιοκτήτη του δημόσιου κλειδιού. Πώς μπορεί ο παραλήπτης ενός μηνύματος να είναι βέβαιος ότι ο αποστολέας είναι αυτός που ισχυρίζεται ότι είναι; Το δημόσιο κλειδί του αποστολέα μπορεί να ανακληθεί από μια ιστοσελίδα ή από ένα σχετικό ευρετήριο, αλλά πόσο αξιόπιστο μπορεί να είναι αυτό; Καθένας θα μπορούσε να ζητήσει την έκδοση ενός ζεύγους κλειδιού υπό άλλο όνομα και στη συνέχεια να ανακοινώσει ότι το τάδε δημόσιο κλειδί είναι δικό του. Ο παραλήπτης θα πρέπει να διαθέτει περισσότερες και πραγματικά αξιόπιστες πληροφορίες για τον ιδιοκτήτη του κλειδιού. Η σημαντικότερη μέθοδος στην κατεύθυνση αυτή βασίζεται στην ύπαρξη Έμπιστων Τρίτων Φορέων - ΕΤΦ (Trusted Third Parties, TTP), οι οποίοι και παρέχουν ηλεκτρονικά (ή ψηφιακά) πιστοποιητικά - (certificates) (Gritzalis, 1999; Ferguson et al., 2011). Τα σχετικά ζητήματα, λόγω της σημαντικότητάς τους, θα αναλυθούν διεξοδικά σε επόμενη παράγραφο.

Διαχείριση των Κλειδιών

Αρκετά συστήματα κρυπτογράφησης έχουν την ιδιότητα ότι τόσο το δημόσιο κλειδί, όσο και το ιδιωτικό, μπορούν να χρησιμοποιηθούν και για την κρυπτογράφηση και για την αποκρυπτογράφηση. Κατά συνέπεια, το ίδιο ζεύγος κλειδιών θα μπορούσε να χρησιμοποιηθεί και για την εμπιστευτικότητα (κρυπτογράφηση) των μηνυμάτων και για τις ψηφιακές υπογραφές τους. Η πρακτική αυτή όμως δημιουργεί προβλήματα στη διαχείριση των κλειδιών. Όταν το ζεύγος κλειδιών χρησιμοποιείται για ψηφιακές υπογραφές, το ιδιωτικό κλειδί θα πρέπει να καταστρέφεται μετά το τέλος της ενεργούς ζωής του, καθώς αν ποτέ αυτό ανακαλυφθεί, μπορεί να χρησιμοποιηθεί για πλαστογράφηση υπογραφών. Αντίθετα, αν χρησιμοποιείται για κρυπτογράφηση θα πρέπει να φυλάσσεται για όσο το δυνατόν περισσότερο, καθώς η απώλειά του θα έχει ως αποτέλεσμα να μη γίνεται αποκρυπτογράφηση πληροφοριών που έγιναν με το δημόσιο ανάλογό του. Γι' αυτό στην πράξη υπάρχουν δύο ζεύγη κλειδιών: ένα για τις υπογραφές και ένα για την κρυπτογράφηση.

4. Υποδομές πιστοποίησης

4.1 Εισαγωγή

Η κρυπτογράφηση δημόσιου κλειδιού από μόνη της δεν μπορεί να εγγυηθεί την ασφαλή διακίνηση πληροφοριών σε ένα ανοικτό δίκτυο όπως είναι το Internet. Στο σημαντικότατο ζήτημα της αυθεντικοποίησης των επικοινωνούντων μερών, το μόνο που πραγματικά διασφαλίζεται είναι ότι το δημόσιο και το ιδιωτικό κλειδί του αποστολέα είναι συμπληρωματικό ζευγάρι κλειδιών. Δεν υπάρχει καμιά εγγύηση για το ποιος είναι αυτός που κρατά το ιδιωτικό κλειδί. Ο παραλήπτης χρειάζεται σίγουρα κάποιες πιο αξιόπιστες πληροφορίες σχετικά με την ταυτότητα του ιδιοκτήτη του κλειδιού. Τη λύση μπορεί να δώσει η λειτουργία κοινώς αποδεκτών φορέων που είναι υπεύθυνοι για τη διαχείριση των δημόσιων κλειδιών. Ένας τέτοιος φορέας θα πρέπει να είναι σε θέση να διασφαλίζει ότι ένα συγκεκριμένο δημόσιο κλειδί ανήκει σε ένα συγκεκριμένο άτομο, οργανισμό ή οντότητα και πουθενά αλλού.

Η διαδικασία αυτή της αντιστοίχισης και δέσμευσης ενός δημόσιου κλειδιού σε μια οντότητα, καλείται πιστοποίηση (certification). Κατ' αναλογία, καλούνται πιστοποιητικά δημόσιου κλειδιού (public key certificates) ή απλά πιστοποιητικά, τα ηλεκτρονικά έγγραφα που χρησιμοποιούνται για την αναγνώριση μιας οντότητας και τη συσχέτισή της με ένα δημόσιο κλειδί. Μια Υποδομή Δημόσιου Κλειδιού (ΥΔΚ), στην απλούστερή της μορφή, είναι ένα τέτοιο σύστημα δημοσιοποίησης δημόσιων κλειδιών, το οποίο ως βασική του λειτουργία έχει την πιστοποίηση όσων επικοινωνούν μέσω Internet.

4.2 Υποδομή δημόσιου κλειδιού

Η μέριμνα για τις υπηρεσίες πιστοποίησης είναι ένας σχετικά νέος τομέας υπηρεσιών. Μια ΥΔΚ (Public Key Infrastructure, PKI) δεν εξαντλείται στην αντιπροσώπευση νόμιμων προσβάσεων στα δημόσια κλειδιά κρυπτογράφησης που απαιτούνται για τη χρήση των ψηφιακών υπογραφών και των ψηφιακών φακέλων. Είναι ικανή να παρέχει ένα ευρύ φάσμα σχετικών υπηρεσιών θωράκισης της παρεχόμενης ασφάλειας, όπως ο καθορισμός των πολιτικών ασφάλειας που θα ορίζουν τους κανόνες σύμφωνα με τους οποίους πρέπει να λειτουργούν τα συστήματα κρυπτογράφησης και ο προσδιορισμός της μεθοδολογίας διαχείρισης των ψηφιακών πιστοποιητικών.

Έτσι, ως ΥΔΚ (PKI) τελικά ορίζεται το σύνολο λογισμικού, υλικού, τεχνολογιών κρυπτογράφησης (ασύμμετρης ή υβριδικής), ανθρώπων, πολιτικών, διαδικασιών και υπηρεσιών, που απαιτούνται για τη διαχείριση πιστοποιητικών δημόσιου κλειδιού. Μια ΥΔΚ έχει σκοπό τη διασφάλιση από κάθε πλευρά (εμπιστευτικότητα, ακεραιότητα, αδυναμία απάρνησης, αυθεντικοποίηση) των επικοινωνιών και των συναλλαγών στο Internet, παρέχοντας το πλαίσιο μέσα στο οποίο εφαρμογές μπορούν να αναπτυχθούν και να λειτουργήσουν με ασφάλεια (Γεωργιάδης, 2002; Pangalos et al., 2002). Παραδείγματα τέτοιων εφαρμογών που αξιοποιούν μια ΥΔΚ (PKI-enabled applications) είναι η ασφαλής επικοινωνία μεταξύ των προγραμμάτων πλοήγησης και των εξυπηρετητών Web, το ηλεκτρονικό ταχυδρομείο, η Ηλεκτρονική Ανταλλαγή Δεδομένων (Electronic Data Interchange, EDI), οι συναλλαγές με πιστωτικές κάρτες στο Internet, τα Εικονικά Ιδιωτικά Δίκτυα (Virtual Private Networks, VPNs) κλπ.

Τα συστατικά μέρη ενός PKI είναι:

- Αρχή Πιστοποίησης (ΑΠ)
- Αρχή Καταχώρησης (ΑΚ)
- Σύστημα διανομής ψηφιακών πιστοποιητικών
- Πολιτική πιστοποίησης

Αρχή Πιστοποίησης (ΑΠ)

Sound 2.4.mp3	Ηχητικό απόσπασμα (audio)
Ορισμός της Αρχής Πιστοποίησης	

Μια ΑΠ (Certification Authority, CA) αποτελεί το βασικό (και το μοναδικό απαραίτητο) μέρος μιας ΥΔΚ. Πρόκειται για μια αρχή, την οποία εμπιστεύονται ένας ή περισσότεροι χρήστες για τη δημιουργία και την αντιστοίχιση πιστοποιητικών δημόσιου κλειδιού. Προαιρετικά, η ΑΠ μπορεί να είναι αυτή που δημιουργεί τα απαιτούμενα κλειδιά. Αξίζει να σημειωθεί ότι μια ΑΠ είναι υπεύθυνη όχι μόνο για την έκδοση των πιστοποιητικών δημόσιων κλειδιών αλλά και για τη διαχείρισή τους για όλον τον κύκλο ζωής τους (X.509, 2008). Συνοπτικά, οι κύριες λειτουργίες της είναι:

1. Η δημιουργία πιστοποιητικών που συνδέουν την ταυτότητα ενός χρήστη ή συστήματος με ένα δημόσιο κλειδί.
2. Ο προγραμματισμός των ημερομηνιών λήξης των πιστοποιητικών.
3. Η δημοσίευση των λιστών ανάκλησης πιστοποιητικών (Certification Revocation Lists, CRLs).

Κάθε οργανισμός που ορίζει μια δική του υποδομή δημόσιου κλειδιού, μπορεί να ορίσει μια δική του ΑΠ ή να χρησιμοποιήσει τις υπηρεσίες μιας υπάρχουσας εμπορικής ΑΠ. Σε επόμενη παράγραφο θα δοθούν περισσότερες λεπτομέρειες που αφορούν την ίδια την πιστοποίηση, τις ιεραρχίες των ΑΠ καθώς και τα είδη των πιστοποιητικών που διατίθενται. Πρόκειται πραγματικά για μια περιοχή που βρίσκεται σε συνεχή εξέλιξη.

Αρχή Καταχώρησης (ΑΚ)

Sound 2.5.mp3	Ηχητικό απόσπασμα (audio)
Ορισμός της Αρχής Καταχώρησης	

Η ΑΚ (Registration Authority, RA) αποτελεί μια προαιρετικά ξεχωριστή από την ΑΠ οντότητα, η οποία αναλαμβάνει την ευθύνη ορισμένων διαχειριστικών διαδικασιών, απαραίτητων για την καταχώριση των οντοτήτων που αποτελούν υποκείμενα πιστοποίησης. Τέτοια διαχειριστικά καθήκοντα είναι (X.509, 2008):

1. Η επιβεβαίωση της ταυτότητας του υποκειμένου πιστοποίησης.
2. Η επικύρωση ότι το υποκείμενο αυτό επιτρέπεται να χρησιμοποιεί ως «δικά του» αυτά που δήλωσε κατά την αίτηση χορήγησης ενός πιστοποιητικού δημόσιου κλειδιού.
3. Η επαλήθευση ότι το υποκείμενο κατέχει το ιδιωτικό κλειδί το οποίο συσχετίζεται με το δημόσιο που ζητά να περιέχεται στο προς έκδοση πιστοποιητικό.

Η ΑΚ υλοποιεί σε ένα βαθμό το περιβάλλον προσέγγισης των χρηστών με μια ΑΠ, μεταφέροντας στην ΑΠ όλες τις αιτήσεις για δημιουργία πιστοποιητικών. Η ποιότητα της διαδικασίας ελέγχου της αυθεντικότητας του χρήστη (π.χ. τηλεφωνική επικοινωνία, φυσική παρουσία κ.ό.κ.), καθορίζει και την τάξη (class) εμπιστοσύνης του πιστοποιητικού.

Σύστημα διανομής πιστοποιητικών

Διάφοροι τρόποι υπάρχουν για τη διανομή των πιστοποιητικών. Αν και οι ίδιοι οι χρήστες θα μπορούσαν να αναλάβουν την ευθύνη της διανομής των πιστοποιητικών τους, η πιο συνηθισμένη μέθοδος είναι μέσω μια υπηρεσίας καταλόγου (directory service).

Έτσι, αντί να χρειάζεται οι χρήστες να αποθηκεύουν τα πιστοποιητικά τους τοπικά, μπορούν αυτά να αποθηκεύονται σε ένα κατάλληλο εξυπηρετητή καταλόγου. Σχεδόν όλα τα PKI διαθέτουν τέτοιους εξυπηρετητές για την αποθήκευση πιστοποιητικών, σύμφωνους με το πρότυπο X.500. Όταν λοιπόν μια ΑΠ εκδώσει ένα πιστοποιητικό, το αποθηκεύει σε μια "αποθήκη" πιστοποιητικών (Certificate Repository). Έτσι, όποτε κάποιος ζητήσει ένα πιστοποιητικό που υπάρχει σε αυτό το αρχείο, μπορεί να το πάρει χωρίς να χρειάζεται να το ζητήσει από τον κάτοχό του.

Πολιτική Πιστοποίησης

Η πολιτική πιστοποίησης (certificate policy, CP) είναι ένα σύνολο κανόνων που υποδεικνύει τη δυνατότητα εφαρμογής (applicability) της χρήσης πιστοποιητικών σε μια ιδιαίτερη ομάδα εφαρμογών, έτσι όπως αυτή προσδιορίζεται από τις κοινές απαιτήσεις ασφάλειάς τους. Περιγράφει τους γενικούς κανόνες που πρέπει να ακολουθούνται για την ασφάλεια των πληροφοριών, και ειδικότερα αυτούς που αφορούν τη χρήση της

κρυπτογραφίας με σκοπό την επίτευξη ενός «αποδεκτού» (acceptable) επίπεδου διασφάλισης. Συνήθως περιλαμβάνει οδηγίες για το χειρισμό των κλειδιών και άλλων σημαντικών πληροφοριών και προσδιορίζει διάφορα επίπεδα ελέγχου ανάλογα με το επίπεδο του κινδύνου κάθε πληροφορίας.

Έκφραση της πολιτικής ασφάλειας αποτελεί η καλούμενη δήλωση πρακτικών της πιστοποίησης (Certificate Practice Statement, CPS), μια παρουσίαση των τρόπων εφαρμογής που χρησιμοποιεί η ΑΠ για την έκδοση πιστοποιητικών. Το CPS είναι ένα αναλυτικό έγγραφο που περιγράφει όλες τις λειτουργικές διαδικασίες που πρέπει να ακολουθηθούν ώστε να εφαρμοστεί η πολιτική ασφάλειας στην πράξη. Περιλαμβάνει πληροφορίες για θέματα όπως η κατασκευή και η λειτουργία των ΑΠ και ΑΚ, η δημιουργία, η ανάκληση και η διανομή των πιστοποιητικών, ή ακόμη και πληροφορίες αναφορικά με τη δημιουργία, την κατοχύρωση, την επιβεβαίωση και την αποθήκευση των κλειδιών.

4.3 Υπηρεσίες μιας υποδομής δημόσιου κλειδιού

Οι υπηρεσίες που μπορεί να παρέχει μια ΥΔΚ, μπορούν να χωριστούν σε τρεις κατηγορίες:

- Υπηρεσίες διαχείρισης πιστοποιητικών
- Υπηρεσίες κρυπτογράφησης
- Βοηθητικές υπηρεσίες

Υπηρεσίες διαχείρισης πιστοποιητικών - Αποτελούν τον πυρήνα ενός PKI και παρέχονται απαραιτήτως από μια ΑΠ. Περιλαμβάνουν:

1. Έκδοση πιστοποιητικού (Certificate Issuance)
2. Ανάκληση πιστοποιητικού (Certificate Revocation)
3. Δημοσίευση πιστοποιητικού (Certificate Publishing)
4. Αρχειοθέτηση πιστοποιητικού (Certificate Archiving)
5. Δημιουργία / έγκριση πολιτικής πιστοποίησης (CP formation / approval)

Υπηρεσίες κρυπτογράφησης - Τέτοιες υπηρεσίες είναι:

1. Η δημιουργία. των ζευγαριών των κλειδιών ασύμμετρης κρυπτογράφησης.
2. Η δημιουργία των κλειδιών εμπιστευτικότητας (συμμετρικής κρυπτογραφίας) για την υποστήριξη των ψηφιακών φακέλων.
3. Η διανομή των κλειδιών εμπιστευτικότητας
4. Η δημιουργία των ψηφιακών υπογραφών.
5. Η επιβεβαίωση της εγκυρότητας των ψηφιακών υπογραφών.

Βοηθητικές υπηρεσίες - Υπάρχουν και ορισμένες άλλες υπηρεσίες που θεωρούνται βασικά μέρη μιας τέτοιας υποδομής:

- Καταχώρηση (Registration): Αναφερθήκαμε στην υπηρεσία αυτή της διαχείρισης των καταχωρήσεων - εγγραφών όταν εξηγήθηκε ο ρόλος της Αρχής Καταχώρισης (ΑΚ), ως συστατικού μέρους μιας ΥΔΚ.
- Ανάκτηση κλειδιού (Key Recovery): Η υπηρεσία αυτή είναι υπεύθυνη για τη δημιουργία απόρρητων και καλά προφυλαγμένων αντιγράφων των κλειδιών. Σκοπός είναι η αντιμετώπιση των περιπτώσεων όπου οι χρήστες χάνουν κλειδιά, ή ξεχνάνε τους κωδικούς που απαιτούνται για να έχουν πρόσβαση σε αυτά.
- Αποθήκευση Πληροφοριών - Αρχειοθέτηση δεδομένων (Information Repository - Data Archiving): Η υπηρεσία αυτή αποθηκεύει, αρχειοθετεί, και διαχειρίζεται διάφορα είδη προσωπικών ή όχι δεδομένων που ανήκουν στους χρήστες της ΥΔΚ. Αποτελεί υποστηρικτική διαδικασία ενός PKI και αφορά νομικές απαιτήσεις, κυβερνητικές ή εσωτερικές ρυθμίσεις, ανάγκες λήψης αντιγράφων ασφαλείας και ανάκαμψης συστήματος, ανάκτησης προσωπικών πληροφοριών κ.ά. Η υπηρεσία αυτή αναλαμβάνει τη διαχείριση των σχετικών ψηφιακών εγγράφων και των αντίστοιχων δεδομένων τους για μεγάλες χρονικές

περιόδους. Είναι υπεύθυνη για την ασφαλή αποθήκευσή τους σε διάφορα μέσα, ώστε να μην παραποιηθούν και να μην παραβιαστεί η εμπιστευτικότητά τους από μη εξουσιοδοτημένους χρήστες. Μια τέτοια υπηρεσία είναι χρήσιμη σε αρκετές περιπτώσεις. Ενδεικτικά, όταν οι χρήστες για να ικανοποιήσουν κάποια νομική υποχρέωση πρέπει να αποδείξουν ότι μια ΥΔΚ έχει λάβει γνώση ενός συγκεκριμένου εγγράφου, όταν οι χρήστες χρειάζονται μια έμπιστη υπηρεσία διατήρησης ενός εφεδρικού αντίγραφου ενός κειμένου, όταν απαιτείται η αποποίηση ευθύνης χωρίς να χρειάζεται η μετάδοση του ηλεκτρονικού εγγράφου κλπ.

- Επιπρόσθετες υπηρεσίες: Μπορεί να χρειαστούν και διάφορες άλλες υπηρεσίες, ανάλογα με την περίπτωση. Για παράδειγμα, όταν χρησιμοποιούνται έξυπνες κάρτες (smart cards) για την αποθήκευση του κλειδιού, απαιτούνται πρόσθετες υπηρεσίες για την εγγραφή τους στην κάρτα.

5. Επισημάνσεις - Συμπεράσματα

Η ασφάλεια των πληροφοριακών συστημάτων, ως κλάδος της επιστήμης της Πληροφορικής, έχει αντικείμενο την πρόληψη μη-εξουσιοδοτημένων ενεργειών των χρηστών ενός πληροφοριακού συστήματος καθώς και την ανίχνευση και την κατάλληλη αντίδραση στις περιπτώσεις εκδήλωσής τους. Τα δίκτυα μπορεί να ειδωθούν ως κάποιες περισσότερο σύνθετες περιπτώσεις πληροφοριακών συστημάτων, και έτσι είναι ουσιαστικά οι γνώριμες απειλές εμπιστευτικότητας, ακεραιότητας και διαθεσιμότητας οι οποίες εκδηλώνονται και σε αυτά αλλά με πολύ περισσότερους και διαφορετικούς τρόπους. Σε ένα μάλιστα ανοικτό δικτυακό περιβάλλον, όπως αυτό του Internet και του παγκόσμιου Ιστού, οι κίνδυνοι πολλαπλασιάζονται λόγω της έλλειψης εμπιστοσύνης προς οποιαδήποτε εξωτερική, ως προς το υπό προστασία σύστημα, οντότητα. Ο τρόπος αντιμετώπισης των προβλημάτων ασφάλειας στηρίζεται σε τρεις θεμελιώδεις αρχές. Σύμφωνα με την "αρχή της ευκολότερης διείσδυσης", ένας επίδοξος "εισβολέας" θα χρησιμοποιήσει τον ευκολότερο για αυτόν τρόπο επίθεσης. Για αυτόν τον λόγο όλες οι αδυναμίες ενός πληροφοριακού συστήματος πρέπει να προφυλαχθούν στον ίδιο βαθμό. Ακόμη περισσότερο, πρέπει τα ζητήματα ασφάλειας, από κάθε άποψη, να μελετηθούν και να απαντηθούν ως ένα ενιαίο σύνολο, έτσι ώστε να είναι δυνατή η επίτευξη ενός ομοιόμορφου επιπέδου ασφάλειας σε όλα τα συστατικά μέρη του πληροφοριακού συστήματος ή δικτύου. Σύμφωνα με τη δεύτερη "αρχή της κατάλληλης προστασίας", τα μέρη ενός συστήματος πρέπει να προστατεύονται πάντα σε ένα βαθμό ανάλογο και συνεπή ως προς την αξία τους. Τέλος, σημαντικό ρόλο διαδραματίζει και η τρίτη "αρχή της αποτελεσματικότητας", η οποία ορίζει ως προϋποθέσεις αποτελεσματικότητας των μέτρων προστασίας, την ευχρηστία, την επάρκεια και την καταλληλότητά τους, έτσι ώστε αυτά να είναι όντως σε ισχύ όταν εκδηλωθούν τα προβλήματα ασφάλειας.

Στο κεφάλαιο αυτό παρουσιάστηκαν οι σημαντικότερες από τις διαθέσιμες τεχνολογίες διασφάλισης στο Διαδίκτυο, οι οποίες αποτελούν πολύ χρήσιμα εργαλεία υποστήριξης μέτρων προστασίας. Η κατάλληλη διαμόρφωση και εφαρμογή τους, πρέπει να γίνεται πάντοτε με γνώμονα τις θεμελιώδεις προαναφερθείσες αρχές, έτσι ώστε αυτές να οδηγούν στο υψηλότερο δυνατό επίπεδο ασφάλειας. Στις κυριότερες διαθέσιμες τεχνολογίες ασφάλειας στο Διαδίκτυο περιλαμβάνονται η κρυπτογράφηση, οι ψηφιακές υπογραφές, και οι υποδομές δημόσιου κλειδιού. Η κρυπτογραφία είναι στις μέρες μας κοινά αποδεκτή σαν το πλέον απαραίτητο εργαλείο ασφάλειας στο Διαδίκτυο. Δύο σημαντικές εφαρμογές κρυπτογραφίας είναι η κρυπτογράφηση και οι ψηφιακές υπογραφές. Η κρυπτογράφηση μπορεί να εξασφαλίσει ότι οι διακινούμενες πληροφορίες είναι εμπιστευτικές. Οι ψηφιακές υπογραφές βοηθούν στην επικύρωση της προέλευσης δεδομένων και επιβεβαιώνουν αν τα δεδομένα έχουν αλλοιωθεί. Περαιτέρω δυνατότητες προσφέρονται μέσω των υποδομών δημοσίου κλειδιού, οι οποίες με την έκδοση των πιστοποιητικών ταυτότητας, αποδεικνύονται ικανές για την υποστήριξη ενός μεγάλου μέρους λειτουργιών ασφάλειας στο Internet.

Βιβλιογραφία / Αναφορές

Ahuja, V. (1997). *Secure Commerce on the Internet*, AP Professional, ISBN: 0-12-045597-8.

Bullock, A., & Benford, S. (1999). An access control framework for multi-user collaborative environments. In *Proceedings of the international ACM SIGGROUP conference on Supporting group work,* ACM, pp. 140-149.

Cherdantseva, Y., & Hilton, J. (2013). A reference model of information assurance & security. In *Availability, Reliability and Security* (ARES), 2013 Eighth International Conference on, IEEE, pp. 546-555.

Ferguson, N., Schneier, B., & Kohno, T. (2011). *Cryptography engineering: design principles and practical applications*. John Wiley & Sons.

Gollmann, D. (2011*). Computer Security, 3rd edition*, Wiley, ISBN: 978-0470741153.

Gritzalis, D. (1999). Trusted Third Parties and Public Key Infrastructure: An Overview, Teaching Material (European Intensive Programme on Information and Communication Technologies Security IPICS 99), University of the Aegean, Greece.

ISO/IEC 27000 (2014). Information technology – Security techniques - Information security management systems - Overview and vocabulary, http://standards.iso.org/ittf/PubliclyAvailableStandards/c063411_ISO_IEC_27000_2014.zip

Laudon, K. & Traver, C. (2014). *E-Commerce, 10/E*, Prentice Hall, ISBN: 978-0133024449.

Mohapatra, K. P. (2000). Public Key Cryptography, *Crossroads ACM Magazine*, Issue 7.1 (Fall 2000), ISSN: 1528-4972, ACM, pp. 14-22.

Pangalos, G. (1992). Security in Medical Database Systems, EU, SEISMED project report, No. INT/S.3/92, Aristotle University of Thessaloniki, Greece.

Pangalos, G. (1998). EU MEDSEC Project Report, deliverable D14: Review of Existing and Emerging Work on Secure Medical Database Systems, Health Care Security and Privacy in the Information Society, ISIS Programme, Aristotle University of Thessaloniki, Greece.

Pangalos, G., Mavridis I., Ilioudis C. & Georgiadis C. (2002). Developing a Public Key Infrastructure for a Secure Regional e-Health Environment, *Methods of Information in Medicine*, Vol. 41/5, Schattauer Publishing Co., pp. 414-418.

Pfleeger, C. P. & Pfleeger, S. L. (2007). *Security in Computing, Fourth Edition*, Prentice Hall, ISBN: 978-0132390774.

Public Key Infrastructure (X.509), (2008). RFC 5280 (IETF's PKIX Certificate), http://tools.ietf.org/pdf/rfc5280.pdf

Recommendation ITU-T X.800 technically aligned with ISO/IEC 7498-2 (1991). Information processing systems – Open systems interconnection – Basic Reference Model – Part 2: Security architecture, http://www.itu.int/ITU-T/recommendations/rec.aspx?id=3102

Spillman, R. (2005). *Classical and Contemporary Cryptology*, Prentice Hall, ISBN: 0131828312.

Γεωργιάδης, Χ. Κ. (2002). *Ασφάλεια Πληροφοριακών Συστημάτων και Εφαρμογές στον Έλεγχο Προσπέλασης Ιατρικών Βάσεων Δεδομένων μέσω Internet*, Διδακτορική Διατριβή, Πολυτεχνική Σχολή, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης.

Γκρίτζαλης, Σ. & Γεωργιάδης, Π. (1997). Ψηφιακές Υπογραφές: Διεθνής Εμπειρία, Τάσεις και Προοπτικές, Εισήγηση στα πλαίσια σεμιναρίου του ανθρώπινου δικτύου ΙΚΑΡΟΣ για την Ασφάλειας, Ποιότητας και Αξιοπιστίας στις Τεχνολογίες Πληροφοριών και Επικοινωνιών, ΕΠΕΤ ΙΙ.

Πάγκαλος, Γ. & Μαυρίδης, Ι. (2002). *Ασφάλεια πληροφοριακών συστημάτων και δικτύων*, Ανίκουλας, ISBN: 978-9605160180.

Quiz2.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Με τον όρο "υποκλοπή" (eavesdropping) αναφερόμαστε στην περίπτωση όπου:

Α) Οι πληροφορίες παραμένουν ανέγγιχτες, αλλά παραβιάζεται η εμπιστευτικότητά τους

Β) Οι πληροφορίες πηγαίνουν σε ένα πρόσωπο που παριστάνει τον νόμιμο αποδέκτη

Γ) Οι πληροφορίες κατά τη μεταφορά τους μεταβάλλονται ή τροποποιούνται και στη συνέχεια στέλνονται στον αποδέκτη

Απάντηση/Λύση

Α) Οι πληροφορίες παραμένουν ανέγγιχτες, αλλά παραβιάζεται η εμπιστευτικότητά τους.

Κριτήριο αξιολόγησης 2

[*] Με τον όρο "εμπιστευτικότητα δεδομένων" (data confidentiality) αναφερόμαστε:

Α) Στην προστασία από τη μη-ανάληψη ευθύνης ενός αποστολέα ότι αυτός έστειλε συγκεκριμένα δεδομένα

Β) Στη δυνατότητα εντοπισμού παραποίησης και ανάκτησης των δεδομένων

Γ) Στην προστασία ενάντια σε μη-εξουσιοδοτημένες αποκαλύψεις πληροφοριών

Απάντηση/Λύση

Γ) Στην προστασία ενάντια σε μη-εξουσιοδοτημένες αποκαλύψεις πληροφοριών.

Κριτήριο αξιολόγησης 3

[*] Η ψηφιακή υπογραφή (digital signature), ως ένα είδος ηλεκτρονικής υπογραφής, είναι μία συμβολοσειρά από bits και εξαρτάται πάντοτε από το μήνυμα που συνοδεύει.

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Α) Σωστό

Κριτήριο αξιολόγησης 4

[*] Με τον όρο "Παραποίηση", αναφερόμαστε στην περίπτωση όπου:

Α) Οι πληροφορίες κατά τη μεταφορά τους μεταβάλλονται ή τροποποιούνται και στη συνέχεια στέλνονται στον αποδέκτη

Β) Οι πληροφορίες πηγαίνουν σε ένα πρόσωπο που παριστάνει τον νόμιμο αποδέκτη

Γ) Οι πληροφορίες παραμένουν ανέγγιχτες, αλλά παραβιάζεται η εμπιστευτικότητά τους

Απάντηση/Λύση
Α) Οι πληροφορίες κατά τη μεταφορά τους μεταβάλλονται ή τροποποιούνται και στη συνέχεια στέλνονται στον αποδέκτη

Κριτήριο αξιολόγησης 5
[*] Ένας από τους επιδιωκόμενους στόχους κατά την κρυπτογράφηση είναι:

Α) Να είναι πολύ δύσκολη η μετατροπή του αρχικού κειμένου σε κρυπτογράφημα

Β) Να κρατηθεί σχετικά απλή η διαδικασία μετατροπής του αρχικού κειμένου σε κρυπτογράφημα

Γ) Να είναι δυνατή σχετικά εύκολα η διαδικασία μετατροπής του κρυπτογραφήματος

Απάντηση/Λύση
Β) Να κρατηθεί σχετικά απλή η διαδικασία μετατροπής του αρχικού κειμένου σε κρυπτογράφημα

Κριτήριο αξιολόγησης 6
[**] Ο κρυπτογραφικός αλγόριθμος κατατεμαχισμού SHA-1, χρησιμοποιεί κωδικοποίηση:

Α) 128 bits

Β) 160 bits

Γ) 224 bits

Δ) 256 bits

Απάντηση/Λύση
Β) 160 bits

Κριτήριο αξιολόγησης 7
[**] Ο όρος "έλεγχος αποδοτικότητας δικτύου" (efficiency control) αναφέρεται σε:

Α) Μηχανισμούς που καταγράφουν τις δηλώσεις ταυτότητας και τις ενέργειες των χρηστών (αλλά και των συστημάτων) που αποκτούν πρόσβαση σε προστατευμένους πόρους

Β) Μηχανισμούς που καταγράφουν και παρακολουθούν τη συνολική απόδοση του συστήματος και την κίνηση του δικτύου, με σκοπό την αποτροπή καταστάσεων άρνησης εξυπηρέτησης (prevention of Denial of Service)

Γ) Εφαρμογές που εκτελούνται στο Διαδίκτυο, διαθέτουν ενδεχομένως χαρακτηριστικά ασφάλειας που πρέπει να μπορούν να κληθούν και να λειτουργούν με ενιαίους τρόπους

Απάντηση/Λύση
Β) Μηχανισμούς που καταγράφουν και παρακολουθούν τη συνολική απόδοση του συστήματος και την κίνηση του δικτύου, με σκοπό την αποτροπή καταστάσεων άρνησης εξυπηρέτησης (prevention of Denial of Service)

Κριτήριο αξιολόγησης 8
[*] Με τον όρο "αδυναμία-απάρνησης" (non-repudiation) αναφερόμαστε:

Α) στην ικανότητα των συστημάτων να λειτουργούν σωστά κάτω από αντίξοες συνθήκες

Β) στην εκούσια ή και ακούσια πρόκληση ζημιών στα πληροφοριακά αγαθά

Γ) στο ότι ένας χρήστης δεν μπορεί να αποποιηθεί της ευθύνης για κάποια πράξη που έκανε

Απάντηση/Λύση
Γ) στο ότι ένας χρήστης δεν μπορεί να αποποιηθεί της ευθύνης για κάποια πράξη που έκανε

Κριτήριο αξιολόγησης 9
[*] Η σύγχρονη κρυπτογραφία στηρίζεται στη μυστικότητα των αλγορίθμων της

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 10
[**] Η κρυπτογραφία μυστικού κλειδιού:

Α) δεν έχει πρόβλημα με τη διαχείριση των κλειδιών που απαιτούνται, αλλά είναι αργή σε σχέση με την κρυπτογραφία δημοσίου κλειδιού

Β) απαιτεί κλειδιά μεγαλύτερου μεγέθους, σε σχέση με αυτά της κρυπτογραφίας δημοσίου κλειδιού, για να παρέχει το ίδιο επίπεδο ασφάλειας, αλλά είναι πιο γρήγορη

Γ) έχει ως κύριο πρόβλημα τη διαχείριση των κλειδιών που απαιτούνται, αλλά είναι πολύ πιο γρήγορη σε σχέση με την κρυπτογραφία δημοσίου κλειδιού

Απάντηση/Λύση
Γ) έχει ως κύριο πρόβλημα τη διαχείριση των κλειδιών που απαιτούνται, αλλά είναι πολύ πιο γρήγορη σε σχέση με την κρυπτογραφία δημοσίου κλειδιού

Κεφάλαιο 3: Ασφαλείς Συναλλαγές στον Παγκόσμιο Ιστό

Σύνοψη
Οι συναλλαγές στον παγκόσμιο Ιστό και κυρίως σε περιβάλλοντα ηλεκτρονικού εμπορίου δέχονται απειλές με διάφορους τρόπους. Απειλές όπως το κακόβουλο λογισμικό αλλά και το ηλεκτρονικό ψάρεμα είναι ανάμεσα στους πιο διαδεδομένους τρόπους υποκλοπής ή παραποίησης δεδομένων. Το παρόν κεφάλαιο, στο πρώτο μέρος του, περιγράφει αναλυτικά τις απειλές αυτές που υπάρχουν σε σύγχρονα περιβάλλοντα ηλεκτρονικού εμπορίου. Στη συνέχεια, η εστίαση μεταφέρεται στο πώς αντιμετωπίζονται οι διάφορες επιθέσεις ασφάλειας και επεξηγείται το πολύ κρίσιμο ζήτημα του ελέγχου προσπέλασης (access control). Παρουσιάζονται τα θεμελιώδη μοντέλα και οι πολιτικές εξουσιοδοτήσεων αυτών. Ιδιαίτερο βάρος δίνεται στην απόδοση προνομίων βάσει ρόλων (RBAC), καθώς και στην αξιοποίηση του πλαισίου αναφοράς (context) κατά τον έλεγχο προσπέλασης. Τέλος, μας απασχολούν τα συστήματα πληρωμών με τα ειδικότερα ζητήματα ασφάλειας και διαφύλαξης της ιδιωτικότητας που αυτά θέτουν. Ειδική αναφορά γίνεται σε δύο σημαντικά σχήματα-πρωτόκολλα ασφαλείας που εμπλέκονται στην υποστήριξη πληρωμών ηλεκτρονικού εμπορίου.

Προαπαιτούμενη γνώση
Το κεφάλαιο 2 του παρόντος συγγράμματος

1. Απειλές ασφαλείας σε περιβάλλοντα ηλεκτρονικού εμπορίου

Στη παράγραφο αυτή θα εστιάσουμε αρχικά στις απειλές λόγω κακόβουλου λογισμικού. Στη συνέχεια, θα αναφερθούμε σε ορισμένες άλλες αρκετά διαδεδομένες απειλές, όπως το ηλεκτρονικό ψάρεμα και η παραποίηση ταυτότητας. Τέλος, θα μιλήσουμε για μια ιδιαίτερη κατηγορία απειλών, στις απειλές λόγω κακής σχεδίασης της υποστήριξης 'κινητού' κώδικα.

1.1 Κακόβουλο λογισμικό

Σημαντικές απειλές εκδηλώνονται μέσω προγραμμάτων που εκμεταλλεύονται μία ή περισσότερες ευπάθειες των συστατικών μερών της υποδομής υποστήριξης στα περιβάλλοντα ηλεκτρονικού εμπορίου. Τέτοια προγράμματα αναφέρονται με τον όρο κακόβουλο λογισμικό (malicious software ή malware). Είναι προγράμματα κατασκευασμένα ειδικά με στόχο την παραβίαση της ασφάλειας του συστήματος. Μια πρώτη κατηγοριοποίηση του κακόβουλου λογισμικού διακρίνει αυτό που χρειάζεται ένα πρόγραμμα-φορέα σε αντιδιαστολή με αυτό που λειτουργεί ανεξάρτητα (Κάτσικας, 2001). Έτσι, στην πρώτη κατηγορία ανήκουν ουσιαστικά τμήματα προγράμματος που δεν είναι δυνατόν να υπάρξουν μόνα τους, χωρίς κάποιο λογισμικό συστήματος ή κάποιο πρόγραμμα εφαρμογής. Ενώ στη δεύτερη κατηγορία ανήκουν όσα είναι αυτόνομα προγράμματα που μπορούν να εκτελεστούν κάτω από τον έλεγχο του λειτουργικού συστήματος, όπως συμβαίνει στα 'κανονικά' προγράμματα.

Μια άλλη κατηγοριοποίηση του κακόβουλου λογισμικού διακρίνει το μη αναπαραγόμενο από το αναπαραγόμενο. Η πρώτη κατηγορία περιλαμβάνει τμήματα προγράμματος που ενεργοποιούνται όταν καλείται το πρόγραμμα-φορέας για να εκτελέσει μια συγκεκριμένη λειτουργία. Η δεύτερη κατηγορία περιλαμβάνει τμήματα προγράμματος, αλλά και αυτόνομα προγράμματα που, όταν εκτελούνται, μπορούν να παραγάγουν ένα ή περισσότερα αντίγραφα του εαυτού τους, τα οποία θα ενεργοποιηθούν αργότερα στον ίδιο ή σε κάποιον άλλον υπολογιστή.

Για την εγκατάσταση (μόλυνση) ενός κακόβουλου λογισμικού σε ένα μηχάνημα, συνήθως απαιτείται η ανθρώπινη συμμετοχή. Η συμμετοχή αυτή μπορεί να είναι άμεση (π.χ. εισαγωγή ενός USB, άνοιγμα συνημμένων αλληλογραφίας, προεπισκόπηση μηνυμάτων αλληλογραφίας, ανταλλαγή αρχείων κλπ.), αλλά μπορεί να είναι και έμμεση (π.χ. μη ενημέρωση του λογισμικού ασφαλείας, επιλογή προφανούς κωδικού σύνδεσης κλπ.). Το τμήμα του κώδικα που είναι υπεύθυνο για τις παρενέργειες του λογισμικού καλείται φορτίο (payload). Το κακόβουλο λογισμικό περιλαμβάνει και επιπρόσθετο κώδικα με σκοπό α) την αναπαραγωγή του (την εξάπλωση του στο σύστημα που προσβάλλει - «μόλυνση» από το ένα πρόγραμμα του

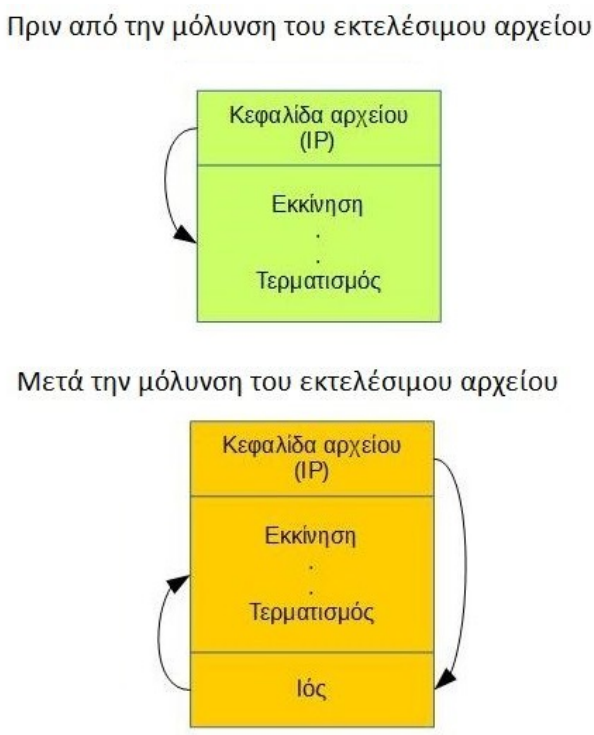
μηχανήματος σε άλλο πρόγραμμα) και β) τη μετάδοσή του (την εξάπλωσή του από το μηχάνημα που μολύνθηκε σε άλλα) (Μάγκος. 2013).

Όλα τα είδη κακόβουλου λογισμικού δίνουν μεγάλη σημασία στον εντοπισμό της πιο κατάλληλης περιοχής για να εγκατασταθούν. Επιδιώκουν η εκτέλεσή τους να μην είναι ανιχνεύσιμη, να εγγράφονται στο μητρώο του συστήματος και να δημιουργούν εμπόδια στις διαδικασίες αφαίρεσής τους.

1.1.1 Μορφές κακόβουλου λογισμικού

Με βάση τον τρόπο λειτουργίας του κακόβουλου λογισμικού, αναφέρονται στην παράγραφο αυτή τα κυριότερα είδη (Kura, 2013; Μάγκος, 2013). Να σημειωθεί βέβαια, ότι συχνά συναντούμε περιπτώσεις συνδυασμών αυτών των ειδών (όπως ιός μαζί με δούρειο ίππο, ιός σε συνδυασμό με σκουλήκι κλπ.).

Ιός (virus): Κακόβουλο λογισμικό το οποίο αφού μολύνει ένα μηχάνημα έχει την ικανότητα να αναπαράγεται και να μολύνει και άλλα προγράμματα στο μηχάνημα αυτό. Ο όρος, όχι τυχαία, προήλθε από τη βιολογία. Ο βιολογικός ιός είναι ένα πολύ μικρό τμήμα γενετικού κώδικα που μπορεί να καταλάβει τον μηχανισμό αναπαραγωγής ενός υγιούς ζωντανού κυττάρου κάποιου οργανισμού και να τον εξαπατήσει, έτσι ώστε να δημιουργήσει χιλιάδες τέλεια αντίγραφα του εαυτού του (του ιού). Όπως και τα βιολογικά τους ανάλογα, οι ιοί του υπολογιστή περιέχουν στον κώδικά τους τη «συνταγή» δημιουργίας τέλειων αντιγράφων του εαυτού τους (Κάτσικας, 2001). Μόλις εγκατασταθεί ο ιός, οποτεδήποτε ο μολυσμένος υπολογιστής έρθει σε επαφή με μη μολυσμένο πρόγραμμα, το πρόγραμμα αυτό μολύνεται με την εισαγωγή στον κώδικά του ενός αντιγράφου του ιού. Για το ιομορφικό κακόβουλο λογισμικό στις σύγχρονες εκφάνσεις του, θα δώσουμε λόγω της σοβαρότατης απειλής που αποτελεί, και κάποια επιπλέον στοιχεία σε ακόλουθη παράγραφο.



Σχήμα 3.1 *Πριν και μετά τη μόλυνση του εκτελέσιμου αρχείου*

Σκουλήκι (Worm): Κακόβουλο λογισμικό το οποίο, αφού μολύνει ένα μηχάνημα, έχει την ικανότητα να μεταδίδεται αυτόματα, κάνοντας χρήση της υπάρχουσας δικτυακής υποδομής. Από τη στιγμή που θα ενεργοποιηθεί μέσα σ' ένα σύστημα, το σκουλήκι μπορεί να συμπεριφερθεί ως ιός ή ως βακτήριο ή να εισαγάγει Δούρειους Ίππους ή να εκτελέσει οποιαδήποτε καταστροφική ενέργεια (Aycock, 2006). Για να αναπαραχθεί ένα σκουλήκι μπορεί να χρησιμοποιήσει την υπηρεσία ηλεκτρονικού ταχυδρομείου (ταχυδρομεί ένα αντίγραφο του εαυτού του σε άλλα συστήματα), την υπηρεσία εκτέλεσης από απόσταση (εκτελεί ένα αντίγραφο του εαυτού του σε κάποιο άλλο σύστημα) ή την υπηρεσία σύνδεσης από απόσταση (συνδέεται με

ένα απομακρυσμένο σύστημα ως χρήστης και μετά χρησιμοποιεί εντολές για να αντιγράψει τον εαυτό του από ένα σύστημα σε άλλο).

Δούρειος Ίππος (Trojan Horse): Κακόβουλο λογισμικό στο οποίο είναι εγγενές το στοιχείο της παραπλάνησης, καθώς συνήθως μεταμφιέζεται σε μια χρήσιμη εφαρμογή, η οποία όμως περιέχει κακόβουλο κώδικα (Μάγκος, 2013). Για παράδειγμα ο χρήστης παροτρύνεται να δει μια φωτογραφία, ή να κατεβάσει ένα δωρεάν εργαλείο που ωστόσο περιέχει κακόβουλο κώδικα. Συνήθως, ένας δούρειος ίππος δημιουργεί μια κερκόπορτα ή πίσω πόρτα (backdoor) στο σύστημα, στην οποία ο επιτιθέμενος θα μπορέσει αργότερα να συνδεθεί ώστε να διαχειριστεί εξ' αποστάσεως το σύστημα. Ως κερκόπορτα ορίζεται κάθε μυστικό σημείο εισόδου σ' ένα πρόγραμμα, που επιτρέπει σε κάποιον που τη γνωρίζει να αποκτήσει δικαιώματα προσπέλασης στο σύστημα, παρακάμπτοντας τις συνήθεις διαδικασίες ελέγχου προσπέλασης. Τις περισσότερες φορές τα trojans δεν αναπαράγονται και για αυτόν τον λόγο δε χαρακτηρίζονται ως ιοί. Όμως χρησιμοποιούνται ως μέσο μεταφοράς για διάφορες μορφές κακόβουλου λογισμικού (spyware, adware, rootkits, ιούς ή σκουλήκια), οπότε εμπίπτουν στην κατηγορία του πολυμερούς (multipartite) κακόβουλου λογισμικού. Να σημειωθεί βεβαίως ότι η ονομασία παραπέμπει στον Δούρειο Ίππο που χρησιμοποίησαν οι αρχαίοι Έλληνες για να παραπλανήσουν τους Τρώες, κατά τον Τρωικό πόλεμο.

Spyware – Adware: Κακόβουλο λογισμικό με χαρακτηριστικά που εντάσσονται στις λειτουργίες ενός Δούρειου Ίππου (κυρίως ως προς τον τρόπο μόλυνσης), με σκοπό την παρακολούθηση - υποκλοπή ευαίσθητων δεδομένων (spyware), ή την αποστολή ανεπιθύμητων διαφημιστικών μηνυμάτων (adware). Αναφέρονται ως μέλη της ίδιας κατηγορίας, καθώς συνήθως συνεργάζονται για να πετύχουν τον σκοπό τους (πχ. παρακολούθηση της αγοραστικής συμπεριφοράς κατά την περιήγηση στον παγκόσμιο Ιστό και στη συνέχεια εμφάνιση διαφημιστικών μηνυμάτων). Στα ευαίσθητα δεδομένα που στοχεύει το spyware ενδεικτικά περιλαμβάνονται προσωπικά στοιχεία, ονόματα χρήστη, κωδικοί πρόσβασης, κλειδιά, αριθμοί πιστωτικής κάρτας, και λεπτομέρειες συναλλαγών. Η χειρότερη εκδοχή του spyware είναι ως λογισμικό **keylogger** που υποκλέπτει κάθε χαρακτήρα που πληκτρολογεί ο χρήστης και τον προωθεί (στο παρασκήνιο, πχ. μέσω e-mail) σε τρίτους. Συνήθως τα spyware συνεργάζονται με λογισμικά adware ή/και με διαφημιστικές εταιρείες στο Internet, με σκοπό τη δημιουργία ενός προφίλ χρήστη και την αποστολή στοχευμένων διαφημίσεων. Οι παρενέργειες ενός λογισμικού adware ποικίλουν (Μάγκος, 2013): εμφάνιση ανεπιθύμητων μηνυμάτων στο πρόγραμμα περιήγησης (browser), ή στην επιφάνεια εργασίας (desktop), αλλαγή της αρχικής σελίδας του browser (browser hijacking), αλλαγή της αρχικής σελίδας αναζήτησης στο Web, ανακατεύθυνση (redirection) σε πλαστό δικτυακό τόπο (web spoofing), κλπ. Όταν οι παρενέργειες των spyware/adware είναι μικρής κλίμακας, αντί του όρου «κακόβουλο λογισμικό» χρησιμοποιείται ο όρος «ανεπιθύμητο λογισμικό» (potentially unwanted programs, PUPs). Άλλος όρος που χρησιμοποιείται είναι **greyware**, ακριβώς για να εστιάσει στο ότι πρόκειται για λογισμικό το οποίο, ενώ δεν είναι πάντα πλήρως κακόβουλο, έχει μια ύποπτη ή πιθανώς ανεπιθύμητη πτυχή σε αυτό.

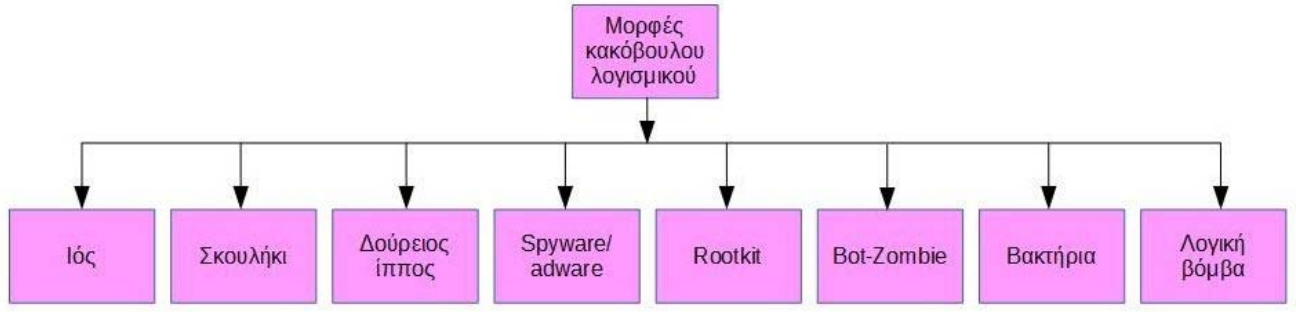
Rootkit: Όπως φαίνεται από την ονομασία τους, ένα rootkit είναι κακόβουλο λογισμικό το οποίο λειτουργεί σε πολύ χαμηλό επίπεδο στο λειτουργικό σύστημα, και συνήθως ενσωματώνει λειτουργίες απόκρυψης (stealth) ώστε να παρακάμπτει τους μηχανισμούς πρόληψης και ανίχνευσης. Αξιοποιεί προνόμια Διαχειριστή (Administrator) με σκοπό ακριβώς την απόκρυψη της παρουσίας του στο σύστημα. Ένα λογισμικό rootkit μπορεί να ανήκει σε οποιαδήποτε από τις παραπάνω κατηγορίες, ωστόσο συνήθως ανοίγει κερκόπορτες που θα επιτρέψουν τη μετέπειτα απομακρυσμένη διαχείριση του μηχανήματος από κάποιον τρίτο. Ο κακόβουλος κώδικας τύπου rootkit (ή αλλιώς τύπου stealth), ενσωματώνει λειτουργίες όπως (Μάγκος, 2013): α) απόκρυψη process (process hiding) – αποκρύπτεται μια διαδικασία αφαιρώντας την από τον πίνακα Διαχείρισης Εργασιών (Task Manager), ή υλοποιείται μια διαδικασία ως ένα σύνολο νημάτων (threads), των οποίων η ανίχνευση είναι δύσκολη, β) απόκρυψη θύρας (port) - αποκρύπτεται η λίστα με τις θύρες που θα χρησιμεύσουν ως «κερκόπορτες» για την απομακρυσμένη διαχείριση του συστήματος και γ) απόκρυψη κλειδιού στο μητρώο (registry) - χρησιμοποιούνται ονόματα κλειδιών που δεν εγείρουν υποψίες (που παραπέμπουν σε «ακίνδυνες» ή χρήσιμες εφαρμογές, πχ. WindowsOS.exe)

Bot – zombie: Κακόβουλο λογισμικό που προσβάλλει υπολογιστές καθιστώντας τους μέλη ενός δικτύου υπολογιστών (botnet) που ελέγχεται εξ' αποστάσεως από τρίτους, με σκοπό την πραγματοποίηση Κατανεμημένων Επιθέσεων Άρνησης Υπηρεσίας (Distributed Denial of Service attacks ή DDOS attacks). Δηλαδή επιθέσεων κατά τις οποίες ένας (συνήθως μεγάλος) αριθμός μολυσμένων υπολογιστών προσπαθεί να συνδεθεί στον υπολογιστή-στόχο μέσω δικτύου προσπαθώντας να τον οδηγήσει σε κατάρρευση: να είναι ανίκανος να λειτουργήσει κανονικά λόγω (κυρίως) του υπερβολικού φόρτου εργασίας για τις αποκρίσεις που καλείται να στείλει καθώς επεξεργάζεται τα πολυπληθή αιτήματα που λαμβάνει. Οι επιθέσεις άρνησης υπηρεσίας είναι επιθέσεις εναντίον της διαθεσιμότητας (availability) του μηχανήματος. Ο όρος «bot»,

προέρχεται από την (Τσεχικής προέλευσης) λέξη «robota» και χρησιμοποιείται για να περιγράψει κάθε είδους αυτοματοποιημένη διαδικασία. Ένας υπολογιστής που έχει μολυνθεί από ένα bot συχνά αναφέρεται ως «zombie». Οι υπολογιστές–zombies μπορεί να χρησιμοποιηθούν για επιθέσεις τύπου 'Άρνησης Υπηρεσίας' (DOS) σε εξυπηρετητές Web, για την αποστολή μηνυμάτων spam, για την πραγματοποίηση επιθέσεων παραπλάνησης (phishing) κλπ.

Βακτήρια (bacteria): Είναι προγράμματα που δεν καταστρέφουν εμφανώς αρχεία. Ο μοναδικός τους σκοπός είναι να πολλαπλασιάζονται. Ένα τυπικό βακτήριο μπορεί να μην κάνει τίποτε περισσότερο από το να τρέχει ταυτόχρονα δύο αντίγραφά του ή πιθανόν να δημιουργεί δύο νέα αρχεία, καθένα απ' τα οποία είναι αντίγραφο του αρχικού αρχείου που περιέχει το βακτήριο. Και τα δύο αυτά προγράμματα μπορούν στη συνέχεια να αντιγράψουν τον εαυτό τους δύο φορές, κοκ. Τα βακτήρια αναπαράγονται εκθετικά και τελικά καταλαμβάνουν όλη τη χωρητικότητα του επεξεργαστή, της μνήμης ή του δίσκου, στερώντας τους πόρους αυτούς από τους χρήστες (επίθεση τύπου 'άρνησης υπηρεσίας').

Λογική βόμβα (logic bomb): Είναι κώδικας ενσωματωμένος σε κάποιο νόμιμο πρόγραμμα εφαρμογής και ρυθμισμένος να «εκραγεί», όταν εκπληρωθούν κάποιες συγκεκριμένες συνθήκες. Παραδείγματα τέτοιων συνθηκών είναι η έλευση μιας συγκεκριμένης μέρας της εβδομάδας ή μιας ημερομηνίας, η παρουσία/απουσία συγκεκριμένων αρχείων, ή η εκτέλεση της εφαρμογής από ένα συγκεκριμένο χρήστη (Ince, 2009; Κάτσικας, 2001). Από τη στιγμή που θα ενεργοποιηθεί, η βόμβα μπορεί να τροποποιήσει ή να διαγράψει δεδομένα ή και ολόκληρα αρχεία, να προκαλέσει το σταμάτημα ενός συστήματος ή να κάνει οποιαδήποτε άλλη ζημιά.



Σχήμα 3.2 *Είδη κακόβουλου λογισμικού*

1.1.2 Σύγχρονο κακόβουλο λογισμικό σε περιβάλλοντα παγκόσμιου Ιστού

Η διείσδυση του παγκόσμιου Ιστού και των υπηρεσιών του, είχε ως συνέπεια τη δημιουργία και την αλματώδη εξάπλωση νέων και πιο ισχυρών μορφών κακόβουλου λογισμικού. Ειδικότερα όσον αφορά τους ιούς, μπορούμε πλέον να διακρίνουμε τους ακόλουθους τύπους (Virus bulletin, 2015; Kura, 2013; Μάγκος, 2013):

Παρασιτικός (parasitic virus): Είναι ο παραδοσιακός (αλλά και πιο διαδεδομένος τύπος ιού). Προσαρτάται σε εκτελέσιμα αρχεία (πχ. αρχεία .exe ή αρχεία .com) και αναπαράγεται, όταν εκτελεστεί το μολυσμένο πρόγραμμα, βρίσκοντας και άλλα εκτελέσιμα αρχεία για να μολύνει. Όταν εκτελεστεί το μολυσμένο πρόγραμμα, εγκαθίσταται συνήθως ως μέρος του λειτουργικού συστήματος και παραμένει στην κύρια μνήμη του συστήματος (**memory-resident**), ώστε να «μολύνει» και άλλα προγράμματα που εκτελεί ο χρήστης. Αντιθέτως, υπάρχουν και (οι λιγότερο βλαπτικοί) μη-παραμένοντες στη μνήμη ιοί (**non memory-resident**) οι οποίοι επειδή δεν εγκαθίστανται στην κεντρική μνήμη, όταν εκτελούνται, σαρώνουν το δίσκο για τους στόχους, τους μολύνουν, και στη συνέχεια σταματά η δράση τους.

Τομέα εκκίνησης (boot virus): Οι ιοί αυτοί μολύνουν τον τομέα εκκίνησης (boot sector) ενός σταθερού ή αφαιρούμενου (removable) αποθηκευτικού μέσου (πχ. του σκληρού δίσκου ή ενός flash drive). Ο τομέας εκκίνησης περιέχει ένα πρόγραμμα μικρού μεγέθους το οποίο το λειτουργικό σύστημα εντοπίζει και «φορτώνει» στην κύρια μνήμη. Τέτοιοι ιοί μπορούν επίσης να μολύνουν την περιοχή MBR (**Master Boot Record**) που περιέχει τον πίνακα κατατμήσεων του δίσκου. Διαδίδονται όταν το σύστημα εκκινήσει από τον δίσκο που περιέχει τον ιό.

Πολυμερής ή Υβριδικός (multipartite, or hybrid virus): Συνδυάζει χαρακτηριστικά δύο ή περισσότερων κατηγοριών. Πχ. χαρακτηριστικά ιών τομέα εκκίνησης και παρασιτικών ιών: ένα μηχάνημα μολύνεται αν χρησιμοποιήσει ένα «μολυσμένο» USB ή αν εκτελέσει ένα μολυσμένο πρόγραμμα. Ο ιός αποτελείται από κώδικα που καλύπτει και τις δύο περιπτώσεις. Οπότε ανάλογα με την περίσταση εκτελείται το αντίστοιχο τμήμα, ενώ αυξάνονται οι πιθανότητες μόλυνσης.

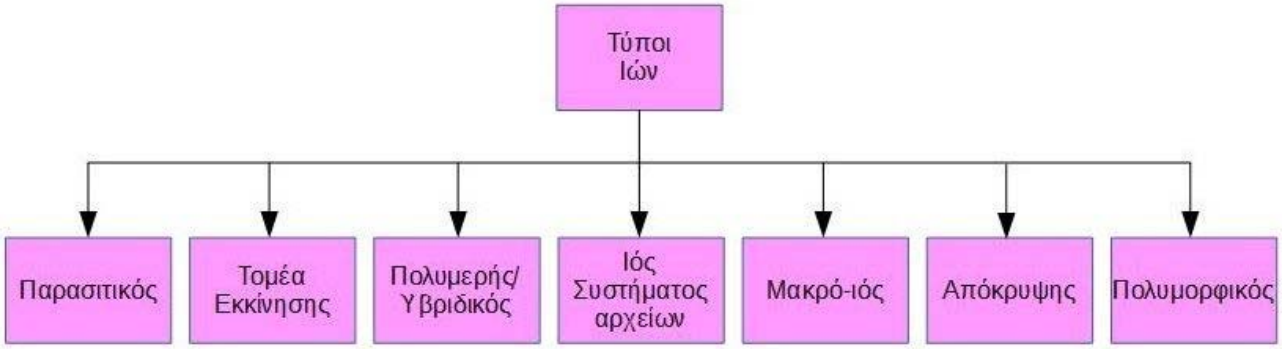
Ιός συστήματος αρχείων (file system virus): Ένας τέτοιος ιός (άλλα ονόματα σε χρήση είναι link virus, ή cluster virus, ή FAT virus) δε συμπεριφέρεται όπως οι παραδοσιακοί ιοί, δηλαδή δε μολύνει τον κώδικα εκτελέσιμων ή άλλων αρχείων. Έχει ωστόσο τη δυνατότητα να παρεμβάλλεται κατά την κλήση ενός προγράμματος και να εκτελεί τον επιβλαβή του κώδικα. Για να το επιτύχει αυτό, τροποποιεί τον πίνακα διευθύνσεων που υπάρχει στον σκληρό δίσκο κάθε υπολογιστή, όπου είναι καταχωρημένη η ακριβής θέση (διεύθυνση) του κάθε αρχείου στον δίσκο του (Μάγκος, 2013). Το λειτουργικό σύστημα χρησιμοποιεί αυτόν τον πίνακα για να οργανώσει τα αρχεία στον δίσκο, καθώς και κάθε φορά που γίνεται κλήση ενός αρχείου. Ο ιός αλλάζει τον πίνακα διευθύνσεων ώστε όταν ζητείται η εκτέλεση ενός «μολυσμένου» προγράμματος, το λειτουργικό σύστημα παραπέμπεται σε μια άλλη θέση όπου βρίσκεται ο κώδικας του ιού, που στη συνέχεια φορτώνεται στη μνήμη και εκτελείται.

Μακρο–ιός (macro virus): Προσβάλλει αρχεία δεδομένων (documents ή emails) που περιέχουν μακροεντολές (macros). Οι μακροεντολές είναι κώδικας εντολών, γραμμένος με εξειδικευμένες γλώσσες μακροεντολών (macro languages). Συνήθως είναι σε μορφή γλώσσας συγγραφής σεναρίων (scripting language). Πχ. VBA (Visual Basic for Applications). Χρησιμοποιούνται κυρίως σε προγράμματα εφαρμογών γραφείου (πχ. Word, Excel, PowerPoint, Outlook, Acrobat) για την αυτοματοποίηση ορισμένων από τις λειτουργίες που εκτελεί ο χρήστης (Aycock, 2006). Οι μακρο-ιοί είναι συνεπώς μακροεντολές που αυτοματοποιούν ένα σύνολο από κακόβουλες (μοχθηρές) ενέργειες. Πχ. όταν σε έναν επεξεργαστή κειμένου εκτελεστεί η μακροεντολή ενός μολυσμένου εγγράφου, ο ιός ενεργοποιείται και απελευθερώνει το καταστροφικό του φορτίο. Αυτός ο τύπος ιών είναι ο κύριος λόγος αύξησης του αριθμού των ιών που εντοπίζονται σε επιχειρηματικά συστήματα. Η δημοτικότητα των εφαρμογών γραφείου, έχει πραγματικά συνεισφέρει στην εξάπλωση αυτού του είδους των ιών. Η επικινδυνότητα του τύπου αυτού οφείλεται κατ' αρχάς στο ότι είναι ανεξάρτητος από πλατφόρμες υλικού. Ο κώδικας που δημιουργείται από μια γλώσσα συγγραφής σεναρίων, μπορεί να εκτελεστεί σε όλες τις πλατφόρμες: ένας μακρο-ιός μπορεί να εκτελεστεί σε ένα PC και σε ένα MAC. Δεύτερο στοιχείο αυξημένης επικινδυνότητας είναι ότι διαδίδονται πολύ εύκολα (με πιο συνηθισμένη μέθοδο διάδοσης το ηλεκτρονικό ταχυδρομείο). Τέλος, επειδή μολύνουν αρχεία δεδομένων (έγγραφα, emails) και όχι εκτελέσιμα προγράμματα, έχουν περισσότερους 'στόχους', αφού η πλειοψηφία της πληροφορίας που εισάγεται σε έναν υπολογιστή είναι σε μορφή τέτοιων αρχείων και όχι σε μορφή εκτελέσιμων προγραμμάτων (Κάτσικας, 2001). Οι περισσότερες μακρο-εντολές ενεργοποιούνται με το άνοιγμα ενός εγγράφου (πχ. λειτουργία auto-open). Για την αναπαραγωγή τους συνήθως οι μακρο-ιοί είναι προγραμματισμένοι να μετατρέπουν τα μολυσμένα έγγραφα σε πρότυπα (templates) ώστε να μολυνθούν όλα τα έγγραφα που θα δημιουργήσει μελλοντικά ο χρήστης.

Απόκρυψης (stealth virus): Ειδικά σχεδιασμένος ώστε να αποφεύγει την ανίχνευση από το αντιβιοτικό λογισμικό. Ένας ιός απόκρυψης χρησιμοποιεί τεχνικές που στοχεύουν την εξαφάνιση των ιχνών του καθώς και των συμπτωμάτων του, παρόμοιες με αυτές που συναντούμε (και ήδη αναφέραμε στην προηγούμενη παράγραφο) στο κακόβουλο λογισμικό τύπου rootkit. Για αυτό και αρκετές φορές ο όρος stealth virus χρησιμοποιείται ως συνώνυμο του rootkit. Μια συνηθισμένη τακτική απόκρυψης είναι η παρεμβολή στις κλήσεις του αντιβιοτικού λογισμικού προς ένα αρχείο (read request intercepts) ώστε να επιστρέφει την 'καθαρή' έκδοσή του, ενώ λίγο αργότερα γίνεται η επαναφορά της μολυσμένης έκδοσης του αρχείου. Μια άλλη ενδεικτική τεχνική απόκρυψης στοχεύει στην υπερπήδηση του ελέγχου ακεραιότητας (integrity checking) που πραγματοποιούν ορισμένα αντιβιοτικά λογισμικά σε όλες τις εφαρμογές του συστήματος (Μάγκος, 2013). Αυτό σημαίνει πως, όταν τροποποιείται ο κώδικας μιας εφαρμογής το αντιβιοτικό λογισμικό ζητάει από τον χρήστη να επιβεβαιώσει την τροποποίηση (πχ. όταν ο χρήστης εγκαθιστά μια επιδιόρθωση, patch, για μια εφαρμογή). Ένας ιός απόκρυψης παραμένει ενεργός στη μνήμη (memory-resident) περιμένοντας την κατάλληλη στιγμή: θα μπορέσει να μολύνει κατ' αυτόν τον τρόπο όσα προγράμματα τροποποιούν τον κώδικα τους κατόπιν μιας καθ' όλα νόμιμης εντολής του χρήστη ή του προγράμματος (πχ. εγκατάσταση μιας αναβάθμισης ή μιας επιδιόρθωσης).

Πολυμορφικός (polymorphic virus): Μαζί με τον προηγούμενο τύπο των ιών απόκρυψης, απαρτίζουν τους καλούμενους δυσανιχνεύσιμους ιούς. Ο πολυμορφικός ιός μεταλλάσσεται με κάθε μόλυνση (αποτελούμενος από σαφώς διαφορετικές ακολουθίες ψηφίων). αλλάζοντας την υπογραφή του και καθιστώντας έτσι αδύνατη την ανίχνευσή του μέσω αυτής (Virus Bulletin, 2015). Για να πετύχει αυτήν τη

διαφοροποίηση, ο ιός μπορεί να εισάγει τυχαίες περιττές εντολές ως θόρυβο ή να αλλάζει τη σειρά εμφάνισης ανεξάρτητων μεταξύ τους εντολών. Όμως, μια πιο αποτελεσματική τακτική είναι να χρησιμοποιήσει κρυπτογράφηση του κώδικά του με ένα συμμετρικό κλειδί που συνεχώς αλλάζει (Κάτσικας, 2001). Πιο συγκεκριμένα, στην τακτική αυτή ένα τμήμα του ιού, που συνήθως ονομάζεται μηχανή μετάλλαξης (mutating engine), δημιουργεί ένα τυχαίο κλειδί κρυπτογράφησης και κρυπτογραφεί τον υπόλοιπο κώδικα του ιού. Το κλειδί αποθηκεύεται μαζί με τον ιό και η μηχανή μετάλλαξης μεταλλάσσεται η ίδια. Όταν κληθεί το μολυσμένο πρόγραμμα, ο ιός χρησιμοποιεί το αποθηκευμένο κλειδί για να αυτοαποκρυπτογραφηθεί. Όταν ο ιός αναπαραχθεί, δημιουργείται νέο κλειδί.



Σχήμα 3.3 *Τύποι ιών*

Πέραν των ιών όμως, το περιβάλλον του παγκόσμιου Ιστού παρέχει πρόσφορο έδαφος και σε κακόβουλο λογισμικό τύπου σκουλήκια, και μάλιστα στην πιο επικίνδυνη τους μορφή, αφού αυτά εκμεταλλεύονται ευπάθειες των δικτυακών εφαρμογών που εκτελούνται σε δικτυωμένους υπολογιστές. Τα πλέον ταχύτερα εξαπλούμενα worms (γνωστά και ως **scanning worms**), εξαπλώνονται μέσω επιθέσεων υπερχείλισης καταχωρητή σε δικτυακές εφαρμογές (Μάγκος, 2013; Ince, 2009). Η υπερχείλιση καταχωρητή (**buffer overflow**) μπορεί να έχει ποικίλα αποτελέσματα: την κατάρρευση μιας εφαρμογής, ή την εκτέλεση κακόβουλου κώδικα με δικαιώματα που είναι ίδια με τα δικαιώματα της εφαρμογής που υπέστη την υπερχείλιση. Οι επιθέσεις αυτές οφείλονται στη λανθασμένη διαχείριση μνήμης από τους προγραμματιστές εφαρμογών: όταν μια ποσότητα πληροφορίας δίνεται ως είσοδος (input) σε ένα πρόγραμμα, το πρόγραμμα θα πρέπει να ελέγξει αν το μέγεθος της τιμής εισόδου είναι μικρότερο ή ίσο από το μέγεθος της μνήμης που έχει δεσμευτεί για συγκεκριμένη μεταβλητή. Εάν, λόγω κακής συγγραφής του κώδικα, κάτι τέτοιο δεν είναι εφικτό, τότε ένας εισβολέας μπορεί να εκτελέσει, μέσω Διαδικτύου, τον κώδικα της αρεσκείας του.

Αυτό γίνεται δίνοντας ως είσοδο μια ποσότητα πληροφορίας (δεδομένα επικάλυψης συν εκτελέσιμο κώδικα) που υπερβαίνει τη μέγιστη ποσότητα που μπορεί να διαχειριστεί το πρόγραμμα. Το λειτουργικό σύστημα τότε μεταφέρει την «περισσευούμενη» πληροφορία σε γειτονικές θέσεις μνήμης, επικαλύπτοντας (overwriting) εκτός των άλλων τα περιεχόμενα του δείκτη που κανονικά περιέχει τη διεύθυνση της επόμενης εντολής του κυρίως προγράμματος που θα εκτελεστεί. Αν η διεύθυνση αυτή παραπέμπει στον κακόβουλο εκτελέσιμο κώδικα του εισβολέα, τότε η μόλυνση είναι επιτυχής. Αν ο δείκτης δείχνει σε περιοχή μνήμης που δεν περιέχει εκτελέσιμο κώδικα, τότε η εφαρμογή καταρρέει, οπότε έχουμε επιτυχή επίθεση διαθεσιμότητας (Denial of Service, DOS). Ένα scanning worm λοιπόν, στέλνει πακέτα κατάλληλου μεγέθους και περιεχομένου στη θύρα της ευπαθούς εφαρμογής, προκαλεί υπερχείλιση, με αποτέλεσμα την εκτέλεση, στο μηχάνημα-στόχος. κώδικα που περιέχει αντίγραφο του εαυτού του.

Μια άλλη σημαντική παράμετρος, έχει σχέση με το ενεργό – δυναμικό περιεχόμενο των σύγχρονων ιστότοπων. Οι περισσότεροι πλέον δικτυακοί τόποι διαθέτουν πλήθος δυνατοτήτων και αλληλεπίδρασης με τον χρήστη. Η λειτουργικότητα μιας δυναμικής ιστοσελίδας οφείλεται στη χρήση γλωσσών συγγραφής σεναρίων (scripting language) με σκοπό τη δημιουργία κώδικα που εκτελείται στην πλευρά του server (πχ. PHP, ASP, κλπ.) ή του client (πχ. Javascript). Οι τεχνολογίες αυτές, πέρα από τη λειτουργικότητα που προσφέρουν, μπορούν να χρησιμοποιηθούν και για την εκτέλεση κακόβουλου λογισμικού στον υπολογιστή του χρήστη. Σχετικό ζήτημα ευπάθειας (vulnerability) είναι και η «αρθρωτή» δομή του κώδικα που συνηθίζεται πλέον σε περιβάλλοντα παγκόσμιου Ιστού: οι περισσότερες εφαρμογές Ιστού / ιστοσελίδες συνίστανται σε ένα πλήθος από ανεξάρτητα τμήματα (components) κώδικα. Η λειτουργικότητα της

εφαρμογής είναι αποτέλεσμα της συνεργασίας μεταξύ των τμημάτων αυτών. Παραδείγματα αρθρωτού κώδικα είναι ακόμη και τα προγράμματα τύπου plug-in που λαμβάνονται και εγκαθίστανται μέσω Διαδικτύου για να βελτιωθεί η λειτουργικότητα των εφαρμογών. Για την αντιμετώπιση κακόβουλου κώδικα που διακινείται μέσω Διαδικτύου, έχουν προταθεί τεχνικές όπως: απομόνωση κώδικα (code isolation – πχ. σε περιβάλλον Java), και η εκτέλεση υπογεγραμμένου κώδικα.

1.2 Άλλες απειλές ασφαλείας στο περιβάλλον του ηλεκτρονικού εμπορίου

Ηλεκτρονικό ψάρεμα (phising): είναι η προσπάθεια παραπλάνησης που γίνεται από κακόβουλους χρήστες προκειμένου να έρθουν στην κατοχή τους ευαίσθητες πληροφορίες με στόχο την αποκόμιση οικονομικών κερδών (Laudon & Traver, 2014). Διαδεδομένη μορφή επίθεσης ψαρέματος είναι τα πλαστά μηνύματα ηλεκτρονικού ταχυδρομείου που ειδοποιούν ανυποψίαστους χρήστες για μια κληρονομιά, ή για μια κλήρωση που έχει γίνει και αυτοί είναι οι 'τυχεροί' ή για μια μεταφορά ενός σημαντικού χρηματικού ποσού στον τραπεζικό λογαριασμό τους κλπ. Στόχος να βρεθούν εύπιστοι χρήστες που θα αποκαλύψουν στοιχεία για τους τραπεζικούς λογαριασμούς τους, ή κωδικούς για την πρόσβαση στους υπολογιστές τους. Η απειλή αυτή δε βασίζεται σε εκτέλεση κακόβουλου κώδικα, αλλά στην εξαπάτηση. Υπάρχει μάλιστα συγκεκριμένος όρος, '**κοινωνική μηχανική**' (*social engineering*) για να περιγράψει τις τεχνικές παραποίησης στοιχείων που στοχεύουν στην εξαπάτηση και οι οποίες προσπαθούν να εκμεταλλευτούν την ανθρώπινη ευπιστία (αλλά και την απληστία…).

Sound 3.1.mp3	Ηχητικό απόσπασμα (audio)
Ηλεκτρονικό ψάρεμα (phising)	

Βανδαλισμοί: καταστροφή σελίδων ή ολόκληρου του ιστότοπου ηλεκτρονικού εμπορίου, σκόπιμη διατάραξη της 'κανονικότητας' των συναλλαγών είναι δυο από τις βασικότερες μορφές κυβερνοβανδαλισμών. Χρήστες με ικανότητα αφενός εντοπισμού των ρηγμάτων ασφαλείας των ιστότοπων και με στόχο αφετέρου την πρόσβαση σε υπολογιστές και δικτυακούς τόπους, χωρίς να το δικαιούνται (οι αποκαλούμενοι χάκερς), και κυρίως όσοι εξ' αυτών έχουν παράνομες προθέσεις (οι αποκαλούμενοι κράκερς), είναι η πηγή των ενεργειών αυτών (Ince, 2009). Πολύ σημαντικός επίσης τύπος κυβερνοβανδαλισμού αποτελεί η '**διαρροή δεδομένων'** ενός οργανισμού, η κλοπή δηλαδή εταιρικών ή και προσωπικών δεδομένων που αυτός περιέχει/διαχειρίζεται, με στόχο την απόκτηση οικονομικών ωφελειών ή απλώς τη δυσφήμιση του οργανισμού ή των χρηστών του (αν πχ. είναι προσωπικές φωτογραφίες διασημοτήτων που αναρτήθηκαν σε λογαριασμούς κοινωνικής δικτύωσης). Πολύ συχνά οι διαρροές δεδομένων οδηγούν σε **κλοπή ταυτότητας**. Με τον όρο αυτόν περιγράφεται η μη εξουσιοδοτημένη χρήση των ευαίσθητων προσωπικών δεδομένων άλλου ατόμου (πχ. κωδικοί πρόσβασης σε μηχανήματα, κωδικοί πρόσβασης σε υπηρεσίες, όπως ο λογαριασμός email του χρήστη, στοιχεία πιστωτικών καρτών, κλπ.) με παράνομα οικονομικά κίνητρα. Μέσω αυτών των πληροφοριών, τις οποίες αποκτούν χρησιμοποιώντας διάφορες τεχνικές που ως τώρα είδαμε (όπως η χρήση κακόβουλου/ανεπιθύμητου κώδικα, το ηλεκτρονικό ψάρεμα), ή τεχνικών που θα δούμε παρακάτω (όπως η κλοπή στοιχείων τραπεζικών καρτών και η παραποίηση), οι παράνομοι χρήστες μπορούν να αγοράζουν προϊόντα και υπηρεσίες χρεώνοντας δικτυακά άλλους. Ειδική μορφή κατηγορίας διαρροής δεδομένων αποτελεί η **κλοπή στοιχείων πιστωτικών και χρεωστικών καρτών** η οποία και οδηγεί στην πολύ σημαντική απειλή ασφαλείας της **απάτης μέσω τραπεζικών καρτών**. Αν και παραδοσιακά οι έρευνες δείχνουν ότι σε πολύ μικρά ποσοστά των συνολικών ηλεκτρονικών συναλλαγών με κάρτες γίνεται κλοπή στοιχείων (κάτω από 1%), για τους χρήστες συνεχώς καταγράφεται η σχετική αυτή απειλή ως ένας από τους σπουδαιότερους ανασταλτικούς παράγοντες για τη χρήση καρτών ως μέσο πληρωμής στις συναλλαγές ηλεκτρονικού εμπορίου. Από την άλλη μεριά, ανησυχητικά στοιχεία είναι ότι αυξάνονται τα περιστατικά που έρχονται στη δημοσιότητα των επιθέσεων και τελικά λεηλασιών εταιρικών διακομιστών που διατηρούν πληροφορίες για αγορές χιλιάδων χρηστών μέσω καρτών, καθώς και ότι η καταγεγραμμένη απάτη μέσω πιστωτικών καρτών είναι διπλάσια στο Διαδίκτυο σε σχέση με τα περιστατικά εκτός Διαδικτύου (Laudon & Traver, 2014).

Προσποίηση ή πλαστογράφηση ή παραπλάνηση ή **παραποίηση ταυτότητας** (spoofing): είναι η απόπειρα απόκρυψης της πραγματικής ταυτότητας ενός χρήστη ή ενός ιστότοπου (ή ενός γενικότερα μηχανήματος) χρησιμοποιώντας μια πλαστή διεύθυνση (e-mail ή URL ή IP). Η πλαστογραφημένη αυτή διεύθυνση προσπαθεί να παραπλανήσει τον χρήστη σχετικά πχ. με το ποιος του έστειλε ένα μήνυμα

ηλεκτρονικού ταχυδρομείου ή το ποιος διακομιστής είναι αυτός με τον οποίο έχει συνδεθεί. Με αυτόν τον τρόπο ο χρήστης, επειδή ακριβώς νομίζει ότι συνομιλεί με μια έμπιστη οντότητα (π.χ. με ένα συνάδελφο από τη δουλειά ή με το μηχάνημα της τράπεζας που συναλλάσσεται), αποκαλύπτει ευαίσθητα στοιχεία, όπως είναι τα στοιχεία της πιστωτικής του κάρτας, τα οποία και βεβαίως γίνονται με αυτόν τον τρόπο κτήμα κακόβουλων χρηστών. Αρκετές φορές η παραποίηση της ταυτότητας ενός μηχανήματος γίνεται μέσω αυτόματης ανακατεύθυνσης (**pharming**) που προκαλεί ένας σύνδεσμος ο οποίος ενώ εμφανίζεται να δείχνει αλλού, τελικά οδηγεί στη διεύθυνση ενός άλλου ιστότοπου (Pfleeger et al., 2015). Έτσι, ο ιστότοπος αυτός προσποιείται και εξαπατά ότι είναι ο σκόπιμος (και αποδεκτός από τον χρήστη) προορισμός, χωρίς βεβαίως να είναι.

Sound 3.2.mp3	Ηχητικό απόσπασμα (audio)
Προσποίηση (spoofing)	

Προγράμματα υποκλοπής (sniffers): είναι προγράμματα που καταγράφουν τα δεδομένα που διακινούνται μέσα σε ένα δίκτυο. Οι διαχειριστές δικτύων τα χρησιμοποιούν νόμιμα για να εντοπίσουν αδυναμίες του δικτύου (πχ. σημεία συμφόρησης). Επίσης και προγραμματιστές τα χρησιμοποιούν όταν χρειαστεί να σχεδιαστεί σε ένα κατανεμημένο σύστημα επεξεργασίας η κίνηση των δεδομένων που παράγεται (Ince, 2009). Όταν όμως χρησιμοποιούνται από χρήστες με παράνομα κίνητρα, αποτελούν ισχυρά εργαλεία υποκλοπής ευαίσθητων δεδομένων. Η εγκατάσταση ενός τέτοιου προγράμματος σε ένα στρατηγικό σημείο ενός δικτύου μπορεί να οδηγήσει στην καταγραφή (και αποστολή σε έναν απομακρυσμένο υπολογιστή) εκατοντάδων συνθηματικών μέσα σε λίγες ώρες. Τα **προγράμματα υποκλοπής συνθηματικών** αποτελούν μια ιδιαίτερα σημαντική απειλή ασφάλειας σε περιβάλλοντα ηλεκτρονικού εμπορίου, καθώς συνδυαζόμενα με τεχνικές παραποίησης οδηγούν σε επιθέσεις τύπου κλοπής ταυτότητας. Οι κακόβουλοι χρήστες χρησιμοποιούν τα προγράμματα υποκλοπής συνθηματικών, μαζί με **προγράμματα ανάλυσης συνθηματικών** (crackers). Τα προγράμματα αυτά επιχειρούν να βρουν την ταυτότητα του χρήστη που αντιστοιχεί σε κάποια συνθηματικά που διατηρούνται στο αρχείο συνθηματικών (password file) ενός διακομιστή. Και αυτά τα προγράμματα έχουν και μη κακόβουλη χρήση: μέσω αυτών οι διαχειριστές ελέγχουν ότι τα συνθηματικά που επέλεξαν οι χρήστες τους είναι αξιόπιστα. Όμως, είναι πολύ διαδεδομένη και η κακόβουλη χρήση τους (πχ. για την απόκτηση πρόσβασης σε συστήματα όπου οι χρήστες είχαν προφανή συνθηματικά όπως 'system' ή 'admin') (Ince, 2009; Pfleeger et al., 2015). Τα προγράμματα ανάλυσης συνθηματικών συνήθως ελέγχουν ιδιότητες ευπάθειας των συνθηματικών (όπως το μικρό μήκος τους ή τη χρήση μόνο γραμμάτων και όχι ψηφίων). Ακόμη, προσπαθούν να ανακαλύψουν ένα συνθηματικό με βάση ένα μεγάλο κατάλογο λέξεων (λεξικό) που συνηθίζονται να χρησιμοποιούνται ως συνθηματικά. Μια παραλλαγή των προγραμμάτων υποκλοπής είναι τα **προγράμματα υποκλοπής emails**. Σε μια μορφή τους, αυτήν που χρησιμοποιούν οι κακόβουλοι χρήστες, είναι ένα τμήμα κρυφού κώδικα που ενσωματώνεται σε ένα email, το οποίο επιτρέπει στον αποστολέα του email να παρακολουθεί όλα τα διαδοχικά mails που προωθούνται με το αρχικό mail. Μια άλλη μορφή τους, η σχετικά νόμιμη, είναι αυτή του προγράμματος-κοριός που εγκαθίσταται από τους παρόχους υπηρεσιών Διαδικτύου στους διακομιστές αλληλογραφίας τους (Laudon & Traver, 2014), ώστε να μπορούν να συλλέγουν τα emails ατόμων όταν αυτό ζητηθεί βάσει της ισχύουσας νομοθεσίας (πχ. αίτημα από την αρχή καταπολέμησης ηλεκτρονικού εγκλήματος).

1.3 Απειλές ασφαλείας λόγω κακής σχεδίασης υποστήριξης 'κινητού' κώδικα

Τα προγράμματα περιήγησης στον παγκόσμιο Ιστό (browsers) έχουν τη δυνατότητα (εκτός αν ο χρήστης αλλάξει τις ρυθμίσεις ασφαλείας τους) να κατεβάζουν και να εκτελούν αυτόματα λογισμικό, με σκοπό τον εμπλουτισμό της εμφάνισης και γενικότερα της συμπεριφοράς του ιστότοπου. Η ικανότητα αυτή αξιοποίησης του λαμβανόμενου κώδικα (downloading code, βλέπε Κεφ.2, παράγραφος 2.2 - Απειλές σε περιβάλλον Internet), αποκαλείται και 'ενεργό περιεχόμενο' (active content). Αρκετά συχνά χρησιμοποιείται αντί του όρου λαμβανόμενος κώδικας, ο όρος '**κινητός κώδικας**' (mobile code) παρόλο που μπορεί να δημιουργήσει μια παρανόηση εμπλέκοντας την έννοια της κινητής συσκευής, με την οποία δεν έχει καμία σχέση. Ο όρος κινητός κώδικας σε περιβάλλοντα παγκόσμιου Ιστού αναφέρεται σε προγράμματα (π.χ., script, μακροεντολές) που μπορεί να αποσταλούν αμετάβλητα από τον διακομιστή και να εκτελεστούν στον υπολογιστή του πελάτη (στο πρόγραμμα περιήγησης) χωρίς ρητή εντολή του χρήστη. Ένα συνηθισμένο παράδειγμα τέτοιου κώδικα είναι ο κώδικας JavaScript που εκτελείται τοπικά από το πρόγραμμα περιήγησης Ιστού. Η υποστήριξη

κινητού κώδικα είναι ένα από τα χαρακτηριστικά που εντατικά προσπαθούν να αξιοποιούν οι ιστότοποι ηλεκτρονικού εμπορίου, στοχεύοντας σε αύξηση της διαδραστικότητάς τους με τους χρήστες, παροχή δυναμικού περιεχομένου σε αυτούς και γενικότερα στην παροχή ποιοτικότερης 'εμπειρίας χρήστη' (user experience). Έτσι όμως, μέσω του κινητού κώδικα, οι μη συνειδητοποιημένοι (ως προς τα προβλήματα ασφαλείας) χρήστες του παγκόσμιου Ιστού, επιτρέπουν την εκτέλεση μη έμπιστου λογισμικού στα μηχανήματά τους (πχ. ιοί, δούρειοι ίπποι, κλπ.). Οι πιο συνηθισμένοι τρόποι αξιοποίησης κινητού κώδικα για κακόβουλες ενέργειες είναι η διατοποθεσιακή δημιουργία δέσμης ενεργειών (cross-site scripting) που θα εξηγήσουμε παρακάτω, η υποστήριξη δυναμικών και ευρείας διάδρασης ιστότοπων, τα συνημμένα του ηλεκτρονικού ταχυδρομείου, και οι λήψεις από αναξιόπιστους ιστότοπους ή λόγω αναξιόπιστου λογισμικού (Stallings, 2014).

Επιπλέον, τα προγράμματα περιήγησης και χωρίς την εκτέλεση κινητού κώδικα, αποτελούν πλέον από μόνα τους ρήγματα ασφαλείας: λόγω του μεγέθους και της πολυπλοκότητάς τους παρουσιάζουν ατέλειες (σημεία ευπάθειας), τις οποίες κακόβουλοι χρήστες αξιοποιούν για να επιτεθούν σε ένα περιβάλλον ηλεκτρονικού εμπορίου, τόσο στο μηχάνημα του πελάτη, όσο και στο μηχάνημα του διακομιστή. Στη δεύτερη βέβαια περίπτωση, χρειάζεται επιπρόσθετα ο εντοπισμός των ευπαθειών στο λογισμικό του διακομιστή του παγκόσμιου Ιστού. Όμως και σε αυτή τη μεριά, αντικρύζουμε παρόμοιες καταστάσεις: το σύγχρονο λογισμικό web server, για να μπορέσει να ανταποκριθεί στις απαιτήσεις υποστήριξης κινητού κώδικα είναι ογκώδες και περίπλοκο, άρα ιδανικό να προσφέρει κενά ασφαλείας προς εκμετάλλευση. Αυτού του τύπου λοιπόν οι απειλές, διακρίνονται στις επιθέσεις που διενεργούνται από την πλευρά του πελάτη (client-side attacks) και αυτές που διενεργούνται από την πλευρά του διακομιστή (server-side attacks).

1.3.1 Επιθέσεις από την πλευρά του πελάτη

Είναι οι επιθέσεις που είτε επωφελούνται από τις αδυναμίες στο λογισμικό που φορτώνεται (mobile code) λόγω του προγράμματος περιήγησης στο μηχάνημα του χρήστη, είτε αυτές που χρησιμοποιούν την εξαπάτηση της κοινωνικής μηχανικής με σκοπό την παραπλάνηση του χρήστη ώστε να 'συνταχθεί' χωρίς να το θέλει με την επίθεση.

Cross-site Scripting (XSS): είναι μια επίθεση που πραγματοποιείται τοποθετώντας κώδικα με τη μορφή μιας γλώσσας σεναρίου (scripting language), πχ. JavaScript, μέσα σε ένα αρχείο ιστοσελίδας (ή σε ένα άλλο αρχείο, πχ. πολυμέσων, από αυτά που μπορεί να 'ερμηνεύει' και παρουσιάσει το πρόγραμμα περιήγησης). Όταν κάποιος χρήστης προβάλλει την ιστοσελίδα (ή το πολυμεσικό αρχείο), εκτελεί αυτόματα το σενάριο (τη δέσμη ενεργειών) του κώδικα και πραγματοποιείται η επίθεση (Stallings, 2014; Stallings, 2014b). Παράδειγμα τέτοιας επίθεσης είναι η αξιοποίηση από έναν κακόβουλο χρήστη του τμήματος σχολίων ενός blog, όπου εισάγει το script επίθεσης (Anders, 2014). Κάθε επισκέπτης του blog που διαβάζει το σχόλιο μέσω του προγράμματος περιήγησης θα εκτελέσει τον κώδικα επίθεσης στο μηχάνημά του.

Cross-site Request Forgery (XSRF): η (διατοποθεσιακή) επίθεση πλαστογραφίας αιτήματος είναι παρόμοια με την επίθεση XSS. Ο επίδοξος εισβολέας τοποθετεί συνδέσμους (links), σε μια ιστοσελίδα με τέτοιον τρόπο ώστε αυτοί να εκτελούνται αυτόματα. Σκοπός είναι μέσω της ανακατεύθυνσης αυτής, να γίνει η εκκίνηση μιας συγκεκριμένης δραστηριότητας σε κάποια άλλη ιστοσελίδα ή εφαρμογή. Ένας τέτοιος σύνδεσμος μπορεί να προκαλέσει για παράδειγμα στο πρόγραμμα περιήγησης την πρόσθεση προϊόντων σε ένα καλάθι αγορών. Επικίνδυνη κατάσταση επίσης είναι η ακόλουθη: κάποιος χρήστης που έχει πιστοποιήσει την ταυτότητά του σε μια έμπιστη ιστοσελίδα, καθώς διατηρεί ταυτόχρονα ανοικτές αρκετές σελίδες στο πρόγραμμα περιήγησης, αν κάποια από αυτές 'φιλοξενεί' επικίνδυνους συνδέσμους τύπου XSRF, τότε ο χρήστης επιτρέπει (χωρίς να το γνωρίζει) την εκτέλεση της επίθεσης στο παρασκήνιο (Anders, 2014).

Click jacking: είναι μια επίθεση που εκμεταλλεύεται τις δυνατότητες γραφικής απεικόνισης του προγράμματος περιήγησης για να παραπλανήσει τον χρήστη να κάνει κλικ σε κάτι που διαφορετικά δε θα επέλεγε να το κάνει. Οι επιθέσεις clickjacking γίνονται τοποθετώντας ένα άλλο στρώμα πάνω από την ιστοσελίδα, ή από τμήματα της ιστοσελίδας, για να μην είναι καθαρό τι πραγματικά κάνουν κλικ. Για παράδειγμα, ο επίδοξος εισβολέας θα μπορούσε να κρύψει ένα κουμπί που λέει "συμφωνώ στην αγορά" κάτω από ένα άλλο στρώμα με ένα κουμπί που λέει απλώς "περισσότερες πληροφορίες".

1.3.2 Επιθέσεις από την πλευρά του διακομιστή

Στην πλευρά του διακομιστή μιας Web συναλλαγής, πολλά προβλήματα και τρωτά σημεία μπορούν να προκύψουν. Πολλά από αυτά εξαρτώνται από το λειτουργικό σύστημα, το λογισμικό διακομιστή Web, τις διάφορες εκδόσεις των scripting γλωσσών και πολλούς άλλους παράγοντες. Υπάρχουν όμως και αρκετοί κοινοί παράγοντες. Πέρα από όλα αυτά, ωστόσο, είναι αρκετοί παράγοντες που ευθύνονται για πολλά θέματα ασφαλείας που είναι κοινά σε όλη τις διάφορες εφαρμογές που ενδέχεται να συναντήσουμε.

Έλλειψη επικύρωσης εισόδου: ένα ενδεικτικό παράδειγμα του τι μπορεί να συμβεί αν λόγω κακού σχεδιασμού δε γίνονται οι κατάλληλοι έλεγχοι επικύρωσης κατά την είσοδο δεδομένων σε φόρμες που βρίσκονται σε ιστότοπους, είναι η έκχυση SQL (SQL injection). Στην περίπτωση των βάσεων δεδομένων που συνδέονται με εφαρμογές ηλεκτρονικού εμπορίου στον Ιστό, η εισαγωγή ειδικά δημιουργημένων δεδομένων σε φόρμες Web, ικανών να αλληλοεπιδρούν με αυτές, μπορεί μερικές φορές να παράγει αποτελέσματα που δεν προβλέπονται από τους προγραμματιστές των εφαρμογών. Το μη σωστό φιλτράρισμα των δεδομένων που εισάγει ένας χρήστης σε μια ιστοσελίδα, μπορεί να γεμίσει τα συστήματα μιας επιχείρησης με κακόβουλο λογισμικό (Anders, 2014; Laudon & Traver, 2014). Σε αρκετές περιπτώσεις, η απομάκρυνση ειδικών χαρακτήρων (πχ. του '*' ή του '%') είναι ένα αποτελεσματικό αντίμετρο σε τέτοιου τύπου επιθέσεις.

Απόδοση ανάρμοστων δικαιωμάτων: στο μηχάνημα του διακομιστή, υπάρχουν ευαίσθητα αρχεία (όπως τα αρχεία ρύθμισης παραμέτρων, configuration files) και φάκελοι αρχείων που η έκθεσή τους σε χρήστες πέραν των υπεύθυνων διαχειριστών μπορεί να δημιουργήσει ζητήματα ασφαλείας. Σε ιστότοπους ηλεκτρονικού εμπορίου, είναι συχνή η ύπαρξη αρχείων ρυθμίσεων που διατηρούν τις πληροφορίες σύνδεσης των εφαρμογών Ιστού με την υποκείμενη βάση δεδομένων. Η μη σωστή διασφάλιση των αρχείων αυτών αποτελεί ουσιαστικά μια διευκόλυνση για όλους τους επίδοξους εισβολείς (Anders, 2014). Να τονιστεί ότι αρκετές φορές δεν είναι ζήτημα κακού σχεδιασμού αλλά αποτέλεσμα σοβαρής αμέλειας: κατά τη μετάβαση ενός διακομιστή Ιστού από την κατάσταση ανάπτυξης λογισμικού στην πλήρη λειτουργία και σύνδεσή του στο Διαδίκτυο, αφήνονται να υπάρχουν (και δεν καθαρίζονται όπως θα έπρεπε) αρχεία που δε συνδέονται με την εύρυθμη λειτουργία του ιστότοπου (πχ. αρχεία πηγαίου κώδικα, αρχεία κειμένου με κρίσιμης σημασίας σημειώσεις κλπ.). Τα αρχεία αυτά αποτελούν πολύτιμο υλικό για όποιον κακόβουλο χρήστη επιθυμεί να παραβιάσει τη σωστή λειτουργία του ιστότοπου.

2. Έλεγχος προσπέλασης και πολιτικές εξουσιοδοτήσεων

Ένα τυπικό σύστημα ελέγχου προσπέλασης περιλαμβάνει υποκείμενα (subjects) που προσπελαύνουν αντικείμενα (objects) διαμέσου κατάλληλων χειρισμών (operations) (Sandhu, 1998). Η θεμελιώδης κατεύθυνση των ελέγχων προσπέλασης, δηλαδή οι αρχές (principles) και οι οδηγίες υψηλού επιπέδου (high level guidelines) που αφορούν τη σχεδίαση και τη διαχείριση των συστημάτων ελέγχου προσπέλασης, προέρχονται από την πολιτική ασφάλειας (security policy) του συστήματος, η οποία και προσπαθεί να δώσει τις κατάλληλες λύσεις σε όλες τις καταγεγραμμένες απαιτήσεις ασφάλειας (Cherdantseva & Hilton, 2013). Υπάρχουν διάφορες πολιτικές, γιατί υπάρχουν διάφορες απαιτήσεις προστασίας. Ο όρος μηχανισμοί ελέγχου προσπέλασης χρησιμοποιείται για όλες τις χαμηλού επιπέδου λειτουργίες λογισμικού και υλικού που μπορούν να διαμορφώνονται κατάλληλα για την υλοποίηση μιας πολιτικής ασφάλειας (Sandhu & Samarati, 1997).

Είναι κοινώς αποδεκτό ότι δεν υπάρχουν "καλύτερες" και "χειρότερες" πολιτικές ελέγχου προσπέλασης. Αυτό οφείλεται στο ότι δεν έχουν όλα τα συστήματα τις ίδιες απαιτήσεις προστασίας. Έτσι, πολιτικές κατάλληλες για ένα δεδομένο σύστημα είναι πολύ πιθανό να είναι ακατάλληλες για κάποιο άλλο. Ως γενική λοιπόν αρχή ισχύει ότι η επιλογή πολιτικής εξαρτάται από τα επιμέρους χαρακτηριστικά του περιβάλλοντος που πρόκειται να προστατευθεί (Γεωργιάδης, 2002). Οι κύριες κατηγορίες πολιτικών ελέγχου προσπέλασης, δηλαδή τα μοντέλα που έχουν προταθεί έως σήμερα και τα οποία συνήθως χρησιμοποιούνται, είναι το κατά-διάκριση (discretionary, DAC) μοντέλο, το κατά-απαίτηση (mandatory, MAC) μοντέλο και το βασισμένο-σε-ρόλους (role-based, RBAC) μοντέλο. Παραδοσιακά, οι πολιτικές ελέγχου προσπέλασης κατατάσσονταν στα πρώτα δυο μοντέλα. Η ανάγκη όμως που θέτει το περιβάλλον του παγκόσμιου Ιστού για υποστήριξη μεγαλύτερης γκάμας εφαρμογών/λειτουργιών, μέσω του συνδυασμού χαρακτηριστικών και από τις δυο κατηγορίες ή ακόμη και μέσω της εισαγωγής νέων, οδήγησε στην εμφάνιση νέων μοντέλων που είτε ήδη απολαμβάνουν ευρείας αναγνώρισης, όπως το μοντέλο RBAC, είτε - εστιάζοντας στην ανάγκη αξιοποίησης των 'συναφών' πληροφοριών (context) - βρίσκονται στο αρχικό ακόμη στάδιο της ανάπτυξής τους.

2.1 Κατά-διάκριση έλεγχος προσπέλασης (μοντέλο DAC)

Το μοντέλο DAC αφήνει την ευθύνη ανάθεσης και ανάκλησης των προνομίων ελέγχου προσπέλασης στη διακριτική ευχέρεια μεμονωμένων χρηστών, οι οποίοι και καλούνται «κάτοχοι» (owners) των αντικειμένων που έχουν υπό τον έλεγχό τους (Gollmann, 2011). Η κατοχή συνήθως αποκτάται ως επακόλουθο της δημιουργίας των αντικειμένων. Ο περιορισμός της προσπέλασης σε αντικείμενα, βασίζεται στην ταυτότητα των υποκειμένων και/ή στις ομάδες όπου αυτά ανήκουν. Στο μοντέλο DAC, όλα τα υποκείμενα και τα αντικείμενα του συστήματος απαριθμούνται, ενώ διευκρινίζονται και όλες οι εξουσιοδοτήσεις προσπέλασης για κάθε υποκείμενο και κάθε αντικείμενο. Αναλυτικότερα, οι βασικές έννοιες είναι οι εξής:

- το σύνολο των υποκειμένων ασφάλειας (security subjects) S,
- το σύνολο αντικειμένων ασφάλειας (security objects) O,
- το σύνολο προνομίων προσπέλασης (access privileges) T, που προσδιορίζουν το είδος της προσπέλασης την οποία μπορεί να έχει ένα υποκείμενο σε ένα συγκεκριμένο αντικείμενο,
- το σύνολο από προϋποθέσεις (predicates) P, που παριστάνουν κανόνες προσπέλασης οι οποίοι βασίζονται (κυρίως, αλλά όχι μόνο πλέον) στο περιεχόμενο των δεδομένων.

Στη περίπτωση αντίστοιχα που στην υποδομή μιας εφαρμογής ηλεκτρονικού εμπορίου βρίσκεται μια σχεσιακή βάση δεδομένων για υποστήριξη Web-based λειτουργιών, το O είναι ένα πεπερασμένο σύνολο τιμών {o1, o2, …, on}, που παριστάνουν ένα σχεσιακό σχήμα (πχ. πίνακες δεδομένων ή όψεις αυτών), και το S είναι ένα πεπερασμένο σύνολο πιθανών υποκειμένων {s1, s2, …, sm} που παριστάνουν χρήστες, ομάδες από χρήστες, ή κινήσεις συναλλαγών (transactions) που ενεργούν εκ μέρους των χρηστών. Τα προνόμια προσπέλασης T είναι το σύνολο των λειτουργιών της βάσης δεδομένων, όπως επιλογή (select), εισαγωγή (insert), διαγραφή (delete), ενημέρωση (update), εκτέλεση (execute), ανάθεση (grant) ή ανάκληση (revoke) προνομίου. Οι προϋποθέσεις P καθορίζουν το παράθυρο προσπέλασης (access window) του υποκειμένου s στο αντικείμενο o (Khair, 1996). Μια τετράδα της μορφής <s, o, t, p> αποτυπώνει τότε έναν κανόνα προσπέλασης (access rule). Κάθε αίτηση χρήστη για προσπέλαση ενός αντικειμένου, αντιπαραβάλλεται με τους προκαθορισμένους κανόνες προσπέλασης. Αν υπάρχει μια εξουσιοδότηση κατάλληλη, η πρόσβαση επιτρέπεται, αλλιώς η αίτηση απορρίπτεται.

2.1.1 Σύνολα χρηστών (User groups)

Το μοντέλο DAC δεν ασχολείται μόνο με ποια μεμονωμένα υποκείμενα μπορούν να προσπελάσουν ποια αντικείμενα, αλλά και με τη συμπεριφορά των ομάδων και των συνόλων υποκειμένων. Σε περιπτώσεις που ένας χρήστης ανήκει σε περισσότερες από μία ομαδοποιήσεις χρηστών, το μοντέλο ελέγχου προσπέλασης μπορεί να καθορίζει ότι το υποκείμενο μπορεί εναλλακτικά:

- να ενεργεί με την ένωση των προνομίων όλων των ομάδων στις οποίες ανήκει,
- να ενεργεί με τα προσωπικά του προνόμια και τα προνόμια μόνο μίας ομάδας κάθε φορά,
- να επιλέξει μεταξύ των προνομίων χρήστη και των προνομίων μίας από τις ομάδες στις οποίες ανήκει,
- να ακολουθεί κάποια άλλη πολιτική.

Η δεύτερη και η τρίτη από τις παραπάνω πολιτικές υποστηρίζουν τη γνωστή προσέγγιση του «ελάχιστου προνομίου» (least privilege). Οι διαχειριστές έχουν τη δυνατότητα της ομαδοποίησης των χρηστών που λειτουργούν παρόμοια, καθώς και την ανάθεση εξουσιοδοτήσεων σε σχέση με τις ομάδες. Το γεγονός αυτό βελτιώνει την αποδοτικότητα του συστήματος, γιατί ο αριθμός των κανόνων εξουσιοδότησης που πρέπει να διαχειρίζονται μειώνεται δραματικά. Επίσης, δίνεται η δυνατότητα της εφαρμογής καθιερωμένων πολιτικών στους καθορισμούς των ομάδων. Για παράδειγμα, όλοι οι νοσηλευτές κατά τη βάρδια τους ενεργούν παρόμοια, οπότε είναι δυνατό να τους αποδοθεί ένα σύνολο κοινών δικαιωμάτων.

Ο συνολικός αριθμός ομάδων είναι σε λογικά πλαίσια μικρός, γιατί ο αριθμός των διαφορετικών λειτουργιών σε έναν οργανισμό δεν είναι πολύ μεγάλος. Οι ομάδες χρήστη είναι ακόμη μία εφαρμογή της έννοιας της υπονοούμενης εξουσιοδότησης (implied authorization), όπου ένας κανόνας εξουσιοδότησης

καθορίζεται βάσει κάποιας σύνθετης δομής (υποκειμένων ή αντικειμένων προσπέλασης) και αποδίδει παρόμοια δικαιώματα σε κάθε συνθετικό στοιχείο (Γεωργιάδης, 2002).

2.1.2 Μορφές Μηχανισμών Ασφάλειας DAC

Οι παραδοσιακοί κατά-διάκριση μηχανισμοί ασφαλείας βασίζονται στη μορφή της λίστας ελέγχου προσπέλασης (access control list). Αυτοί οι μηχανισμοί έχουν την τάση να περιορίζουν τις πολιτικές ελέγχου προσπέλασης στις ικανότητες των μηχανισμών. Μπορούν εύκολα να υλοποιηθούν χρησιμοποιώντας το μοντέλο του πίνακα προσπέλασης (access matrix model, ACM) των Graham-Denning. Σύμφωνα με αυτό, οι κανόνες προσπέλασης διατηρούνται σε ένα πίνακα προσπέλασης, όπου οι γραμμές αναπαριστούν τα υποκείμενα, οι στήλες τα αντικείμενα και τα στοιχεία στην τομή τους περιέχουν τους σχετικούς τρόπους προσπέλασης. Μια παραλλαγή του, το HRU (Harrison, Ruzzo and Ullman) ACM μοντέλο, αποτελεί έναν από τους αξιοσημείωτους εκπροσώπους των μοντέλων DAC. Χρησιμοποιεί ένα σύνολο εντολών το οποίο μπορεί να δημιουργεί και να μεταβάλλει το όλο σχήμα των εξουσιοδοτήσεων. Στο μοντέλο HRU, η επιτευχθείσα ασφάλεια αποτελεί στη γενική περίπτωση ένα όχι πλήρως εκτιμήσιμο μέγεθος (undecidable). Το πρόβλημα αυτό (safety problem) ορίζεται ως η δυνατότητα προσδιορισμού για ένα συγκεκριμένο υποκείμενο να φτάσει να κατέχει ένα συγκεκριμένο προνόμιο, το οποίο δεν κατείχε προηγουμένως.

Αρκετές νέες προσεγγίσεις έχουν προταθεί, για τις οποίες θεωρητικά το προηγούμενο πρόβλημα μπορεί να διευθετηθεί. Οι περισσότερες από αυτές βασίζονται στην έννοια του τύπου ασφάλειας (security type) και περιλαμβάνουν το Schematic Protection Model (SPM), το Typed Access Matrix (TAM) μοντέλο και το Dynamically Typed Access Control (DTAC) μοντέλο. Ενώ τα SPM και TAM υποστηρίζουν τύπους υποκειμένων και αντικειμένων, το DTAC δεν κάνει διακρίσεις ανάμεσα στα υποκείμενα και τα αντικείμενα. Αυτό του το στοιχείο το κάνει περισσότερο κατάλληλο για περιβάλλοντα εναλλαγής των ρόλων υποκειμένων και αντικειμένων, όπως αυτά του παγκόσμιου Ιστού. Δυναμικοί έλεγχοι και στατικές αναλύσεις των διάφορων απόψεων της ασφάλειας του συστήματος, χρησιμοποιούνται για τη διατήρηση ενός σταθερού επιπέδου ασφάλειας. Αυτό το χαρακτηριστικό, κάνει το DTAC ικανό να μοντελοποιήσει μηχανισμούς ασφάλειας βασισμένους-σε-δραστηριότητες. Ομαδοποιώντας οντότητες σε τύπους, το μοντέλο αυτό μπορεί να μειώσει το μέγεθος των πληροφοριών διαμόρφωσης της ασφάλειας και μπορεί επίσης να διευρύνει τις διαχειριστικές λειτουργίες. Όλες αυτές οι επεκτάσεις έχουν αντικειμενικό σκοπό να διευρύνουν την γκάμα των εφαρμογών που μπορούν να υποστηριχθούν από τα βασισμένα-στο-ACM μοντέλα.

2.1.3 Εξουσιοδοτήσεις για τις όψεις (authorizations for views)

Το κατά-διάκριση μοντέλο υποστηρίζεται από τα περισσότερους τύπους συστημάτων βάσεων δεδομένων και βασίζεται στην έννοια των "όψεων" (database views). Οι όψεις είναι οι διαφορετικοί τρόποι απεικόνισης των δεδομένων της βάσης, οι οποίοι προκύπτουν ως αποτέλεσμα της εκτέλεσης ερωτημάτων. Αποκαλούνται και εικονικές σχέσεις, σε αντίθεση με τις υπαρκτές βασικές σχέσεις, δηλαδή τους πίνακες δεδομένων της βάσης. Οι όψεις χρησιμοποιούνται ως μέσο προσαρμογής των βασικών σχέσεων σε συγκεκριμένες ανάγκες, αποσκοπώντας ουσιαστικά στο να προσδώσουν μια "προσωπικού χαρακτήρα" εικόνα-ερμηνεία της βάσης δεδομένων στους χρήστες της.

Οι όψεις μπορούν να αποτελέσουν και βάση μηχανισμών προστασίας. Πρόκειται για την αποκαλούμενη βασισμένη-σε-όψεις προστασία (view-based protection). Ένας χρήστης μπορεί να είναι εξουσιοδοτημένος για μία όψη της βάσης, χωρίς να έχει εξουσιοδότηση για τον πίνακα (ή τους πίνακες) στον οποίο βασίζεται η όψη. Έτσι, η ανάθεση εξουσιοδότησης σε μία όψη αποτελεί ένα μέσο περιορισμού της εξουσιοδότησης σε ένα υποσύνολο των δεδομένων που περιέχονται στους πίνακες δεδομένων. Οι μη υλοποιημένες αυτές ερωτήσεις που βασίζονται στις βασικές σχέσεις καλούνται σχέσεις όψεων (view relations). Οι περιορισμοί ασφάλειας υλοποιούνται μέσω της ενσωμάτωσης κατάλληλων προϋποθέσεων κατά τον ορισμό των όψεων της βάσης δεδομένων. Έτσι, οι στήλες του πίνακα ελέγχου προσπέλασης, αναπαριστούν τις διαθέσιμες όψεις της βάσης.

Ένας άλλος τρόπος εκμετάλλευσης των όψεων σε ζητήματα ασφαλείας, είναι μέσω της προσέγγισης της τροποποίησης των ερωτήσεων (query modification). Στην περίπτωση αυτή, το ερώτημα της αίτησης προσπέλασης του χρήστη μεταβάλλεται μέσω της προσθήκης κατάλληλων προϋποθέσεων που προσδιορίζουν τις απαιτήσεις ασφάλειας.

2.1.4 Μειονεκτήματα του μοντέλου DAC

Παρά τη μεγάλη ευελιξία του κατά-διάκριση μοντέλου, υπάρχει και μια έμφυτη αδυναμία, γνωστή ως το πρόβλημα της αντιγραφής (copy problem). Δηλαδή, οι πληροφορίες μπορεί να αντιγραφούν από το ένα αντικείμενο σε ένα άλλο, ούτως ώστε να καθίσταται δυνατή η πρόσβαση στο αντίγραφο ακόμη και αν ο κάτοχος του πρωτότυπου δεν παρέχει δικαίωμα προσπέλασης στο πρωτότυπο. Τέτοια αντίγραφα μάλιστα, μπορούν να πολλαπλασιάζονται από κακόβουλο λογισμικό, όπως οι Δούρειοι Ίπποι (trojan horses) που αναφέρθηκαν σε προηγούμενη παράγραφο, χωρίς να είναι απαραίτητη η ρητή συνεργασία των χρηστών που διαθέτουν νόμιμες εξουσιοδοτήσεις για τα πρωτότυπα. Τα μοντέλα τύπου DAC δεν προσφέρουν λοιπόν επαρκείς τρόπους διασφάλισης της ροής των πληροφοριών, αφού στην πραγματικότητα καθιστούν τους χρήστες υπεύθυνους για την επιβολή της πολιτικής ασφάλειας, πράγμα που αποτελεί σοβαρότατο ρήγμα ασφάλειας.

Ειδικότερα κατά την εφαρμογή του μοντέλου DAC σε βάσεις δεδομένων, ο έλεγχος της μετάδοσης των εξουσιοδοτήσεων αποτελεί ένα αρκετά σύνθετο πρόβλημα. Έρευνες στον χώρο, καταλήγουν στο ότι οι μηχανισμοί απόδοσης αρνητικών εξουσιοδοτήσεων (negative authorizations), η υποστήριξη των μη-διαδοχικών ανακλήσεων (non-cascading revoke) και η χρήση εξουσιοδοτήσεων χρονοσήμανσης (time-stamped authorizations) μπορούν να υλοποιήσουν (κατά περίπτωση) ευέλικτες πολιτικές επίλυσης των αντικρουόμενων προνομίων προσπέλασης.

2.2 Κατά-απαίτηση έλεγχος προσπέλασης (μοντέλο MAC)

Το μοντέλο MAC παρέχει τα μέσα για τον περιορισμό της προσπέλασης σε αντικείμενα, με βάση τόσο την ευαισθησία (sensitivity) της πληροφορίας που περιέχεται σε αυτά, όσο και την εμπιστευτικότητα (clearance) των υποκειμένων που επιθυμούν να τα προσπελάσουν (Gollmann, 2011). Ετικέτες (labels) ασφάλειας αποδίδονται και στα δεδομένα και στους χρήστες, για την αποτύπωση αντίστοιχα της ευαισθησίας και της εμπιστευτικότητάς τους. Όπως θα περιγραφεί αναλυτικά σε επόμενη παράγραφο, τα κατά-απαίτηση μοντέλα μπορούν να εγγυηθούν μια συγκεκριμένη κατεύθυνση στη ροή των πληροφοριών: με βάση το δικτυωτό (lattice) των ετικετών ασφάλειας, είναι σε θέση να αποτρέπουν τη ροή των πληροφοριών χαμηλής ακεραιότητας προς αντικείμενα υψηλότερης ακεραιότητας, καθώς και τη ροή πληροφοριών υψηλής εμπιστευτικότητας προς αντικείμενα χαμηλότερης εμπιστευτικότητας. Επίσης, φροντίζει ώστε να προστατεύονται τα δεδομένα όχι μόνο από τις άμεσες προσπελάσεις των μη εξουσιοδοτημένων χρηστών, αλλά και από τις έμμεσες διαρροές πληροφοριών, οι οποίες συμβαίνουν είτε μέσω των καναλιών επικάλυψης σημάτων (covert signaling channels), είτε λόγω της έμμεσης προσπέλασης από εξαγωγή συμπερασμάτων (inference) (Gollmann, 2011; Khair, 1996).

2.2.1 Γενική περιγραφή του MAC

Οι κατά-απαίτηση πολιτικές ελέγχου προσπέλασης χρησιμοποιούνται όταν σε ένα σύστημα περιέχονται πληροφορίες με ποικιλία διαβαθμίσεων ασφάλειας (security classifications), και υπάρχουν χρήστες οι οποίοι δεν είναι εξουσιοδοτημένοι για την ανώτατη διαβάθμιση των πληροφοριών που περιέχονται στο σύστημα. Η βασική φιλοσοφία του MAC έχει τις ρίζες της στα στρατιωτικά περιβάλλοντα, όπου είναι κοινή πρακτική η κατηγοριοποίηση των χρηστών και των πληροφοριών (πχ. αδιαβάθμητο, εμπιστευτικό, απόρρητο κλπ.) σε διάφορα επίπεδα ασφαλείας. Για αυτόν τον λόγο χρησιμοποιείται και ο όρος military ως συνώνυμο του mandatory στον έλεγχο προσπέλασης.

Η προσπέλαση πληροφοριών περιορίζεται από την αρχή της αναγκαίας γνώσης. Δηλαδή, η προσπέλαση σε ευαίσθητα δεδομένα επιτρέπεται μόνο σε εκείνα τα υποκείμενα που χρειάζεται να γνωρίζουν αυτά τα δεδομένα προκειμένου να εκτελέσουν τις εργασίες τους. Κάθε ευαίσθητη πληροφορία μπορεί να συσχετιστεί με ένα ή περισσότερα έργα που καλούνται υποδιαιρέσεις (compartments) ή κατηγορίες (categories) και οι οποίες περιγράφουν την αντικειμενική σημασία της πληροφορίας. Μια επέκταση τότε της αρχής need-to-know, η οποία ακολουθεί τους παρακάτω κανόνες, μπορεί να χρησιμοποιηθεί για να καλύψει τη γενική αρχή των ζητημάτων ροής πληροφοριών:

- Το s επιτρέπεται να «διαβάσει» το o, εάν categories(o) $\subseteq$ need-to-know(s)
- Το s επιτρέπεται να «γράψει» στο o, εάν need-to-know(s) $\subseteq$ categories(o)

Τα μοντέλα τύπου MAC κάνουν χρήση της προηγούμενης αρχής, συνδυάζοντάς τη με τις δυνατότητες κατάταξης των χρηστών σε επίπεδα εξουσιοδότησης. Έτσι, η βασική ορολογία στους κατά-απαίτηση ελέγχους προσπέλασης είναι:

- Διαβάθμιση (classification): Η διαβάθμιση των πληροφοριών που πρέπει να προστατευθούν αντανακλά την πιθανή ζημιά που θα μπορούσε να προκληθεί από τη μη εξουσιοδοτημένη διαρροή των πληροφοριών.
- Βαθμός εξουσιοδότησης (clearance): Ο βαθμός εξουσιοδότησης που αντιστοιχεί σε ένα χρήστη αντικατοπτρίζει την αξιοπιστία του να μη διαρρεύσουν πληροφορίες σε μη έμπιστα άτομα.

Τα μοντέλα MAC προϋποθέτουν την απόδοση συγκεκριμένων ετικετών ασφάλειας στα αντικείμενα και τα υποκείμενα. Η ετικέτα ενός αντικειμένου ο καλείται τάξη του αντικειμένου (class(o)) και η ετικέτα ενός αντικειμένου s καλείται εξουσιοδότηση του υποκειμένου (clear(s)). Μια ετικέτα ασφάλειας S αποτελείται από δύο στοιχεία:

- ένα επίπεδο ασφάλειας (L), από μια πλήρως διατεταγμένη λίστα τάξεων προσπέλασης. Πχ. άκρως απόρρητο (top secret) > απόρρητο (secret) > εμπιστευτικό (confidential) > αδιαβάθμητο (unclassified)
- ένα μέλος (C) ενός μη ιεραρχημένου συνόλου κατηγοριών (categories set), που αντιπροσωπεύουν γενικά τάξεις των τύπων των αντικειμένων.

Τα επίπεδα διαβάθμισης δεδομένων και βαθμού εξουσιοδότησης χρηστών, είναι πλήρως διατεταγμένα σύνολα, ενώ οι ετικέτες που προκύπτουν από αυτά είναι μερικώς διατεταγμένα, λόγω των συνόλων κατηγοριών. Καλείται δικτυωτό (lattice), το πλέγμα που σχηματίζεται από το σύνολο όλων των ετικετών ασφάλειας, στο οποίο δεν είναι συγκρίσιμες μεταξύ τους όλες οι ετικέτες. Μια ετικέτα S1 = (L1, C1) κυριαρχεί (dominates) μιας άλλης ετικέτας S2 = (L2, C2), δηλαδή ισχύει S1 $\subseteq$ S2, αν και μόνον εάν ισχύει L1 $\geq$ L2 και C2 $\geq$ C1.

2.2.2 Το μοντέλο εμπιστευτικότητας Bell-La Padula

Οι απαιτήσεις εμπιστευτικότητας του κατά-απαίτηση ελέγχου προσπέλασης εκφράζονται συχνά μέσω του μοντέλου των Bell και LaPadula (BLP), το οποίο περιγράφεται από τους εξής κανόνες (Pfleeger et al., 2015):

1. Η απλή ιδιότητα (simple property), η οποία προστατεύει τα δεδομένα από μη εξουσιοδοτημένη ανάγνωση (no read-up rule). Σύμφωνα με αυτή το υποκείμενο s επιτρέπεται να διαβάσει τα δεδομένα ο, μόνον εάν clear(s) $\subseteq$ class(o)
2. Η ιδιότητα αστερίσκου (*-property), η οποία προστατεύει τα δεδομένα από νοθεύσεις ή μη εξουσιοδοτημένες μετατροπές περιορίζοντας τη ροή των πληροφοριών από τα ανώτερα προς τα κατώτερα επίπεδα (no write-down rule). Σύμφωνα με αυτή στο υποκείμενο s επιτρέπεται να γράψει το δεδομένο ο, μόνον εάν class(o) $\geq$ clear(s).

Ο στόχος του μοντέλου είναι ο προσδιορισμός επιτρεπτών επικοινωνιών όταν βασικό μέλημα είναι η διατήρηση της εμπιστευτικότητας. Όμως παρουσιάζονται προβλήματα ακεραιότητας, επειδή σε έναν χρήστη "από χαμηλά" επιτρέπεται να εισάγει μη-ορθά ενδεχομένως δεδομένα. Για την αντιμετώπιση αυτού του φαινομένου, οι μηχανισμοί του μοντέλου BLP συνδυάζονται με αυτούς που παρέχει το ακόλουθο μοντέλο Biba.

2.2.3 Το μοντέλο ακεραιότητας Biba

Πρόκειται για ένα μοντέλο πρόληψης των μη εξουσιοδοτημένων μεταβολών των δεδομένων, δίδυμο του BLP μοντέλου. Σε αναλογία με τα επίπεδα εμπιστευτικότητάς του BLP, το μοντέλο Biba καθορίζει επίπεδα ακεραιότητας. Υποκείμενα και αντικείμενα ταξινομούνται με ένα σχήμα κατάταξης με σκοπό την

ακεραιότητα των δεδομένων. Σημειώνοντας ως I(s) και I(o) τις ετικέτες ακεραιότητας αντίστοιχα των υποκειμένων και των αντικειμένων, οι κανόνες που επιβάλει το μοντέλο Biba είναι (Pfleeger et al., 2015):

1. Η απλή ιδιότητα ακεραιότητας (simple integrity property), η οποία προστατεύει τα δεδομένα από μη εξουσιοδοτημένη μεταβολή (no write-up rule). Σύμφωνα με αυτή το υποκείμενο s επιτρέπεται να μεταβάλλει το αντικείμενο o, μόνον εάν $I(s) \geq I(o)$.
2. Η ιδιότητα αστερίσκου ακεραιότητας (integrity *-property), καλείται και no read-down rule. Σύμφωνα με αυτή, εάν ένα υποκείμενο s, έχει δικαίωμα ανάγνωσης σε ένα αντικείμενο o, τότε το s επιτρέπεται να μεταβάλλει ένα άλλο αντικείμενο k, μόνον εάν $I(o) \geq I(k)$.

Οι ιδιότητες αυτές εκφράζουν δυο βασικές αλήθειες αναφορικά με την αξιοπιστία των πληροφοριών. Η πρώτη αφορά την περίπτωση που ένα αναξιόπιστο υποκείμενο μεταβάλλοντας ένα αντικείμενο, ουσιαστικά προκαλεί μείωση της ακεραιότητας αυτού του αντικειμένου. Η δεύτερη αφορά το ότι χαμηλού επιπέδου ακεραιότητα ενός αντικειμένου, συνεπάγεται και χαμηλή ακεραιότητα για κάθε άλλο αντικείμενο που βασίζεται σε αυτό. Το μοντέλο Biba αγνοεί τις απαιτήσεις εμπιστευτικότητας, όμως η τάση είναι να αντιμετωπίζονται ταυτόχρονα τα ζητήματα μυστικότητας και ακεραιότητας μέσω του συνδυασμού των μοντέλων BLP και Biba.

2.2.4 Έλεγχος προσπέλασης πολλαπλών επιπέδων

Η υποστήριξη κατά-απαίτηση ελέγχων προσπέλασης στις βάσεις δεδομένων, οδηγεί στις βάσεις πολλαπλών επιπέδων ασφάλειας (multi-level security, MLS). Σε αυτές, οι πίνακες δεδομένων είναι δυνατόν να εμφανίζονται διαφορετικοί σε χρήστες με διαφορετικούς βαθμούς εξουσιοδότησης. Αυτό οφείλεται στο ότι συνήθως όλοι οι βαθμοί εξουσιοδότησης δεν επιτρέπουν σε όλα τα υποκείμενα να έχουν προσπέλαση σε όλα τα αντικείμενα. Πραγματικά, η ανάγκη για μια πολιτική ελέγχου προσπέλασης πολλαπλών επιπέδων, προκύπτει όταν η βάση περιέχει πληροφορίες με διαφορετικούς βαθμούς διαβάθμισης (εμπιστευτικότητας) και οι χρήστες κατατάσσονται σε διαφορετικούς βαθμούς εξουσιοδότησης. Σε συστήματα με υψηλές ανάγκες ασφάλειας, η προστασία των εμπιστευτικών πληροφοριών οφείλει να αφορά όχι μόνο τις προσπάθειες άμεσης προσπέλασης, αλλά και αυτές της έμμεσης, καλύπτοντας όλες τις πιθανές ροές πληροφοριών. Οι MLS βάσεις δεδομένων διαθέτουν τους κατάλληλους μηχανισμούς για την επίτευξη αυτών των στόχων.

Η απλούστερη τεχνική αποκάλυψης πληροφοριών από εξουσιοδοτημένο χρήστη είναι η ανάκτηση τους, η αντιγραφή τους σε ένα αντικείμενο που του ανήκει και η διάθεση του αντίγραφου σε άλλα άτομα. Για να αποφευχθεί αυτό, είναι απαραίτητο να ελέγχεται η ικανότητα του εξουσιοδοτημένου χρήστη να δημιουργεί αντίγραφα. Δηλαδή για παράδειγμα, όταν μία κίνηση έχει ολοκληρώσει μια προσπάθεια ανάγνωσης, το σύστημα προστασίας πρέπει να εξασφαλίζει ότι δεν έγινε εγγραφή σε κατώτερο επίπεδο ασφάλειας από τον χρήστη που είναι εξουσιοδοτημένος να εκτελεί κίνηση ανάγνωσης. Σε μια MLS βάση, οι έλεγχοι εγγραφής και ανάγνωσης που εφαρμόζονται βασίζονται στους κατά-απαίτηση κανόνες των μοντέλων BLP και Biba, οπότε όπως ήδη αναφέρθηκε, μπορεί να ελεγχθεί αποτελεσματικά η ροή των πληροφοριών μεταξύ των υποκειμένων με διαφορετικούς βαθμούς εξουσιοδότησης.

Ο κατά-απαίτηση έλεγχος προσπέλασης στηρίζεται στις ετικέτες ασφάλειας των δεδομένων, για αυτό και καλείται και βασισμένος-σε-ετικέτες (label-based) έλεγχος προσπέλασης. Υπάρχουν δυο τρόποι εκχώρησης επιπέδων διαβάθμισης στα δεδομένα. Ο πρώτος απονέμει τα επίπεδα εμπιστευτικότητας ανά πίνακα δεδομένων. Η προσέγγιση αυτή παρουσιάζει το μικρότερο δυνατό επίπεδο διακριτότητας, για αυτό και τις περισσότερες φορές οδηγεί σε μη ευέλικτες λύσεις. Ο δεύτερος τρόπος διαβαθμίζει τα δεδομένα, εξετάζοντάς τα ανάλογα με το επιθυμητό επίπεδο λεπτομέρειας σε επίπεδο εγγραφής (tuple), στήλης (attribute), ή ακόμη και μεμονωμένου στοιχείου. Ένα παράδειγμα πολλαπλών επιπέδων σχέσης (πίνακα δεδομένων) είναι:

$R (a_1, c_1, a_2, c_2, a_3, a_4, c_{34}, a_5, c_5, c_t)$

Στη σχέση αυτή, η στήλη a_1 έχει επίπεδο διαβάθμισης c_1, η a_2 έχει c_2, οι στήλες a_3 και a_4 έχουν διαβάθμιση c_{34}, η στήλη a_5 έχει c_5, ενώ το επίπεδο διαβάθμισης ολόκληρης της εγγραφής είναι c_t.

Οι MLS σχέσεις δεδομένων, επηρεάζουν τη ίδια τη δομή του σχεσιακού μοντέλου, λόγω του ότι δεν είναι όλα τα δεδομένα διαθέσιμα σε όλους τους χρήστες. Μια σημαντική επίδραση είναι η ύπαρξη των αποκαλούμενων πολυστιγμιότυπων (*polyinstantiation*)

2.2.5 Πολυστιγμιότυπα

Είναι η ταυτόχρονη ύπαρξη στη βάση, αντικειμένων δεδομένων τα οποία έχουν ίδιο όνομα αλλά διαφορετικά επίπεδα διαβάθμισης (Γεωργιάδης, 2002). Μπορούν να προκύψουν σε πολλές διαφορετικές περιπτώσεις: όταν ένας χρήστης με κατώτερο βαθμό εξουσιοδότησης προσπαθεί να εισάγει μία εγγραφή που υπάρχει ήδη σε κάποιο ανώτερο επίπεδο, ή όταν ένας χρήστης επιθυμεί να μεταβάλλει τιμές σε κατώτερα διαβαθμισμένες εγγραφές. Τα πολυστιγμιότυπα αφορούν είτε ολόκληρες οντότητες (polyinstantiated entities), είτε ιδιότητες των οντοτήτων (polyinstantiated attributes of entities). Υπάρχουν δύο ακραίες θέσεις για τα πολυστιγμιότυπα:

1. Τα πολυστιγμιότυπα είναι ένα αναπόφευκτο φαινόμενο των δεδομένων πολλαπλών επίπεδων. Το πρόβλημα είναι η εύρεση ενός βέλτιστου τρόπου χειρισμού των πλαστών εγγραφών.
2. Τα πολυστιγμιότυπα δε συμβαδίζουν λειτουργικά με την ακεραιότητα. Τα πολυστιγμιότυπα δεν πρέπει να είναι αποδεκτά από ένα σύστημα, γιατί θα μπορούσαν να εμποδίσουν τη σωστή διεκπεραίωση κάποιας εργασίας. Πρέπει να βρεθούν λύσεις αποφυγής του φαινομένου.

Η αλήθεια ίσως βρίσκεται κάπου ενδιάμεσα. Οι Jajodia και Sandhu έχουν υποδείξει τον τρόπο με τον οποίο μπορεί να εμποδιστεί με ασφάλεια η ύπαρξη πολυστιγμιοτύπων (δηλαδή, χωρίς τη διαρροή απόρρητων πληροφοριών ή άρνησης εξυπηρέτησης). Έτσι, όταν το φαινόμενο είναι ανεπιθύμητο, είναι δυνατό να απαλλαγούμε πλήρως από αυτό. Συνοπτικά, δεν υπάρχει θεμελιώδης ασυμβατότητα μεταξύ της ύπαρξης πολυστιγμιοτύπων και της ακεραιότητας, εφόσον η χρησιμότητά τους είναι αδιαμφισβήτητη κυρίως όταν σκόπιμα επιζητείται επικάλυψη των πληροφοριών (cover stories), όπως συμβαίνει όταν δεν πρέπει να δίνονται στους χρήστες με κατώτερους βαθμούς εξουσιοδότησης οι σωστές τιμές ορισμένων δεδομένων. Είναι η περίπτωση που τα πολυστιγμιότυπα δημιουργούνται σκόπιμα για να παρέχονται στους μη εξουσιοδοτημένους χρήστες αληθοφανείς εξηγήσεις για πληροφορίες που αναπόφευκτα πέφτουν στην αντίληψη τους και θα μπορούσαν αλλιώς να οδηγήσουν σε μερική ή ολική έμμεση προσπέλαση ευαίσθητων πληροφοριών.

2.2.6 Λειτουργικότητα των βάσεων δεδομένων πολλαπλών επιπέδων

Η χρησιμοποίηση κατά-απαίτηση ελέγχων προσπέλασης σε μια βάση δεδομένων, οδηγεί σε μία επέκταση του καθιερωμένου σχεσιακού μοντέλου δεδομένων, έτσι ώστε τα μεμονωμένα στοιχεία δεδομένων μιας σχέσης να μπορούν να περιλαμβάνουν τη διαβάθμισή τους. Επιπλέον, και κάθε ολόκληρη εγγραφή μπορεί και πρέπει να έχει μία ετικέτα διαβάθμισης, η οποία συνήθως για τις εγγραφές που ανήκουν σε ένα πίνακα δεδομένων, είναι το κατώτερο από τα άνω όρια των ετικετών των μεμονωμένων στοιχείων της εγγραφής. Κάθε πραγματική σχέση (πίνακας δεδομένων) έχει ένα πρωτεύον κλειδί που έχει καθοριστεί για αυτή, και το οποίο αποτελείται είτε από ένα χαρακτηριστικό (attribute) είτε από ένα όμοια διαβαθμισμένο σύνολο χαρακτηριστικών.

Τα πολυστιγμιότυπα είναι διαφορετικές εκδόσεις της ίδιας πραγματικής οντότητας μιας σχέσης σε μία βάση δεδομένων, που παριστάνουν το τι είναι γνωστό σε χρήστες με διαφορετικούς βαθμούς εξουσιοδότησης. Οι πληροφορίες της εγγραφής, που προσδιορίζουν μια οντότητα (εγγραφή) σε ένα πίνακα, είναι ο συνδυασμός της τιμής και της διαβάθμισης του πρωτεύοντος κλειδιού. Τα πολυστιγμιότυπα των εγγραφών παριστάνουν διαφορετικές πραγματικές οντότητες, που έχουν το ίδιο πρωτεύον κλειδί, αλλά διαφορετικές διαβαθμίσεις του πρωτεύοντος κλειδιού. Τα πολυστιγμιότυπα στοιχείων αναφέρονται όλα στην ίδια πραγματική οντότητα και παριστάνονται με ένα σύνολο εγγραφών, που έχουν όλες την ίδια τιμή και την ίδια διαβάθμιση πρωτεύοντος κλειδιού. Πέρα από το ζήτημα της αντιμετώπισης των πολυστιγμιότυπων, οι κυριότεροι περιορισμοί σε μια βάση που υιοθετεί κατά-απαίτηση μηχανισμούς είναι:

- Η διασφάλιση της ακεραιότητας (μέσω του συνδυασμού των μοντέλων BLP και Biba) καθίσταται ένα πολύ σύνθετο πρόβλημα.

- Απαιτούνται από την αρχική ακόμη φάση σχεδίασης, υποκείμενα και αντικείμενα τα οποία να είναι εφοδιασμένα με ετικέτες ασφάλειας.
- Δεν υποστηρίζουν κανόνες προσπέλασης που απαιτούν την ταυτόχρονη παρουσία διαφορετικών προσώπων (n-persons access rules), όπως αυτοί που χρησιμοποιούνται σε τράπεζες, επιχειρήσεις κλπ.

2.3 Απόδοση προνομίων βάσει ρόλων (RBAC)

Ένας ρόλος (role) είναι ένα σύνολο από ενέργειες και ευθύνες, που έχουν σχέση με τη συγκεκριμένη λειτουργία ενός οργανισμού. Κάθε μεμονωμένο άτομο μπορεί να λειτουργήσει με ένα ή και περισσότερους ρόλους. Ο διαχωρισμός (με σκοπό τον έλεγχο προσπέλασης) των χρηστών από τους ρόλους τους, προσφέρει ευελιξία στους τρόπους συνδυασμού των χρηστών και των ρόλων. Επιτρέπει στους χρήστες να λειτουργούν ως κάποιος ρόλος, χωρίς αυτό να επηρεάζει άλλες δραστηριότητες τους που εξαρτώνται από άλλους ρόλους. Επομένως, ο καθορισμός των κανόνων προσπέλασης στα δεδομένα μπορεί να υλοποιηθεί ανεξάρτητα από τα συγκεκριμένα άτομα που έχουν σχέση με τον οργανισμό. Αρχικά, η υποστήριξη των ρόλων των χρηστών εντασσόταν στα κατά-διάκριση μοντέλα ως προσέγγιση ελέγχου προσπέλασης, αφού ο ρόλος αντιμετωπιζόταν καθαρά ως ένα σύνολο, μια ομαδοποίηση χρηστών (user group). Η κατάσταση αυτή άλλαξε με την έλευση-διατύπωση του αποκαλούμενου βασισμένου-σε-ρόλους ελέγχου προσπέλασης (role-based access control, RBAC). Το μοντέλο αυτό (Sandhu, 1998), κάνει χρήση της (ευρέως χρησιμοποιούμενης από πολλά χρόνια) έννοιας του ρόλου χρήστη (user role), την οποία αναβαθμίζει μέσω ενός ισχυρού φορμαλισμού που συνδυάζει επιπρόσθετα χαρακτηριστικά όπως οι ιεραρχίες ρόλων και οι περιορισμοί.

Το μοντέλο RBAC είναι ουδέτερο πολιτικής (policy neutral). Επιτρέπει να παραχωρείται σε χρήστες ο προσδιορισμός των εξουσιοδοτήσεων σε αντικείμενα (όπως στις κατά-διάκριση πολιτικές ελέγχου), και επιπλέον μπορεί να επιβάλλει περιορισμούς, είτε άμεσα, είτε μέσω των ιεραρχιών των ρόλων, στη χρήση τέτοιων εξουσιοδοτήσεων με σκοπό τον έλεγχο ροής πληροφοριών (όπως στις κατά-απαίτηση πολιτικές ελέγχου). Ο έλεγχος προσπέλασης που τελικά επιβάλλεται σε ένα σύστημα, είναι το αποτέλεσμα της κατάλληλης διαμόρφωσης των επιμέρους μηχανισμών του RBAC (Sandhu & Samarati, 1997). Έτσι, τα μοντέλα τύπου RBAC απολαμβάνουν συνεχώς αυξανόμενης προσοχής ως μια πολλά υποσχόμενη προσέγγιση για επέκταση των παραδοσιακών μοντέλων MAC και DAC.

Με το RBAC, οι αποφάσεις προσπέλασης στηρίζονται στους ρόλους που οι μεμονωμένοι χρήστες αναλαμβάνουν στα πλαίσια ενός οργανισμού. Η διαδικασία καθορισμού των ρόλων βασίζεται σε μια λεπτομερή ανάλυση του τρόπου λειτουργίας του οργανισμού, συμπεριλαμβάνοντας στοιχεία προερχόμενα από όσο το δυνατόν ευρύτερο φάσμα χρηστών. Τα δικαιώματα προσπέλασης ομαδοποιούνται "κάτω" από κοινά ονόματα (τα ονόματα των ρόλων), και η χρήση των αντικειμένων-δεδομένων περιορίζεται μόνο σε όσα υποκείμενα-χρήστες είναι εξουσιοδοτημένα να αναλαμβάνουν τους αντίστοιχους ρόλους. Έτσι για παράδειγμα, σε ένα νοσοκομείο ο ρόλος του θεράποντος ιατρού μπορεί να περιλαμβάνει λειτουργίες όπως η διατύπωση διαγνώσεων, η παραγγελία εργαστηριακών εξετάσεων, ο προσδιορισμός της φαρμακευτικής αγωγής κλπ., ενώ ο ρόλος του ερευνητή ιατρού μπορεί να περιορίζεται στη συλλογή ανώνυμων κλινικών δεδομένων με σκοπό την ανάλυση και μελέτη τους.

Η προσέγγιση αυτή προσφέρει σημαντικότατα πλεονεκτήματα σε διαχειριστικά ζητήματα. Αρχικά, το RBAC προσφέρει δυνατότητες εφαρμογής κεντρικού ελέγχου και διαχείρισης των δικαιωμάτων προσπέλασης, σύμφωνα με τις κατευθυντήριες οδηγίες προστασίας του οργανισμού. Αυτό είναι κάτι που συμβαδίζει με τις επιθυμίες των περισσότερων οργανισμών, αφού θεωρούν ότι οι τελικοί χρήστες στους οποίους επιτρέπεται η προσπέλαση σε πληροφορίες, δεν είναι και οι κάτοχοι των πληροφοριών αυτών (όπως συχνά υπονοεί η επιβολή κατά-διάκριση μηχανισμών ελέγχου). Αντίθετα, αποζητούν μηχανισμούς που καθιστούν τους ίδιους τους οργανισμούς, τους πραγματικούς ιδιοκτήτες των πληροφοριών αυτών (Sandhu, 1998).

Έπειτα, το κόστος και η πολυπλοκότητα της διαχείρισης ασφάλειας μειώνεται σε μεγάλο βαθμό. Σε κάθε χρήστη αναθέτονται ένας ή περισσότεροι ρόλοι και σε κάθε ρόλο εκχωρούνται μια ή περισσότερες άδειες προσπέλασης, οι οποίες και μπορούν να χρησιμοποιηθούν από τους χρήστες που αναλαμβάνουν τον συγκεκριμένο ρόλο. Η διαχείριση ασφάλειας έχει λοιπόν ως έργο τον προσδιορισμό των εργασιών που επιτρέπεται να εκτελούν οι χρήστες οι οποίοι λειτουργούν ως συγκεκριμένοι ρόλοι του οργανισμού και συνεπώς τον καθορισμό των αδειών που εκχωρούνται στους ρόλους αυτούς. Επιπρόσθετα, οφείλει να επιβλέπει την εκχώρηση των κατάλληλων ρόλων σε κάθε χρήστη. Να σημειωθεί ότι οι άδειες (permissions)

στο RBAC είναι οι εγκρίσεις για επιμέρους τρόπους προσπέλασης αντικειμένων, για αυτό και είναι πάντοτε θετικές. Ισοδύναμα με τον όρο άδειες, χρησιμοποιούνται οι όροι προνόμια (privileges), δικαιώματα προσπέλασης (access rights) και εξουσιοδοτήσεις (authorizations).

Οι ρόλοι εκχωρούνται στους χρήστες βάσει των ειδικοτήτων τους (ικανότητες, εμπειρία) και των αρμοδιοτήτων τους (υπευθυνότητες που ανέλαβαν). Οι μεταβολές στις αντιστοιχίες ρόλων με χρήστες, που υπαγορεύονται από τις αλλαγές στην ανάθεση των εργασιών ενός οργανισμού, δεν αποτελούν πρόβλημα. Πράγματι, η εκχώρηση ενός ρόλου σε ένα χρήστη μπορεί να ανακληθεί εύκολα, όπως εύκολα διευθετείται και το αντίστροφο, δηλαδή το να εκχωρηθούν σε αυτόν νέοι ρόλοι. Ακόμη, όταν καθιερώνονται νέου τύπου δράσεις σε έναν οργανισμό, νέοι ρόλοι μπορούν εύκολα να δημιουργηθούν, και καθώς οι οργανωτικές λειτουργίες αλλάζουν και εξελίσσονται οι χωρίς λόγο ύπαρξης δράσεις-ρόλοι μπορούν να διαγραφούν χωρίς δυσκολία. Οι ρόλοι λοιπόν μπορούν να ενημερώνονται από κάθε άποψη, χωρίς να χρειάζεται να μεταβάλλονται οι άδειες για κάθε χρήστη μεμονωμένα.

Η εκχώρηση ενός ρόλου σε έναν χρήστη γίνεται με τέτοιο τρόπο ώστε αυτός να μην αποκτήσει περισσότερα προνόμια από όσα του είναι απαραίτητα για να προχωρήσει στην εργασία του. Η έννοια αυτή της αρχής των ελάχιστων προνομίων (least privilege), προϋποθέτει την αναγνώριση των εργασιακών λειτουργιών του χρήστη, τον προσδιορισμό του ελάχιστου συνόλου προνομίων που απαιτούνται για την ολοκλήρωση των λειτουργιών αυτών και τον περιορισμό του χρήστη στην επικράτεια (domain) αυτών και μόνο των προνομίων. Σε λιγότερο ακριβή συστήματα ελέγχου, είναι συχνά δύσκολο ή απαιτεί μεγάλο κόστος η υποστήριξη αυτής της αρχής. Από την άλλη μεριά, πέρα από την αποτροπή απόδοσης μη-απαραίτητων δικαιωμάτων σε χρήστες, η αρχή των ελάχιστων προνομίων μπορεί να διασφαλίσει τη μικρότερη δυνατή καταστροφή σε ένα σύστημα, στις περιπτώσεις ακούσιων λαθών.

2.3.1 Οι Ιεραρχίες Ρόλων στο Μοντέλο RBAC

Οι ρόλοι είναι δυνατόν να έχουν επικαλυπτόμενες αρμοδιότητες και συνεπώς άδειες προσπέλασης. Δηλαδή χρήστες που ανήκουν σε διαφορετικούς ρόλους μπορεί να απαιτείται να έχουν ορισμένες κοινές εργασίες, το οποίο άλλωστε δεν είναι και σπάνιο φαινόμενο αφού όλοι οι χρήστες συνήθως έχουν τη δυνατότητα να εκτελούν ορισμένες κοινές για όλους, στοιχειώδεις λειτουργίες. Σε τέτοιες καταστάσεις, θα ήταν μεγάλος διαχειριστικός φόρτος εργασίας να έπρεπε επαναληπτικά για κάθε ρόλο να προσδιοριστούν αυτές οι γενικές λειτουργίες. Οι ιεραρχίες των ρόλων (role hierarchies) χρησιμοποιούνται ακριβώς για την αντιμετώπιση αυτών των καταστάσεων. Αποτελούν τα φυσικά μέσα δόμησης των ρόλων, προκειμένου αυτοί να αποτυπώσουν τους σχηματισμούς αρμοδιοτήτων ενός οργανισμού.

Μια ιεραρχία ρόλων βασίζεται στις γνωστές αρχές της συνένωσης-γενίκευσης (καθώς κινούμαστε προς τα πάνω) και του επιμερισμού-εξειδίκευσης (καθώς κινούμαστε προς τα κάτω). Έτσι οι πλέον ισχυροί και ανώτεροι (senior) ρόλοι τοποθετούνται προς την κορυφή, ενώ οι λιγότερο ισχυροί και κατώτεροι (junior) ρόλοι τοποθετούνται προς τη βάση των διαγραμμάτων απεικόνισής τους (Sandhu et al., 2000). Σε μια ιεραρχία ρόλων ένας ρόλος καθορίζεται τόσο από τα μοναδικά χαρακτηριστικά (δικαιώματα προσπέλασης) που αυτός διαθέτει, όσο και ενδεχομένως από τους ρόλους που περιέχει.

Η χρήση των ιεραρχιών ρόλων προσφέρει επιπρόσθετα πλεονεκτήματα στο RBAC μοντέλο, διότι ένας ρόλος είναι δυνατόν έμμεσα να συμπεριλαμβάνει τα δικαιώματα που έχουν εκχωρηθεί σε έναν άλλο ρόλο. Όταν ένας ρόλος λοιπόν περιλαμβάνει άλλους ρόλους, αυτό σημαίνει ότι υποστηρίζονται συγκεκριμένες σχέσεις κληρονομικότητας των εξουσιοδοτήσεων, αφού οι άδειες των κατώτερων ρόλων αποτελούν υποσύνολο των αδειών του ανώτερου ρόλου. Έτσι, οι χρήστες που αναλαμβάνουν ανώτερους ρόλους μπορούν να κληρονομούν όλες τις εξουσιοδοτήσεις των υφισταμένων τους και αντίστροφα οι χρήστες των κατώτερων ρόλων κληρονομούν κάθε απαγόρευση που ισχύει στους προϊσταμένους τους.

2.3.2 Η Έννοια των Περιορισμών στο Μοντέλο RBAC

Σε έναν οργανισμό, για την αποφυγή καταχρήσεων στις προσπελάσεις πληροφοριών και γενικώς αθέμιτων δραστηριοτήτων, απαιτούνται αρκετοί περιορισμοί στις εξουσιοδοτήσεις των χρηστών. Ο διαχωρισμός των καθηκόντων (separation of duties) αποτελεί ένα τυπικό παράδειγμα περιορισμού εξουσιοδοτήσεων, το οποίο είναι πολύ γνωστό στην περιοχή της ασφάλειας. Εκφράζει μια πρακτική πολλών οργανισμών, οι οποίοι θέλοντας να μειώσουν τον κίνδυνο απάτης, δεν επιτρέπουν σε κανένα άτομο να συγκεντρώνει πάνω του τις

απαιτούμενες για αυτόν τον σκοπό εξουσιοδοτήσεις. Έτσι η αρχή του διαχωρισμού καθηκόντων επιβάλλει στη διαχείριση ασφάλειας ότι σε κανένα χρήστη δεν πρέπει να εκχωρούνται δικαιώματα τέτοια ώστε αυτός να μπορεί στη συνέχεια να χρησιμοποιήσει καταχρηστικά το σύστημα για το συμφέρον του.

Ένα μοντέλο RBAC, πέρα από τη δυνατότητα υποστήριξης της αρχής των ελάχιστων προνομίων μέσω των αντιστοιχίσεων χρηστών σε ρόλους και δικαιωμάτων σε ρόλους, διαθέτει ένα ακόμη χρήσιμο διαχειριστικό εργαλείο, τους περιορισμούς (constraints). Οι περιορισμοί σύγκρουσης συμφερόντων (conflict of interest) είναι οι κατάλληλοι για την ικανοποίηση της αρχής του διαχωρισμού καθηκόντων (Sandhu et al., 2000), διότι αρκεί να εισαχθεί σε όσους ρόλους απαιτείται ένας περιορισμός της δήλωσής τους ως "αμοιβαία αποκλειστικοί" (mutually disjoint). Έτσι αποτρέπεται η ανάθεση σε έναν χρήστη που ήδη του έχει ανατεθεί κάποιος ρόλος, ενός άλλου ρόλου του οποίου τα επιπρόσθετα δικαιώματα μπορούν να τον καταστήσουν πηγή κινδύνου (static separation of duty). Το RBAC υποστηρίζει αμοιβαία αποκλειστικούς ρόλους όχι μόνο κατά την ανάθεση ρόλων σε χρήστες αλλά και κατά την ενεργοποίηση των ρόλων από τους χρήστες. Δηλαδή, μπορούν να εισαχθούν περιορισμοί έτσι ώστε ένας χρήστης να μη μπορεί να ενεργεί χρησιμοποιώντας ταυτόχρονα συγκεκριμένους ρόλους (dynamic separation of duty).

Στους ρόλους μπορεί να επιβληθούν ακόμη και προσωρινοί (temporal) περιορισμοί, όπως ο χρόνος και η διάρκεια ενεργοποίησης ενός ρόλου από ένα χρήστη, ή ακόμη και δυνατότητα χρησιμοποίησης της ενεργοποίησης ενός ρόλου ως χρονικό έναυσμα (timed-triggering) για την ενεργοποίηση ενός άλλου ρόλου. Τέλος, η διάρθρωση του μοντέλου RBAC επιτρέπει τη δυνατότητα επιβολής περιορισμών στην ανάθεση ρόλων σε χρήστες. Για παράδειγμα, ένας ρόλος διαχειριστικός συνήθως απαιτείται να δέχεται ένα ανώτατο όριο σχετικά με το πλήθος των χρηστών που τον αναλαμβάνουν (role cardinality). Συνοψίζοντας, μέσω των περιορισμών ένας οργανισμός μπορεί να διαφυλάξει τις επιδιώξεις της γενικότερης πολιτικής του, αποκεντρώνοντας τις διαδικασίες ανάθεσης ρόλων σε χρήστες. Οι περιορισμοί αποτελούν συνεπώς, έναν ισχυρό μηχανισμό επιβολής οργανωτικών επιλογών υψηλότερου επιπέδου.

2.4 Αξιοποίηση του context κατά τον έλεγχο προσπέλασης

Παρά το γεγονός ότι τα βασισμένα σε ρόλους μοντέλα παρέχουν έναν σχετικά επαρκή τρόπο εφαρμογής ενός συστήματος ελέγχου, δεν έχουν τη δυνατότητα να εκμεταλλευτούν σημαντικά στοιχεία που αποτελούν το πλαίσιο αναφοράς (context) μέσα στο οποίο εξελίσσονται οι αιτήσεις προσπέλασης του χρήστη. Σημαντική παράμετρος για παράδειγμα, είναι αυτή της διαπίστωσης κατά πόσο εξελίσσεται μια δραστηριότητα συνεργασίας των χρηστών με στόχο την αλληλοσυμπλήρωση των ικανοτήτων τους και συνεπώς των ευθυνών που έχουν αναλάβει στα πλαίσια ενός οργανισμού. Ακόμη, το αντικείμενο, ο χώρος και ο χρόνος της αίτησης προσπέλασης αποτελούν επίσης 'συναφείς' πληροφορίες, χρήσιμες για την υλοποίηση ακόμη αποδοτικότερων μηχανισμών ελέγχου, ικανών για σαφή διαχωρισμό ανάμεσα στην απονομή αδειών προσπέλασης και την ενεργοποίησή τους την κατάλληλη στιγμή. Για αυτόν τον λόγο, και έχουν προταθεί επεκτάσεις στα βασισμένα σε ρόλους μοντέλα που συμπληρώνουν ένα αριθμό δυναμικών συστατικών τα οποία είναι ικανά να προσδιορίσουν το τρέχον πλαίσιο αναφοράς των αιτήσεων προσπέλασης. Αυτό βεβαίως θα χρησιμοποιείται κατά τον χρόνο εκτέλεσης (runtime) των εφαρμογών του συστήματος με σκοπό τον έλεγχο με υψηλή ακρίβεια της ενεργοποίησης των προνομίων προσπέλασης.

Η αναγκαιότητα είναι για ένα σύστημα ελέγχου προσπέλασης ικανού να υποστηρίζει δυναμικά μεταβαλλόμενες εξουσιοδοτήσεις, χωρίς όμως να επιβαρύνει τόσο το έργο των χρηστών όσο και των διαχειριστών ασφάλειας. Το σύστημα αυτό χρειάζεται να έχει ένα υβριδικό χαρακτήρα αφού θα προσπαθεί να συνδυάσει τα πλεονεκτήματα διαχείρισης των βασισμένων σε ρόλους αδειών, με τη δυνατότητα αυστηρότερων ελέγχων πάνω στη ενεργοποίηση αυτών των αδειών του κάθε χρήστη ανάλογα με την τρέχουσα ενασχόλησή του (Georgiadis et al., 2001).

Video 3.1.mp4	Βίντεο (video)
Ενεργοποίηση context-based δικαιωμάτων	

2.4.1 Ενδεικτική μελέτη περίπτωσης: το μοντέλο ελέγχου προσπέλασης C-TMAC

Το μοντέλο C-TMAC (Context-based Team Access Control) στηρίζεται στα μοντέλα RBAC (Sandhu, 1998), και TMAC (Thomas, 1997). Είναι ένας αρκετά αποτελεσματικός και ταυτόχρονα ευέλικτος έλεγχος

προσπέλασης που βασίζεται στις γενικότερες αρχές ενεργητικής ασφάλειας που δίνει περισσότερο βάρος στα RBAC χαρακτηριστικά των μελών της ομάδας. Προσδιορίζει με μεγαλύτερη σαφήνεια τον τρόπο αλληλεπίδρασης των βασισμένων σε ρόλους αδειών στα όρια μιας ομάδας χρηστών, έτσι ώστε να είναι δυνατή η όσο το δυνατόν πληρέστερη εκμετάλλευση των πλεονεκτημάτων διαχείρισης που αυτά προσφέρουν. Σημαντική προσθήκη αποτελεί η λειτουργικότητα που αποδίδεται στις συναφείς προς μια δραστηριότητα πληροφορίες context. Οι παράγοντες καθορισμού, το περιεχόμενο, ο τρόπος χρήσης και εν τέλει το ίδιο το νόημα των πληροφοριών context, με τον τρόπο με τον οποίο ενσωματώνονται στους μηχανισμούς ελέγχου προσπέλασης, επιδρούν καταλυτικά στη συμπεριφορά του βασισμένου σε ομάδες μοντέλου. Βασικό πλεονέκτημα που προκύπτει ως επακόλουθο της χρήσης context, είναι η μείωση του όγκου του διαχειριστικού έργου που αφορά τις ομάδες και τα δικαιώματα των μελών τους. Αξίζει τέλος να υπογραμμισθεί ότι κατ' αυτόν τον τρόπο αποκτά δυνατότητα εφαρμογής ακόμη και σε δραστηριότητες όπου οι χρήστες δρουν μεμονωμένα και όχι σαν μέλη ομάδων.

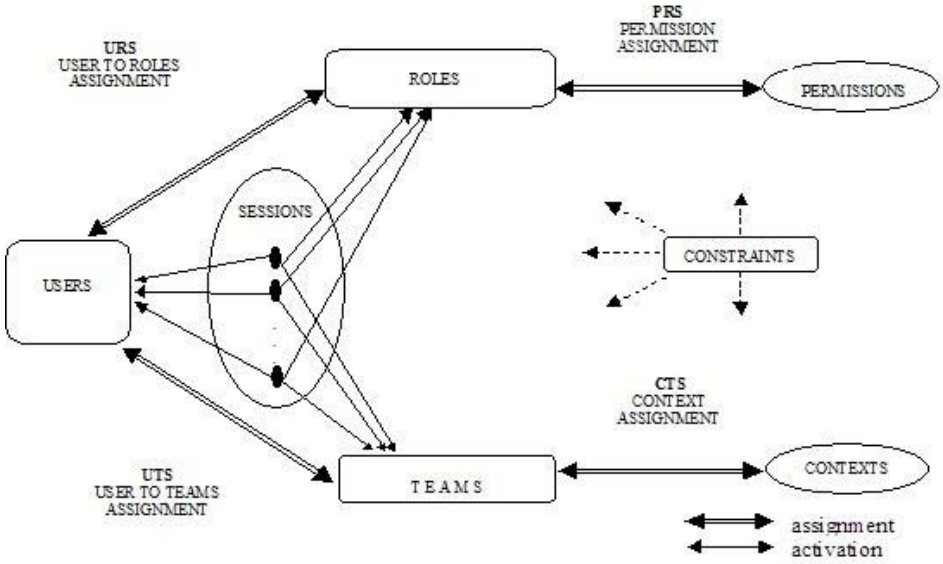
Η ενοποίηση (integration) χαρακτηριστικών RBAC και Contexts (Georgiadis et al., 2001) οδηγεί στον σχηματισμό του μοντέλου C-TMAC, του οποίου βασικά σημεία είναι:

- Αναγνωρίζεται ως δραστηριότητα συνεργασίας ένα συγκεκριμένο έργο όταν αποτελεί προϊόν της συνεργασίας μιας ομάδας χρηστών που λειτουργούν βάσει συγκεκριμένων ρόλων σε έναν οργανισμό. Η σύνθεση της ομάδας αυτής επηρεάζει τα συνολικά βασισμένα σε ρόλους δικαιώματα προσπέλασης των μελών της. Τα πραγματικά όμως δικαιώματα της ομάδας που τελικά πρέπει να ενεργοποιηθούν για τα μέλη της, οφείλουν να έχουν άμεση σχέση με το αντικείμενο της ομάδας και μόνο με αυτό.
- Αναγνωρίζεται η σημασία συγκεκριμένων παραγόντων που επιδρούν στον προσδιορισμό του αντικειμένου της ομάδας και συνεπώς επιδρούν και στην επιθυμητή συμπεριφορά του συστήματος ελέγχου προσπέλασης ενόσω λειτουργεί σε πραγματικό χρόνο. Ενδεικτικά, ο χώρος που βρίσκονται οι χρήστες ή ο χρόνος αίτησης προσπέλασης είναι σημαντικοί παράγοντες. Τέτοιου είδους παράγοντες διαφοροποιούν τη συνάρτηση απόδοσης των δικαιωμάτων προσπέλασης ανάλογα με τη σχέση που αποκτούν οι χρήστες με τις διαχειριστικές επικράτειες (administrative domains) που οι παράγοντες αυτοί ορίζουν.
- Κάθε μέλος ομάδας είναι δυνατόν, ανάλογα και με την πολιτική που θέλουμε να ακολουθηθεί, για μικρό χρονικό διάστημα και μόνον όσο βρίσκεται στον χώρο δράσης της ομάδας, να αποκτήσει επιπρόσθετα βασισμένα σε ρόλους δικαιώματα, προερχόμενα από τους ρόλους που έχουν τα υπόλοιπα ενεργά μέλη της ομάδας. Έτσι, θα ήταν δυνατό κάτω από ορισμένες προϋποθέσεις, τα βασισμένα σε ρόλους προνόμια προσπέλασης των μελών μιας ομάδας να είναι ίδια για όλα τα μέλη. Σε κάθε περίπτωση όμως, το σύνολο αυτό των δικαιωμάτων λόγω ρόλων (role-based permissions) δεν είναι παρά ένα ενδιάμεσο υπερσύνολο των πραγματικών δικαιωμάτων του χρήστη. Τα τελικά δικαιώματα προσπέλασης που ενεργοποιούνται για κάθε χρήστη, τα οποία ονομάζουμε βασισμένα στο πλαίσιο δικαιώματα (context-based permissions), προκύπτουν μέσω μιας διαδικασίας φιλτραρίσματος στα αντικείμενα προσπέλασης που παρέχουν οι παράγοντες context. Κάθε δηλαδή συγκεκριμένη αίτηση πρόσβασης μπορεί να επιτραπεί αν και μόνον αν δεν παραβιάζει τον στόχο, πλαίσιο αναφοράς της εργασίας της ομάδας. Αυτό αποτελεί την τελική γραμμή άμυνας του συστήματος ελέγχου προσπέλασης.
- Ο συνδυασμός των context παραγόντων σχηματίζει ένα context-πρότυπο, το οποίο μπορεί να προσδιοριστεί στατικά. Αποδίδοντας συγκεκριμένες τιμές σε ένα πρότυπο, σχηματίζεται το context ενός συγκεκριμένου χρήστη. Όμως με αυτή τη λογική, ένας τεράστιος όγκος διαχειριστικού έργου απαιτείται για την εισαγωγή σε κάθε χρήστη και σε καθημερινή βάση των πληροφοριών context. Απάντηση σε αυτό το πρόβλημα δίνει η χρήση της έννοιας της ομάδας χρηστών (team), ως διαχειριστικό εργαλείο.
- Για τη διαχείριση αυτών των παραγόντων, χρειάζεται ένα σύστημα ελέγχου προσπέλασης, ικανό να υποστηρίζει την ενεργοποίηση αδειών ανάλογα βεβαίως με αυτούς τους παράγοντες αλλά παράλληλα να είναι σε θέση να συνεκτιμήσει και την ομαδικότητα που αρκετές φορές απαιτεί μια δραστηριότητα. Οι παράγοντες χρησιμοποιούνται για τον προσδιορισμό του πλαισίου αναφοράς (context) που αντιστοιχεί σε κάθε συγκεκριμένη δραστηριότητα. Το πλαίσιο αυτό τελικά, δεν είναι τίποτε άλλο παρά οι σαφώς καθορισμένες αναγκαίες γνώσεις

για ένα χρήστη που του επιτρέπουν να ενεργήσει αποδοτικά. Στο σημείο αυτό έρχεται να "δέσει" λογικά και η έννοια της ομάδας, αφού οι απαιτήσεις αναγκαίας γνώσης με τον τρόπο που προηγουμένως ορίστηκαν, είναι κοινές σε όλα τα μέλη της ομάδας που ο χρήστης ανήκει.

Η έννοια της ομάδας εξυπηρετεί πραγματικές διαχειριστικές ανάγκες. Επειδή τα επιμέρους πλαίσια αναφοράς των ενεργειών των χρηστών είναι τόσο πολλά για να μπορεί κανείς να τα χειρίζεται ξεχωριστά επί καθημερινής βάσης, οι ομάδες μπορεί να θεωρηθούν ως 'ομαδοποιήσεις πλαισίων'. Με τον ίδιο τρόπο που οι ρόλοι στο μοντέλο RBAC παρέχουν μια αποδοτική μέθοδο χειρισμού των διάφορων υπευθυνοτήτων των χρηστών στους οργανισμούς, οι ομάδες αναλογικά μπορούν να διευκολύνουν τη διαχείριση των πληροφοριών context στις διάφορες δραστηριότητες που εξελίσσονται.

Υπάρχουν βεβαίως, πολλές περιπτώσεις όπου οι χρήστες λειτουργούν όχι στα πλαίσια μιας ομάδας, αλλά μόνοι τους και απομονωμένα. Οι περιπτώσεις αυτές μπορούν επίσης να αντιμετωπιστούν από το μοντέλο αυτό, επιτρέποντας ομάδες ενός μέλους. Έτσι, ένας χρήστης που δε θέλει να δράσει ως μέλος ομάδας, ενεργοποιεί χωρίς να χρειάζεται να το ζητήσει ρητά, την προκαθορισμένη ιδιωτική ομάδα του. Με αυτό τον τρόπο, το μοντέλο αυτό μπορεί να χρησιμοποιηθεί σε όλα τα περιβάλλοντα, αφού είναι ικανό να υποστηρίξει και την ομαδική και την ατομική δραστηριότητα των χρηστών.



Σχήμα 3.4 *Το μοντέλο C-TMAC*

Υπάρχουν διάφοροι τρόποι με τους οποίους μπορούν να συνδυαστούν τα βασισμένα σε ρόλους δικαιώματα των μελών μιας ομάδας:

- Συνάθροιση (aggregation): Συνενώνονται σε ένα ευρύτερο υπερσύνολο, τα σύνολα δικαιωμάτων από όλα τα μέλη της ομάδας και έτσι προκύπτουν οι άδειες πρόσβασης που διατίθενται στην ομάδα.
- Μέγιστο/ελάχιστο (maximum/minimum): Εξετάζοντας τα δικαιώματα του κάθε μέλους της ομάδας, χρησιμοποιείται το ανώτατο ή το κατώτατο όριο αυτών για να εκχωρηθούν δικαιώματα στην ομάδα ως σύνολο. Υπάρχουν περιπτώσεις όπου αρκεί ένα μόνο μέλος της ομάδας να έχει υψηλό βαθμό εξουσιοδότησης (clearance) έτσι ώστε αυτό το μέλος να εγγυηθεί και για τα υπόλοιπα. Από την άλλη μεριά, υπάρχουν περιπτώσεις όπου το γεγονός ότι η ομάδα συμπεριλαμβάνει ένα ή περισσότερα μέλη κατέχοντα χαμηλού βαθμού διαπιστευτήρια πρόσβασης, δρα περιοριστικά στις ενέργειες της ομάδας ως σύνολο.

- Διάρθρωση ομάδας (team structure): Πέρα από το ποια είναι τα δικαιώματα των μελών της ομάδας, εξετάζεται και το πλήθος αυτών ως παράγοντας που επηρεάζει τα δικαιώματα που αποκτά η ομάδα. Για παράδειγμα, η δυνατότητα να αλλάξει κάποια κρίσιμα προστατευμένα δεδομένα μια ομάδα, είναι δυνατόν να ισχύει μόνο όταν δυο χρήστες με τις κατάλληλες εξουσιοδοτήσεις είναι μέλη της και όχι όταν κάποιο μεμονωμένο μέλος το ζητήσει, όποιον βαθμό εξουσιοδότησης και αν έχει αυτό.

Προφανώς, συνδυασμοί των τριών αυτών μεθόδων μπορούν να χρησιμοποιηθούν με τελικό στόχο την προσαρμογή των δικαιωμάτων πρόσβασης σε ειδικές ανάγκες και προϋποθέσεις.

2.4.2 Ενεργοποίηση των τελικών δικαιωμάτων χρήστη στο μοντέλο C-TMAC

Για την αντιμετώπιση του προβλήματος του ελέγχου προσπέλασης σε περιβάλλοντα όπου εξελίσσονται και δραστηριότητες συνεργασίας των χρηστών, χρησιμοποιούμε βασισμένη σε ρόλους απονομή αδειών και βασισμένη σε ομάδες ενεργοποίηση των αδειών πρόσβασης. Σκοπός μας ο έλεγχος πρόσβασης συγκεκριμένων δεδομένων σε μικρά χρονικά διαστήματα. Η λειτουργία του συστήματος ελέγχου, έχει ως εξής:

Μετά την ολοκλήρωση της διαδικασίας αναγνώρισης και πιστοποίησης χρήστη από το σύστημα, ο χρήστης πρέπει να επιλέξει τους αρχικούς ρόλους με τους οποίους θα λειτουργήσει κατά την τρέχουσα συνεδρία (session). Επιλέγει έναν ή περισσότερους ρόλους, αλλά πάντα από τους ρόλους που η διαχείριση του οργανισμού έχει ορίσει ως διαθέσιμους για αυτόν, ανάλογα με τα προσόντα, την εμπειρία και τις ικανότητες που διαθέτει. Σύμφωνα με αυτή του την επιλογή, ένα συγκεκριμένο σύνολο (βασισμένων σε ρόλους) αδειών του εκχωρείται, το οποίο ονομάζεται άδειες ρόλων συνεδρίας (session-roles permissions). Μέχρι αυτό το σημείο, το μοντέλο συμπεριφέρεται όπως ακριβώς ένα τυπικό μοντέλο παθητικής ασφάλειας: το τρέχον σύνολο αδειών πρόσβασης δεν είναι ικανό να ανιχνεύσει καμίας μορφής πλαίσιο πληροφοριών.

Μετά την επιλογή ρόλου, ο χρήστης επιλέγει την ομάδα ή τις ομάδες που θα συμμετάσχει κατά την τρέχουσα συνεδρία. Έτσι σε πρώτη φάση τροποποιούνται οι άδειες πρόσβασης που κατέχει, αφού αυτές συνδυάζονται (με έναν από τους τρόπους που καθορίζονται στον ορισμό του C-TMAC) με τις άδειες τις βασισμένες σε ρόλους που προέρχονται από τους ρόλους που έχουν επιλέξει οι υπόλοιποι χρήστες, μέλη εκείνη τη στιγμή της ίδιας ομάδας. Η πρώτη λοιπόν φάση προσδιορισμού των βασισμένων σε ρόλους δικαιωμάτων ενός χρήστη ολοκληρώνεται με τον συνδυασμό των session-roles προσωπικών δικαιωμάτων του και των team-roles δικαιωμάτων των μελών της ομάδας του.

Οι ομάδες θεωρούνται ως σύνολα πληροφοριών contexts σχετικών με τις τρέχουσες δραστηριότητες. Επιλέγοντας λοιπόν ο χρήστης μια ομάδα, αποκτά επιπλέον (σε δεύτερη φάση) και το σύνολο των contexts που αντιστοιχεί σ΄ αυτήν. Το πλαίσιο αναφοράς της ομάδας (team context) αποτελείται από συγκεκριμένα αντικείμενα δεδομένων (στόχοι αιτήσεων πρόσβασης) και από συνθήκες που εκφράζονται με όρους πεδίων τιμών (ranges of values) για τους παράγοντες context. Για κάθε ομάδα υπάρχουν διαθέσιμες μεταβλητές συστήματος, ικανές να διατηρούν σύνολα τιμών για τις παραμέτρους που επιλέξαμε. Το ‘δέσιμο’ αυτών των μεταβλητών με τις πραγματικές τιμές τους, πραγματοποιείται σε πραγματικό χρόνο (runtime) από το προσωπικό διαχείρισης του οργανισμού, ενώ και αρκετές από τις προβλεπόμενες διαδικασίες εκχώρησης-ανάθεσης μπορούν έως ένα βαθμό να αυτοματοποιηθούν με τη χρήση κατάλληλων διατάξεων ελέγχου φυσικής παρουσίας των χρηστών, ενός ελέγχου που πραγματοποιείται σε συνδυασμό με τη δυνατότητα καθορισμού διάρθρωσης των ομάδων με όρους ρόλων για συχνά εμφανιζόμενες καταστάσεις.

Το context ομάδας λειτουργεί τελικά ως σύνολο περιορισμών πάνω στα αντικείμενα προσπέλασης καθώς και στις συνθήκες κατά την αίτηση προσπέλασης. Επιτρέπει το φιλτράρισμα, την επιλογή μέρους δηλαδή των δεδομένων που ο χρήστης ζητάει, και τελικά είναι υπεύθυνο για το αποτέλεσμα της αίτησης που λαμβάνει ο χρήστης. Το φιλτράρισμα αυτό ουσιαστικά γίνεται πάνω στα βασισμένα σε ρόλους δικαιώματα της ομάδας του χρήστη. Έτσι, τα τελικά δικαιώματα που ενεργοποιούνται για ένα χρήστη κινούνται πάντα μέσα στα πλαίσια της δραστηριότητας της ομάδας του. Κάθε αίτησή του για πρόσβαση, επιτρέπεται μόνο αν τα αναγκαία βασισμένα σε ρόλους δικαιώματα του έχουν ήδη αποδοθεί και μόνον όταν οι τρέχουσες τιμές των μεταβλητών context περιέχονται στα πεδία τιμών που προσδιορίζουν το context της ομάδας του. Με αυτό τον τρόπο, κρίσιμες πληροφορίες μιας δραστηριότητας γίνονται διαθέσιμες μόνο στα μέλη της ομάδας χρηστών που την έχει αναλάβει και μόνο κατά τη διάρκεια που αυτή εξελίσσεται.

Η διαδικασία φιλτραρίσματος (filtering process) προσδίδει αυξημένη δυναμικότητα στο μοντέλο C-TMAC. Στηρίζεται σε κανόνες οι οποίοι με απλό αλλά ταυτόχρονα και αυστηρό όπου χρειάζεται τρόπο είναι ικανοί να προσδιορίσουν το τελικό σύνολο δικαιωμάτων ενός χρήστη. Μπορεί να θεωρηθεί ως ένας μηχανισμός εξαγωγής μεστών σε νόημα (meaningful) υποσυνόλων των βασισμένων σε ρόλους συνόλων αδειών προσπέλασης. Τα υποσύνολα αυτά βασίζονται στις τιμές των μεταβλητών που προσδιορίζουν τα πεδία τιμών για όλους τους επιλεγμένους παράγοντες προσδιορισμού του context της ομάδας και είναι με αυτό τον τρόπο ικανά να εξυπηρετήσουν τους εξειδικευμένους στόχους του συστήματος ελέγχου προσπέλασης. Απαιτείται βέβαια από το τμήμα διαχείρισης του υπό θεώρηση οργανισμού, ο προσδιορισμός των "νομίμων" (valid) και αποδεκτών (acceptable) πεδίων τιμών των πληροφοριών context για κάθε ομάδα.

Συμπερασματικά, η ενεργοποίηση των αδειών πρόσβασης ενός χρήστη, διενεργείται με την ακόλουθη διαδικασία δυο βημάτων:

Βήμα 1: Ένας χρήστης που έχει επιλέξει ένα σύνολο ρόλων και συμμετέχει σε ένα σύνολο ομάδων, αρχικά αποκτά τα βασισμένα σε ρόλους δικαιώματα σύμφωνα με την ακόλουθη έκφραση, όπου ≥ σημαίνει *"σε συνδυασμό με"*:

Role-based Permissions = *Session-Roles Permissions* ⊕ *Team-Roles Permissions*

Βήμα 2: Τα τελικά δικαιώματα που ενεργοποιούνται για τον χρήστη, είναι τα *context-based permissions*, τα οποία προέρχονται από τα *role-based permissions* του 1ου βήματος, σύμφωνα με την ακόλουθη έκφραση, όπου ⊕ σημαίνει *"φιλτραρισμένα από"*:

Context-based Permissions = Role-based Permissions ⊗ Team-Context

3. Συστήματα πληρωμών ηλεκτρονικού εμπορίου

Σημαντική πλευρά των τεχνολογιών ηλεκτρονικού εμπορίου αποτελεί η υποστήριξη των ηλεκτρονικών πληρωμών. Άλλωστε, η αξιοποίηση των δυνατοτήτων του ηλεκτρονικού εμπορίου σε ολοένα και μεγαλύτερη κλίμακα, εξαρτάται από τη διαθεσιμότητα και την ευρεία διάδοση των συστημάτων ηλεκτρονικών πληρωμών. Στην τελευταία αυτή ενότητα του κεφαλαίου θα αναφερθούμε επιλεκτικά και με συντομία στις κυριότερες μορφές συστημάτων πληρωμών ηλεκτρονικού εμπορίου (e-payment systems). Τα συστήματα αυτά μπορεί να εμπλέκουν διάφορους 'παίκτες' – συμμετέχουσες οντότητες και μια ποικιλία μεθόδων πληρωμής. Ειδική αναφορά θα γίνει σε δύο σημαντικά σχήματα-πρωτόκολλα ασφαλείας που εμπλέκονται στην υποστήριξη πληρωμών ηλεκτρονικού εμπορίου: το πρωτόκολλο γενικής χρήσης SSL και το πρωτόκολλο 3D-Secure για τις online πληρωμές με χρήση (πιστωτικών/χρεωστικών) καρτών.

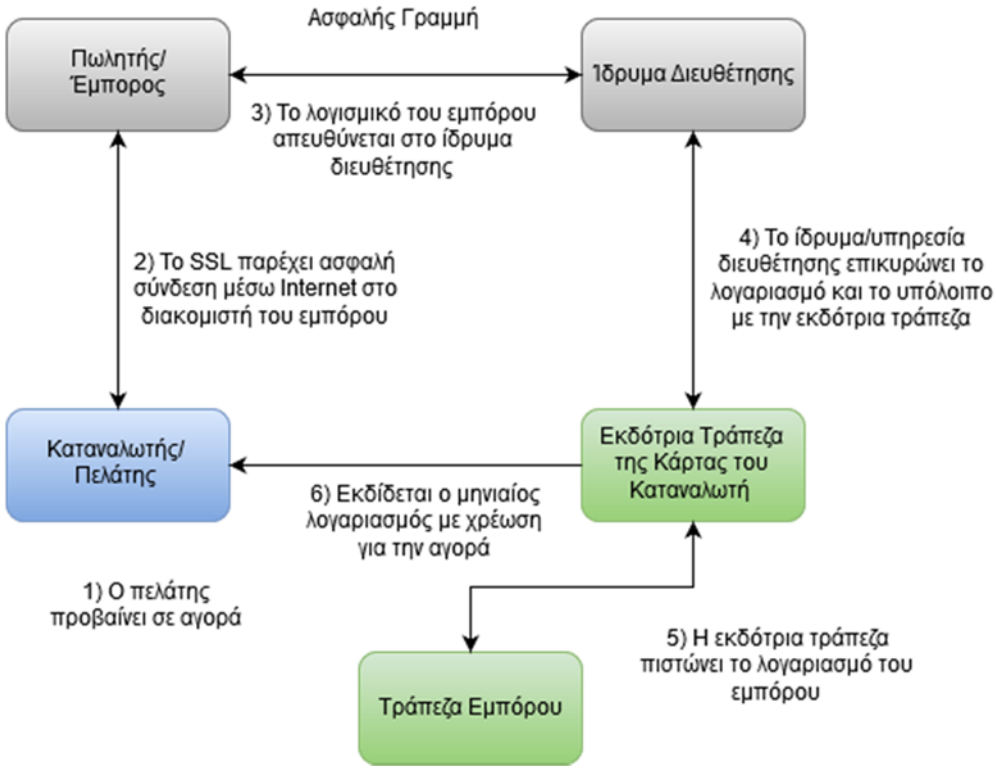
3.1 Είδη και μέσα πληρωμών ηλεκτρονικού εμπορίου

Κυρίαρχη τάση αποτελούν οι πληρωμές που βασίζονται σε κάρτες (card-based payments), όμως σημαντικό ρόλο παίζει και η χρήση ψηφιακών πορτοφολιών (e-wallets) όπως το PayPal. Επίσης, συστήματα ψηφιακού χρήματος, όπως το Bitcoin, αποτελούν ενδιαφέρουσες εναλλακτικές λύσεις. Μια πιο καθαρή ίσως εικόνα των συστημάτων πληρωμών μπορεί να δοθεί, χρησιμοποιώντας ως κριτήριο κατηγοριοποίησης το πότε ο αγοραστής πληρώνει, πότε ουσιαστικά μετατρέπει τα χρήματά του σε ένα κατάλληλο μέσο για ηλεκτρονικές συναλλαγές: πριν να αποφασίσει μια συγκεκριμένη αγορά, κατά τη διάρκεια της συναλλαγής, ή μετά την απόφασή του να προβεί σε μια αγορά:

Πίστωσης ή **πληρωμής μετά την αγορά** (credit or postpaid): το μηχάνημα του διακομιστή (server) από την πλευρά του πωλητή/εμπόρου ελέγχει τη γνησιότητα των πελατών, δηλαδή πιστοποιεί την ταυτότητά τους (authentication), και ταυτόχρονα ελέγχει στην τράπεζα για τα κατάλληλα υπόλοιπα πίστωσης πριν την ολοκλήρωση της συναλλαγής. Σημαντικό σημείο είναι ότι έχει προ-συμφωνηθεί μια διαδικασία επίλυσης διαφορών μεταξύ των εμπλεκόμενων μερών. Η πιο διαδεδομένη μορφή μηχανισμών πίστωσης είναι η υποστήριξη online συναλλαγών με πιστωτικές κάρτες (credit cards). Διατραπεζικοί όμιλοι, όπως η Visa, η MasterCard ή η American Express καθορίζουν τους κανόνες στις τράπεζες που εκδίδουν τις κάρτες και έχουν την επίβλεψη της επεξεργασίας των συναλλαγών (Laudon & Traver, 2014). Στην επιβεβαίωση των στοιχείων της πιστωτικής κάρτας και τον έλεγχο υπολοίπου εμπλέκονται ιδρύματα διευθέτησης (clearinghouses, γραφεία εκκαθάρισης συναλλαγών, κέντρα επεξεργασίας) ως 'τρίτα αξιόπιστα μέρη' (third party verification).

Για τη διευθέτηση, τα ιδρύματα επικοινωνούν με κέντρα διαχείρισης πιστοποιητικών. Για λόγους μεγαλύτερης διασφάλισης των συναλλαγών, η χρήση των ευαίσθητων στοιχείων της κάρτας γίνεται αποκλειστικά σε κρυπτογραφημένη μορφή. Με αυτό τον τρόπο οι πληροφορίες της πιστωτικής κάρτας, όχι μόνο προφυλάσσονται κατά τη διακίνησή τους στο Διαδίκτυο, αλλά δεν αποκαλύπτονται ούτε και στον ίδιο τον έμπορο (Κάτσικας, 2001). Ως συνολική λοιπόν εικόνα, εκτός των ομίλων που θέτουν τα πρότυπα, διακρίνουμε πέντε οντότητες που συμμετέχουν στα ηλεκτρονικά συστήματα πληρωμής με πιστωτικές κάρτες: ο καταναλωτής/πελάτης (κάτοχος της κάρτας), ο πωλητής/έμπορος, η τράπεζα που εκδίδει την κάρτα (εκδότρια τράπεζα, η τράπεζα του πελάτη), η τράπεζα του εμπόρου, και το ίδρυμα διευθέτησης.

Gif 3.1.gif	Κινούμενη εικόνα (interactive)
Φάσεις συναλλαγής με πιστωτική κάρτα	



Σχήμα 3.5 *Φάσεις συναλλαγής με πιστωτική κάρτα*

Να σημειωθεί ότι οι λογαριασμοί πιστωτικών καρτών (όπως οι λογαριασμοί άλλων υπηρεσιών, πχ. τηλεφώνου) αποκαλούνται **πληρωμές συσσώρευσης υπολοίπου** (accumulating balance) επειδή συσσωρεύουν δαπάνες μέσα σε μια προκαθορισμένη χρονική περίοδο (συνήθως ένα μήνα) και κατόπιν εξοφλούνται (πλήρως ή εν μέρει) στο τέλος της περιόδου.

Χρέωσης ή **προπληρωμής** (debit or prepaid): οι χρήστες πληρώνουν προκαταβολικά για να αποκτήσουν αντίστοιχα δικαιώματα. Υπάρχει σχέση με τα αποκαλούμενα συστήματα πληρωμών αποθηκευμένης αξίας (stored value), αφού αυτά περιλαμβάνουν χρήση λογαριασμών (που περιέχουν χρήματα κατατεθειμένα εκ των προτέρων) από τους οποίους διενεργούνται πληρωμές. Παραδείγματα πληρωμών αποθηκευμένης αξίας είναι οι χρεωστικές κάρτες, οι προπληρωμένες κάρτες, οι έξυπνες κάρτες και οι δωροεπιταγές (Laudon & Traver, 2014). Η χρήση των **χρεωστικών καρτών** (debit cards) είναι μια πολύ διαδεδομένη μορφή πληρωμών ηλεκτρονικού εμπορίου σε αυτήν την κατηγορία. Η ιδιαιτερότητά τους, σε σχέση με τις άλλες πληρωμές της ίδιας κατηγορίας, είναι ότι χρεώνουν απευθείας (την ώρα της συναλλαγής) έναν λογαριασμό, που όμως απαιτείται να έχει ήδη κατατεθειμένα χρήματα σε επάρκεια. Μια εναλλακτική, και ευρέως διαδεδομένη μορφή πληρωμών σε περιβάλλοντα ηλεκτρονικού εμπορίου σε αυτήν την κατηγορία είναι οι αποκαλούμενες **online πληρωμές αποθηκευμένης αξίας**. Το PayPal είναι το πιο διαδεδομένο σύστημα πληρωμής αυτής της υποκατηγορίας, που από τη μια μεριά δεν προϋποθέτει ακριβώς την προπληρωμή, αλλά από την άλλη μεριά απαιτεί αποθηκευμένη αξία χρημάτων: ένας λογαριασμός PayPal για

έναν αγοραστή ή έναν πωλητή, προϋποθέτει τον προσδιορισμό ενός (χρεωστικού, τρεχούμενου ή πιστωτικού) λογαριασμού στον οποίο θα γίνει η χρέωση ή η πίστωση χρημάτων όταν ολοκληρωθεί η online συναλλαγή. Στην ουσία του, είναι μια υπηρεσία διαμεσολάβησης, μια πύλη πληρωμών (Trautman, 2014), ικανή να υποστηρίξει ακόμη και πληρωμές μικροποσών. Χρησιμοποιείται και ο όρος **ομότιμο σύστημα πληρωμών** (P2P), επειδή επιτρέπει τη διεκπεραίωση πληρωμών (έως ένα καθορισμένο ποσό) χωρίς τη μεσολάβηση τραπεζών. Όταν ο αγοραστής κάνει μια πληρωμή PayPal, στέλνει μέσω email την πληρωμή στον λογαριασμό PayPal του πωλητή, χωρίς να διακινούνται ευαίσθητες πληροφορίες για πιστωτικές κάρτες, λογαριασμούς κλπ. μεταξύ των δυο συναλλασσόμενων μερών: το PayPal αναλαμβάνει να μεταφέρει το χρηματικό ποσό από τον λογαριασμό του αγοραστή στον λογαριασμό του πωλητή (Laudon & Traver, 2014), αξιοποιώντας τη συνεργασία του με το αυτοματοποιημένο δίκτυο διευθέτησης/εκκαθάρισης λογαριασμών (ACH). Πρόκειται για έναν αυτόνομο ιδιωτικό οργανισμό που έχει ρόλο διαμεσολαβητή των συναλλαγών και διεκπεραιώνει τις μεταφορές χρημάτων μεταξύ τραπεζών ή άλλων οικονομικών ιδρυμάτων.

Μετρητά ή **πραγματικού χρόνου** (cash or real-time): η συναλλαγή εξοφλείται με την ανταλλαγή ηλεκτρονικού χρήματος (e-currency). Στην κατηγορία αυτή πληρωμών ανήκουν τα ψηφιακά μετρητά (digital cash, e-cash), τα οποία είναι το ηλεκτρονικό ανάλογο του φυσικού χρήματος. με ενδεικτικά συστήματα πληρωμής το Bitcoin και το Ukash, καθώς και το εικονικό χρήμα (virtual currency) το οποίο χρησιμοποιείται σε εσωτερικές κοινότητες εικονικών κόσμων (πχ. Facebook credits). Οι πληρωμές αυτής της μορφής βασίζονται σε αλγορίθμους που παράγουν μοναδικά αυθεντικοποιημένα τεκμήρια (tokens) τα οποία αναπαριστούν χρηματική αξία και τα οποία χρησιμοποιούνται στις συναλλαγές. Για παράδειγμα, τα Bitcoins είναι κρυπτογραφημένοι αριθμοί που παράγονται από ένα σύνθετο αλγόριθμο, ο οποίος κάνει χρήση ομότιμων δικτύων, όπως και το PayPal που είδαμε προηγουμένως. Όμως εν αντιθέσει με το συγκεντρωτικό (centralized) χαρακτήρα του PayPal, τα Bitcoins είναι ένα **αποκεντρωμένο** (decentralized) σύστημα έκδοσης ψηφιακών μετρητών που προϋποθέτει μάλιστα μια διαδικασία εξόρυξης πολύ απαιτητική σε όρους υπολογιστικής ισχύος (Kazan et al., 2014). Σημαντικά στοιχεία των Bitcoins είναι η ανωνυμία τους (δεν απαιτούν στοιχεία ταυτοποίησης, εκτός από μια αλφαριθμητική διεύθυνση 34 χαρακτήρων την οποία ο χρήστης χρησιμοποιεί για να μπορεί να τα ανταλλάξει), και η αυξομείωση της αξίας τους, βάσει κανόνων προσφοράς-ζήτησης, όπως άλλωστε συμβαίνει σε κάθε μορφή νομίσματος (Laudon & Traver, 2014).

3.2 Το πρωτόκολλο επιπέδου ασφαλών υποδοχών Secure Sockets Layer (SSL)

Τα σχήματα ασφαλείας που χρησιμοποιούνται συνήθως σε μεθόδους ηλεκτρονικών πληρωμών ικανοποιούν τις τέσσερις ακόλουθες βασικές απαιτήσεις ασφάλειας για ασφαλείς πληρωμές:

1. **Πιστοποίηση**: μια μέθοδος επαλήθευσης της ταυτότητας (αυθεντικοποίηση) του καταναλωτή πριν να εξουσιοδοτηθεί η πληρωμή, αλλά και του εμπόρου που δέχεται την πληρωμή.
2. **Κρυπτογράφηση**: μια διαδικασία να γίνονται τα μηνύματα ακατάληπτα για όλους, εκτός εκείνων που έχουν ένα κλειδί αποκρυπτογράφησης.
3. **Ακεραιότητα**: επιβεβαίωση ότι οι πληροφορίες δε θα αλλάξουν ή δε θα καταστραφούν κατά λάθος ή σκόπιμα κατά τη μετάδοση.
4. **Μη άρνηση αποδοχής χρέους**: προστασία από την άρνηση των καταναλωτών να πληρώσουν παραγγελίες που έδωσαν, ή από την άρνηση εμπόρων ότι έλαβαν μια πληρωμή.

Από τα πιο σημαντικά και ευρέως χρησιμοποιούμενα σχήματα ασφαλείας είναι το πρωτόκολλο γενικής χρήσης SSL. Και όχι μόνο για συστήματα πληρωμών: για παράδειγμα, το πρωτόκολλο HTTPS (Hypertext Transfer Protocol Secure) που θα δούμε αναλυτικότερα τη λειτουργία του αργότερα, είναι ουσιαστικά «HTTP over SSL», δηλαδή εφαρμόζει το SSL ανάμεσα σε Web servers και σε προγράμματα πλοήγησης που επικοινωνούν με το πρωτόκολλο HTTP. Η προδιαγραφή Transport Layer Security (TLS), είναι ένα Internet Standard (RFC 5246) από την IETF (Internet Engineering Task Force) που προέκυψε ως επακόλουθο του SSL και είναι σχεδόν ταυτόσημη με την τρέχουσα έκδοση 3 του SSL (SSLv3). Γι' αυτόν τον λόγο, το πρωτόκολλο είναι γνωστό και ως SSL/TLS. Το SSL/TLS λοιπόν ανήκει στις λύσεις γενικής σχετικά χρήσης, οι οποίες στην ιεραρχία των πρωτοκόλλων βρίσκονται (και ουσιαστικά εφαρμόζουν ασφάλεια) ακριβώς επάνω από το επίπεδο Μεταφοράς (TCP/IP) και κάτω από το επίπεδο Εφαρμογής (Μάγκος, 2013), με στόχο την υποστήριξη πρωτοκόλλων υψηλότερου επιπέδου. Βεβαίως, λόγω της θέσης του, το SSL/TLS

μπορεί να βασίζεται στις ιδιότητες που εγγυάται το TCP και, για παράδειγμα, δε χρειάζεται να ασχοληθεί με την αξιόπιστη παράδοση των δεδομένων.

HTTP	Telnet	FTP	SMTP	και άλλα …
SSL/TLS				
TCP/IP				

Πίνακας 3.1 *Η θέση του πρωτοκόλλου SSL*

Σε αυτό το επίπεδο, υπάρχουν δύο επιλογές υλοποίησης. Για πλήρη γενικότητα, το SSL (ή το TLS) θα μπορούσε να παρασχεθεί ως μέρος της υποκείμενης οικογένειας πρωτοκόλλων και ως εκ τούτου να είναι διαφανές (transparent) για εφαρμογές. Εναλλακτικά, το SSL μπορεί να ενσωματωθεί σε συγκεκριμένες εφαρμογές. Για παράδειγμα, τα περισσότερα προγράμματα περιήγησης έρχονται εξοπλισμένα με SSL, και οι περισσότεροι διακομιστές Ιστού (Web servers) υλοποιούν το πρωτόκολλο (Stallings, 2014).

Το πρωτόκολλο SSL χρησιμοποιεί υβριδικό σύστημα κρυπτογράφησης, συνδυάζει δηλαδή κρυπτογραφία μυστικού και δημόσιου κλειδιού. Απλοποιημένα, η λειτουργία του βασίζεται στην ανταλλαγή συμμετρικού κλειδιού (υιοθετώντας τη λογική του αλγορίθμου ανταλλαγής κλειδιών Diffie-Hellman με τη δημιουργία ενός ψηφιακού φακέλου, όπως είδαμε στο προηγούμενο κεφάλαιο στο σχήμα 2.4). Βασικά βήματα λοιπόν είναι:

1. Ο Χρήστος επισκέπτεται τον ιστότοπο της Μαριάννας, και λαμβάνει το πιστοποιητικό του δημόσιου κλειδιού της.
2. Ο Χρήστος επιλέγει ένα μυστικό κλειδί K, το οποίο κρυπτογραφεί με το δημόσιο κλειδί της Μαριάννας PU_M. Παράγει έτσι το κρυπτογραφημένο μήνυμα Y.
3. Αποστέλλει στη Μαριάννα αυτό το κρυπτογραφημένο μήνυμα Y.
4. Η Μαριάννα αποκρυπτογραφεί το Y με το ιδιωτικό της κλειδί PR_M και αποκτά το κλειδί K.
5. Η Μαριάννα και ο Χρήστος χρησιμοποιούν το κλειδί K και έναν συμμετρικό αλγόριθμο για την επικοινωνία τους.
6. Πλέον, πχ. ο Χρήστος στέλνει τον αριθμό της πιστωτικής του κάρτας για την αγορά που θέλει να κάνει.

Το SSL είναι σχεδιασμένο ώστε να παρέχει διαφανείς (transparent) υπηρεσίες στον χρήστη. Ειδικότερα, ο ‘συνδυασμός’ των δυνατοτήτων κρυπτογράφησης που παρέχει το πρωτόκολλο SSL και του HTTP πρωτοκόλλου αναφέρεται ως πρωτόκολλο HTTPS. Η κρυπτογράφηση που χρησιμοποιείται διασφαλίζει ότι τα κρυπτογραφημένα δεδομένα δε θα μπορούν να υποκλαπούν από άλλους κακόβουλους χρήστες. Ας δούμε τη λειτουργία του πρωτοκόλλου HTTPS πιο αναλυτικά: ένας SSL Web διακομιστής (server) δέχεται μία αίτηση από έναν πελάτη (client) για «ασφαλή» σύνδεση σε μια θύρα διαφορετική από αυτήν των απλών HTTP αιτήσεων (port 80). Εξ’ ορισμού είναι η port 443. Το URL για συνδέσεις στη θύρα 443 είναι της μορφής: *https://www.server.com*. Όταν ο πελάτης συνδέεται σε αυτήν τη θύρα, αρχικοποιεί τη σύνοδο SSL. Η αρχικοποίηση αυτή αποκαλείται «χειραψία» (SSL handshake). Να σημειωθεί ότι η SSL-χειραψία χρειάζεται να πραγματοποιηθεί μόνο μια φορά: αυτό που συμβαίνει όταν ολοκληρωθεί η χειραψία είναι η δημιουργία μιας SSL συνόδου (session) κατά τη διάρκεια της οποίας η επικοινωνία πλέον κρυπτογραφείται και οι έλεγχοι ακεραιότητας εκτελούνται έως ότου εκπνεύσει η σύνοδος SSL. Κατά τη διάρκεια της SSL-χειραψίας συμβαίνουν τα ακόλουθα:

1. Ο πελάτης (προαιρετικά) και ο διακομιστής (υποχρεωτικά) ανταλλάσσουν ψηφιακά πιστοποιητικά (X.509) ώστε να αποδείξουν την ταυτότητά τους. Αυτή η ανταλλαγή ενδεχομένως περιλαμβάνει μια αλυσίδα πιστοποιητικών, έως το πιστοποιητικό ρίζας (του οποίου το δημόσιο κλειδί εμπιστεύονται όλοι οι χρήστες του συστήματος). Η αποστολή του ψηφιακού πιστοποιητικού από τον διακομιστή είναι υποχρεωτική επειδή ο πελάτης χρειάζεται να αποκτήσει το δημόσιο κλειδί του διακομιστή.
2. Ο πελάτης δημιουργεί ένα τυχαίο ζεύγος κλειδιών που θα το χρησιμοποιήσει για την κρυπτογράφηση και τον υπολογισμό των απαιτούμενων Κωδικών Αυθεντικοποίησης Μηνυμάτων (Message Authentication Codes, MACs), οι οποίοι είναι συναρτήσεις σύνοψης (hash) ειδικής μορφής που θα εξηγηθούν αμέσως μετά. Χρησιμοποιούνται ξεχωριστά κλειδιά

για τις επικοινωνίες πελάτη-διακομιστή (client write), και διακομιστή-πελάτη (server write), δηλαδή σύνολο τέσσερα κλειδιά:

- κλειδί server write: το μυστικό κλειδί που χρησιμοποιεί ο διακομιστής για να κρυπτογραφεί τα δεδομένα
- κλειδί client write: το μυστικό κλειδί που χρησιμοποιεί ο πελάτης για να κρυπτογραφεί τα δεδομένα
- κλειδί server write MAC: το μυστικό κλειδί που χρησιμοποιεί ο διακομιστής για τους απαιτούμενους υπολογισμούς MAC
- κλειδί client write MAC: το μυστικό κλειδί που χρησιμοποιεί ο πελάτης για τους απαιτούμενους υπολογισμούς MAC

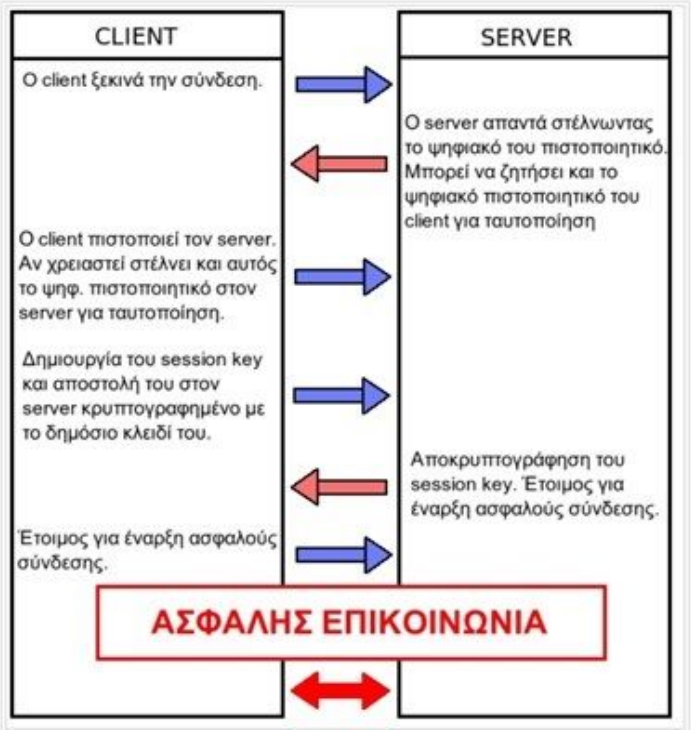
3. Τα κλειδιά κρυπτογραφούνται με το δημόσιο κλειδί του διακομιστή και αποστέλλονται με ασφάλεια στον διακομιστή.
4. Τίθενται υπό διαπραγμάτευση ένας αλγόριθμος κρυπτογράφησης και ένας αλγόριθμος (συνάρτηση) δημιουργίας τιμών hash (για την πιστοποίηση των δύο μερών και τον έλεγχο ακεραιότητας). Ο πελάτης παρουσιάζει μια λίστα με όλους τους αλγόριθμους που υποστηρίζει (πχ. κρυπτογράφηση: AES, DES,… και σύνοψη: SHA, MD5, …), και ο διακομιστής επιλέγει έναν εκ των αλγορίθμων που είναι διαθέσιμοι.

Η συνάρτηση Κώδικα Αυθεντικοποίησης Μηνύματος (MAC), χρησιμοποιείται για την αυθεντικοποίηση των πληροφοριών (δηλαδή την πιστοποίηση της ταυτότητας των μερών που επικοινωνούν) και ως προϋπόθεση έχει το ότι τα δύο μέρη που επικοινωνούν ήδη έχουν συμφωνήσει ένα μυστικό κλειδί. Ο MAC είναι γνωστός και ως συνάρτηση keyed hash, διότι συνδυάζει μια ισχυρή (ή αλλιώς κρυπτογραφική) συνάρτηση σύνοψης (hash), στις οποίες αναφερθήκαμε στο προηγούμενο ($2^{ο}$) κεφάλαιο, στην παράγραφο της Ακεραιότητας, με ένα μυστικό (συμμετρικό) κλειδί που γνωρίζουν μόνον ο αποστολέας και ο παραλήπτης.

Ειδικότερα στο SSL, οι παραγόμενοι κώδικες MAC δε χρησιμοποιούνται μόνο για την πιστοποίηση/αυθεντικοποίηση των μηνυμάτων, αλλά και για τον έλεγχο ακεραιότητάς τους. Για να επιτευχθεί αυτό, ακολουθείται μια διαδικασία δύο βημάτων στο SSL: πρώτα η αυθεντικοποίηση και μετά η κρυπτογράφηση. Συγκεκριμένα, στην περίπτωση για παράδειγμα της επικοινωνίας από SSL διακομιστή σε πελάτη, ο διακομιστής αρχικά αυθεντικοποιεί το μήνυμα Μ, υπολογίζοντας τη σύνοψη (MAC value) Τ = MAC(κλειδί server write MAC, Μ). Στη συνέχεια κρυπτογραφεί (Μ') και το μήνυμα Μ και τον MAC του Μ'=Ε(κλειδί server write, Μ+Τ).

Ο πελάτης λαμβάνοντας το Μ', έχει διασφαλισμένη την εμπιστευτικότητα του μηνύματος, αφού μόνο αυτός γνωρίζει το κλειδί server write, και έτσι λαμβάνει τα Μ και Τ. Επίσης ο πελάτης χρησιμοποιεί το Μ και το δεύτερο κλειδί του διακομιστή που γνωρίζει (το server write MAC) για να υπολογίσει από μόνος του την τιμή Τ=MAC(κλειδί server write MAC, Μ). Συγκρίνει τις δύο τιμές MAC: αν είναι ίδιες, ο πελάτης επαληθεύει την ταυτότητα του διακομιστή ως αποστολέα του μηνύματος, αφού μόνον ο διακομιστής γνωρίζει το κλειδί server write MAC, για να μπορεί να κατασκευάσει το MAC(Μ). Χάρη στη χρήση της κρυπτογραφικής συνάρτησης hash, ο αλγόριθμος επιπλέον διασφαλίζει την ακεραιότητα των μηνυμάτων που ανταλλάσσονται. Για παράδειγμα, αν το μέγεθος της τιμής εξόδου της συνάρτησης hash είναι μεγάλο, πχ. 256 bit, η εύρεση ενός μηνύματος Μ'' ώστε Η(Μ)=Η(Μ'') είναι πρακτικά αδύνατη (Μάγκος, 2013).

Gif 3.2.gif	Κινούμενη εικόνα (interactive)
Η διαδικασία της χειραψίας σύμφωνα με το πρωτόκολλο SSL	



Σχήμα 3.6 *Η διαδικασία της χειραψίας σύμφωνα με το πρωτόκολλο SSL*

Sound 3.3.mp3	Ηχητικό απόσπασμα (audio)
Δημιουργία μιας SSL συνόδου (session)	

3.3 Το πρωτόκολλο 3D-Secure

Η χρησιμοποίηση των πιστωτικών/χρεωστικών καρτών ως μέσα πληρωμών στις συναλλαγές ηλεκτρονικού εμπορίου εγκυμονεί ορισμένους κινδύνους: υπάρχει ο κίνδυνος αφενός να γίνουν γνωστά τα στοιχεία της κάρτας του πελάτη/καταναλωτή εφόσον η επιχείρηση μπορεί να τα διαβάσει (ζήτημα ιδιωτικότητας), και αφετέρου η κάρτα του καταναλωτή να είναι πλαστή, ενόσω η επιχείρηση δεν μπορεί να επιβεβαιώσει τα στοιχεία της άμεσα. Στο σημείο αυτό επεμβαίνουν πρωτόκολλα υποστήριξης πληρωμών και εκμηδενίζουν την πιθανότητα αυτή. Στόχος, η επιχείρηση να μην αποθηκεύει τα εμπιστευτικά στοιχεία του καταναλωτή, μιας και αυτό δεν είναι απαραίτητο για την ορθή λειτουργία του συστήματος ηλεκτρονικών πληρωμών. Επίσης, η επιχείρηση/έμπορος μπορεί μέσω του διατραπεζικού συστήματος χρέωσης να ελέγξει την εγκυρότητα της πιστωτικής κάρτας. Το πρωτόκολλο SSL που αναφέρθηκε προηγουμένως βρίσκεται στην καρδιά των σύγχρονων λύσεων υποστήριξης συναλλαγών μέσω καρτών, όπως θα αναδειχθεί και από την περιγραφή του πρωτοκόλλου 3D-Secure.

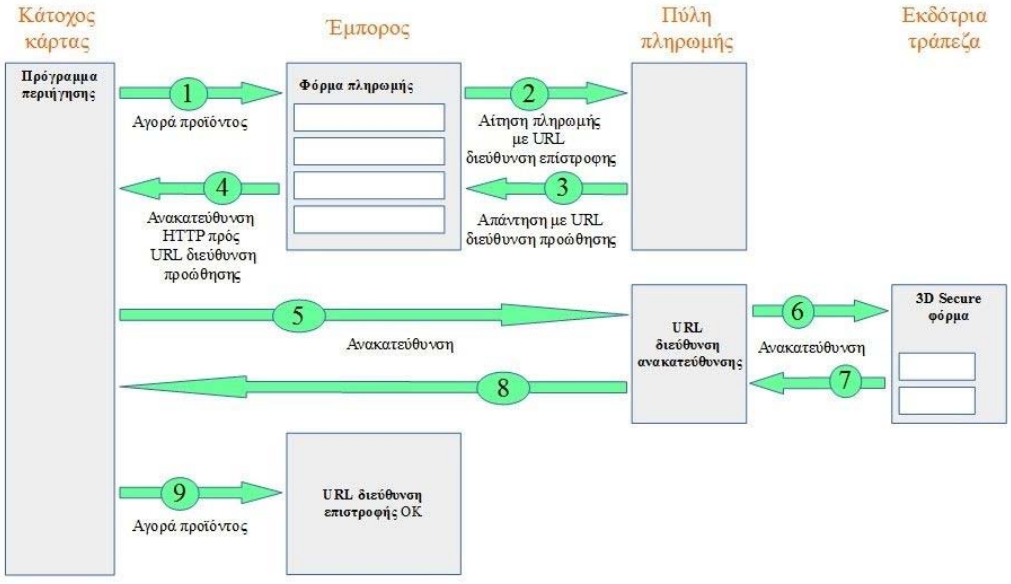
Για ‘ιστορικούς’ κυρίως λόγους, θα κάνουμε μια αναφορά για μια πρώτη προσπάθεια που έγινε στις αρχές του 2000 από τις Visa και MasterCard για να προσφέρουν συναλλαγές με πιστωτική κάρτα στον παγκόσμιο Ιστό: Ήταν το πρωτόκολλο SET (Secure Electronic Transactions). Η βασική του φιλοσοφία, της οποίας αρκετά στοιχεία μπορεί κανείς να συναντήσει και στη νεότερη προσέγγιση 3D-Secure, είναι: η επιχείρηση/έμπορος ανοίγει λογαριασμό σε τράπεζα (τράπεζα αποδέκτη, acquiring bank), η τράπεζα αποδέκτης καθορίζει ποιες κάρτες γίνονται δεκτές στις συναλλαγές, ο καταναλωτής δίνει τα στοιχεία της κάρτας του στην επιχείρηση μέσω ασφαλούς σύνδεσης του Διαδικτύου, και τέλος η επιχείρηση μεταβιβάζει με ασφάλεια τα στοιχεία που δέχτηκε στο διατραπεζικό σύστημα επεξεργασίας χρεώσεων και διαπιστώνει την πιστοληπτική ικανότητα του καταναλωτή χάρη στην αυτόματη επικοινωνία με την τράπεζα έκδοσης (issuing bank) της κάρτας του καταναλωτή. Το SET είναι πρωτόκολλο βασισμένο σε ψηφιακές υπογραφές, οπότε επιλύονται οι παρεξηγήσεις «αποποίησης παραγγελίας». Αλλά το SET απέτυχε τελικά, δεν μπόρεσε να υιοθετηθεί από τη μεγάλη μερίδα των χρηστών καρτών στον παγκόσμιο Ιστό. Αυτό συνέβη κυρίως λόγω της

επιτυχίας του SSL και της μη δυνατότητας αξιοποίησης από το SET των δυνατοτήτων του SSL. Το σχήμα 3D-Secure ήρθε ακριβώς να 'καλύψει' αυτό το κενό.

Το 3D-Secure είναι ένα XML-based πρωτόκολλο σχεδιασμένο να αποτελεί ένα πρόσθετο στρώμα ασφάλειας για τις online συναλλαγές πιστωτικών (credit) και χρεωστικών (debit) καρτών. Αναπτύχθηκε από τη Visa με σκοπό τη βελτίωση της ασφάλειας των πληρωμών στο Internet (verified by Visa service). Υπηρεσίες βασισμένες στο πρωτόκολλο έχουν επίσης υιοθετηθεί από τη MasterCard, ως MasterCard SecureCode, ενώ και η American Express έχει προσθέσει το SafeKey σε ορισμένες χώρες.

Το 3D-Secure προσθέτει ένα βήμα αυθεντικοποίησης στις online πληρωμές. Η βασική έννοια του πρωτοκόλλου είναι να 'δέσει' την οικονομική διαδικασία εξουσιοδότησης με μια διαδικασία online αυθεντικοποίησης. Η αυθεντικοποίηση στηρίζεται σε 3 μέρη (domains), εξ' ου και το 3D στο όνομα. Τα 3 μέρη είναι: Αποδέκτης (Acquirer Domain), Εκδότης (Issuer Domain), και Διαλειτουργικότητα (Interoperability Domain). Το πρωτόκολλο χρησιμοποιεί XML μηνύματα που στέλνονται πάνω από SSL συνδέσεις με client authentication (έτσι εξασφαλίζεται η αυθεντικότητα των δυο μερών, διακομιστή και πελάτη, μέσω ψηφιακών πιστοποιητικών). Μια συναλλαγή που χρησιμοποιεί διαδικασίες 'Verified by Visa' ή 'MasterCard SecureCode' εκκινεί μια ανακατεύθυνση (redirect) στον ιστότοπο της τράπεζας που εκδίδει την κάρτα για να εξουσιοδοτήσει τη συναλλαγή. Κάθε εκδότρια τράπεζα μπορεί να χρησιμοποιήσει όποια μορφή αυθεντικοποίησης επιθυμεί (το πρωτόκολλο δε δεσμεύει), αλλά συνήθως χρησιμοποιείται μια password-based μέθοδος. Έτσι, το 'buy on the Internet' ουσιαστικά σημαίνει χρήση ενός κωδικού για την κάρτα.

Η έκδοση 'Verified by Visa' του πρωτοκόλλου συστήνει τη φόρτωση της σελίδας επικύρωσης της τράπεζας σε ένα inline frame session. Με αυτόν τον τρόπο το σύστημα της τράπεζας μπορεί να διατηρεί την ευθύνη για τα περισσότερα ρήγματα ασφάλειας. Η βασική διαφορά ανάμεσα στις υλοποιήσεις Visa και MasterCard αφορά τη μέθοδο δημιουργίας του UCAF (Universal Cardholder Authentication Field). Πρόκειται για μια κρυπτογραφική τιμή που συνδέει πληροφορίες αναγνώρισης του χρήστη της κάρτας (cardholder's personal identity information) με τη συγκεκριμένη ηλεκτρονική πληρωμή. Η MasterCard χρησιμοποιεί AAV (Accountholder Authentication Value) και η Visa χρησιμοποιεί CAVV (Cardholder Authentication Verification Value). Στο 3D-Secure σχήμα, ο ACS (Access Control Server) βρίσκεται στην πλευρά των εκδοτριών τραπεζών (issuers). Οι περισσότερες τράπεζες χρησιμοποιούν τρίτα μέρη για την υποστήριξη λειτουργιών ACS, δηλαδή χρησιμοποιούν εξειδικευμένες εταιρίες ως εξωτερικούς συνεργάτες (outsourcing).



Σχήμα 3.7 *Ροή Συναλλαγής 3D-Secure*

4. Επισημάνσεις – Συμπεράσματα

Η διασφάλιση των συναλλαγών στο ηλεκτρονικό εμπόριο είναι μια αυτονόητη αναγκαιότητα: τα μεγαλύτερα εμπόδια περαιτέρω υιοθέτησης αυτών από τους καταναλωτές οφείλονται σε δισταγμούς και ανησυχίες που έχουν σχέση με ζητήματα ασφάλειας και ιδιωτικότητας. Οι συναλλαγές ηλεκτρονικού εμπορίου, καθώς χρησιμοποιούν τεχνολογίες παγκόσμιου Ιστού, απειλούνται και οι κίνδυνοι όπως το κακόβουλο λογισμικό και το ηλεκτρονικό ψάρεμα παραμένουν πάντα επίκαιροι. Η ασφάλεια σε συνδυασμό με την προστασία της ιδιωτικότητας των χρηστών έχει καταστήσει αναγκαία τη χρήση πολιτικών και μοντέλων ελέγχου προσπέλασης, τα οποία αξιοποιώντας κατάλληλες έννοιες και δομές (όπως οι κάτοχοι, οι ετικέτες, τα επίπεδα προσπέλασης, οι ρόλοι, το πλαίσιο αναφοράς, κλπ.), διασφαλίζουν την εμπιστευτικότητα, την ακεραιότητα και τη διαθεσιμότητα των εμπλεκομένων συναλλασσόμενων μερών (δεδομένα και μηχανήματα καταναλωτών, εμπόρων και άλλων ενδεχομένως ενδιάμεσων οντοτήτων).

Το κεφάλαιο αυτό κλείνει με θέματα υποστήριξης ηλεκτρονικών πληρωμών. Αναφέρονται με συντομία οι κυριότερες μορφές συστημάτων πληρωμών ηλεκτρονικού εμπορίου. Ειδική μέριμνα υπάρχει σε δύο σημαντικά σχήματα ασφαλείας που εμπλέκονται στην υποστήριξη πληρωμών ηλεκτρονικού εμπορίου: το πρωτόκολλο γενικής χρήσης SSL και το πρωτόκολλο 3D-Secure για τις online πληρωμές με χρήση πιστωτικών ή/και χρεωστικών καρτών.

Βιβλιογραφία / Αναφορές

Anders J. (2014). *The Basics of Information Security: Understanding the Fundamentals of InfoSec in Theory and Practice, 2nd Edition*, Syngress - Elsevier, ISBN: 978-0128007440.

Aycock J. (2006). *Computer Viruses and Malware*, Springer, ISBN: 978-0387341880.

Cherdantseva, Y., & Hilton, J. (2013). A reference model of information assurance & security. In *Availability, Reliability and Security* (ARES), 2013 Eighth International Conference on, IEEE, pp. 546-555.

Georgiadis, C. K., Mavridis, I., Pangalos, G., & Thomas, R. K. (2001). Flexible team-based access control using contexts, *Proc. 6th ACM symposium on Access control models and technologies*, pp. 21-27.

Gollmann, D. (2011). *Computer Security, 3rd edition*, Wiley, ISBN: 978-0470741153.

Ince, D. (2009). *Developing Distributed & E-commerce Applications, Third Edition*, Prentice Hall, ISBN: 0321417194.

Kazan, E., Tan, C.W., & Lim, E.T. (2014). Towards a Framework of Digital Platform Disruption: A Comparative Study of Centralized & Decentralized Digital Payment Providers, in *Proc. of ACIS*, Auckland, New Zealand.

Khair, M. (1996). *Design and Implementation of Secure Database Systems with Application on Healthcare Information Systems*, Ph.D. Dissertation, Aristotle University of Thessaloniki, Greece.

Kura, D. (2013). *Categorization of Large Corpora of Malicious Software*, MSc. Thesis, University of New Orleans.

Laudon, K. & Traver, C. (2014). *E-Commerce, 10/E*, Prentice Hall, ISBN: 978-0133024449.

Pfleeger, C. P., Pfleeger, S. L. & Margulies, J. (2015). *Security in Computing, Fifth Edition*, Prentice Hall, ISBN: 0134085043.

Sandhu, R. & Samarati, P. (1997). Authentication, Access Control, and Intrusion Detection, *The Computer Science and Engineering Handbook*.

Sandhu, R. (1998). Role-Based Access Control, *Advances in Computers*, Vol.46, Academic Press.

Sandhu, R., Ferraiolo, D. & Kuhn, R. (2000). The NIST Model for Role-Based Access Control: Towards a Unified Standard, in *Proceedings of the Fifth ACM Workshop on RBAC* (RBAC 2000), Berlin, Germany, pp.47-63, ACM.

Stallings, W. (2014). *Cryptography and Network Security: Principles and Practice, Sixth Edition*, Pearson, ISBN: 0-13-335469-5.

Stallings, W. (2014b). *Network Security Essentials Applications and Standards, Fith Edition*, Pearson, ISBN: 0-13-337043-7.

Thomas, K. R. (1997). Team-Based Access Control (TMAC): A Primitive for Applying Role-Based Access Controls in Collaborative Environments, in *Proceedings of the Second ACM Workshop on RBAC*, November 6-7, Fairfax, VA, USA, ACM.

Trautman, L. J. (2014). E-Commerce and Electronic Payment System Risks: Lessons from PayPal. *SMU Science and Technology Law Review* 17.2.

Virus Bulletin (2015). Glossary, http://www.virusbtn.com.

Γεωργιάδης, Χ. Κ. (2002). *Ασφάλεια Πληροφοριακών Συστημάτων και Εφαρμογές στον Έλεγχο Προσπέλασης Ιατρικών Βάσεων Δεδομένων μέσω Internet*, Διδακτορική Διατριβή, Πολυτεχνική Σχολή, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης.

Κάτσικας, Σ. (2001). *Ασφάλεια Υπολογιστών*, Ελληνικό Ανοικτό Πανεπιστήμιο, ISBN: 960–538–226–1.

Μάγκος, Ε. (2013). Ασφάλεια Υπολογιστών και Προστασία Δεδομένων, Παν. Σημειώσεις, Ιόνιο Πανεπ/μιο.

Quiz3.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Με τον όρο "κακόβουλο λογισμικό" αναφερόμαστε σε:

Α) Λογισμικό που παραβιάζει την ασφάλεια ενός συστήματος με στόχο την πρόκληση προβλημάτων στους νόμιμους χρήστες

Β) Λογισμικό που ελέγχει τις απόπειρες σύνδεσης των χρηστών για να απομονώσει τους χρήστες που έχουν ύποπτη συμπεριφορά

Γ) Όλα τα παραπάνω

Απάντηση/Λύση

Α) Λογισμικό που παραβιάζει την ασφάλεια ενός συστήματος με στόχο την πρόκληση προβλημάτων στους νόμιμους χρήστες

Κριτήριο αξιολόγησης 2

[*] Ο όρος "Έλεγχος προσπέλασης βασισμένος-σε-ρόλους" (Role-Based Access Control) αναφέρεται σε:

Α) Ανάθεση των δεδομένων ενός συστήματος σε ρόλους

Β) Ανάθεση των χρηστών σε ρόλους, όπου κάθε ρόλος έχει περιορισμούς όσον αφορά την πρόσβαση σε δεδομένα ή υπηρεσίες

Γ) Ανάθεση των διαχειριστών (Administrators) ενός συστήματος σε διαφορετικούς διαχειριστικούς ρόλους

Απάντηση/Λύση

Β) Ανάθεση των χρηστών σε ρόλους, όπου κάθε ρόλος έχει περιορισμούς όσον αφορά την πρόσβαση σε δεδομένα η υπηρεσίες

Κριτήριο αξιολόγησης 3

[*] Το «πρωτόκολλο επιπέδου ασφαλών υποδοχών» (SSL) διασφαλίζει την επικοινωνία:

Α) Ανάμεσα σε ένα Web server και σε προγράμματα πλοήγησης που επικοινωνούν με το πρωτόκολλο HTTP

Β) Ανάμεσα σε ένα server και σε εφαρμογές που επικοινωνούν με το πρωτόκολλο FTP

Γ) Ανάμεσα σε ένα server και σε εφαρμογές που επικοινωνούν με το πρωτόκολλο Telnet

Δ) Όλα τα παραπάνω

Απάντηση/Λύση
Δ) Όλα τα παραπάνω

Κριτήριο αξιολόγησης 4
[*] Ποια από τις παρακάτω, δεν είναι κατηγορία ηλεκτρονικών πληρωμών (E-payments);

Α) Πίστωσης

Β) Μετρητά

Γ) Κατά απαίτηση

Δ) Χρέωσης

Απάντηση/Λύση
Γ) Κατά απαίτηση

Κριτήριο αξιολόγησης 5
[*] Το μοντέλο ελέγχου προσπέλασης C-TMAC βασίζεται σε:

Α) RBAC+DAC

Β) RBAC+TMAC

Γ) TMAC+MAC

Απάντηση/Λύση
Β) RBAC+TMAC

Κριτήριο αξιολόγησης 6
[**] Η γενική περιγραφή του MAC πρωτοκόλλου ορίζει πως:

Α) Το s επιτρέπεται να «γράψει» στο o, εάν need-to-know(s) $\subseteq$ categories(o)

Β) Το s δεν επιτρέπεται να «διαβάσει» το o, εάν categories(o) $\subseteq$ need-to-know(s)

Γ) Το s επιτρέπεται να «γράψει» στο o, εάν categories(o) $\subseteq$ need-to-know(s)

Απάντηση/Λύση
Α) Το s επιτρέπεται να «γράψει» στο o, εάν need-to-know(s) $\subseteq$ categories(o)

Κριτήριο αξιολόγησης 7
[**] Τι είναι το πρωτόκολλο 3-D Secure;

Α) Είναι ένα πρωτόκολλο το οποίο χωρίς χρήση SSL προσφέρει ένα πρόσθετο στρώμα ασφάλειας για τις online συναλλαγές πιστωτικών ανάμεσα στον web server και στο πρόγραμμα πλοήγησης

Β) Είναι ένα HTML-based πρωτόκολλο σχεδιασμένο να αποτελεί ένα πρόσθετο στρώμα ασφάλειας για τις online συναλλαγές πιστωτικών (credit) και χρεωστικών (debit) καρτών

Γ) Είναι ένα XML-based πρωτόκολλο σχεδιασμένο να αποτελεί ένα πρόσθετο στρώμα ασφάλειας για τις online συναλλαγές πιστωτικών (credit) και χρεωστικών (debit) καρτών

Απάντηση/Λύση
Γ) Είναι ένα XML-based πρωτόκολλο σχεδιασμένο να αποτελεί ένα πρόσθετο στρώμα ασφάλειας για τις online συναλλαγές πιστωτικών (credit) και χρεωστικών (debit) καρτών

Κριτήριο αξιολόγησης 8
[*] Ποια από τις ακόλουθες δεν είναι μορφή κακόβουλου λογισμικού:

Α) Biba

Β) Bot - Zombie

Γ) Λογική Βόμβα

Δ) Ιός

Ε) Δούρειος Ίππος

Απάντηση/Λύση
Α) Biba

Κριτήριο αξιολόγησης 9
[**] Ποια από τις ακόλουθες δεν είναι λειτουργία κακόβουλου κώδικα τύπου rootkit/stealth;

Α) Απόκρυψη θύρας

Β) Απόκρυψη κλειδιού στο μητρώο

Γ) Απόκρυψη χρήστη

Δ) Απόκρυψη διαδικασίας (process)

Απάντηση/Λύση
Γ) Απόκρυψη χρήστη

Κριτήριο αξιολόγησης 10
[**] Ποια από τις ακόλουθες προτάσεις είναι σωστή;

Α) Οι κυριότερες επιθέσεις λόγω κακής σχεδίασης κώδικα από την πλευρά του πελάτη είναι η επίθεση Cross-Site Scripting (XSS) και η επίθεση Cross-Site Request Forgery (XSRF)

Β) Οι κυριότερες επιθέσεις λόγω κακής σχεδίασης κώδικα από την πλευρά του διακομιστή είναι η επίθεση Cross-site Scripting (XSS) και η επίθεση Cross-site Request Forgery (XSRF)

Γ) Οι κυριότερες επιθέσεις λόγω κακής σχεδίασης κώδικα από την πλευρά του πελάτη είναι η έλλειψη επικύρωσης εισόδου και η απόδοση ανάρμοστων δικαιωμάτων

Απάντηση/Λύση
Α) Οι κυριότερες επιθέσεις λόγω κακής σχεδίασης κώδικα από την πλευρά του πελάτη είναι η επίθεση Cross-Site Scripting (XSS) και η επίθεση Cross-Site Request Forgery (XSRF)

Κεφάλαιο 4: Εξατομίκευση και Συστήματα Συστάσεων

Σύνοψη
Το κεφάλαιο αυτό παρέχει μια επισκόπηση των συστημάτων συστάσεων, των ωφελειών που προσφέρουν τόσο στην επιχείρηση όσο και στους πελάτες αυτής, και των προκλήσεων, οι οποίες αν αντιμετωπιστούν αποτελεσματικά, μπορούν να βελτιώσουν τη διαδικασία εξατομίκευσης και για τα δύο μέρη. Γίνεται επίσης μια περιγραφή των κύριων αλγόριθμων που χρησιμοποιούν τα συστήματα αυτά καθώς και του τρόπου αξιολόγησής τους. Συζητούνται δυο ενδιαφέρουσες πτυχές των συστημάτων συστάσεων: τα ζητήματα ιδιωτικότητας που προκύπτουν και οι τρόποι αξιοποίησης δεδομένων από κοινωνικά δίκτυα. Τέλος, γίνεται μια πρώτη διερεύνηση της εφαρμογής των συστημάτων συστάσεων σε κινητά περιβάλλοντα.

Προαπαιτούμενη γνώση
Το κεφάλαιο 1 του παρόντος συγγράμματος

1. Εισαγωγή στα συστήματα συστάσεων και εξατομίκευσης

Λόγω της ταχείας ανάπτυξης του Διαδικτύου σε συνδυασμό με το πρόβλημα της συσσώρευσης πληροφοριών, η χρήση των συστημάτων συστάσεων (recommender systems) έχει αρχίσει να γίνεται απαραίτητη και για τις ηλεκτρονικές επιχειρήσεις και για τους πελάτες. Ωστόσο, υπάρχουν και άλλοι παράγοντες, όπως η προστασία της ιδιωτικής ζωής και η εμπιστοσύνη που κάνουν τους πελάτες να μη θέλουν να χρησιμοποιήσουν αυτά τα συστήματα.

Η εξέλιξη των υπολογιστών σε συνδυασμό με την ταχεία ανάπτυξη των σχετικών υποδομών δικτύωσης έφερε το Ηλεκτρονικό Εμπόριο (ΗΕ) σε ένα νέο επίπεδο παροχής υπηρεσιών. Η χρήση του Διαδικτύου αυξάνει συνεχώς και η ανάγκη για το ΗΕ γίνεται όλο και πιο μεγάλη και με διαφορετικούς τρόπους (Jannach et al., 2010). Ωστόσο, δεδομένου ότι ο όγκος των πληροφοριών μεγαλώνει και οι άνθρωποι που χρησιμοποιούν τον Παγκόσμιο Ιστό (ΠΙ) γίνονται περισσότεροι, υπάρχει η ανάγκη για να αντιμετωπιστεί η πρόκληση της παροχής διαφορετικού περιεχομένου σε χρήστες με διαφορετικά ενδιαφέροντα. Η ανάγκη για να αντιμετωπιστεί η υπερφόρτωση πληροφοριών είναι ένα από τα πιο σημαντικά στις μέρες μας προβλήματα και οδηγεί στην αναγκαιότητα χρήσης των συστημάτων συστάσεων (Konstan & Riedl, 2012).

Sound 4.1.mp3	Ηχητικό απόσπασμα (audio)
Εισαγωγή στα συστήματα συστάσεων	

Τα συστήματα συστάσεων και εξατομίκευσης ασχολούνται με τη δυναμική προσαρμογή των δεδομένων που λαμβάνονται μέσω του ΠΙ και προσαρμόζονται με βάση τις προτιμήσεις του χρήστη (Shi et al., 2014; Ricci et al., 2011). Στόχος των συστάσεων μπορεί να είναι να βοηθήσουν τον χρήστη να αποφασίσει τι να αγοράσει, ποιόν να κάνει φίλο σε ένα κοινωνικό δίκτυο ή το τι να διαβάσει (Konstan & Riedl, 2012; Polatidis & Georgiadis, 2013; Prasad & Kumari, 2012). Λόγω του μεγάλου όγκου πληροφοριών στο Διαδίκτυο, η τεχνολογία εξατομίκευσης (personalization) αποτελεί ένα από τα πιο πολύτιμα εργαλεία στις μέρες μας. Θα πρέπει να σημειωθεί ότι πρόκειται για μια πολύ απαιτητική διαδικασία: η σχεδίαση και ανάπτυξη ενός τέτοιου συστήματος απαιτεί συνδυασμό γνώσεων και δεξιοτήτων από διαφορετικούς τομείς της επιστήμης των υπολογιστών (Konstan & Riedl, 2012; Ricci, 2011). Ένας αξιόλογος αριθμός αλγορίθμων έχει αναπτυχθεί τα τελευταία χρόνια, με τους περισσότερους να χρησιμοποιούνται σε εμπορικά περιβάλλοντα.

Επιπλέον, στον τομέα των κινητών συσκευών, το πρόβλημα πρόσβασης στην πληροφορία γίνεται ακόμη πιο δύσκολο λόγω των δυσκολιών που οφείλονται σε περιορισμούς του υλικού. Είναι σημαντικό να σημειωθεί ότι οι αλγόριθμοι που εφαρμόζονται σε συστήματα που βασίζονται στον ΠΙ δεν μπορούν να μεταφερθούν αυτούσια σε συστήματα που προορίζονται για μια κινητή συσκευή, αφού υπάρχουν διαφορετικές ανάγκες, χαρακτηριστικά και περιορισμοί.

Η ανάγκη για την προστασία της ιδιωτικής ζωής έχει γίνει μια πολύ σημαντική πτυχή των τεχνικών εξατομίκευσης (Kobsa, 2007; Shyong et al., 2006; Benats et al., 2011; Jeckmans et al., 2013). Είναι βέβαια

ζωτικής σημασίας για το σύστημα το να χρησιμοποιηθούν κάποια ιδιωτικά δεδομένα προκειμένου να παραχθούν ακριβείς συστάσεις. Ωστόσο, θα πρέπει να ληφθεί υπόψη ότι η προστασία της ιδιωτικής ζωής είναι ένα τεράστιο ζήτημα των συστημάτων αυτών διότι η άρνηση των χρηστών να συμβάλλουν δίνοντας δεδομένα, καθιστά τα εξατομικευμένα περιβάλλοντα μη χρήσιμα (Jeckmans et al., 2013; Polatidis & Georgiadis, 2013). Το μεγαλύτερο μέρος των απλών χρηστών δε γνωρίζει πώς χρησιμοποιούνται τα δεδομένα αυτά και αντιδρούν με διάφορους τρόπους (Ricci, 2011).

Υπάρχουν ανοιχτά σχετικά ζητήματα που χρειάζονται εκτενή διερεύνηση, όπως οι επιχειρηματικές πτυχές των συστημάτων συστάσεων και ο περιορισμός του κόστους αναζήτησης. Επιπλέον θα πρέπει να διερευνηθούν και άλλοι παράμετροι όπως η επίγνωση θέσης και η αξιοποίηση γενικότερα της περιβάλλουσας κατάστασης (context) από τα συστήματα συστάσεων (Konstan & Riedl, 2012).

2. Η σημασία των συστημάτων συστάσεων για τις ηλεκτρονικές επιχειρήσεις

Τα συστήματα συστάσεων και εξατομίκευσης είναι αλγόριθμοι που χρησιμοποιούνται ευρέως στο ηλεκτρονικό εμπόριο για να προτείνονται προϊόντα ή υπηρεσίες σε χρήστες. Οι συστάσεις είναι για το τι αγορές να γίνουν, τι ανάγνωση ειδήσεων να γίνει, τι συνδέσεις κοινωνικής δικτύωσης να γίνουν και τι ταινίες να παρακολουθήσει ο χρήστης μεταξύ πολλών άλλων. Ανάμεσα στις πιο δημοφιλείς ιστοσελίδες που χρησιμοποιούν συστήματα συστάσεων είναι η Amazon.com, η οποία παρέχει μια εξατομικευμένη ιστοσελίδα για κάθε μεμονωμένο χρήστη. Το Netflix είναι ένα άλλο παράδειγμα ιστοσελίδας που χρησιμοποιεί εξατομικευμένα συστήματα για να προταθούν ταινίες και τηλεοπτικές εκπομπές. Τέτοια συστήματα γενικά δείχνουν μια λίστα με κορυφαία αντικείμενα που σχετίζονται με τον χρήστη. Τα στοιχεία που ανακτώνται είναι σύμφωνα με τους κανόνες που καθορίζονται από τον αλγόριθμο και προτείνονται ως κορυφαία από τη λίστα, ανάλογα με το περιβάλλον. Τα συστήματα συστάσεων αναπτύχθηκαν για να κάνουν τις καθημερινές αποφάσεις απλούστερες. Οι αποφάσεις αυτές είναι ως επί το πλείστον για υπηρεσίες χαμηλού κόστους, όπως τα βιβλία και οι προτάσεις ταινιών, με πρωταρχικό στόχο τη διευκόλυνση του χρήστη στη διαδικασία της αναζήτησης (Jannach et al., 2010; Ricci et al., 2011).

Sound 4.2.mp3	Ηχητικό απόσπασμα (audio)
Σημασία των συστημάτων συστάσεων	

Αν και τα συστήματα συστάσεων και εξατομίκευσης αποτελούν ένα σχετικά νέο πεδίο μελέτης στη βιβλιογραφία, τεχνικές τους έχουν ευρέως υιοθετηθεί και έχουν λύσει καλά ως ένα βαθμό το πρόβλημα της υπερφόρτωσης πληροφοριών (Oulasvirta et al., 2012). Να σημειωθεί ότι η χρήση των συστημάτων συστάσεων είναι απαραίτητη για τους παρόχους υπηρεσιών και όχι μόνο για τους χρήστες (Polatidis & Georgiadis, 2013; Karimov & Brengman, 2011). Οι λόγοι για τους οποίους οι ηλεκτρονικές επιχειρήσεις χρησιμοποιούν τέτοια συστήματα είναι οι εξής (Polatidis & Georgiadis, 2013):

1. Αύξηση πωλήσεων: Ο πιο σημαντικός λόγος για να χρησιμοποιηθεί μια τεχνολογία συστάσεων είναι η αύξηση των πωλήσεων και των εσόδων του. Αυτό επιτυγχάνεται επειδή το σύστημα συστάσεων προτείνει συνήθως τα στοιχεία που είναι σχετικά με το χρήστη, σύμφωνα με το ιστορικό και τις προτιμήσεις του.
2. Προώθηση ειδών που ανήκουν σε μία ευρύτερη γκάμα προϊόντων: Ένα σύστημα συστάσεων θα προτείνει συνήθως τα στοιχεία από μια μεγάλη ποικιλία προϊόντων που διαφορετικά ο χρήστης θα ήταν πολύ δύσκολο να εντοπίσει.
3. Αύξηση της ικανοποίησης των χρηστών: Ο χρήστης είναι περισσότερο ικανοποιημένος από τη συνολική υπηρεσία που προσφέρεται και είναι πιθανό να την προτείνει και σε άλλους.
4. Αύξηση πίστης: Είναι πιο πιθανό για έναν χρήστη να επισκεφθεί ξανά μια ιστοσελίδα ή να ξανά χρησιμοποιήσει μια κινητή εφαρμογή, αν είναι ικανοποιημένος με την ποιότητα και γενικότερα την εμπειρία διάδρασης.

Σύμφωνα με τους Hinz & Eckert (2010) τα δύο πιο σημαντικά ζητήματα για μια ηλεκτρονική επιχείρηση που χρησιμοποιεί συστήματα συστάσεων είναι:

5. Μείωση του κόστους αναζήτησης
6. Υψηλότερες πωλήσεις

Ως εκ τούτου είναι σαφές σε αυτό το στάδιο ότι οι επιχειρήσεις που θέλουν να είναι καινοτόμες, να αυξήσουν τις πωλήσεις τους και να είναι πιο αξιόπιστες για τον πελάτη, πρέπει να παρέχουν πιο εξατομικευμένες υπηρεσίες. Η εξατομίκευση του ίδιου του λειτουργικού συστήματος μπορεί να συνεισφέρει στην παραγωγή πιο κατάλληλων συστάσεων (Davidson & Livshits, 2012). Επιπλέον, οι κινητές συσκευές τείνουν να γίνουν η κύρια πηγή πρόσβασης σε κοινωνικά δίκτυα διότι υποστηρίζουν άμεση πρόσβαση από παντού στον ΠΙ (Jabeur et al., 2013; Oulasvirta et al., 2012).

Σε μια μελέτη τους οι Karimov & Brengman (2011) δείχνουν ότι τα συστήματα συστάσεων μπορούν να αποφέρουν σημαντικό κέρδος σε μια ηλεκτρονική επιχείρηση. Ωστόσο, οι ίδιοι ερευνητές διαπίστωσαν ότι μόνο το 1,4% των εσόδων των 210 κορυφαίων ιστοσελίδων οφείλεται στα συστήματα συστάσεων, διότι οι περισσότερες ιστοσελίδες δε χρησιμοποιούσαν κάποιο. Για να γίνει πιο αποτελεσματική η εξατομίκευση, πρέπει να δίνει τη δυνατότητα σε έναν ηλεκτρονικό πωλητή να αλληλοεπιδρά αυτόματα με τους πιθανούς πελάτες και να τους προσφέρει μια ποικιλία υπηρεσιών, αυξάνοντας την ικανοποίηση των πελατών (Riemer & Totz, 2001). Οι επιλογές που προσφέρονται από τα συστήματα συστάσεων πρέπει να προσαρμόζονται στις ανάγκες της κάθε επιχείρησης.

Η αξιοποίηση των μέσων κοινωνικής δικτύωσης είναι ένας σοβαρός παράγοντας επιτυχίας στις ηλεκτρονικές επιχειρήσεις. Οι κορυφαίες ιστοσελίδες προσπαθούν να αναπτύξουν μια κοινωνική παρουσία, με τη δημιουργία σελίδων σε δίκτυα όπως το Facebook και το Twitter. Αν και υπάρχουν αρκετές δυσκολίες, τα δεδομένα από τα κοινωνικά δίκτυα θα πρέπει να ενσωματώνονται σε συστήματα συστάσεων προκειμένου να βελτιώσουν τις συστάσεις, την ικανοποίηση των χρηστών και την εμπιστοσύνη σε ένα ηλεκτρονικό κατάστημα. Έχει καταγραφεί ότι όταν τα συστήματα συστάσεων χρησιμοποιούνται τότε η εμπιστοσύνη του πελάτη προς ένα ηλεκτρονικό κατάστημα είναι υψηλότερη (Qiu & Benbasat, 2009; Wang et al., 2007). Σε μια έρευνα τους οι Ochi et al. (2010) παρουσίασαν στοιχεία που δείχνουν ότι οι συστάσεις επηρεάζονται από τα δεδομένα των κοινωνικών δικτύων (χρησιμοποιείται τότε και ο όρος "κοινωνικές συστάσεις").

3. Αλγόριθμοι συστημάτων συστάσεων και εξατομίκευσης

Ένα σύστημα συστάσεων παίρνει συνήθως ως είσοδο προσωπικές πληροφορίες από τον χρήστη, χρησιμοποιώντας έναν αλγόριθμο δημιουργεί τις συστάσεις, είτε τοπικά είτε σε ένα κατανεμημένο περιβάλλον και προσφέρει τις "προβλέψεις" στη διασύνδεση των υπηρεσιών που ο χρήστης χρησιμοποιεί. Οι δύο πιο σημαντικοί και ευρέως χρησιμοποιούμενοι αλγόριθμοι συστάσεων είναι οι εξής:

Συνεργατικό φιλτράρισμα (Collaborative filtering)

Η βασική ιδέα είναι να βρούμε ποιοι χρήστες μοιράζονται τα ίδια ενδιαφέροντα με τον ενδιαφερόμενο χρήστη στο παρελθόν. Η κατηγορία αυτή των αλγορίθμων εξατομίκευσης στηρίζεται στο ότι οι χρήστες που έχουν παρόμοιες προτιμήσεις, βαθμολογούν και αξιολογούν με παρόμοιο τρόπο. Οι τεχνικές αυτές συνήθως λαμβάνουν ένα σύνολο με τις βαθμολογίες των χρηστών του συστήματος και παράγουν προβλέψεις σχετικά με το τι χρειάζεται ένας χρήστης, βασιζόμενες στους πιο κοντινούς (ως προς τις προτιμήσεις) σε αυτόν χρήστες (Jannach et al., 2010; Polatidis & Georgiadis, 2013).

Ο ακόλουθος πίνακας αντιπροσωπεύει μια βάση δεδομένων με αξιολογήσεις για την Αλίκη και τέσσερις άλλους χρήστες.

Users	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	Null
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4

User4	1	5	5	2	1

Πίνακας 4.1 *Βάση δεδομένων με αξιολογήσεις*

Ένας αριθμός μεθόδων ομοιότητας υφίστανται. Η μέθοδος ομοιότητας Pearson (Pearson correlation similarity) χρησιμοποιείται ευρέως. Μια άλλη μέθοδος ομοιότητας είναι η ομοιότητα συνημίτονου (Cosine similarity). Κάθε μέθοδος επιστρέφει έναν αριθμό από το -1 ως το 1. Όσο μεγαλύτερος είναι ο αριθμός αυτός τόσο μεγαλύτερη είναι η ομοιότητα ανάμεσα στους χρήστες. Η εικόνα 4.1 αποτυπώνει την ομοιότητα Pearson ενώ η εικόνα 4.2 αποτυπώνει την ομοιότητα συνημιτόνου.

$$Sim\ (a,b) = \frac{\sum p \in P(ra, p - \bar{r}a)(rb, p - \bar{r}b)}{\sqrt{\sum p \in P(ra, p - \bar{r}a)^2}\sqrt{\sum p \in P(rb, p - \bar{r}b)^2}}$$

Εικόνα 4.1 *Pearson correlation similarity*

$$Sim\ (a,b) = \frac{\sum p \in P(ra, p - \overline{ra})(rb, p - \overline{rb})}{\sqrt{\sum p \in P(ra, p - \overline{ra})^2}\sqrt{\sum p \in P(rb, p - \overline{rb})^2}}$$

Εικόνα 4.2 *Cosine similarity*

Sim (a,b) είναι η ομοιότητα ανάμεσα στον χρήστη a και στον χρήστη b, r a,p είναι η βαθμολογία του χρήστη a για το προϊόν p, ενώ r b,p είναι η βαθμολογία του χρήστη b για το προϊόν p. Το P είναι το σύνολο των προϊόντων.

Φιλτράρισμα με βάση το περιεχόμενο (**Content-based filtering)**
Η εξατομίκευση με βάση το περιεχόμενο δε χρησιμοποιεί στοιχεία που αφορούν άλλους χρήστες πέραν του τρέχοντος χρήστη για τον οποίο παράγονται συστάσεις. Βασίζεται σε μεταδεδομένα (metadata) των πραγματικών δεδομένων, δηλαδή των προϊόντων/υπηρεσιών για τα οποία παράγονται συστάσεις. Αυτά μπορούν να είναι κάποια τεχνική περιγραφή, το είδος της ταινίας, ο τίτλος, τύπος, συγγραφέας ή άλλο καθορισμένο σύνολο των λέξεων-κλειδιών (Jannach et al., 2010). Ένας κατάλογος των ιδιοτήτων των προϊόντων/υπηρεσιών διατηρείται. Ο αλγόριθμος στηρίζει τη λειτουργία του με τον εντοπισμό προϊόντων/υπηρεσιών τα οποία έχουν ιδιότητες παρόμοιες με αυτές που ένας χρήστης προτίμησε στο παρελθόν. Είναι αποτελεσματικός σε συστήματα όπως πύλες ειδήσεων και καταστήματα βιβλίων, δηλαδή σε συστήματα που υπάρχει προφανής κατηγοριοποίηση των στοιχείων. Επιπλέον, η χρησιμότητά του γίνεται ακόμη μεγαλύτερη όταν δεν υπάρχουν αρκετές αξιολογήσεις για να εκτελεστεί αλγόριθμος κατηγορίας συνεργατικού φιλτραρίσματος.

Φιλτράρισμα με βάση τη γνώση (Knowledge-based filtering)
Σε ένα μέρος της βιβλιογραφίας, αναφέρεται ως διακριτή τρίτη κατηγορία το φιλτράρισμα με βάση τη γνώση. Τα συστήματα αυτά, βασίζονται σε δεδομένα που παρέχονται από τον χρήστη και χρησιμοποιούνται ως περιορισμοί, προκειμένου να του δοθούν συστάσεις.

Name	Price	Mega Pixels	Zoom	Screen size	Quality
Camera1	100	6	2x	2cm	Low
Camera2	119	8	2x	2.5cm	Medium
Camera3	200	12	4x	3cm	High
Camera4	150	10	3x	3cm	Medium

Camera5	140	8	4x	2.7cm	Medium

Πίνακας 4.2 *Τεχνικά χαρακτηριστικά ψηφιακής κάμερας*

Ο παραπάνω πίνακας περιγράφει τα χαρακτηριστικά ενός συνόλου δεδομένων σχετικά με τις ψηφιακές φωτογραφικές μηχανές, όπως διαπιστώθηκε στη βάση δεδομένων του συστήματος. Ο χρήστης μπορεί στη συνέχεια να προσθέσει ένα σύνολο ειδικών κανόνων για το σύστημα προκειμένου να λάβει συστάσεις. Εξετάστε το ακόλουθο παράδειγμα που περιγράφεται με τη χρήση κανόνων λογικής:

(Price <= 150) AND (MegaPixels >= 10) AND (Quality >= Medium)

Το σύστημα θα εμφανίσει όλες τις κάμερες με τις τιμές τους να είναι μικρότερες ή ίσες με 150 και τα mega pixel τους μεγαλύτερα από ή ίσο με 10 και η ποιότητά τους να είναι τουλάχιστον μέση. Στην περίπτωση αυτή, η κάμερα νούμερο 4 θα εμφανιστεί διότι είναι η μόνη που έχει τις προϋποθέσεις.

Υβριδικοί αλγόριθμοι

Ο υβριδισμός είναι ο συνδυασμός ή η χρήση στοιχείων από διαφορετικούς αλγόριθμους, προκειμένου να παραχθούν συστάσεις. Οι υβριδικοί αλγόριθμοι έχουν κατηγοριοποιηθεί σύμφωνα με τον σχεδιασμό τους (Jannach et al., 2010):

- **Monolithic:** Ένας μονολιθικός αλγόριθμος ενσωματώνει διαφορετικά χαρακτηριστικά/ παραμέτρους από διαφορετικούς αλγόριθμους. Θα μπορούσε να περιλαμβάνει κάθε χαρακτηριστικό που ένας αλγόριθμος υποστηρίζει ή έναν αριθμό από αυτά. Τα δεδομένα εισάγονται προς επεξεργασία υπό ένα κοινό σύνολο παραμέτρων και οι συστάσεις παρέχονται σε ένα κοινό σημείο εξόδου.
- **Parallelized:** Ένα σύνολο τουλάχιστον δύο ή και παραπάνω αλγόριθμων τρέχει παράλληλα και στο τέλος οι συστάσεις ενσωματώνονται σε ένα κοινό σημείο εξόδου.
- **Pipelined:** Ένας τέτοιος αλγόριθμος αποτελείται από δύο ή και παραπάνω αλγόριθμους. Κάθε αλγόριθμος τρέχει και οι συστάσεις που παρέχονται στο σημείο εξόδου χρησιμοποιούνται από τον επόμενο αλγόριθμο ως στοιχεία εισόδου, μέχρι τον τελευταίο.

Sound 4.3.mp3	Ηχητικό απόσπασμα (audio)
Κοινά χαρακτηριστικά των αλγόριθμων συστάσεων	

4. Οφέλη ηλεκτρονικών επιχειρήσεων

Μελέτες έχουν δείξει ότι οι πελάτες εκτιμούν τα συστήματα συστάσεων και αισθάνονται πιο άνετα όταν επισκέπτονται μια ιστοσελίδα που είναι προσαρμοσμένη στις ανάγκες τους (ChoiceStream Personalization Survey, 2008). Αισθάνονται περισσότερη εμπιστοσύνη προς τον πωλητή και είναι πιθανό να τον επισκεφθούν ξανά και να κάνουν περισσότερες αγορές. Τα οφέλη για μία ηλεκτρονική επιχείρηση που χρησιμοποιεί εξατομικευμένα συστήματα είναι σημαντικά όταν συγκρίνονται με μη-εξατομικευμένες ιστοσελίδες, και είναι κυρίως οικονομικά. Υπάρχουν πράγματι πολλές μελέτες που δείχνουν ότι τα συστήματα συστάσεων είναι αρκετά κερδοφόρα για τις ηλεκτρονικές επιχειρήσεις (Wu et al., 2011; Cooperstain et al., 1999; Hinz & Eckert, 2010).

Σε ένα χαρακτηριστικό άρθρο του το CNN (Mangalindan, 2012) έχει συγκεντρώσει πληροφορίες για μία από τις πλέον σημαντικές ηλεκτρονικές επιχειρήσεις στον κόσμο, την Amazon, η οποία χρησιμοποιεί συστήματα συστάσεων και έχει να επιδείξει αύξηση στις συνολικές πωλήσεις της κατά 29% το 2011. Τα συστήματα συστάσεων μπορούν να μειώσουν το κόστος αναζήτησης για τις επιχειρήσεις, παρέχοντας στον χρήστη τα πιο κατάλληλα προϊόντα. Σύμφωνα με τους Wu et al. (2011), επειδή τα ηλεκτρονικά καταστήματα έχουν συνήθως μια μεγαλύτερη ποικιλία προϊόντων, τα συστήματα συστάσεων μπορούν να μειώσουν το κόστος των συναλλαγών με τη στόχευση προς τους χρήστες: προτείνουν οποιοδήποτε σχετικό προϊόν σε κάθε πελάτη, ο οποίος (ο ίδιος ή άλλοι πελάτες με ίδια ενδιαφέροντα με αυτόν) μπορεί να έχει δείξει προτίμηση σε παρόμοια προϊόντα. Σύμφωνα με την ίδια έρευνα, η σύσταση μπορεί να γίνει σε διάφορα στάδια της

διαδικασίας πώλησης. Οι Fleder και Hosanager (2009) σε μελέτη τους έδειξαν ότι τα συστήματα συστάσεων μπορούν να αυξήσουν την πώληση των νέων προϊόντων προς τους πελάτες.

Οι Dias et al. (2008) έχουν δείξει μέσα από μία περίπτωση μελέτης 21 μηνών με πραγματικά δεδομένα ότι τα συστήματα συστάσεων όχι μόνο αυξάνουν τα οικονομικά οφέλη άμεσα, αλλά πάνε πολύ πέρα από αυτό. Υποστηρίζουν ότι υπάρχει άμεση σχέση των συστημάτων συστάσεων με τα επιπλέον έσοδα, τα οποία σχετίζονται με την αγορά ενός προϊόντος και έμμεσα επιπλέον έσοδα που σχετίζονται με την αγορά ενός προϊόντος που βρίσκεται στην ίδια κατηγορία με το προτεινόμενο προϊόν. Και επιχειρηματολογούν ότι η έμμεση αξία πωλήσεων παραμένει σταθερά υψηλότερη από την άμεση.

4.1 Η προστασία της ιδιωτικής ζωής και η εμπιστοσύνη σε σχέση με τα οφέλη

Η μελέτη των Chellappa and Sin (2005) έδειξε ότι η εξατομίκευση είναι ένας πολύ σημαντικός παράγοντας για τους πελάτες όταν επισκέπτονται μια ιστοσελίδα και είναι πιθανό να κερδίσει την εμπιστοσύνη τους προς την κατεύθυνση του ηλεκτρονικού επιχειρείν. Τελικά, θα κάνουν περισσότερες αγορές και ως εκ τούτου το οικονομικό όφελος για τον πωλητή θα είναι υψηλότερο. Παράλληλα όμως, έρευνες σχετικά με την ιδιωτικότητα (privacy) έχουν δείξει ότι οι πελάτες θέλουν να ξέρουν πώς θα χρησιμοποιηθούν από την επιχείρηση τα στοιχεία που θα δώσουν (Kobsa & Teltzrow, 2005; Turow, 2003).

Sound 4.4.mp3	Ηχητικό απόσπασμα (audio)
Προστασία ιδιωτική ζωής στα συστήματα συστάσεων	

Ο Kobsa (2007) υπογραμμίζει ότι θα πρέπει να ληφθούν σοβαρά υπόψη οι νόμοι προστασίας της ιδιωτικής ζωής: η προστασία της ιδιωτικής ζωής είναι ζωτικής σημασίας για την εμπιστοσύνη, οπότε η παρουσία μίας δήλωσης προστασίας προσωπικών δεδομένων στο δικτυακό τόπο συναλλαγών κρίνεται απαραίτητη.

Ένας άλλος πολύ σημαντικός παράγοντας στη διαδικασία της δημιουργίας εμπιστοσύνης μεταξύ μιας ηλεκτρονικής επιχείρησης και ενός πελάτη είναι οι θετικές εμπειρίες. Μια μελέτη του Pavlou (2003) έχει δείξει ότι η εμπειρία παίζει ζωτικό ρόλο για την εμπιστοσύνη στις ιστοσελίδες. Ο σχεδιασμός και η λειτουργία ενός δικτυακού τόπου επηρεάζουν το κλίμα εμπιστοσύνης. Για να αυξηθεί η εμπιστοσύνη θα πρέπει να μειωθούν τα σφάλματα και η όλη διαδικασία της αγοράς να είναι εύχρηστη.

Άλλοι παράγοντες περιλαμβάνουν τη συνολική φήμη του δικτυακού τόπου (Schoenbachler & Gordon, 2002). Επίσης, ο Fogg (2002) αναφέρει ότι όσο περισσότερες είναι οι πληροφορίες σχετικά με την επιχείρηση τόσο μεγαλύτερη είναι η εμπιστοσύνη και ότι οι γρήγορες απαντήσεις σε διάφορα ερωτήματα δρουν θετικά. Τέλος, μια μελέτη του Turow (2003) έχει δείξει ότι οι οικονομικές ανταμοιβές για τους πελάτες μπορούν να αυξήσουν την εμπιστοσύνη τους και την πρόθεσή τους να εισάγουν περισσότερες ιδιωτικές πληροφορίες στην ιστοσελίδα.

5. Περίπτωση μελέτης βασισμένη στην ηλεκτρονική επιχείρηση Amazon

Η Amazon είναι ένα από τα κορυφαία ηλεκτρονικά καταστήματα που έχει χρησιμοποιήσει συστήματα συστάσεων με επιτυχία για να αυξήσει τα οικονομικά οφέλη της. Παρέχει με διαφορετικό τρόπο συστάσεις σε διάφορα στάδια της διαδικασίας αγοράς. Όπως φαίνεται στην εικόνα 4.3 παρακάτω, εξατομικευμένες συστάσεις εμφανίζονται αφότου ο χρήστης έχει εισέλθει με επιτυχία στο Amazon και στο στάδιο αυτό παρέχονται συστάσεις σύμφωνα με προηγούμενες αγορές του. Οι υπολογισμοί γίνονται και προς τις δυο κατευθύνσεις: με βάση αλγορίθμους βασισμένους στο περιεχόμενο υπολογίζονται ομοιότητες ανάμεσα στα αντικείμενα που έχουν καταγραφεί στο ιστορικό του ως αγορές ή ακόμη και ως θετικές αξιολογήσεις και στα διαθέσιμα προς πώληση αντικείμενα. Και προτείνονται βέβαια τα πιο 'σχετικά'. Επίσης, υπολογισμοί γίνονται και με βάση αλγορίθμους συνεργατικού φιλτραρίσματος: υπολογίζονται ομοιότητες ανάμεσα στο προφίλ του χρήστη (δημογραφικά στοιχεία, ρητές δηλώσεις προτιμήσεων και καταγεγραμμένη στο δικτυακό τόπο συμπεριφορά/προτιμήσεις) με τα προφίλ άλλων καταγεγραμμένων χρηστών. Και προτείνονται οι πρόσφατες επιλογές των πιο ΄κοντινών' ως προς τις προτιμήσεις χρηστών. Υπάρχουν και γενικές συστάσεις, με βάση την έκπτωση και την εμπορική επιτυχία αντικειμένων (στη δεξιά πλευρά).

Εικόνα 4.3 *Οι συστάσεις όπως εμφανίζονται στην αρχική σελίδα της Amazon*

Η εικόνα 4.4 παρουσιάζει τις προσφερόμενες συστάσεις, αφού έκανε μια αγοραστική επιλογή ο χρήστης. Εμφανίζονται δύο διαφορετικοί τύποι συστάσεων: αντικείμενα που συχνά αγοράστηκαν μαζί με αυτό που μόλις επέλεξε ο χρήστης και τι άλλο έχουν αγοράσει άλλοι πελάτες που έχουν αγοράσει αυτό που επέλεξε ο χρήστης.

Εικόνα 4.4 *Οι συστάσεις όπως εμφανίζονται κατά τη διάρκεια επιλογής προϊόντος*

6. Προκλήσεις

Στις επόμενες παραγράφους θα συζητηθούν δυο ενδιαφέροντα θέματα της σχετικής βιβλιογραφίας των συστημάτων συστάσεων.

6.1 Ιδιωτικότητα

Οι χρήστες ανησυχούν για τα στοιχεία που χρειάζεται να προσφέρουν στα συστήματα συστάσεων. Για τον λόγο αυτό, αναπτύσσονται κατάλληλες πολιτικές προστασίας προσωπικών δεδομένων, που στοχεύουν να διασφαλίσουν ότι τα δεδομένα χρήστη θα παραμείνουν απολύτως ιδιωτικά. Χωρίς την πρόνοια αυτών, οι

χρήστες τείνουν να παρουσιάζουν αρνητική συμπεριφορά, όταν τους ζητείται να εισάγουν τα δεδομένα τους προκειμένου να λάβουν πιο εξατομικευμένο περιεχόμενο (Shyong et al., 2006; Polatidis & Georgiadis, 2013; Jeckmans et al., 2013). Στα συστήματα συστάσεων, οι χρήστες χωρίζονται σε τρεις βασικές κατηγορίες, όσον αφορά τις αποφάσεις και τις επιλογές τους σχετικά με την ιδιωτικότητα (Shyong et al., 2006):

1. Ο χρήστης που θα δώσει κάθε είδους πληροφορία σε ένα σύστημα συστάσεων με αντάλλαγμα περισσότερο εξατομικευμένο περιεχόμενο.
2. Ο χρήστης που θα δώσει κάποιες πληροφορίες για ένα σύστημα συστάσεων, προκειμένου να πάρει βελτιωμένες προτάσεις.
3. Ο χρήστης που δε θα δώσει καμία πληροφορία σε ένα σύστημα συστάσεων, λόγω των ανησυχιών του για την ιδιωτικότητά του.

Η κατηγορία χρηστών που επιτρέπει εισαγωγή κάποιων πληροφοριών, περιορίζεται σε γενικά στοιχεία όπως το φύλο, η ηλικία και η εκπαίδευση. Αυτά τα δεδομένα δίνονται ευκολότερα σχετικά με περισσότερο συγκεκριμένα προσωπικά δεδομένα (Jeckmans et al., 2013). Ωστόσο, για να βελτιωθούν οι συστάσεις πρέπει να πεισθούν οι χρήστες να εισάγουν περισσότερα στοιχεία. Και αυτό σημαίνει να πεισθούν ότι τα δεδομένα τους θα είναι ασφαλή. Υπάρχουν διάφορες μέθοδοι διαφύλαξης της ιδιωτικότητας στα συστήματα συστάσεων, που στηρίζονται σε τεχνικές εξόρυξης δεδομένων Ιστού (Web mining) και στις οποίες θα αναφερθούμε σε επόμενο κεφάλαιο, εστιάζοντας στα κινητά συστήματα συστάσεων.

Σε ορισμένες πάντως περιπτώσεις, η διεξαγωγή έρευνας σχετικά με τις ανησυχίες ιδιωτικότητας των τρεχόντων χρηστών, βοηθά στη διερεύνηση των λύσεων που θα ακολουθηθούν. Η έρευνα αυτή μπορεί να γίνει είτε σε ένα εργαστήριο, χρησιμοποιώντας τεχνικές παρατήρησης, είτε σε πραγματικό περιβάλλον, ζητώντας από τους χρήστες να απαντήσουν σε κατάλληλες ερωτήσεις. Επίσης, βοηθά στην 'διασκέδαση' των όποιων ανησυχιών, το να καταστούν απολύτως σαφή προς τους χρήστες τα οφέλη της εξατομίκευσης, και βεβαίως η παροχή μιας πολύ σαφούς, πιστοποιημένης, δήλωσης προστασίας προσωπικών δεδομένων. Οι χρήστες θέλουν να είναι σίγουροι για τον τρόπο χρήσης των δεδομένων τους.

6.2 Ενσωμάτωση δεδομένων από κοινωνικά δίκτυα

Η ενσωμάτωση των κοινωνικών δικτύων στα κινητά λειτουργικά συστήματα, σε συνδυασμό με την ανάπτυξη και την ταχύτητα του Internet έχει οδηγήσει στην παραγωγή τεράστιων ποσοτήτων δεδομένων κοινωνικής δικτύωσης. Για τις επιχειρήσεις είναι ένας από τους ευκολότερους τρόπους συλλογής δεδομένων: οι χρήστες είναι πρόθυμοι να μοιραστούν απόψεις, αξιολογήσεις κλπ. Ωστόσο, η επιλογή και ο διαχωρισμός των χρήσιμων στοιχείων για τα συστήματα εξατομίκευσης και γενικότερα για τη βελτίωση των διαδικασιών ΗΕ, είναι μια διαδικασία γεμάτη προκλήσεις. Τα δεδομένα από τα κοινωνικά δίκτυα θα πρέπει να μπορούν να χρησιμοποιηθούν αποτελεσματικά για να βοηθηθούν τελικά οι πελάτες μέσω της παραγωγής πιο εύστοχων συστάσεων προς αυτούς.

Ένα τεράστιο ποσό δεδομένων δημιουργείται στα κοινωνικά δίκτυα καθημερινά και αυτό είναι μια τάση που αυξάνεται με γεωμετρική πρόοδο. Ως ενδεικτικό ποσοστό, ας αναφέρουμε ότι ο αριθμός των χρηστών Facebook και Twitter αυξήθηκε κατά 112% και 347% αντίστοιχα την περίοδο Ιανουάριος 2009 - Ιανουάριος 2010 (Jabeur et al., 2013). Ωστόσο, έχουν αναφερθεί και ζητήματα ποιότητας των στοιχείων: υπάρχει ένας αριθμός κακόβουλων χρηστών που δημιουργούν δεδομένα και αλλοιώνονται έτσι τα στοιχεία των πραγματικών χρηστών (Gundecha et al., 2012).

Τα κοινωνικά δίκτυα προσφέρουν διεπαφές προγραμματισμού εφαρμογών (APIs) που μπορούν να χρησιμοποιηθούν για την επικοινωνία μαζί τους, και την εξόρυξη πληροφοριών χρήσης τους. Εκτός από τις πληροφορίες που σχετίζονται με τον χρήστη (όπως φύλο, ηλικία, υπόβαθρο, κατάσταση σχέσης), υπάρχει ένας αριθμός (πολύ χρήσιμων για την εξατομίκευση) δυναμικών στοιχείων που μεταβάλλεται συνεχώς. Σε αυτό το δυναμικό περιεχόμενο περιλαμβάνονται ενδεικτικά η διάθεση, η τοποθεσία, οι θέσεις και τα μηνύματα που στάλθηκαν από άλλους χρήστες (Liu & Maes, 2004). Κάνοντας όμως σύνδεση και με τα ζητήματα ιδιωτικότητας της προηγούμενης παραγράφου, απαιτείται η ανάκτηση των δεδομένων από τα κοινωνικά δίκτυα να γίνεται χωρίς τη διακύβευση της ιδιωτικής ζωής των χρηστών.

7. Αξιολόγηση των συστημάτων συστάσεων

Η αξιολόγηση συστημάτων συστάσεων και οι αλγόριθμοι τους είναι εγγενώς δύσκολο να αξιολογηθούν για διάφορους λόγους (Herlocker et al., 2004):

1. Διαφορετικοί αλγόριθμοι μπορεί να είναι καλύτεροι ή χειρότεροι για διαφορετικά σύνολα δεδομένων.
2. Ένας αριθμός αλγόριθμών έχει σχεδιαστεί ειδικά για σύνολα δεδομένων όπου ο αριθμός των χρηστών είναι περισσότερος από τον αριθμό των προϊόντων. Αυτοί οι αλγόριθμοι δεν αποδίδουν το μέγιστο όταν τα προϊόντα είναι περισσότερα από τους χρήστες.
3. Η αξιολόγηση είναι δύσκολη όταν οι στόχοι για τους οποίους γίνεται η αξιολόγηση διαφέρουν.
4. Οι περισσότεροι αλγόριθμοι έχουν επικεντρωθεί στην ποιότητα (accuracy) των συστάσεων, ενώ θα έπρεπε να ληφθούν υπόψιν και άλλοι παράγοντες όπως πχ. το τι τελικά αγοράζει ο χρήστης.
5. Μία πρόκληση είναι το τι μονάδα μέτρησης θα χρησιμοποιηθεί για να αξιολογηθεί ένας αλγόριθμος, αλλά και το γεγονός ότι οι ίδιοι χρήστες δίνουν διαφορετικές βαθμολογίες σε διαφορετικούς χρόνους.

Για να αξιολογηθεί σωστά ένα σύστημα συστάσεων, είναι σημαντικό να κατανοήσουμε τους στόχους και τα καθήκοντα για τα οποία χρησιμοποιείται. Η επιλογή των συνόλων δεδομένων που θα χρησιμοποιηθούν επηρεάζει την επιτυχή αξιολόγηση του αλγόριθμου συστάσεων. Ακόμη, μπορεί η αξιολόγηση να πραγματοποιηθεί χωρίς σύνδεση στο Διαδίκτυο χρησιμοποιώντας έτοιμα σύνολα δεδομένων ή να απαιτεί δοκιμές χρηστών σε online περιβάλλοντα. Εάν ένα σύνολο δεδομένων δεν είναι επί του παρόντος διαθέσιμο, μπορεί η αξιολόγηση να πραγματοποιηθεί σε προσομοιωμένα, συνθετικά σύνολα δεδομένων (synthetic data sets).

Για παράδειγμα, όταν σχεδιάζουμε έναν αλγόριθμο ο οποίος προορίζεται για τη σύσταση ταινιών, το πιο πιθανό είναι ότι δε θα προσφέρει ικανοποιητικά αποτελέσματα σε άλλα δεδομένα. Ή όταν θέλουμε να φτιάξουμε έναν αλγόριθμο που θα συστήνει νέα, καινοτόμα, προϊόντα τότε ένα σύνολο δεδομένων για offline αξιολόγηση δεν είναι αρκετό. Επίσης, όταν αξιολογούμε έναν αλγόριθμο σε ένα νέο περιβάλλον όπου δεν υπάρχουν έτοιμα σύνολα δεδομένων, τότε μπορούμε να δημιουργήσουμε συνθετικά σύνολα δεδομένων. Ο πίνακας 4.3 μας δίνει μία λίστα με τα πιο γνωστά σύνολα δεδομένων και τα χαρακτηριστικά τους. Ενώ ο πίνακας 4.4 μας δίνει μία όψη αξιολογήσεων ενός χρήστη σε μία βάση δεδομένων.

Όνομα	**Πεδίο**	**Χρήστες**	**Προϊόντα**	**Αξιολογήσεις**
BX	Βιβλία	278,858	271,379	1,149,780
EachMovie	Ταινίες	72,916	1,628	2,811,983
Entrée	Εστιατόρια	50,672	4,160	Δεν υπάρχουν
Epinions	Εμπόριο	49,000	140,000	665 ΧΙΛ
Jester	Ανέκδοτα	73,421	101	4,1 ΕΚ.
MovieLens 100k	Ταινίες	967	4,700	100 ΧΙΛ.
MovieLens 1m	Ταινίες	6,040	3,900	1 ΕΚ.
MovieLens 10m	Ταινίες	71,657	10,681	10 ΕΚ.
Netflix	Ταινίες	480,000	18,000	100 ΕΚ.
Ta-Feng	Εμπόριο	32,266	Μη διαθέσιμο	800 ΧΙΛ.

***Πίνακας* 4.3** *Σύνολα δεδομένων*

Στήλη	ID του χρήστη	ID προϊόντος	Βαθμολογία
1	1	200	3
2	1	150	5
3	1	100	5
4	1	102	2
5	1	3	3
6	1	99	4
7	1	399	1

8	1	408	5
9	1	1	2
10	1	11	3

***Πίνακας* 4.4** *Αξιολογήσεις προϊόντων*

Χρησιμοποιώντας ένα σύνολο δεδομένων όπως ένα από αυτά που αναφέρονται στον πίνακα 4.3 θα μπορέσουμε να διαπιστώσουμε πόσο καλός είναι ο αλγόριθμος μας. Ένα ολοκληρωμένο σύνολο δεδομένων αποτελείται από έναν αριθμό χρηστών που έχουν βαθμολογήσει κάποια προϊόντα (στη μορφή των εγγραφών του πίνακα 4.4, υπάρχουν εκτός του χρήστη 1 και πολλοί άλλοι χρήστες με τις βαθμολογίες τους).

Το επόμενο βήμα είναι να χρησιμοποιηθεί μία μονάδα μέτρησης ακρίβειας (accuracy metric) για να διαπιστωθεί το πόσο καλός είναι ο αλγόριθμος. Οι πιο διαδεδομένες μονάδες μέτρησης είναι το Mean Absolute Error (MAE) και το Root Mean Square Error (RMSE) (Jannach et al., 2010). Επίσης χρησιμοποιούνται και οι μονάδες μέτρησης Precision και Recall που προέρχονται από τα παραδοσιακά συστήματα εξόρυξης δεδομένων. Η εικόνα 4.5 δείχνει την έκφραση της μονάδας μέτρησης MAE. Η εικόνα 4.6 δείχνει την έκφραση της μονάδας μέτρησης RMSE. Οι εικόνες 4.7 και 4.8 δείχνουν τις εκφράσεις των μετρικών Precision και το Recall αντίστοιχα.

Η μονάδα μέτρησης MAE υπολογίζει την απόκλιση (deviation) ανάμεσα στις υπάρχουσες βαθμολογίες και στις υπολογισμένες από τον αλγόριθμο. Στους τύπους των εικόνων 4.5 και 4.6, το pi εκφράζει την υπολογισμένη βαθμολογία ενώ το ri εκφράζει την υπάρχουσα βαθμολογία που έχει δώσει ο χρήστης. Επίσης η μονάδα RMSE είναι παρόμοια με τη MAE αλλά δίνει μεγαλύτερο βάρος στη μεγαλύτερη απόκλιση. Στις μονάδες μέτρησης MAE και RMSE οι μικρότερες μετρήσεις είναι οι καλύτερες.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|pi - ri|$$

Εικόνα 4.5 *Mean Absolute Error*

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(pi - ri)^2}$$

Εικόνα 4.6 *Root Mean Square Error*

Η μονάδα μέτρησης **Ακρίβεια** (Precision) μετράει το ποσοστό των συστάσεων που ορθά προτάθηκαν από τον αλγόριθμο (True Positive) σε σχέση με το σύνολο των συστάσεων. Το μεγαλύτερο ποσοστό είναι το καλύτερο. Η Ακρίβεια ορίζεται λοιπόν ως ο λόγος των ορθών συστάσεων προς το σύνολο των συστάσεων, ορθών και λανθασμένων που παρήγαγε ο αλγόριθμος (True Positive + False Positive).

Η μονάδα μέτρησης **Ανάκληση** (Recall) μετράει το ποσοστό των συστάσεων που ορθά προτάθηκαν από τον αλγόριθμο (True Positive) σε σχέση με το (ιδανικό) σύνολο όλων των συστάσεων που θα μπορούσαν να προταθούν. Η Ανάκληση ορίζεται λοιπόν ως ένας λόγος που ο αριθμητής της είναι ίδιος με την Ακρίβεια (είναι και πάλι το πλήθος True Positive), αλλά διαφέρει στον παρονομαστή, όπου έχει το σύνολο των συστάσεων που προτάθηκαν ορθά και αυτών που λανθασμένα δεν προτάθηκαν (True Positive + False Negative). Επίσης οι μεγαλύτερες τιμές στην Ανάκληση, είναι οι καλύτερες.

Στις παρακάτω εικόνες υπάρχουν και δυο εναλλακτικές εκφράσεις των μετρικών αυτών, στις οποίες το σύνολο (True Positive + False Positive) αντιστοιχεί στο πλήθος των retrieved recommendations, δηλαδή των συστάσεων που παρήγαγε ο αλγόριθμος, ενώ το σύνολο (True Positive + False Negative) αντιστοιχεί στο πλήθος των relevant recommendations, δηλαδή των συστάσεων που θα έπρεπε να δοθούν. Με το ακόλουθο, απλοϊκό ως ένα βαθμό, παράδειγμα, θα γίνει κατανοητή η διαφορά ανάμεσα στις δυο αυτές μετρικές: έστω ότι ο αλγόριθμος παρήγαγε 3 συστάσεις (retrieved recommendations), τις α, β και γ, ενώ ο χρήστης τελικά ως ενέργειες ακολούθησε 4 ενέργειες (relevant recommendations), τις α, β, δ και ε. Έχουμε λοιπόν στα δυο σύνολα, δυο μόνο κοινά στοιχεία, τα α και β, οπότε η True Positive έχει τιμή 2. Η τιμή της Precision είναι 2/3, δηλαδή περίπου 67%. Η τιμή της Recall αντίστοιχα είναι 2/4, δηλαδή 50%. Η σύσταση γ προτάθηκε

λανθασμένα, οπότε η τιμή της False Positive είναι 1. Οι ενέργειες δ και ε δεν προτάθηκαν από τον αλγόριθμο, οπότε η τιμή της False Negative είναι 2.

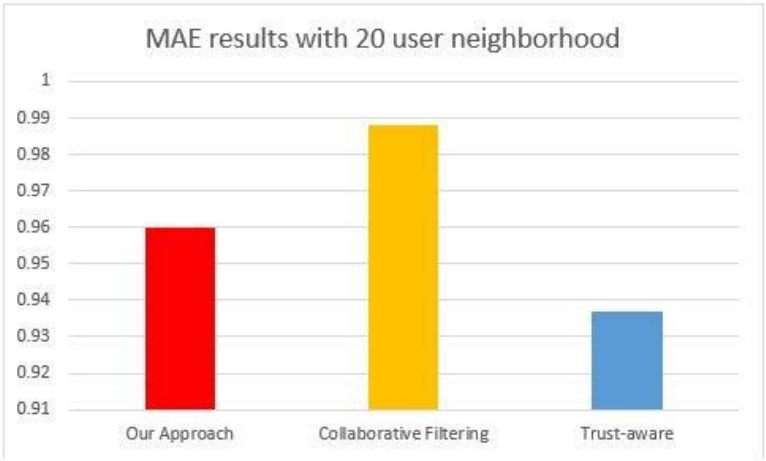
$$precision = \frac{|\{relevant\ recommendations\} \cap \{retrieved\ recommendations\}|}{|\{retrieved\ recommendations\}|}$$

Εικόνα 4.7 *Precision*

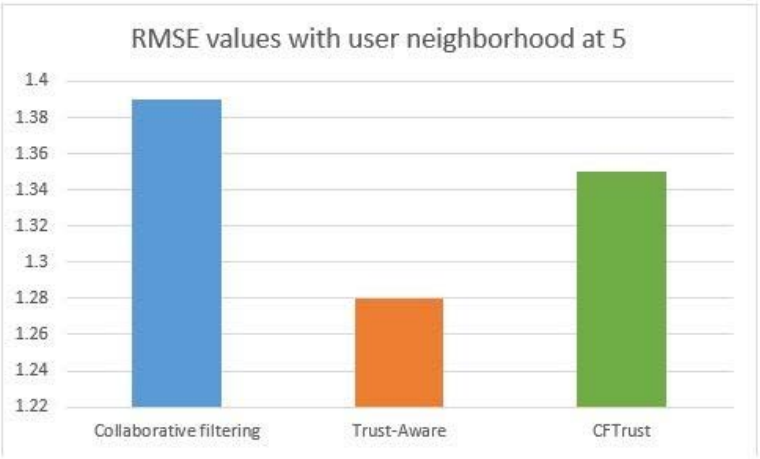
$$recall = \frac{|\{relevant\ recommendations\} \cap \{retrieved\ recommendations\}|}{|\{relevant\ recommendations\}|}$$

Εικόνα 4.8 *Recall*

Στην εικόνα 4.9 μπορούμε να δούμε μία σύγκριση τριών αλγόριθμων τύπου συνεργατικού φιλτραρίσματος (χρήση μετρικής MAE), στους οποίους για τα ζητήματα ομοιότητας μεταξύ χρηστών χρησιμοποιήθηκε η τιμή 20, δηλαδή ορίσθηκε όπως λέγεται γειτονιά 20 χρηστών. Η μικρότερη τιμή είναι η καλύτερη. Στην εικόνα 4.10 μπορούμε να δούμε άλλη σύγκριση των τριών αλγόριθμων (θα μιλήσουμε σε λίγο για αυτούς), με χρήση της μετρικής RMSE και γειτονιάς 5 χρηστών. Η μικρότερη τιμή είναι η καλύτερη και εδώ. Και στις δύο περιπτώσεις το σύνολο των πιο κοντινών χρηστών (neighborhood) έχει οριστεί από πριν έτσι ώστε να μπορέσουν να γίνουν οι υπολογισμοί. Θα πρέπει να επισημανθεί επίσης ότι για τις μετρήσεις χρησιμοποιήθηκε το Epinions dataset.



Εικόνα 4.9 *Παράδειγμα μέτρησης MAE*



Εικόνα 4.10 *Παράδειγμα μέτρησης RMSE*

Ένας παράγοντας που θα πρέπει να οριστεί στο προγραμματιστικό περιβάλλον για να γίνουν οι μετρήσεις είναι το ποσοστό του συνόλου που θα χρησιμοποιηθεί για το τεστ. Δηλαδή αν υποθέσουμε ότι το σύνολο των χρηστών του συνόλου δεδομένων που χρησιμοποιούμε ορίζεται με μία μεταβλητή X η οποία έχει τιμή 1.0 (100%) θα πρέπει να σπάσουμε αυτή την τιμή σε ένα σύνολο εκπαίδευσης (training test) που θα μπορούσε να είναι πχ. 0.30 (30%) ή 0.70 (70%) και το υπόλοιπο ποσοστό προορίζεται για να γίνουν οι μετρήσεις. Αυτό ισχύει για όλες τις μονάδες μέτρησης (MAE, RMSE, Precision, Recall). Σημαντική παράμετρος βέβαια είναι η μονάδα μέτρησης ομοιότητας (similarity) πχ. Pearson ή Cosine. Θα ήταν χρήσιμο να αναφερθούν επίσης εδώ, οι βιβλιοθήκες/APIs LensKit και Apache Mahout που μπορούν να χρησιμοποιηθούν σε περιβάλλον Java για να γίνουν δοκιμές και μετρήσεις επί προσεγγίσεων συστάσεων.

Video 4.1.mp4	Βίντεο (video)
Αξιολόγηση των συστημάτων συστάσεων	

Στα προηγούμενα διαγράμματα φαίνονται μετρήσεις για τρεις αλγορίθμους. Ο Collaborative filtering είναι ένας τυπικός αλγόριθμος συνεργατικού φιλτραρίσματος, έτσι όπως παρέχεται από τη βιβλιοθήκη Apache Mahout, με χρήση ομοιότητας Pearson. Ο Trust-Aware είναι μια προσέγγιση όπου κάθε σύσταση παράγεται από χρήστες που ανήκουν στο δίκτυο των χρηστών τους οποίους εμπιστεύεται ο χρήστης που ζητά συστάσεις. Ο τρίτος αλγόριθμος (CFTrust ή Our Approach) έχει αναπτυχθεί από τον συγγραφέα με στόχο να συνδυάσει τη φιλοσοφία του συνεργατικού φιλτραρίσματος με στοιχεία των κοινωνικών δικτύων των χρηστών. Χρησιμοποιεί τις αξιολογήσεις των χρηστών για να βρει τους πιο κοντινούς χρήστες. Αρχικά, ανάμεσα στο χρήστη που ζητάει τις συστάσεις και σε κάθε χρήστη υπάρχει μια τιμή ομοιότητας από -1 έως 1. Ωστόσο ο αλγόριθμος ελέγχει εάν ανάμεσα στο σύνολο των χρηστών υπάρχει κάποιος που να ανήκει στο κοινωνικό δίκτυο του χρήστη ως φίλος, έτσι ώστε να του αυξήσει τον βαθμό της ομοιότητας (με ανώτατο όριο το 1).

Algorithm 1 Combining rating and trust network for user a ∈ U

```
1: Input
2: UR → the set of all users and ratings
3: UTA → the set of the user-trust network of user a
4: for (i=0; i<UR; i++)
5: Sim (a, i) // the similarity function using equation 1
6: double tempSimilarity = Sim (a, i) // a value between -1 to 1
7: if (i.isIn (UTA))
8: tempSimilarity + 0.50
9: finalSimilarity = tempSimilarity
10: else finalSimilarity=tempSimilarity
11: end for
12: return finalSimilarity
13: output: finalSimilarity
```

Το παρακάτω τμήμα κώδικα είναι ένα παράδειγμα για το πώς γίνεται αξιολόγηση RMSE και MAE σε περιβάλλον Netbeans και χρησιμοποιώντας τη βιβλιοθήκη Apache Mahout. Ενώ η εικόνα 4.13 δείχνει το περιβάλλον Netbeans, το πώς λειτουργεί ο αλγόριθμος αλλά και το πώς εμφανίζεται το αποτέλεσμα.

Sound 4.5.mp3	Ηχητικό απόσπασμα (audio)
Αξιολόγηση σε περιβάλλον NetBeans	

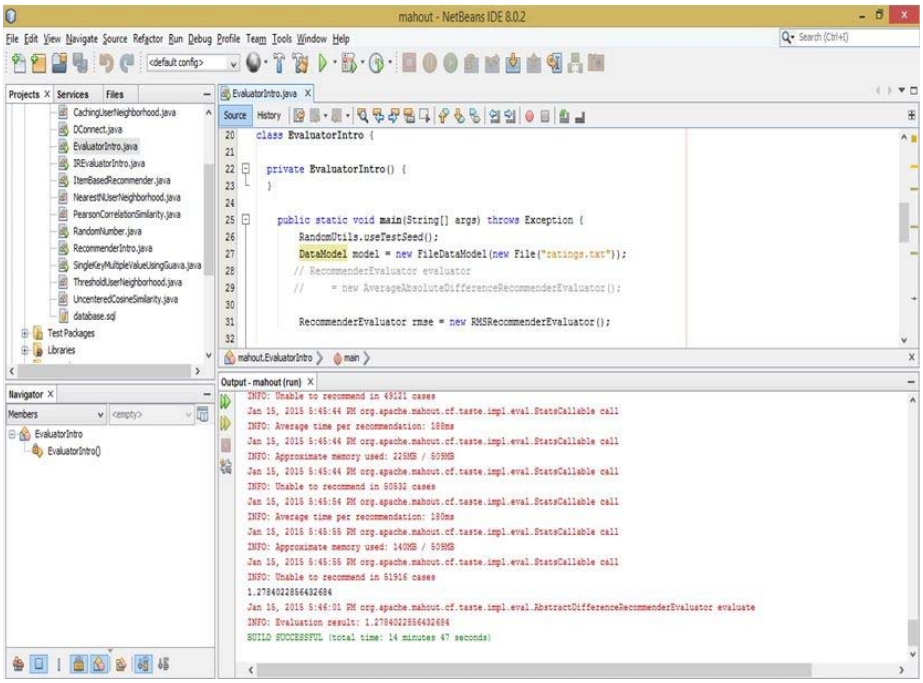
```
class EvaluatorIntro {

 private EvaluatorIntro() {
 }

   public static void main(String[] args) throws Exception {
      RandomUtils.useTestSeed();
      DataModel model = new FileDataModel(new File("ratings.txt"));
      RecommenderEvaluator rmse = new RMSRecommenderEvaluator();
```

```
    // Build the same recommender for testing that we did last time:
    RecommenderBuilder recommenderBuilder = new RecommenderBuilder() {
    @Override
public Recommender buildRecommender(DataModel model) throws TasteException {

UserSimilarity similarity = new PearsonCorrelationSimilarity(model);
UserNeighborhood neighborhood =
        new NearestNUserNeighborhood(60, similarity, model);
return new GenericUserBasedRecommender(model, neighborhood, similarity);
  }
};

// Use 90% of the data to train; test using the other X% e.g 0.1 is 10% 1.0 is 100%
    double measure = rmse.evaluate(recommenderBuilder, null, model, 0.9, 1.0);
    System.out.println(measure);
  }
}
```

Video 4.2.mp4	Βίντεο (video)
Επεξήγηση πηγαίου κώδικα στην αξιολόγηση των συστημάτων συστάσεων	

Τώρα, σε περίπτωση που θέλουμε να μετρήσουμε χρησιμοποιώντας τη μονάδα μέτρησης MAE, θα πρέπει να κάνουμε τις εξής αλλαγές:

- Η γραμμή κώδικα:
  RecommenderEvaluator rmse = new RMSRecommenderEvaluator();
  θα πρέπει να αντικατασταθεί με:
  RecommenderEvaluator mae = new AverageAbsoluteDifferenceRecommenderEvaluator();
- Η γραμμή κώδικα:
  double measure = rmse.evaluate(recommenderBuilder, null, model, 0.9, 1.0);
  θα πρέπει να αντικατασταθεί με:
  double measure = mae.evaluate(recommenderBuilder, null, model, 0.9, 1.0);



Εικόνα 4.11 *Netbeans και Apache Mahout RMSE and MAE Evaluator*

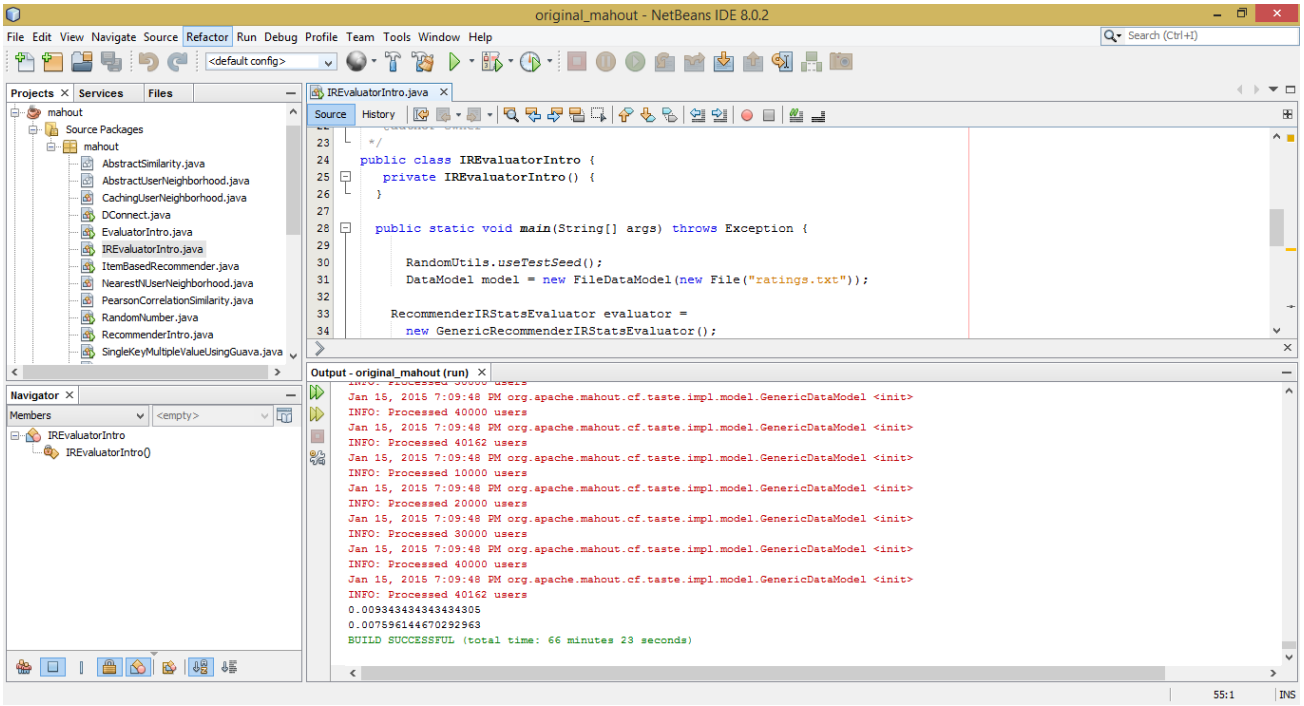
Το παρακάτω τμήμα κώδικα είναι ένα παράδειγμα για το πώς γίνεται αξιολόγηση Precision και Recall σε περιβάλλον Netbeans και χρησιμοποιώντας τη βιβλιοθήκη Apache Mahout. Ενώ η εικόνα 4.12 δείχνει το περιβάλλον Netbeans, το πώς λειτουργεί ο αλγόριθμος αλλά και το πώς εμφανίζεται το αποτέλεσμα. Να σημειωθεί ότι, οι τιμές που προκύπτουν από το προηγούμενο παράδειγμα στις μετρικές Precision και Recall είναι πολύ μικρές, λόγω επιλογών που έγιναν στις διαθέσιμες παραμέτρους (πχ. πλήθος κοντινών γειτόνων=3, πλήθος ζητούμενων συστάσεων=5, κατώφλι συνάφειας=0.8). Σκοπός αυτών των επιλογών ήταν η γρήγορη εκτέλεση των αλγορίθμων και η συγκριτική παρουσίαση των διαφορετικών επιδόσεών τους.

Sound 4.6.mp3	Ηχητικό απόσπασμα (audio)
Αξιολόγηση σε περιβάλλον NetBeans	

```
public class IREvaluatorIntro {
  public static void main(String[] args) throws Exception {
     RandomUtils.useTestSeed();
     DataModel model = new FileDataModel(new File("ratings.txt"));

RecommenderIRStatsEvaluator evaluator =
   new GenericRecommenderIRStatsEvaluator();
//Build the same recommender for testing that we did last time:
   RecommenderBuilder recommenderBuilder = new RecommenderBuilder() {
   @Override
public Recommender buildRecommender(DataModel model) throws TasteException {
   UserSimilarity similarity = new PearsonCorrelationSimilarity(model);
   UserNeighborhood neighborhood =
        new NearestNUserNeighborhood(3, similarity, model);
                        return new GenericUserBasedRecommender(model, neighborhood, similarity);
  }
};
//Evaluate precision and recall "at X recommendations":
//e.g. X = 5 products and 0.8 is the training set
IRStatistics stats = evaluator.evaluate(recommenderBuilder,
                    null, model, null, 5, GenericRecommenderIRStatsEvaluator.CHOOSE_THRESHOLD, 0.8);
          System.out.println(stats.getPrecision());
          System.out.println(stats.getRecall());
  }
}
```

Κάντε κλικ στα εικονίδια του σχήματος για επεξήγηση

Dynamic 4.1.zip	Διαδραστική εικόνα (interactive)
Εικόνα 4.12 Netbeans και Apache Mahout Precision and Recall Evaluator	



Εικόνα 4.12 *Netbeans και Apache Mahout Precision and Recall Evaluator*

8. Συστάσεις σε κινητά περιβάλλοντα

Η εξέλιξη των συστημάτων συστάσεων ήταν αναμενόμενο να επηρεαστεί σε μεγάλο βαθμό από το περιβάλλον του ΠΙ. Ωστόσο, οι πληροφορίες πλέον λαμβάνονται σε οποιοδήποτε μέρος και οποιαδήποτε ώρα χρησιμοποιώντας μία κινητή συσκευή και μία ασύρματη σύνδεση στο Διαδίκτυο. Τα κινητά συστήματα συστάσεων (mobile recommender systems) έχουν δημιουργήσει ένα απαιτητικό πεδίο έρευνας.

Οι κινητές εφαρμογές από την πρώτη στιγμή αποτέλεσαν ένα πεδίο όπου τα συστήματα συστάσεων και εξατομίκευσης βρήκαν πρόσφορο έδαφος, παρά τις αρχικά περιορισμένες δυνατότητες μίας κινητής συσκευής (όπως το μικρό μέγεθος της οθόνης και οι επεξεργαστικές δυνατότητες). Κύριοι παράγοντες υιοθέτησης της τεχνολογίας εξατομίκευσης από τους χρήστες κινητών συσκευών, η επιθυμία για μια ‘προσωπική’ συσκευή, αλλά και η αξιοποίηση της επίγνωσης της μεταβαλλόμενης τοποθεσίας τους για παροχή περιεχομένου προσαρμοσμένου στην τρέχουσα θέση τους. Η σημερινή βέβαια τεχνολογία έχει ξεπεράσει πολλούς από τους αρχικούς περιορισμούς. Για παράδειγμα, η τελευταία γενιά έξυπνων κινητών τηλεφώνων έχει συνήθως πολύ γρήγορους επεξεργαστές, αρκετή μνήμη αλλά και οθόνη υψηλής ευκρίνειας. Επίσης, η πλειοψηφία έχει ενσωματωμένο δέκτη εντοπισμού μέσω δορυφόρου (GPS) ο οποίος μπορεί να χρησιμοποιηθεί ανά πάσα στιγμή για να βρεθεί η γεωγραφική τοποθεσία του χρήστη. Ένα ενδεικτικό παράδειγμα που χρησιμοποιεί αυτή την τεχνολογία είναι οι χάρτες Google. Επιπροσθέτως, οι μοντέρνες ασύρματες συνδέσεις παρέχουν υψηλές ταχύτητες σε πολύ προσιτές τιμές, κάνοντας έτσι την τεχνολογία διαθέσιμη σε μια μεγάλη μερίδα χρηστών.

Κάποια ανοιχτά ερωτήματα που απασχολούν ακόμη την ερευνητική κοινότητα είναι (Jannach et al., 2010):

- Ποιοι είναι οι συγκεκριμένοι, τελικοί, στόχοι ενός συστήματος εξατομίκευσης που λειτουργεί σε ένα κινητό περιβάλλον; Περιμένουν οι χρήστες συστάσεις που θα μπορούσαν να βρεθούν χρησιμοποιώντας ένα κοινό περιβάλλον Διαδικτύου ή είναι σημαντικό (και πόσο) το να προταθούν προϊόντα ή υπηρεσίες που βρίσκονται κοντά στην τοποθεσία του χρήστη;
- Ποιες είναι οι επιπτώσεις του να χρησιμοποιηθούν παράμετροι του περιβάλλοντος (όπως η τοποθεσία) για τη σχεδίαση ενός αλγόριθμου συστάσεων; Είναι εντέλει η τοποθεσία μία απλή παράμετρος ή αποτελεί η αξιοποίησή της μία σοβαρή παράβαση της ιδιωτικότητας του χρήστη;

- Υπάρχει όντως ένα νέο περιβάλλον για κινητές συσκευές, που πρέπει να ανιχνευθεί και να μοντελοποιηθεί ή υπάρχουν διαφορετικά σενάρια χρήσης που απλά απαιτούν μία απλή προσαρμογή αλγόριθμων που ήδη υπάρχουν στον ΠΙ; Ενδεικτικά σενάρια διερεύνησης προς τον σκοπό αυτό, είναι οι οδηγοί πόλεων και μουσείων;
- Όταν ο χρήστης κινείται, πόσο σημαντικό είναι να του δίνονται εξατομικευμένες πληροφορίες αυτόματα; Η άδεια του χρήστη είναι κατά κανόνα απαραίτητη για να συμβεί αυτό, αλλά υπάρχουν εξαιρέσεις;

8.1 Συστάσεις βασισμένες στην περιβάλλουσα κατάσταση

Οι συστάσεις στα κινητά περιβάλλοντα είναι κυρίως βασισμένες στην περιβάλλουσα κατάσταση (context). Η περιβάλλουσα κατάσταση (ή αλλιώς πλαίσιο) είναι κυρίως η τοποθεσία, η ώρα, αλλά και άλλοι παράγοντες όπως ο καιρός, η παρέα αλλά και η διάθεση του χρήστη. Συνήθως οι παράγοντες της περιβάλλουσας κατάστασης θα πρέπει να είναι άμεσα διαθέσιμοι στο σύστημα ως παράμετροι. Οπότε:

- Το μοντέλο χρήστη δεν είναι πλέον βασισμένο μόνο σε δημογραφικά χαρακτηριστικά ή/και προτιμήσεις.
- Λειτουργούν ως παράμετροι φυσικές ιδιότητες (όπως η τοποθεσία και η ώρα).
- Λειτουργούν ως παράμετροι περιβαλλοντικές ιδιότητες (όπως ο καιρός, αν είναι μέρα η νύχτα, αν βρέχει ή όχι, κ.ά.).
- Λειτουργούν ως παράμετροι πληροφορίες που βασίζονται στα ενδιαφέροντα του χρήστη, οι οποίες μπορεί να μεταβάλλονται συνεχώς.
- Λειτουργούν ως παράμετροι προσωπικές πληροφορίες (όπως η υγεία, η διάθεση αλλά και το προσωπικό πρόγραμμα του χρήστη).
- Λαμβάνονται υπόψη επίσης πληροφορίες για την παρέα του χρήστη (με ποιον ή ποιους βρίσκεται μαζί).
- Λαμβάνονται υπόψη χαρακτηριστικά της κινητής συσκευής αλλά και του δικτύου σύνδεσης (πχ. η ταχύτητα του ασύρματου δικτύου).

Μία ενδιαφέρουσα προσέγγιση για την ενοποίηση των πληροφοριών του πλαισίου στα συστήματα συστάσεων έχει γίνει από τους Adomavicius et al. (2005), όπου προτείνεται η χρήση ενός μοντέλου που βασίζεται σε πολλές διαστάσεις. Ο φορμαλισμός έχει δοθεί από τους ερευνητές σε μια γενική θεώρηση που περιλαμβάνει και άλλες διαστάσεις εκτός από τον χρήστη και τις προτιμήσεις.

Έστω f μια συνάρτηση παραγωγής συστάσεων. Αρχικά μπορεί να εκφρασθεί με βάση τις δυο διαστάσεις, των χρηστών (U) και των στοιχείων (I), δηλαδή των αντικειμένων/προϊόντων (Εικόνα 4.13):

$$f = U \times I \dashrightarrow R$$

Εικόνα 4.13 *Συμβολισμός δύο διαστάσεων για τη συνάρτηση παραγωγής συστάσεων*

Όπου *U* είναι ο χρήστης, *I* είναι το στοιχείο και *R* είναι οι συστάσεις. Μπορούμε να αντικαταστήσουμε/αναλύσουμε το *U x I* (Εικόνα 4.14):

$$f = D1 \times \ldots \times Dn \dashrightarrow R$$

Εικόνα 4.14 *Συμβολισμός n διαστάσεων για τη συνάρτηση παραγωγής συστάσεων*

Όπου D1 ως Dn είναι οι διαστάσεις (n πλήθους) που περιλαμβάνουν το *U,* το *I*, αλλά και άλλους παράγοντες όπως η τοποθεσία, η ώρα κλπ. Για παράδειγμα, θα μπορούσε η συνάρτηση να είναι ως εξής (Εικόνα 4.15):

$$f = U \times I \times L \dashrightarrow R$$

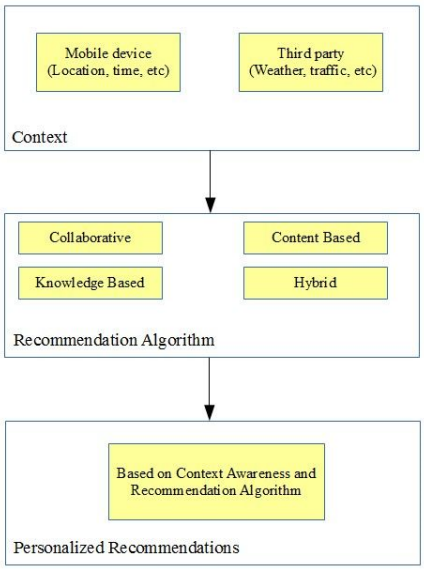
Εικόνα 4.15 *Συμβολισμός τριών διαστάσεων για τη συνάρτηση παραγωγής συστάσεων*

Είναι συνάρτηση τριών διαστάσεων: περιλαμβάνει τον χρήστη, το στοιχείο και την τοποθεσία (δηλαδή ο χρήστης είναι το *D1,* το προϊόν το *D2* και η τοποθεσία το *D3)*.

Η εικόνα 4.16 οπτικοποιεί, σε ένα αφαιρετικό επίπεδο, ένα σύστημα συστάσεων που χρησιμοποιεί πληροφορίες πλαισίου. Τέλος, στον πίνακα 4.5 αναγράφονται συνοπτικά οι διαφορές ανάμεσα στα παραδοσιακά συστήματα συστάσεων και σε αυτά που λειτουργούν σε κινητά περιβάλλοντα. Να σημειώσουμε εδώ ότι μεγαλύτερη ανάλυση του κινητού περιβάλλοντος θα γίνει σε επόμενο κεφάλαιο. Με βάση την ανάλυση αυτή, θα 'επισκεφθούμε' ξανά τα κινητά συστήματα συστάσεων για να συζητήσουμε τις ιδιαιτερότητές τους.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 4.1.gif	Κινούμενη εικόνα (interactive)
Σύστημα συστάσεων βασισμένο στην περιβάλλουσα κατάσταση	



Εικόνα 4.16 *Σύστημα συστάσεων βασισμένο στην περιβάλλουσα κατάσταση*

Πρόκληση	**Διαδικτυακά συστήματα**	**Κινητά συστήματα**
Θέματα ενέργειας	ΟΧΙ	ΝΑΙ
Μέγεθος αποθηκευτικής μνήμης	ΟΧΙ	ΝΑΙ
Ασύρματη σύνδεση στο Διαδίκτυο	ΟΧΙ	ΝΑΙ
Άλλα θέματα συνδεσιμότητας	ΟΧΙ	ΝΑΙ
Διεπαφή με τον χρήστη	ΝΑΙ	ΝΑΙ
Λειτουργία χωρίς φυσική παρουσία χρήστη	ΟΧΙ	ΝΑΙ
Φιλικό περιβάλλον προς τον μέσο χρήστη	ΝΑΙ	ΝΑΙ
Μπορεί να βελτιωθεί η εμπειρία του χρήστη	ΝΑΙ	ΝΑΙ
Μπορεί να βελτιωθεί η αλληλεπίδραση	ΝΑΙ	ΝΑΙ
Να μαντέψει την πρόθεση του χρήση	ΝΑΙ	ΝΑΙ

Μικρότερο μέγεθος εισαγωγής δεδομένων	ΟΧΙ	ΝΑΙ
Δύσχρηστο	ΝΑΙ	ΝΑΙ
Θέματα ιδιωτικότητας	ΝΑΙ	ΝΑΙ
Πρόβλημα νέου χρήστη	ΝΑΙ	ΝΑΙ
Πρόβλημα νέου προϊόντος	ΝΑΙ	ΝΑΙ
Οι χρήστες ξοδεύουν λιγότερο χρόνο	ΟΧΙ	ΝΑΙ

Πίνακας 4.5 *Σύγκριση διαδικτυακών και κινητών συστημάτων συστάσεων*

9. Συμπεράσματα

Η εξατομίκευση είναι μια τεχνολογία που επιτρέπει τις δυναμικές προσαρμογές στην παρεχόμενη πληροφορία, υπηρεσία ή προϊόν. Οι προσαρμογές βασίζονται στις προτιμήσεις ενός χρήστη (ή κάποιας προγενέστερης στάσης αυτού), στις προτιμήσεις άλλων χρηστών, αλλά και σε παραμέτρους της περιβάλλουσας κατάστασης. Στόχος, να παρέχει στους χρήστες ό,τι τους ταιριάζει περισσότερο και όπως το προτιμούν, αντί της παροχής του ίδιου περιεχομένου στο ίδιο ύφος. Η παροχή συστάσεων και εξατομικευμένων πληροφοριών είναι ένας κρίσιμος παράγοντας σχετικά με την αποτελεσματικότητα ενός δικτυακού τόπου συναλλαγών: έχοντας τη δυνατότητα να ‘κατανοήσει’ τις ανάγκες του κάθε χρήστη, προσαρμόζει τους πόρους του για να ανταποκριθεί καλύτερα στις ανάγκες του. Το κεφάλαιο αυτό περιγράφει τα συστήματα συστάσεων, τους κυριότερους αλγορίθμους αυτών, καθώς και τα οφέλη που τόσο η επιχείρηση όσο και οι πελάτες έχουν. Ιδιαίτερη σημασία δίνεται στις προκλήσεις που τα σύγχρονα συστήματα συστάσεων καλούνται να αντιμετωπίσουν: στη διαφύλαξη της ιδιωτικότητας των χρηστών και στην εύρεση κατάλληλων τρόπων αξιοποίησης των δεδομένων που συλλέγονται στον ΠΙ μέσω των κοινωνικών δικτύων. Ο χώρος των κινητών συστημάτων συστάσεων, με το ακόμη πιο σύνθετο αλλά και πλούσιο πλαίσιο παραμέτρων που προσφέρει η κινητή συσκευή, είναι πολύ ενδιαφέρουσα περιοχή μελέτης, και μια πρώτη διερεύνηση των σχετικών θεμάτων γίνεται σε αυτό το κεφάλαιο επίσης. Περαιτέρω ανάλυση για τα συστήματα συστάσεων σε κινητές συσκευές και τα σχετικά ζητήματα ασφάλειας-ιδιωτικότητας θα γίνει στο κεφάλαιο 7.

Βιβλιογραφία / Αναφορές

Adomavicius, G., Sankaranarayanan, R., Sen, S. & Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems* (TOIS), 23(1), 103-145.

Benats, G., Bandara, A., Yu, Y., Colin, J.-N. & Nuseibeh, B. (2011). PrimAndroid: privacy policy modelling and analysis for Android applications. In: IEEE International Symposium on Policies for Distributed Systems and Networks (POLICY 2011) (pp. 129-132). Washington, DC: IEEE.

Chellappa, R. K. & Sin, R. G. (2005). Personalization versus Privacy: An Empirical Examination of the Online Consumer's Dilemma. *Information Technology and Management*, 6(2-3), 181-202.

ChoiceStream Personalization Survey: Consumer Trends and Perceptions (2008). Retrieved October 21 2013 from: http://www.choicestream.com/pdf/ChoiceStream_ 2008_Personalization_Survey.pdf

Cooperstain, D., Delhagen, K., Aber. A. & Levin, K. (1999). Making Net Shoppers Loyal. *Forrester Research*, Cambridge, MA.

Davidson, D. & Livshits, B. (2012). MoRePriv: Mobile OS Support for Application Personalization and Privacy. TechReport, Microsoft Research, Redmond, WA 98052, United States.

Fleder, D. M. & Hosanagar, K. (2009). Blockbuster Culture's Next Rise or Fall: The Impact of Recommender Systems on Sales Diversity. *Management Science*, 55(5), 697-712.

Fogg, B. J. (2002). Persuasive Technology: Using Computers to Change what We Think and Do. *Ubiquity*. December (2002): 5

Gundecha, P., & Liu, H. (2012). Mining Social Media: A Brief Introduction. *Tutorials in Operations Research*.

Herlocker, J. L., Konstan, J. A., Terveen, L. G. & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, *22*(1), 5-53.

Hinz, O. & Eckert, J. (2010). The Impact of Search and Recommendation Systems on Sales in Electronic Commerce. *Business & Information Systems Engineering*, 2(2), 67-77.

Jabeur, N., Zeadally, S. & Sayed, B. (2013). Mobile social networking applications. *Communications of the ACM*, 56(3), 71-79.

Jannach, D., Zanker, M., Felfernig, A. & Friedrich, G. (2010). *Recommender Systems: An Introduction*, Cambridge University Press.

Jeckmans, A., Beye, M., Erkin, Z., Hartel, P., Lagendijk, R. & Tang, Q. (2013) Privacy in recommender systems. In: *Social media retrieval. Computer Communications and Networks*. Springer Verlag, London, 263-281.

Karimov, F. P. & Brengman, M. (2011). Adoption of Social Media by Online Retailers: Assessment of Current Practices and Future Directions. *International Journal of E-Entrepreneurship and Innovation*, 2(1), 26-45.

Kobsa, A. (2007) Privacy-Enhanced Web Personalization. *The Adaptive Web*, LNCS 4321, 628-670.

Kobsa, A. & Teltzrow, M. (2005). Contextualized Communication of Privacy Practices and Personalization Benefits: Impacts on Users' Data Sharing and Purchase Behavior. In D. Martin & A. Serjantov, *Privacy Enhancing Technologies*, 4th International Workshop, Toronto, Canada (PET 2004) Revised Selected Papers (pp. 329—343).

Konstan, J. Riedl, J. (2012). Recommender systems: from algorithms to user experience. User Modeling and User-Adapted Interaction, 22(1-2), 101-123.

Liu, H. & Maes, P. (2004). InterestMap: Harvesting Social Network Profiles for Recommendations. *Beyond Personalisation-IUI*, December:56.

Mangalindan, J. P. (2012). Amazon's Recommendations secrets. Retrieved October 21 2013 from: http://tech.fortune.cnn.com/2012/07/30/amazon-5/.

Ochi, P., Rao, S., Takayama, L. & Nass, C. (2010). Predictors of user perceptions of web recommender systems: How the basis for generating experience and search product recommendations affects user responses. *International Journal of Human Computer Studies*, 68(8), 472-482.

Oulasvirta, A., Rattenbury, T., Ma, L. & Raita, E. (2012). Habits make smartphone use more pervasive. *Personal and Ubiquitous Computing*, 16(1), 105-114.

Pavlou, P. A. (2003). Consumer Acceptance of Electronic Commerce: Integrating Trust and Risk with the Technology Acceptance Model. *International Journal of Electronic Commerce* 7(3), 101-134.

Polatidis, N. & Georgiadis, C. K. (2013). Mobile Recommender Systems: An Overview of Technologies and Challenges. In: *IEEE Proceedings of the Second International Conference on Informatics and Applications* (ICIA 2013) (pp. 282-287) Washington, DC: IEEE.

Prasad, R. & Kumari, V. V. (2012). A Catagorical Review of Recommender Systems. *International Journal of Distributed and Parallel Systems* (IJDPS) 3(5), 73-83.

Qiu, L. & Benbasat, I. (2009). Evaluating Anthropomorphic Product Recommendation Agents: A Social Relationship Perspective to Designing Information Systems. *Journal of Management Information Systems*, 25(4), 145-182.

Ricci, F., Rokach, L., Shapira, B. & Kantor, P. (2011). *Recommender systems handbook*, Springer.

Ricci, F. (2011). Mobile Recommender Systems. *International Journal of Information Technology and Tourism*. 12(3), 205-231.

Riemer, K. & Totz, C. (2001). The many faces of personalization – An Integrated economic overview of mass customization and personalization. In *Proceedings of the World Congress of Mass Customization and Personalization* (MCPC 2001).

Schoenbachler, D. D. & Gordon, G. L. (2002). Trust and customer willingness to provide information in database-driven relationship marketing. *Journal of Interactive Marketing*, 16(3), 2-16.

Shi, Y., Larson, M. & Hanjalic, A. (2014). Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)* 47.1: 3.

Shyong, K., Lam, T., Frankowski, D. & Riedl, J. (2006). Do you trust your recommendations? An exploration of security and privacy issues in recommender systems. In *Proceedings of the International Conference on Emerging Trends in Information and Communication Security* (ETRICS), LNCS 3995. 14-29.

Turow, J. (2003) Americans amd Online Privacy: The system is broke. Aneenberg Public Policy Center, University of Pennnsylvania. Retrieved October 21 2013 from: http://www.securitymanagement.com/archive/library/Anneberg_privacy1003.pdf

Wang, L. C., Baker, J., Wagner, J. A. & Wakefield K. (2007). Can a Retail Web Site Be Social? *Journal of Marketing,* July, 71(3), 143-157.

Wu, L.-L., Joung, Y.-J. & Chiang, T.-E. (2011). Recommendation Systems and Sales Concentration: The Moderating Effects of Consumers' Product Awareness and Acceptance to Recommendations. System Sciences (HICSS), 2011 44th Hawaii International Conference on. IEEE, pp. 1-10.

Quiz4.html	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Τα συστήματα συστάσεων και εξατομίκευσης είναι απαραίτητα λόγω της υπερφόρτωσης πληροφοριών στο Διαδίκτυο.

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Α) Σωστό

Κριτήριο αξιολόγησης 2

[*] Η προστασία των προσωπικών δεδομένων του χρήστη είναι χαμηλής σημασίας.

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 3

[*] Τα συστήματα συστάσεων και εξατομίκευσης που λειτουργούν σε κινητές συσκευές:

Α) Λειτουργούν όπως ακριβώς σε ένα περιβάλλον Ηλεκτρονικού Υπολογιστή

Β) Λαμβάνουν υπόψιν τους περιορισμούς υλικού που υπάρχουν

Γ) Βασίζονται αποκλειστικά στην τοποθεσία

Απάντηση/Λύση

Β) Λαμβάνουν υπόψιν τους περιορισμούς υλικού που υπάρχουν

Κριτήριο αξιολόγησης 4

[*] Οι επιχειρήσεις χρησιμοποιούν συστήματα εξατομίκευσης κυρίως για:

Α) Αύξηση πωλήσεων

Β) Προώθηση και άλλων προϊόντων

Γ) Και τα δύο

Απάντηση/Λύση
Γ) Και τα δύο

Κριτήριο αξιολόγησης 5
[*] Η πιο διαδεδομένη μονάδα μέτρησης ομοιότητας είναι:

Α) Cosine

Β) Content Based Filtering

Γ) Pearson Correlation

Απάντηση/Λύση
Γ) Pearson Correlation

Κριτήριο αξιολόγησης 6
[**] Η εκτενής αξιολόγηση των συστημάτων συστάσεων γίνεται με:

Α) Precision και Recall

Β) MAE και RMSE

Γ) Συνδυασμό των παραπάνω

Απάντηση/Λύση
Γ) Συνδυασμό των παραπάνω

Κριτήριο αξιολόγησης 7
[**] Η μονάδα μέτρησης Precision μετράει:

Α) το ποσοστό των σχετικών προϊόντων που προτάθηκαν σε σχέση με το σύνολο όλων των σχετικών προϊόντων που θα μπορούσαν να προταθούν

Β) το ποσοστό των σχετικών προϊόντων που προτάθηκαν σε σχέση με το σύνολο των προϊόντων που προτάθηκαν

Γ) το ποσοστό των σχετικών προϊόντων που δεν προτάθηκαν σε σχέση με το σύνολο των προϊόντων που προτάθηκαν

Απάντηση/Λύση
Β) το ποσοστό των σχετικών προϊόντων που προτάθηκαν σε σχέση με το σύνολο των προϊόντων που προτάθηκαν

Κριτήριο αξιολόγησης 8
[*] Το συνεργατικό φιλτράρισμα για να παράγει συστάσεις σε έναν χρήστη, λαμβάνει υπόψη:

Α) τις προτιμήσεις των άλλων χρηστών που μοιράζονται τα ίδια ενδιαφέροντα με τον τρέχοντα χρήστη

Β) τις ιδιότητες των προϊόντων/υπηρεσιών για τα οποία παράγονται συστάσεις

Γ) και τα δυο τα αναφερόμενα από τις προηγούμενες επιλογές

Απάντηση/Λύση
Α) τις προτιμήσεις των άλλων χρηστών που μοιράζονται τα ίδια ενδιαφέροντα με τον τρέχοντα χρήστη

Κριτήριο αξιολόγησης 9
[**] Ποια από τις ακόλουθες προτάσεις δεν είναι σωστή;

Α) Μια μεγάλη κατηγορία χρηστών είναι πρόθυμη να δώσει κάθε είδους πληροφορία σε ένα σύστημα συστάσεων με αντάλλαγμα περισσότερο εξατομικευμένο περιεχόμενο.

Β) Όλοι οι χρήστες είναι πρόθυμοι να δώσουν όσες πληροφορίες χρειάζεται ένα σύστημα συστάσεων, προκειμένου να πάρουν βελτιωμένες συστάσεις.

Γ) Μια μεγάλη κατηγορία χρηστών δε θα δώσει καμία πληροφορία σε ένα σύστημα συστάσεων, λόγω των ανησυχιών της για ζητήματα ιδιωτικότητας

Απάντηση/Λύση
Β) Όλοι οι χρήστες είναι πρόθυμοι να δώσουν όσες πληροφορίες χρειάζεται ένα σύστημα συστάσεων, προκειμένου να πάρουν βελτιωμένες συστάσεις.

Κριτήριο αξιολόγησης 10
[**] Τα δεδομένα αξιολογήσεων από τα κοινωνικά δίκτυα μπορούν να επηρεάσουν:

Α) τη λειτουργία των αλγορίθμων συνεργατικού φιλτραρίσματος.

Β) τη λειτουργία των αλγορίθμων με βάση το περιεχόμενο.

Γ) τη λειτουργία τόσο των αλγορίθμων συνεργατικού φιλτραρίσματος όσο και αυτών με βάση το περιεχόμενο, ανάλογα με το κοινωνικό δίκτυο.

Απάντηση/Λύση
Α) τη λειτουργία των αλγορίθμων συνεργατικού φιλτραρίσματος.

Κεφάλαιο 5: Ανάπτυξη εφαρμογών Ιστού – Υποστήριξη Λειτουργιών Ηλεκτρονικού Εμπορίου: Εργαστηριακές Ασκήσεις

Σύνοψη
Στο κεφάλαιο αυτό παρουσιάζεται ένα σύνολο εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών), με σκοπό την κατανόηση τεχνικών προγραμματισμού από την πλευρά του διακομιστή (server-side scripting). Η επιμέρους τεχνολογία που αξιοποιείται είναι η τεχνολογία ASP.NET (Active Server Pages), η οποία υποστηρίζεται από διακομιστές (web servers) της εταιρίας Microsoft (IIS, Internet Information Services). Στα παραδείγματα χρησιμοποιείται ως βασική γλώσσα προγραμματισμού στον Ιστό η γλώσσα C#, και συνδυάζεται η λειτουργικότητά της με στοιχεία προγραμματισμού από την πλευρά του πελάτη, με αποσπάσματα κώδικα JavaScript. Χρησιμοποιούνται επίσης αντικείμενα ADO.NET (ActiveX Data Objects) για τη σύνδεση και αλληλεπίδραση με βάσεις δεδομένων SQL.

Προαπαιτούμενη γνώση
Το κεφάλαιο 1 του παρόντος συγγράμματος, και επιπλέον θα είναι χρήσιμη κάποια προηγούμενη εμπειρία σε ζητήματα Προγραμματισμού Υπολογιστών και Βάσεων Δεδομένων.

1. Εισαγωγή

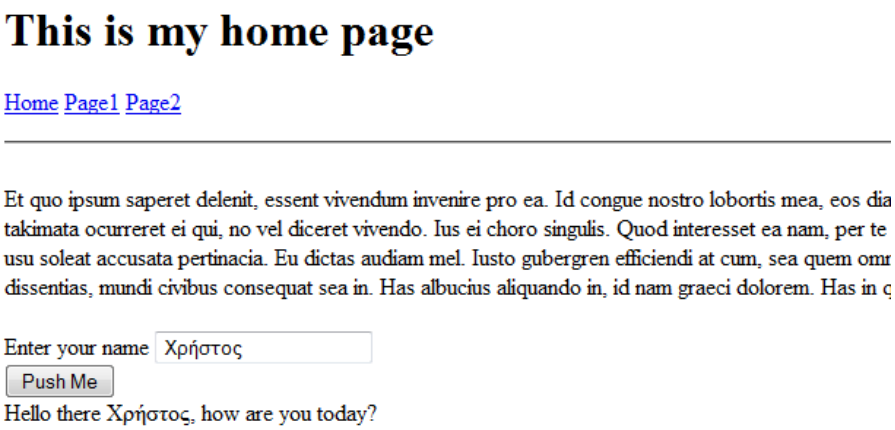
Στηριζόμενοι στα ενορατικά/οπτικά (visual) εργαλεία, στους προσφερόμενους μηχανισμούς διάδρασης, αλλά και γενικότερα στην προσφερόμενη ευχρηστία ενός ολοκληρωμένου περιβάλλοντος προγραμματισμού (όπως αυτό του Visual Studio), ακολουθούμε τη μέθοδο εκπαίδευσης από παράδειγμα (example-based learning) για να παρουσιάσουμε σημαντικές έννοιες και τεχνικές σχετικές με την ανάπτυξη εφαρμογών Ιστού και την υποστήριξη λειτουργιών ηλεκτρονικού εμπορίου, όπως: χρήση και επεξεργασία πρότυπων σελίδων (master pages), HTML tags και κανόνες μορφοποίησης CSS, δημιουργία μενού πλοήγησης, αντικειμενοστραφής και καθοδηγούμενος από συμβάντα προγραμματισμός για ανάπτυξη εφαρμογών στον Ιστό, φόρμες στον Ιστό και επικύρωση στοιχείων φόρμας, αξιοποίηση των application και session events, σύνδεση Ιστότοπου με βάση δεδομένων και επεξεργασία στοιχείων βάσης δεδομένων μέσω Ιστοσελίδων, διαχείριση ζητημάτων ασφάλειας (δημιουργία ρόλων και χρηστών, εγγραφή χρηστών, αυθεντικοποίηση και έλεγχος πρόσβασης χρηστών), λειτουργία καλαθιού αγορών.

Να σημειώσουμε εδώ ότι η πολύ σημαντική πλευρά του προγραμματισμού στον Ιστό από την πλευρά του πελάτη (client-side scripting), με άξονα τις ισχυρές δυνατότητες της γλώσσας JavaScript, θα γίνει σε επόμενο κεφάλαιο που πραγματεύεται την ανάπτυξη περιεχομένου για κινητές συσκευές. Ό σημαντικότερος λόγος για την απόφασή μας αυτή, είναι η ευρεία χρήση πλέον της JavaScript για την ανάπτυξη εφαρμογών κινητού Ιστού (mobile Web), δυναμικών Ιστότοπων που γίνονται προσβάσιμοι από τους χρήστες κινητών συσκευών μέσω των φυλλομετρητών (browsers) τους.

2. Προγραμματισμός από την πλευρά του Διακομιστή - ASP.NET Τεχνολογία και Περιβάλλον Προγραμματισμού Visual Studio

2.1. Βασικοί μηχανισμοί διάδρασης (UI controls) ενός Ιστότοπου

Εκφώνηση:
Δημιουργήστε έναν Ιστότοπο με τρεις σελίδες, με μενού βασισμένο σε λέξεις-συνδέσμους που τις συνδέει (η παρακάτω εικόνα παρουσιάζει την αρχική του σελίδα). Στο κάτω μέρος της αρχικής σελίδας και της Page 2 παρέχεται η δυνατότητα να δώσει ο χρήστης το όνομά του και να λάβει απάντηση (μηχανισμοί label, textbox, button). Στις σελίδες Page1 και Page 2 να εμφανίζεται στο πάνω μέρος τους δυναμικά ‘Καλημέρα!’ ή ‘Καλησπέρα!’ (με script μέσα στον κώδικα της σελίδας)



Εικόνα 5.1 *Βασική, αρχική Ιστοσελίδα*

Υποδειγματική λύση:

- Δημιουργία καινούργιου Ιστότοπου (New Empty Web Site), Γλώσσα C#, όνομα: Lab1
- Εισαγωγή 3 web forms: Default.aspx, Page1.aspx, Page2.aspx
- Στην Default.aspx
  - Μέσα στο (συνήθως υπάρχει ήδη, αλλιώς εισαγωγή από το toolbox – HTML), εισάγουμε <h1> (από toolbar: Formatting), και βάζουμε ως κείμενο: "Αυτή είναι η αρχική μου σελίδα"
  - Εισαγωγή <p> (νέα παράγραφος)
    - Εισαγωγή (από καρτέλα source) ενός συνδέσμου <a> με κείμενο "Home", και ενημέρωση της ιδιότητάς του href με την αντίστοιχη σελίδα default.aspx.
    - Εισαγωγή με άλλο τρόπο των συνδέσμων για τις σελίδες "Page1" και "Page2": γράφουμε το κείμενο π.χ. Page1 και πατάμε από την μπάρα εργαλείων το εργαλείο 'Convert to Hyperlink'
  - Εισαγωγή <hr /> (toolbar – HTML) (Horizontal Rule)
  - Εισαγωγή
 (αλλαγή γραμμής)
  - Εισαγωγή κειμένου (copy –paste) από Internet (έλεγχος μορφοποίησης)
  - Εισαγωγή 2
 (ή 2 <p> με enter στο design)
  - Εισαγωγή "Enter your name:" και δίπλα …
  - Εισαγωγή Textbox (toolbox – Standard) και
  - Εισαγωγή Button (toolbox – Standard) και Text="Push Me" και
  - Εισαγωγή Label (toolbox – Standard) και σβήσιμο του κειμένου «label» στην ιδιότητα text (για να μην είναι ορατό κάποιο κείμενο) και
  - Εκτέλεση – δοκιμή - έλεγχος
    Εισαγωγή click event στο button (με διπλό κλικ πάνω στο κουμπί):

    Label1.Text = "Hello there " + TextBox1.Text + ", πώς αισθάνεσαι σήμερα;";

  - Εκτέλεση – δοκιμή - έλεγχος

- Στην Page1.aspx
  - Μέσα στο (υπάρχει ήδη, αλλιώς εισαγωγή του από toolbox – HTML)
  - Εισαγωγή <h1> (από toolbar: Formatting), και βάζουμε ως κείμενο: "Είναι η Σελίδα 1"
  - Εισαγωγή <p> (με enter σε Design view ή από Formatting)
  - Εισαγωγή του 'μενού' και της διαχωριστικής γραμμής με Αντιγραφή-Επικόλληση από την default.aspx.
  - Εισαγωγή Label (από toolbox – Standard) και Text="Good morning!"

- Εισαγωγή script **μέσα** στο κώδικα της σελίδας (βάλτε το πριν το <html>)

```
<script runat="server">
    protected void Page_Load(object sender, EventArgs e)
   {
      if (DateTime.Now.Hour < 12)
         Label1.Text = "Καλημέρα!";
      else
         Label1.Text = "Καλησπέρα!";

   }

</script>
```

- Εκτέλεση – δοκιμή – έλεγχος. Υπάρχει πρόβλημα; Κάντε την Ιδιότητα AutoEventWireup="True"

- Στην Page2.aspx
  - Μέσα στο (υπάρχει ήδη, αλλιώς από το toolbox – HTML)
  - Εισαγωγή <h1> "Είναι η Σελίδα 2"
  - Εισαγωγή <p> (με enter σε Design view ή από Formatting)
  - Εισαγωγή του 'μενού' και της διαχωριστικής γραμμής με Αντιγραφή-Επικόλληση από την default.aspx.
  - Εισαγωγή
  - Εισαγωγή του 'Enter your name:', του textbox, του button και του label με Αντιγραφή-Επικόλληση από την default.aspx.
  - Υποστήριξη του γεγονότος κλικ για το κουμπί με εναλλακτικό τρόπο - Εισαγωγή script **μέσα** στον κώδικα (πριν το <html>):

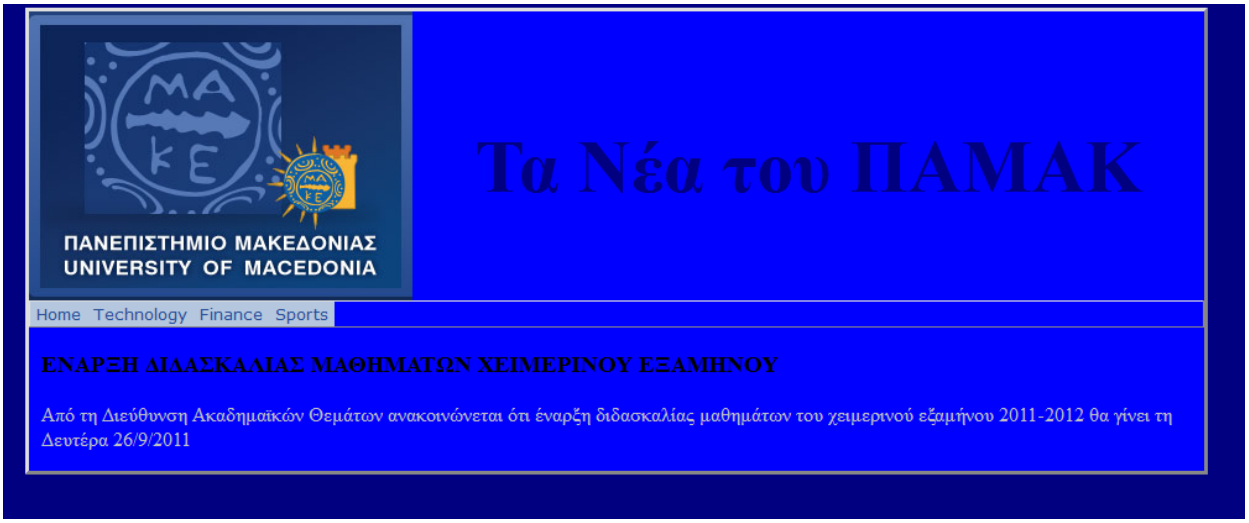
```
<script runat="server">
 protected void Button1_Click(object sender, EventArgs e)
   {
      Label1.Text = "Hello there " + TextBox1.Text  + ", πώς αισθάνεσαι σήμερα;";
   }
</script>
```

  - Εκτέλεση – δοκιμή - έλεγχος
  - Αλλάξτε τον τίτλο και το χρώμα υποβάθρου στις 3 σελίδες (βάζοντας <body bgcolor=…>)
  - Εκτέλεση – δοκιμή – έλεγχος

2.2 Χρήση και επεξεργασία προτύπων και μορφοποιήσεων CSS

Εκφώνηση:
Δημιουργήστε έναν Ιστότοπο τεσσάρων σελίδων (Home, Technology, Finance, Sports) παρόμοιο με την παραπάνω εικόνα, κάνοντας χρήση σελίδας-πρότυπο (master page) και μορφοποιήσεων Cascading Style Sheet (CSS). Για το μενού επιλογών χρησιμοποιείστε το εξειδικευμένο σχετικό εργαλείο (Menu control στην κατηγορία Navigation).

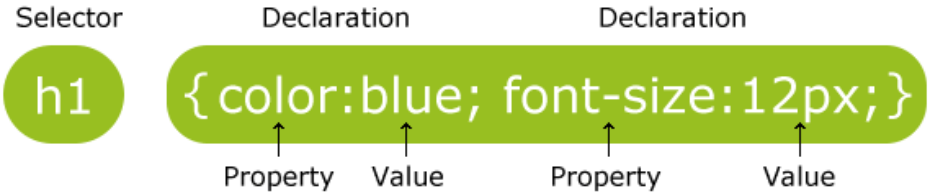
Sound 5.1.mp3	Ηχητικό απόσπασμα (audio)
Επεξήγηση τεχνολογίας CSS	



Εικόνα 5.2 *Βασική, αρχική Ιστοσελίδα με μορφοποίηση*

Υποδειγματική λύση:

- Δημιουργία καινούργιου άδειου Ιστότοπου (New Empty Web Site), γλώσσα C#, με όνομα "NewsSite"
- Εισαγωγή σελίδας-πρότυπο: επιλέγουμε new Master Page (με δεξί κλικ πάνω στο όνομα του site στο παράθυρο Solution Explorer και επιλογή Add New Item). Εξέταση του κώδικα (υπάρχει ένας ContentPlaceHolder στο <head> με id="head", και ένας στο <body> με id = "ContentPlaceHolder1")
- Αλλαγή του id "ContentPlaceHolder1" (που ανήκει στο <body>) σε "PageContent"
- Εισαγωγή εξωτερικού αρχείου μορφοποιήσεων CSS, με όνομα StyleSheet.css (File -> New -> File -> Style Sheet)
- Η γενική μορφή των κανόνων μορφοποίησης CSS δίνεται στην παρακάτω εικόνα



Εικόνα 5.3 *Γενική μορφή κανόνων CSS*

- Επεξεργασία μορφοποιήσεων: επιλογή στο αρχείο StyleSheet.css του selector 'body', εμφάνιση (αν δεν είναι εμφανής ήδη) της toolbar Style Sheet (από το μενού View), και click στο εικονίδιο του toolbar "build style" για να εμφανιστεί το παράθυρο/πλαίσιο διαλόγου μορφοποιήσεων CSS:
- Ας επιλέξουμε background -> background color, με ένα χρώμα της αρεσκείας μας (Navy). Προκύπτει:

```
body
{
   background-color: navy;
}
```

Video 5.1.mp4	Βίντεο (video)
Χρήση και επεξεργασία προτύπων και μορφοποιήσεων CSS	

Προαιρετικά μπορούμε να βάλουμε και font-family, font-size, κ.α. Σε ότι δεν ορίσουμε, ισχύουν οι προεπιλογές (“browser default”)

- Στην Master Page
  - Σύνδεση με το StyleSheet, βάζουμε μέσα στο <head> ένθετη ετικέτα:

    <link href="StyleSheet.css" rel="stylesheet" type="text/css" />

  - Εμφάνιση των αλλαγών στο Design – Παρατήρηση αποτελεσμάτων
  - Εισαγωγή ενός σε συγκεκριμένη θέση: Πηγαίνουμε στον κώδικα, επιλογή όλων των στοιχείων ανάμεσα στο <body></body>, δεξί κλικ και surround with -> html -> div). Εισαγωγή παραμέτρου id=”page” στο div
- Στο Style Sheet
  - Προσθέτουμε selector #page { }. Στο selector page κάνουμε build style και βάζουμε τις ακόλουθες ρυθμίσεις:

```
#page
{
   border-style: outset;
   margin: 0px auto auto auto;
   width: 950px;
   color: silver;
   background-color: blue;
}
```

- Δοκιμάζουμε να γράψουμε κείμενο στο Master Page και βλέπουμε στο design το αποτέλεσμα των μορφοποιήσεων #page
- Δημιουργία της επικεφαλίδας με το λογότυπο του ΠΑΜΑΚ και το τίτλο. Στο Master Page:
  - Μετά το άνοιγμα του div “page” εισαγωγή div με id=”header” και μέσα σε αυτό εισαγωγή της εικόνας του ΠΑΜΑΚ. [Έχουμε φροντίσει να ‘κατεβάσουμε’ την εικόνα ως αρχείο (university_logo.jpg) και την έχουμε αποθηκεύσει σε έναν φάκελο images που δημιουργήσαμε για αυτόν τον σκοπό.]

  - Εισαγωγή div με id=”title” (μέσα στο header) που περιέχει το κείμενο: ‘Τα νέα του ΠΑΜΑΚ’
- Μορφοποιήσεις (Στοίχιση, μέγεθος, χρώματα…). Στο αρχείο StyleSheet:
  - Εισαγωγή κανόνων CSS για το id=”header”, την εικόνα, και το id=”title”

```
#header
{
   height: 234px;
}

img
{
  float: left;  /* απαραίτητο ώστε να έχουμε εικόνα και κείμενο μαζί */
           /* προσέξτε ότι το img δεν έχει πριν ‘#’ γιατί δεν είναι id */

}

#title
{
  float: right;  /* απαραίτητο ώστε να έχουμε εικόνα και κείμενο μαζί */
  width: 600px;
```

```
  height: 234px;
  color: navy;
  text-align: center;
  line-height: 250px;  /* γρήγορη τεχνική για κάθετο κεντράρισμα */
  font-size: 60px;
  font-weight: bold;
}
```

- Παρατηρούμε στο Master Page το αποτέλεσμα των μορφοποιήσεων
- Συνεχίζουμε στο Master Page, για να βάλουμε το μενού:
    - Μετά το κλείσιμο του "header" div, και μέσα στο form, πριν το υπάρχον περιεχόμενο, εισαγωγή νέου div με id="menu" και μέσα σε αυτό εισαγωγή Menu από το Toolbox (κατηγορία: Navigation)
- Εισαγωγή τεσσάρων ακόμα σελίδων (Home, Technology, Finance, Sports) ως content pages (με δεξί click στο master page: add content page)
- Διαμόρφωση του navigation menu (Edit Menu Items) και σύνδεση με τις τέσσερις σελίδες. Προσοχή: Text – το ορατό κείμενο της επιλογής, NavigateUrl – η σύνδεση με την αντίστοιχη σελίδα. Επίσης βάλτε την ιδιότητα Orientation του Menu στην τιμή Horizontal. Παρατηρήστε τον κώδικα που παράχθηκε.

- Στο αρχείο StyleSheet.css, εισάγουμε κανόνες CSS για το id='menu'

```
#menu
{
   border-style: outset;
   border-width: thin;
}
```

- Προβολή των σελίδων και επιβεβαίωση ότι ακολουθούν τη ζητούμενη μορφοποίηση

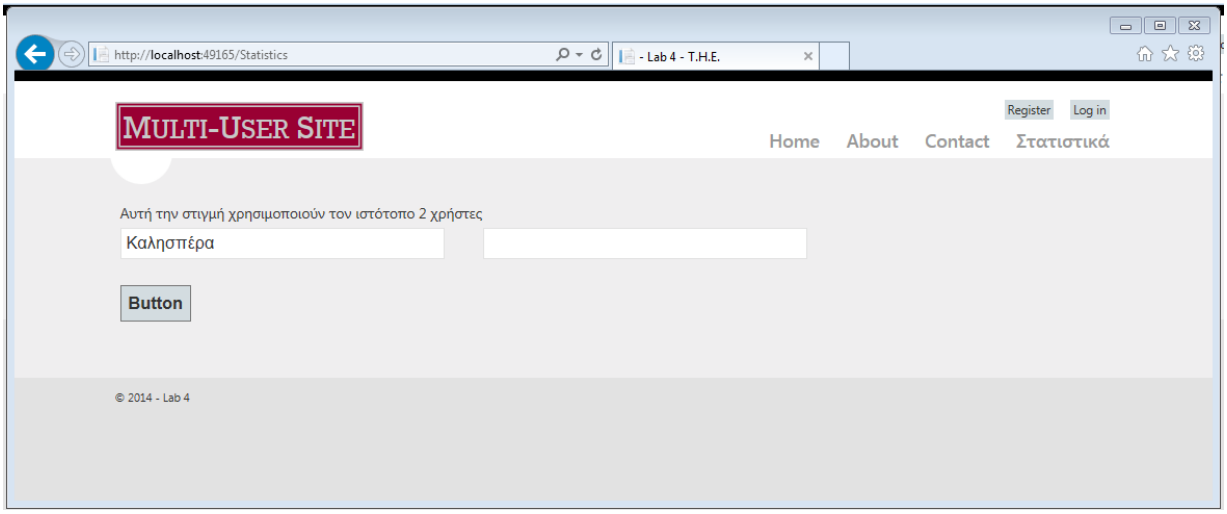
- Στο Master Page, κάνουμε το ContentPlaceHolder να περικλείεται από div με id="content" ώστε να το μορφοποιήσουμε με CSS κανόνες
- Μορφοποίηση του content με συμπληρωματικούς κανόνες στο Style Sheet:

```
#content
{
   margin: 10px;
}

#content img   /* img που εμπεριέχεται σε content*/
{
   border: thick double #800000;
   height: 68px;
   width: 100px;
}

#content h3   /* h3 που εμπεριέχεται σε content*/
{
   color:black;
}
```

- Ελέγξτε τις μορφοποιήσεις αυτές. Βάλτε για παράδειγμα κείμενο με μορφοποίηση <h3> εντός της περιοχής 'content', και άλλο κείμενο με μορφοποίηση <h3> εκτός αυτής. Παρατηρήστε ότι στην πρώτη μόνο περίπτωση τα γράμματα γίνονται μαύρα.

2.3 Προσαρμογή έτοιμου Ιστότοπου - χρήση των application και session events

Εκφώνηση:
Δημιουργήστε έναν Ιστότοπο (με κεφαλίδα 'Multi-User Site') αξιοποιώντας και προσαρμόζοντας την παρεχόμενη λειτουργικότητα των συστατικών που αυτόματα δημιουργεί το περιβάλλον ανάπτυξης VS 2012. Στη συνέχεια προσθέστε μια νέα σελίδα Statistics.aspx, σύμφωνη με το υπάρχον πρότυπο, και συνδέστε την επιλογή 'Στατιστικά' του μενού. Στη σελίδα αυτή αρχικά μέσω 2 textboxes δώστε ένα παράδειγμα χρήσης της ιδιότητας IsPostBack που ελέγχει την ύπαρξη/διατήρηση ενός session και στη συνέχεια δημιουργείστε κατάλληλα application και session events για να εμφανίζετε ένα μήνυμα που ενημερώνει πόσοι χρήστες χρησιμοποιούν αυτή τη στιγμή τον Ιστότοπο (δείτε την παρακάτω εικόνα).



Εικόνα 5.4 *Ιστοσελίδα με πολλούς χρήστες ταυτόχρονα*

Υποδειγματική λύση:

- Δημιουργία νέου έτοιμου Ιστότοπου (ASP.NET Web Forms site) με όνομα "Lab3" και γλώσσα C#.
- Παρατήρηση – πρώτη επαφή με τα στοιχεία/συστατικά που δημιουργούνται από το περιβάλλον ανάπτυξης.
- Επεξεργασία της σελίδας-πρότυπο (Master Page, αρχείο Site.master)
  Εντοπισμός στον κώδικα και αλλαγή της επικεφαλίδας: είναι ως class="site-title" και έχει τιμή "your logo here". Αλλάξτε το σε "Multi-User Site"

```
<p class="site-title">
        <a runat="server" href="~/">Multi-User Site</a>
 </p>
```

  - Αλλαγή του υποσέλιδου. Θέλουμε το σύμβολο copyright, δίπλα το τρέχον έτος και δίπλα το κείμενο " – Lab 3". Προσέξτε ότι για τον υπολογισμό του τρέχοντος έτους ζητούμε την εκτέλεση της κατάλληλης συνάρτησης, χρησιμοποιώντας στην ετικέτα το σύμβολο '%':

```
<p>&copy; <%: DateTime.Now.Year %> - Lab 3 </p>
```

  - Αλλαγή του title (εμφανίζεται στον browser ως τίτλος της καρτέλας/παραθύρου). Και εδώ, θέλουμε να συνδυάσουμε 'δυναμικό κείμενο', όπως ποιος είναι ο τίτλος της τρέχουσας σελίδας, με το σταθερό κείμενο " - Lab 3 – T.H.E.". Πάλι το σύμβολο '%' μπορεί να 'υπολογίσει' την τιμή μιας ιδιότητας (Page.Title):

```
<title><%: Page.Title %> - Lab 3 – T.H.E.</title>
```

- Εντοπισμός διασύνδεσης με αρχεία μορφοποίησης (υπάρχει αρχείο Content/Site.css, το εντοπίζουμε κάνοντας: Edit->Find "Site.css")
  (… και βρίσκουμε στο αρχείο Bundle.config την αναφορά που συνδέει το αρχείο μορφοποιήσεων με τη σελίδα-πρότυπο.)

    - Προσθήκη κανόνα CSS ώστε να εμφανίζει την επικεφαλίδα σε μορφή "small-caps" (εμφάνιση του CSS properties παραθύρου – βρίσκεται από το View-> CSS Properties, click στην επικεφαλίδα "Multi-User Site" και πάτημα του Summary στο CSS properties. Δεξί κλικ στην κατάλληλη γραμμή του Applied Rules και επιλογή Modify Style για το γνώριμο πλαίσιο διαλόγου κανόνων CSS).
    - Εντοπισμός του κανόνα και στο .css αρχείο:

```
.site-title {
    color: #c8c8c8;
    font-family: Rockwell, Consolas, "Courier New", Courier, monospace;
    font-size: 2.3em;
→   font-variant: small-caps;
     margin: 0;
}
```

    - Εντοπίστε το background-color και αλλάξτε το σε χρώμα σκούρο κόκκινο:
      background-color: #990033;

    - Βάλτε στην επικεφαλίδα διπλό περίγραμμα (border), παχύ (thick), και για να φανεί αυτό δείτε ότι πρέπει να βάλετε κάποια τιμή στο margin (έστω auto). Όταν τελειώσετε την άσκηση, επιστρέψτε σε αυτό το σημείο και πειραματιστείτε στην αλλαγή κανόνων CSS

- Εισαγωγή νέας σελίδας βασισμένη στο πρότυπο Site.master με το όνομα Statistics.aspx
- Εντοπισμός του μενού (δίνοντας Edit->Find "Home") και προσθήκη της σελίδας Statistics στο menu του προτύπου (με copy την επιλογή ενός υπάρχοντος στοιχείου του μενού, πχ. του Contact, και προσαρμογή του):

```
<nav>
        <ul id="menu">
          <li><a runat="server" href="~/">Home</a></li>
          <li><a runat="server" href="~/About">About</a></li>
          <li><a runat="server" href="~/Contact">Contact</a></li>
→         <li><a runat="server" href="~/Statistics">Στατιστικά</a></li>
        </ul>
</nav>
```

- Αντικαταστήστε το favicon με ένα άλλο favicon2 (πρέπει να εντοπίσετε/κατεβάσετε ένα άλλο αρχείο .ico, να το προσθέσετε (add existing item) στο site και να ενημερώσετε την αντίστοιχη εντολή:

  ```
  <link href="~/favicon2.ico" rel="shortcut icon" type="image/x-icon" />
  ```

  Βρείτε/δείτε που εμφανίζεται, κάνοντας εκτέλεση του Ιστότοπου.

Ιδιότητα IsPostBack και έλεγχος για session:

- Στην σελίδα Statistics:
    - Εισαγωγή δύο textbox και ενός κουμπιού από το toolbox

- Προσθήκη κώδικα ώστε το περιεχόμενου του δεύτερου textbox να γεμίζει με το περιεχόμενο του πρώτου textbox με το πάτημα του κουμπιού και εκτέλεση για δοκιμή

```
protected void Button1_Click(object sender, EventArgs e)
{
TextBox2.Text = TextBox1.Text;
}
```

- Έστω ότι θέλουμε να αρχικοποιούμε το TextBox1 και γνωρίζουμε την τιμή εκ των προτέρων, π.χ. «Καλημέρα». Δοκιμάζουμε αρχικά κάνοντας χρήση του παράθυρου ιδιοτήτων (property window), και βάζοντας την τιμή «Καλημέρα». Κάντε δοκιμές στην εκτέλεση: πατήστε κατευθείαν το κουμπί για να δείτε το «Καλημέρα» και στο TextBox2, και δείτε αν αυτό συμβαίνει και όταν αλλάξετε το περιεχόμενο του TextBox1, πχ. σε Θεσσαλονίκη, και μετά πατήστε το κουμπί να δείτε τι εμφανίζεται στο TextBox2. Κανονικά δε θα έχετε πρόβλημα.
- Έστω όμως ότι δε γνωρίζουμε από την αρχή με ποια τιμή θέλουμε να αρχικοποιήσουμε το TextBox1. Οπότε δεν μπορούμε να χρησιμοποιήσουμε το παράθυρο ιδιοτήτων. Σβήστε λοιπόν στο παράθυρο ιδιοτήτων του TextBox1 την τιμή «Καλημέρα» από την ιδιότητα Text. Η αρχικοποίηση θα γίνει δυναμικά, με κώδικα: στο Page_Load event κάνουμε αρχικοποίηση του περιεχομένου του TextBox1:

```
string minima="Καλησπέρα";
        protected void Page_Load(object sender, EventArgs e)
        {
       TextBox1.Text = minima;
      }
```

- Εκτέλεση και δοκιμές ξανά. Παρατηρούμε ότι αλλάζοντας το περιεχόμενο του TextBox1, με το πάτημα του κουμπιού, ξαναεπιστρέφει η τιμή «Καλησπέρα». Αυτό μας 'δείχνει' ότι το Page_Load event προηγείται του Button1_ Click και έχουμε τελικά πάντα «Καλησπέρα» ως τιμή
- Για να αντιμετωπίσουμε αυτό το πρόβλημα μπορούμε να ελέγχουμε αν η σελίδα εκτελείται για πρώτη φορά ή όχι μέσω της ιδιότητας IsPostBack

```
string minima="Καλησπέρα";
protected void Page_Load(object sender, EventArgs e)
{
if (!Page.IsPostBack) {
  TextBox1.Text = minima;
}
}
```

- Εκτέλεση και δοκιμή

Συμβάντα εφαρμογής (καθολικά για όλον τον Ιστότοπο - application events) και συμβάντα συνεδρίας χρήστη (user session events):

- Application Object – Session Object
  - Ένα Application Object δημιουργείται όταν ο πρώτος χρήστης κάνει το πρώτο request στον Ιστότοπό μας. Έχει εμβέλεια σε όλες τις σελίδες και σε όλους τους χρήστες. Καταστρέφεται όταν κλείσει ο web server ή μετά από μεγάλη αδράνεια
  - Ένα Session Object δημιουργείται ξεχωριστά για **κάθε** χρήστη όταν κάνει το πρώτο του request στον Ιστότοπό μας. Έχει εμβέλεια σε όλες τις σελίδες για κάθε χρήστη. Καταστρέφεται μετά από αδράνεια του χρήστη (π.χ. 20 min)
  - Μπορούμε να χειριστούμε τα events της δημιουργίας και της καταστροφής των αντικειμένων του Application και του Session από το αρχείο Global.asax. Παράδειγμα:
- Στο Global.asax

- Εισαγωγή μεταβλητής που μετράει τους χρήστες στο Application_Start event

```
void Application_Start(object sender, EventArgs e)
{
// Code that runs on application startup
…
Application["UserCount"] = 0;

}
```

- Αύξηση της μεταβλητής όταν χρησιμοποιεί τον Ιστότοπό μας νέος χρήστης. Μείωση της μεταβλητής όταν λήξει το Session του χρήστη

```
void Session_Start(object sender, EventArgs e)
{
// Code that runs when a new session is started
Application["UserCount"] = Convert.ToUInt32(Application["UserCount"]) + 1;
}

   void Session_End(object sender, EventArgs e)
{
     // Code that runs when a session ends.
      // Note: The Session_End event is raised only when the sessionstate mode
      // is set to InProc in the Web.config file. If session mode is set to StateServer
     // or SQLServer, the event is not raised.
    Application["UserCount"] = Convert.ToUInt32(Application["UserCount"]) - 1;
   }
```

- Στην σελίδα Statistics
  - Εισαγωγή Label από το toolbox
  - Μετατροπή του Page_Load event ώστε να εμφανίζει τους τρέχοντες χρήστες στο label

```
protected void Page_Load(object sender, EventArgs e)
{
  …
  Label1.Text = "Αυτή τη στιγμή χρησιμοποιούν τον Ιστότοπο " +
   Convert.ToString(Application["UserCount"]) + " χρήστες";
 }
```

  - Εκτέλεση και δοκιμή (refresh σελίδας, άνοιγμα νέου browser και αντιγραφή του url του Ιστότοπου για να προσομοιώσουμε 2ο χρήστη, refresh ξανά κτλ.)

Προσοχή: Δε βλέπουμε τον αριθμό των χρηστών να ελαττώνεται. Για να γίνει το συμβάν Session_End, σύμφωνα και με τα σχόλια που υπάρχουν στο αντίστοιχο τμήμα του αρχείου Global.asax, πρέπει να πάμε στο αρχείο Web.config, μέσα στο τμήμα του <system.web>, και να βάλουμε:

```
<sessionState    mode = "inProc" …    timeout="1" />
```

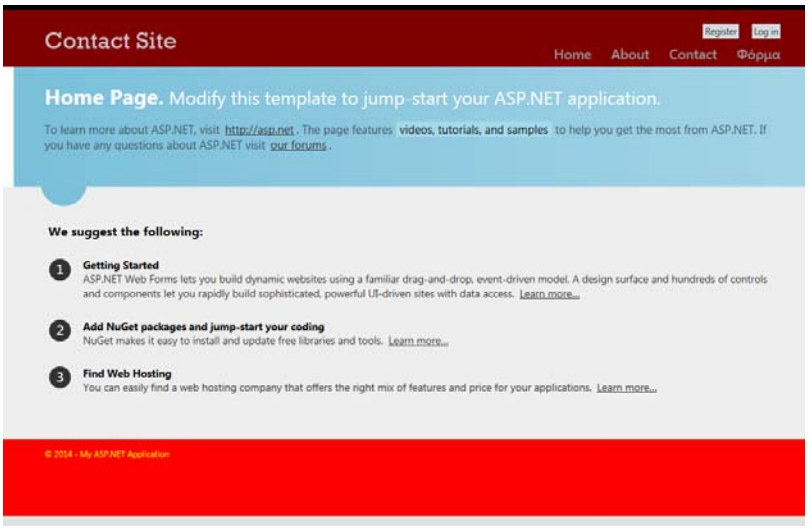
Έτσι θέτουμε το mode στην επιθυμητή κατάσταση και αλλάζουμε σε 1 λεπτό τον χρόνο αδράνειας για τη λήξη του session (για να μπορούμε να κάνουμε έλεγχο αμέσως μετά από ένα λεπτό αδράνειας).

2.4. Φόρμες και επικύρωση δεδομένων (validation)

Εκφώνηση:
Δημιουργήστε έναν Ιστότοπο (με κεφαλίδα 'Contact Site') αξιοποιώντας την παρεχόμενη λειτουργικότητα των συστατικών που αυτόματα δημιουργεί το περιβάλλον ανάπτυξης VS 2012. Κάντε μια σειρά αλλαγών: αλλάξτε το υπόβαθρο του κύριου μέρους των σελίδων σε καφέ, ενώ στο υποσέλιδο το υπόβαθρο κάντε το κόκκινο. Τα γράμματα στο υπόβαθρο να είναι κίτρινα. Τέλος, όταν περνάει το ποντίκι πάνω από το μενού

(event ‘hover’) τα γράμματα κάντε τα να γίνονται κίτρινα. Στη συνέχεια προσθέστε μια νέα σελίδα Validators.aspx, σύμφωνη με το υπάρχον πρότυπο, και συνδέστε την επιλογή ‘Φόρμα’ του μενού. Στη σελίδα αυτή βάλτε έναν πίνακα 18 γραμμών και 2 στηλών για να δώσετε σε μια στήλη περιγραφές στοιχείων (π.χ. όνομα, πόλη κλπ.) που πρέπει ο χρήστης να γεμίσει στα πεδία της διπλανής στήλης (δείτε την παρακάτω εικόνα). Χρησιμοποιείστε εργαλεία επικύρωσης (validators) για να ελέγξετε την εισαγωγή των στοιχείων που κάνει ο χρήστης (έλεγχος ότι δεν έμεινε ένα πεδίο της φόρμας άδειο, έλεγχος ότι έγινε μια επιλογή από μια πτυσσόμενη λίστα, έλεγχος ότι το mail που δόθηκε έχει σωστή ‘μορφοποίηση’, έλεγχος ότι πατήθηκε ένα checkbox).

Sound 5.2.mp3	Ηχητικό απόσπασμα (audio)
Επεξήγηση φορμών εισαγωγής δεδομένων	



Εικόνα 5.5 *Σελίδα επικοινωνίας*

Υποδειγματική λύση:

- Δημιουργία νέου έτοιμου Ιστότοπου (ASP.NET Web Forms site) με όνομα “Lab4” και γλώσσα C#.
- Εισαγωγή καινούργιας σελίδας βασισμένη στο πρότυπο με το όνομα Validators.aspx
- Προσθήκη της σελίδας Validators στο menu του προτύπου, με κείμενο ‘Φόρμα’ και εκτέλεση
- Επεξεργασία του προτύπου Master Page
  - Αλλαγή της κεφαλίδας σε “Contact Site”
  - Αλλαγές στη μορφοποίηση του Προτύπου και εκτέλεση
- Επεξεργασία του Site.css για αλλαγές στο πρότυπο .master
  - Στο **body** καφέ υπόβαθρο: background-color: brown;
  - Στο υποσέλιδο (**footer**) κόκκινο υπόβαθρο: background-color: red;
  - Θέλουμε κίτρινα τα γράμματα στο υποσέλιδο:

```
color: yellow;
```

  - Εναλλακτικά:

```
.float-left {
    …
    color: yellow;
}
```

  - Προσέξτε ότι και η επικεφαλίδα Contact Site είναι μέσα σε περιοχή .float-left. Γιατί όμως δεν έγιναν και εκεί κίτρινα τα γράμματα; Ποια είναι η ειδικότερη ρύθμιση/κανόνας css που ισχύει;
  - Θέλουμε όταν περνάει το ποντίκι πάνω από το μενού (event ‘hover’) τα γράμματα να γίνονται κίτρινα:

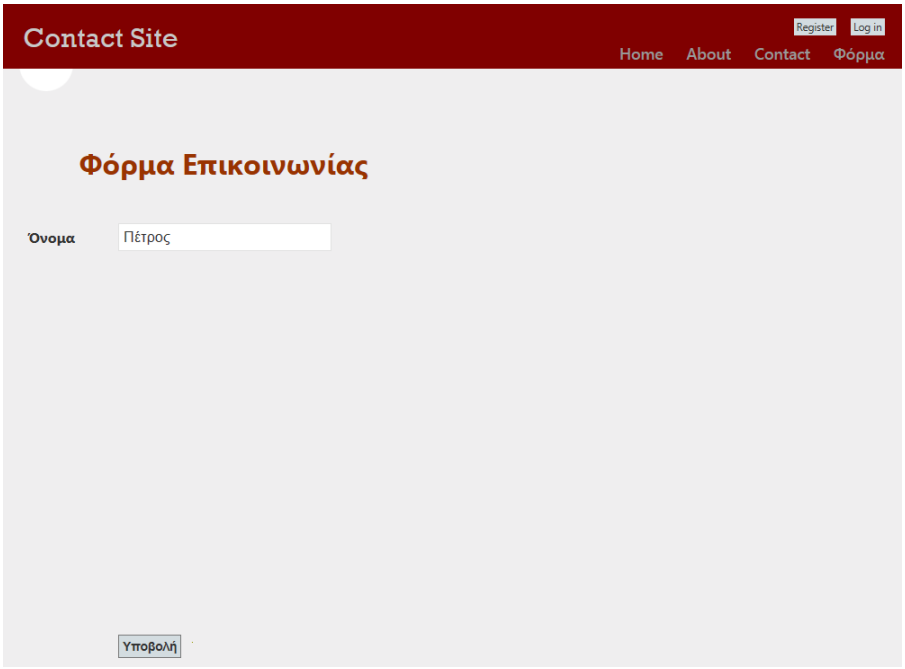
```
ul#menu li a:hover {
        …
        color: yellow;
    }
```

- Μπορείτε να δοκιμάσετε και άλλες αλλαγές

- Στην σελίδα Validators:
  - Εισαγωγή πίνακα 18x2 (από μενού Table)
  - Στην πρώτη γραμμή ενώνουμε τα κελιά (επιλέγουμε τη γραμμή και από το μενού επιλέγουμε Table – Modify – Merge Cells) και γράφουμε με καφέ μεγάλα γράμματα Φόρμα Επικοινωνίας
  - Αφήνουμε τη δεύτερη γραμμή κενή
  - Στην τρίτη γραμμή προσθέτουμε το κείμενο Όνομα στο αριστερό κελί και στο δεξί κελί ένα Textbox. Δίνουμε το ID txtName στις ιδιότητες του Textbox.
    Έστω ότι θέλουμε να μην μπορεί να αφήνει κενό το textbox ο χρήστης, όταν συμπληρώνει τη φόρμα. Από την κατηγορία Validation στο Toolbox εισάγουμε δίπλα στο Textbox το στοιχείο **RequiredFieldValidator**. Στις Ιδιότητες του RequiredFieldValidator αλλάζουμε τα
    - ErrorMessage: Πρέπει να συμπληρώσετε το όνομά σας
    - Text: *
    - ForeColor: Red

    ControlToValidate: txtName
  - Για να δοκιμάσουμε τη συμπεριφορά του validator, στην τελευταία γραμμή του πίνακα, στο δεξί κελί προσθέτουμε ένα button από το toolbox με text: Υποβολή (προσέξτε την ιδιότητα CausesValidation να είναι True)
    Κάτω από τον πίνακα εισάγουμε το στοιχείο Validation Summary (είναι και αυτό στην κατηγορία Validation του Toolbox) και σε αυτό αλλάζουμε τα
    - ForeColor: Red
    - HeaderText: Προσοχή:

- Δοκιμή της φόρμας μας
  - αφήστε κενό το textbox και πατήστε Υποβολή. Παρατηρήστε τι εμφανίζεται κάτω από το κουμπί 'Υποβολή'
  - Βάλτε κάποιους χαρακτήρες στο textbox και πατήστε Υποβολή. Τι άλλαξε;



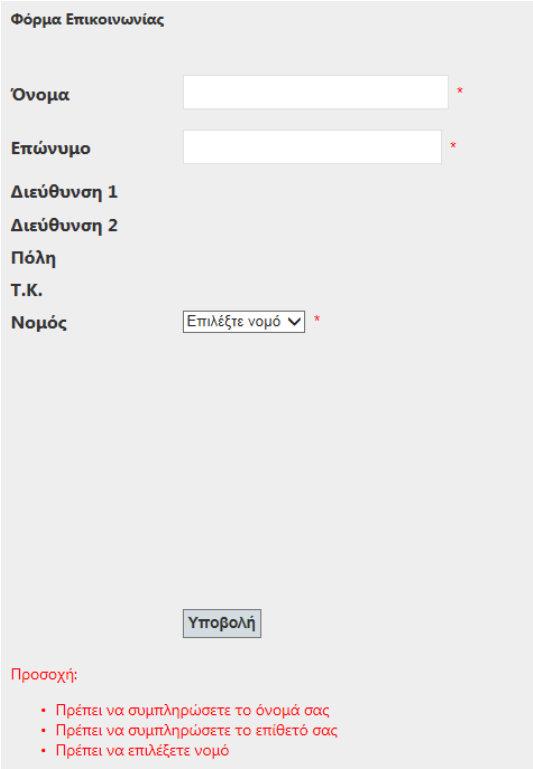
Εικόνα 5.6 *Φόρμα επικοινωνίας*

- Όμοια με το Όνομα, κάνουμε για εξάσκηση τις αντίστοιχες ενέργειες για την επόμενη γραμμή με το Επώνυμο
- Στις επόμενες γραμμές (5η-8η) βάζουμε Διεύθυνση 1, Διεύθυνση 2, Πόλη και Τ.Κ.
  Θυμηθείτε όταν τελειώσουμε το παράδειγμα, για εξάσκηση, να τοποθετήσετε ελέγχους RequiredFieldValidator και στα πεδία $5^{ης}$, $7^{ης}$ και $8^{ης}$ γραμμής
- Βρισκόμαστε στην $9^{η}$ γραμμή και προσθέτουμε στο αριστερό κελί τη λέξη Νομός. Στο δεξί κελί εισάγουμε το αντικείμενο DropDownList από το Toolbox. Δίνουμε το ID lstStateList στις ιδιότητες του DropDownList και εισάγουμε σε αυτό (Edit Items) διάφορους Νομούς (π.χ. Θεσσαλονίκης, Καβάλας, Μαγνησίας)
- Για να μπορέσουμε να δοκιμάσουμε Validators σε λίστα τιμών, καταχωρούμε και μια τιμή του τύπου: 'Επιλέξτε νομό'. Την τιμή αυτή την ορίζουμε ως προεπιλογή (ζητάμε να έχει στην ιδιότητα Selected την τιμή True, και χωρίς να είναι υποχρεωτικό, αλλά για λόγους αισθητικούς τη βάζουμε στη θέση 0).
- Από την κατηγορία Validator στο Toolbox εισάγουμε δίπλα στο DropDownList το στοιχείο **CompareValidator**. Στις Ιδιότητες του CompareValidator αλλάζουμε τα
  - ErrorMessage: Πρέπει να επιλέξετε τον νομό σας
  - Text: *
  - ForeColor: Red
  - ControlToValidate: lstStateList
  - Operator: NotEqual
  - ValueToCompare: Επιλέξτε νομό

  Δοκιμή της φόρμας μας (με κενά όλα τα πεδία, με κενό ένα από τα πεδία, με κανένα κενό πεδίο)

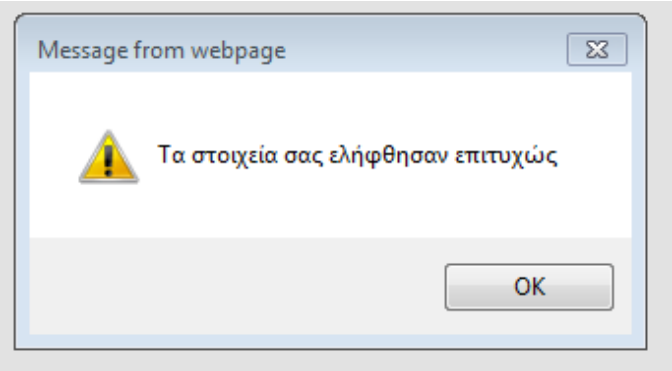
Κάντε κλικ στα εικονίδια του σχήματος για επεξήγηση

Dynamic 5.1.zip	Διαδραστική εικόνα (interactive)
Εικόνα 5.7 *Χρήση Validators*	



Εικόνα 5.7 *Χρήση Validators*

- Έστω ότι όταν όλα είναι έγκυρα, δε θέλουμε απλώς να μη βγαίνουν τα κόκκινα μηνύματα (του Validation Summary), αλλά θέλουμε να βγαίνει και ένα μήνυμα



Εικόνα 5.8 *Επιτυχής καταχώρηση δεδομένων φόρμας*

- Για να βγει το μήνυμα, βάζουμε μέσα στη συνάρτηση Button1_Click τον ακόλουθο κώδικα:

```
if (Page.IsValid) {
    ScriptManager.RegisterClientScriptBlock(Page, typeof(Page), "ClientScript",
          "alert('Τα στοιχεία σας ελήφθησαν επιτυχώς')", true);
 }
```

- Κάνουμε τις αντίστοιχες ενέργειες και για τη Χώρα
- Αφήνουμε μια γραμμή κενή

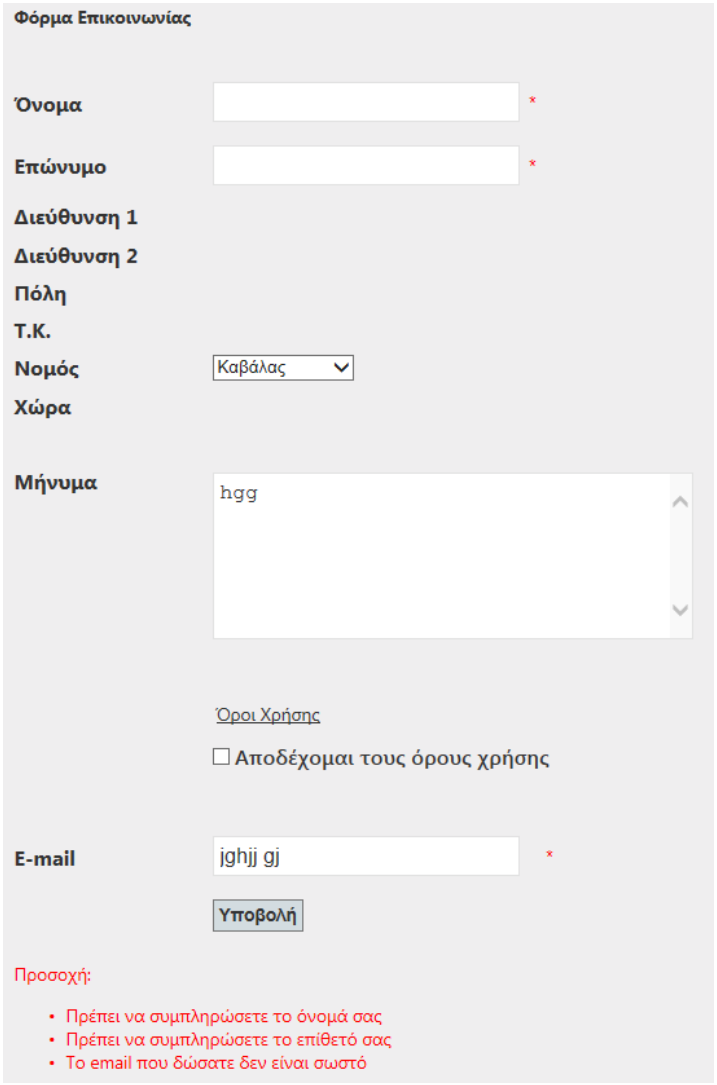
- Εισάγουμε τη λέξη Μήνυμα αριστερά (12η γραμμή) και ένα μεγαλύτερο σε ύψος Textbox δεξιά. Για να είναι πολλαπλών γραμμών, βάζουμε στην ιδιότητά του TextMode την τιμή Multiline.
- Για να βγει η λέξη *Μήνυμα* στο ύψος της πρώτης γραμμής του multiline textbox, στην ιδιότητα style, επιλέγουμε Block-> vertical-align: top
- Βάζουμε τον αντίστοιχο RequiredFieldValidator και αφήνουμε μια γραμμή κενή
- Στο δεξί κελί της 14ης γραμμής εισάγετε ένα Hyperlink από το Toolbox με κείμενο Όροι Χρήσης και να οδηγεί στη σελίδα About
- Στην επόμενη γραμμή, στο δεξί κελί εισάγετε ένα checkbox από το Toolbar με text: Αποδέχομαι τους Όρους Χρήσης
- Για να μην είναι το κείμενο κάτω από το checkbox, απενεργοποιούμε τον κανόνα css σχετικά με το display block:

```
label {
     /*   display: block; */
      font-size: 1.2em;
      font-weight: 600;
   }
```

- Αφήνουμε μια γραμμή κενή
- Στο αριστερό κελί της 17ης γραμμής, εισάγετε το κείμενο *E-mail* και στο δεξί κελί ένα Textbox (με ID txtEmail – έχουμε δώσει σε όλα τα στοιχεία δικό μας ID) και εισάγουμε το αντίστοιχο **RequiredFieldValidator**. Εισάγουμε επίσης και ένα **RegularExpressionValidator**. Στο RegularExpressionValidator αλλάζουμε τα
    - ErrorMessage: Το email που δώσατε δεν είναι σωστό
    - Text: *
    - ForeColor: Red
    - ControlToValidate: txtEmail

- ValidationExpression: (επιλέγουμε Internet e-mail address)

- Δοκιμή της φόρμας μας (με κενά όλα τα πεδία, με κενό ένα από τα πεδία, με κανένα κενό πεδίο, με λανθασμένο mail)



Εικόνα 5.9 *Παράδειγμα με λάθη σε χρήση φόρμας*

- Επικύρωση για checkbox (χρήση **CustomValidator** μέσω κώδικα)
  - Στο Validators.aspx, στο τέλος μετά το

    …

    <asp:ValidationSummary ID="ValidationSummary1" runat="server" ForeColor="Red" HeaderText="Προσοχή:" />

    προσθέτουμε πριν το </asp:Content>, ένα CustomValidator (από το Toolbox-> Validation). Παρατηρούμε ότι δημιουργήθηκε ο ακόλουθος κώδικας:

    <asp:CustomValidator ID="CustomValidator1" runat="server" ErrorMessage="CustomValidator">
    </asp:CustomValidator>

  - Ενημερώνουμε τις ιδιότητες ForeColor, ErrorMessage και το event OnServerValidate με τις ακόλουθες τιμές:

```
<asp:CustomValidator ID="CustomValidator1" runat="server" ForeColor =
   "Red" ErrorMessage=" Παρακαλώ πατήστε το Checkbox!"
    OnServerValidate="CustomValidator1_ServerValidate">
 </asp:CustomValidator>
```

- Στο Validators.aspx.cs προσθέτουμε τον ακόλουθο κώδικα:

```
Protected void CustomValidator1_ServerValidate
                    (object source, ServerValidateEventArgs args)
{
        args.IsValid = CheckBox1.Checked;
}
```

- Δοκιμή ξανά της φόρμας μας
- Παρατηρούμε ότι βγαίνει 2 φορές το μήνυμα λάθους από τον CustomValidator. Η μία φορά είναι από το ErrorMessage και η άλλη από το ValidationSummary. Οπότε για να γλιτώσουμε από τη μια εμφάνιση αφαιρούμε την τιμή από το ErrorMessage και το μήνυμα λάθους το βάζουμε στην ιδιότητα Text. Και προκύπτει:

```
<asp:CustomValidator ID="CustomValidator1" runat="server"
ForeColor="Red"  OnServerValidate="CustomValidator1_ServerValidate">  Παρακαλώ
πατήστε το Checkbox!
</asp:CustomValidator>
```

2.5. Σύνδεση σε βάση δεδομένων

Εκφώνηση:
Δημιουργήστε έναν Ιστότοπο (με κεφαλίδα 'Database Site') αξιοποιώντας την παρεχόμενη λειτουργικότητα των συστατικών που αυτόματα δημιουργεί το περιβάλλον ανάπτυξης VS 2012. Δημιουργήστε μια SQL Server βάση δεδομένων 'Customers.mdf' με έναν πίνακα πελατών 'Customer'. Ο πίνακας θα έχει 5 πεδία (δείτε την παρακάτω εικόνα). Εισάγετε σε αυτόν 4 γραμμές (εγγραφές) με στοιχεία. Στη συνέχεια προσθέστε μια νέα σελίδα 'Dbase.aspx', σύμφωνη με το υπάρχον πρότυπο, και συνδέστε τη με το μενού. Στη σελίδα αυτή να εμφανίσετε το περιεχόμενο του πίνακα Customer (στη κλασική μορφή πλέγματος στηλών και γραμμών). Προσθέστε ακόμη μια σελίδα 'Customers.aspx', σύμφωνη με το υπάρχον πρότυπο, και συνδέστε τη με το μενού. Στη σελίδα αυτή να εμφανίζονται μόνο τα πεδία 'Όνομα' και 'Επώνυμο' όλων των πελατών σε μορφή πλέγματος και επιπλέον να δίνετε τη δυνατότητα ο χρήστης να επιλέγει μια εγγραφή (έναν πελάτη) και από κάτω να εμφανίζονται στη σελίδα όλα τα πεδία-στοιχεία του πελάτη αυτού (δείτε την τελευταία εικόνα της παραγράφου).

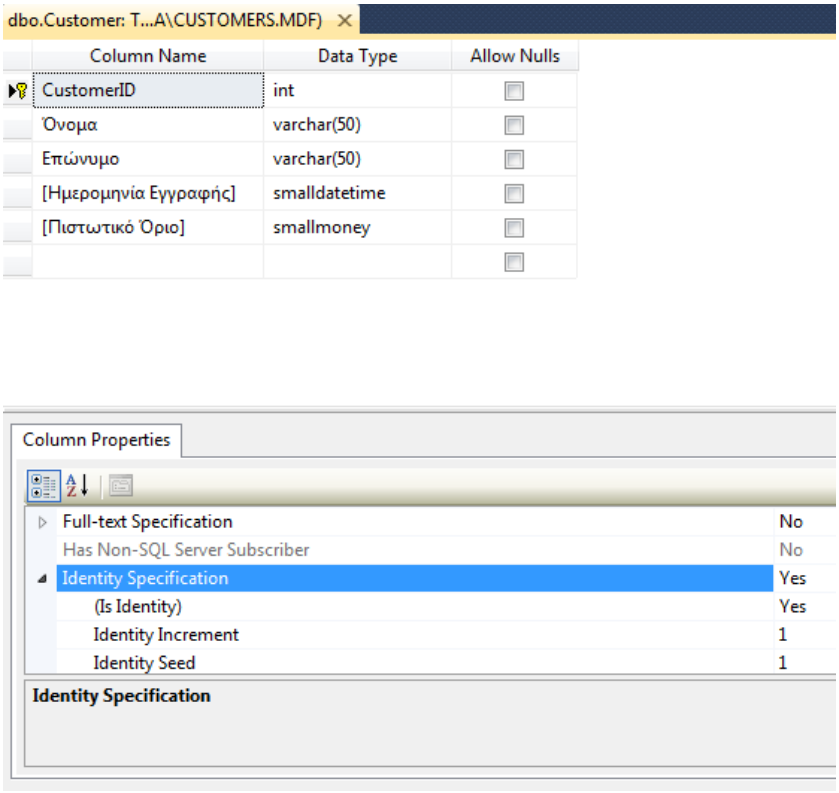
Υποδειγματική λύση:

- Δημιουργία νέου έτοιμου Ιστότοπου (ASP.NET Web Forms site) με όνομα "Lab5" και γλώσσα C#.
- Επεξεργασία του Master Page
  - Αλλαγή της επικεφαλίδας σε "Database Site"
- Δημιουργία και εμπλουτισμός ΒΔ
  - Επιλέγουμε το φάκελο App_Data (Application Data) του Ιστότοπου, κάνουμε δεξί κλικ, Add New Item, και προσθέτουμε SQL Server Database με το όνομα Customers.mdf.
  - Στο Server Explorer (View-> Server Explorer) αρχικά με δεξί κλικ πάνω στο Customers.mdf ζητάμε Refresh ώστε να μην υπάρχει πάνω του η ένδειξη του κόκκινου 'x' (που δηλώνει μη σύνδεση)
  - Επιλέγουμε Tables, δεξί κλικ, και Add New Table.

- Προσοχή: για να αποθηκευτεί ο πίνακας με το όνομα Customer, πρέπει από αυτό το σημείο να πάμε στο κάτω παράθυρο (Design) και να αλλάξουμε την CREATE TABLE σε:

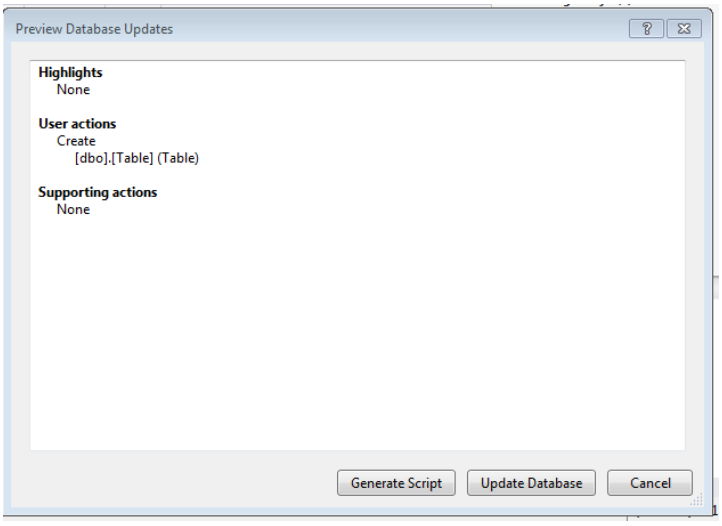
CREATE TABLE [dbo].[Customer]

- Εισάγουμε τα στοιχεία του πίνακα όπως φαίνονται στην παρακάτω εικόνα. Στις ιδιότητες του CustomerID , στο πεδίο Identity Specification επιλεγούμε True στο Is Identity και ελέγχουμε ότι το CustomerID είναι primary key



Εικόνα 5.10 *Δημιουργία βάσης δεδομένων*

- Για την αποθήκευση, πατάμε Update και στη συνέχεια Update Database.



Εικόνα 5.11 *Αποθήκευση βάσης δεδομένων*

- Κάνουμε δεξί κλικ στον πίνακα Customer από τον Server Explorer και επιλέγουμε Show Table Data. Γεμίζουμε τον πίνακα με στοιχεία.

Customer: Query...A\CUSTOMERS.MDF) × dbo.Customer: T...A\CUSTOMERS.MDF)

	CustomerID	Όνομα	Επώνυμο	Ημερομηνία Εγγραφής	Πιστωτικό Όριο
	1	Νίκος	Παπαδόπουλος	14/3/2006 12:00:00 πμ	3000,0000
	2	Μαρία	Μάρου	3/6/2008 12:00:00 πμ	6500,0000
	3	Παναγιώτης	Βροντάκης	22/8/2010 12:00:00 πμ	2000,0000
	4	Γιάννης	Ντοΐδης	3/5/2010 12:00:00 πμ	10000,0000
▸*	NULL	NULL	NULL	NULL	NULL

Εικόνα 5.12 *Εισαγωγή δεδομένων στη βάση*

- Εισαγωγή καινούργιας σελίδας βασισμένη στο πρότυπο με το όνομα Dbase.aspx
- Προσθήκη της σελίδας Dbase στο menu του προτύπου, με κείμενο 'Βάση Δεδομένων' και εκτέλεση

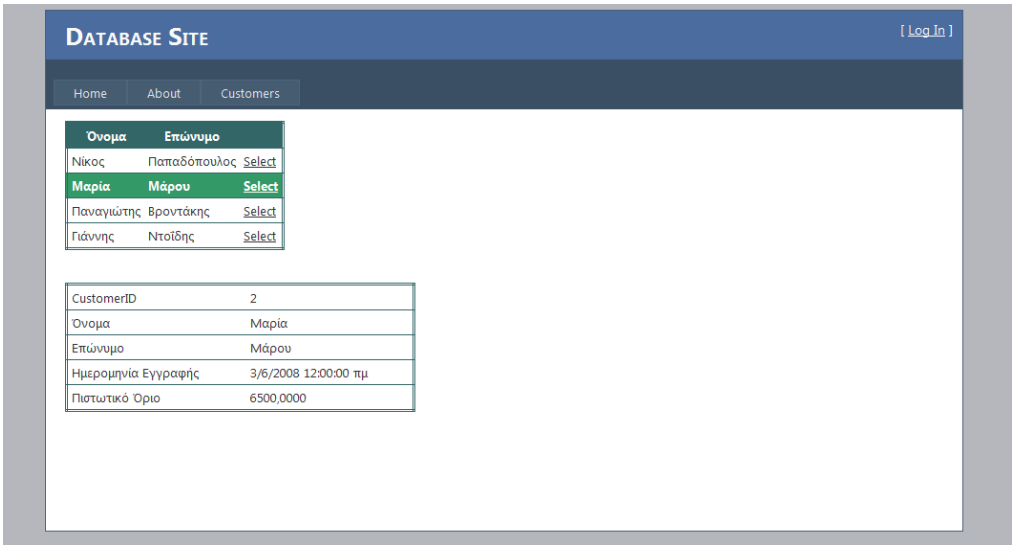
- Στην σελίδα Dbase.aspx
  - Εισάγουμε από το toolbox – Data το στοιχείο SqlDataSource. Το control αυτό δεν είναι ορατό. Ο σκοπός του είναι να αποκαταστήσει τη σύνδεση της Ιστοσελίδας με τη βάση δεδομένων. Κάνουμε κλικ στο Configure Data Source
    - επιλέγουμε τη βάση μας (επόμενο)
    - αποθηκεύουμε τη σύνδεση μας στο application configuration file όπως μας προτείνει (επόμενο)
    - επιλέγουμε την ανάκτηση όλων των στηλών του πίνακα (tick στο *) (επόμενο)
    - κάνουμε test query. Αφού δούμε τα δεδομένα μας πατάμε το Finish.
  - Τώρα θα βάλουμε ένα control που θα εμφανίζει τα περιεχόμενα των στοιχείων που φέρνει η προηγούμενη σύνδεση.
  - Εισάγουμε από το toolbox – Data το στοιχείο GridView. Σε αυτό επιλέγουμε τη σύνδεση που μόλις κάναμε ως Data Source (Choose data Source) η οποία έχει όνομα SqlDataSource1.
  - Εκτελούμε και βλέπουμε το αποτέλεσμα. Επιλέγουμε στο GridView μια μορφοποίηση της αρεσκείας μας (επιλογή Auto Format) και εκτέλεση ξανά.

- Δημιουργούμε ακόμη μια νέα σελίδα Customers.aspx και την προσθέτουμε στο menu.
  - Μπορούμε πρώτα να βάλουμε ένα control για την εμφάνιση των στοιχείων και στη συνέχεια να διαμορφώσουμε την πηγή δεδομένων (data source):
  - Στην σελίδα που δημιουργήσαμε προσθέτουμε το στοιχείο DetailsView. Είναι μηχανισμός που δείχνει μια γραμμή κάθε φορά από ένα σύνολο στοιχείων (πχ, από ένα πίνακα). Επιλέγουμε New Data Source στο Choose Data Source του DetailsView.
    - επιλέγουμε SQL Database (επόμενο)
    - επιλέγουμε το ConnectionString που έχουμε ήδη δημιουργήσει και αποθηκεύσει προηγουμένως (επόμενο)
    - επιλέγουμε την ανάκτηση όλων των στηλών του πίνακα (tick στο *) (επόμενο)
    - κάνουμε test query. Αφού δούμε τα δεδομένα μας πατάμε το Finish.
    - αλλάζουμε το μέγεθος του στοιχείου και τη μορφοποίησή του (Auto Format). Κάνουμε tick την επιλογή Enable Paging (είναι απαραίτητο σε αυτό το σημείο για το control DetailsView, για να μπορούμε να περιδιαβαίνουμε στις διάφορες γραμμές του πίνακα.
    - εκτελούμε

- Προσθέτουμε και ένα Gridview Control πάνω από το DetailsView
  - επιλέγουμε ως Data Source (Choose data Source) τη σύνδεσή μας με τη βάση (SqlDataSource1).
  - Το μορφοποιούμε και κάνουμε tick στην επιλογή enable Selection.
  - Εκτελούμε
  - Στο Gridview επιλέγουμε Edit Columns. Με την επιλογή Auto-generate fields αποεπιλεγμένη προσθέτουμε στα Selected fields μόνο το Όνομα και το Επώνυμο.
  - Ελέγχουμε ότι έχει επιλεγεί και η στήλη Select και την πηγαίνουμε τελευταία.
  - Αποεπιλέγουμε το Enable Paging από το DetailsView (θα κάνουμε την επιλογή μέσω του GridView)
- Επιλέγουμε το GridView, βρίσκουμε στο properties window (στα events) το selectedIndexChanged event και το κάνουμε διπλό click. Προσθέτουμε κώδικα στο event που δημιουργήθηκε ώστε η επιλογή στο GridView να μας πηγαίνει στην αντίστοιχη σελίδα του DetailsView.

```
protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
  {
    DetailsView1.PageIndex = GridView1.SelectedIndex;
  }
```

- Εκτελούμε



Εικόνα 5.13 *Προβολή δεδομένων από μια βάση*

2.6 Διαχείριση βάσης δεδομένων μέσω ιστοσελίδας

Εκφώνηση:
Δημιουργήστε έναν Ιστότοπο (με κεφαλίδα 'Διαχείριση Database Site') αξιοποιώντας την παρεχόμενη λειτουργικότητα των συστατικών που αυτόματα δημιουργεί το περιβάλλον ανάπτυξης VS 2012. Δημιουργήστε μια SQL Server βάση δεδομένων 'Companies.mdf' με έναν πίνακα εταιριών 'Company'. Ο πίνακας θα έχει 5 πεδία (δείτε την παρακάτω εικόνα). Εισάγετε σε αυτόν 9 γραμμές (εγγραφές) με στοιχεία, για να μπορέσετε στη συνέχεια όταν παρουσιάζετε τα δεδομένα σε μια Ιστοσελίδα να 'πειραματιστείτε' με τη σελιδοποίηση των δεδομένων (θέτοντας πχ. το page size = 4). Στη συνέχεια προσθέστε μια νέα σελίδα 'Companies.aspx', σύμφωνη με το υπάρχον πρότυπο, και συνδέστε τη με το μενού. Στη σελίδα αυτή να εμφανίσετε το περιεχόμενο του πίνακα 'Company' (στη κλασική μορφή πλέγματος στηλών και γραμμών), όμως επιπλέον να δώσετε στους χρήστες του Ιστότοπου αυξημένες δυνατότητες –

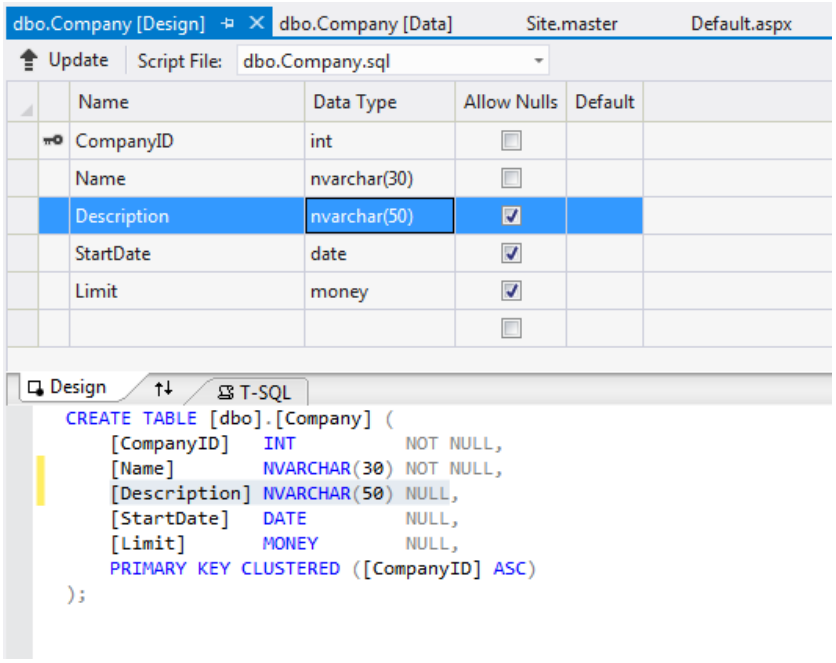
εισαγωγής/ενημέρωσης/διαγραφής των δεδομένων στον πίνακα αυτό. Ειδικότερα για την 'επικίνδυνη' ενέργεια της 'Διαγραφής', δημιουργήστε με κώδικα μια πιο ασφαλή έκδοση της εντολής 'Delete', αφού πριν την όποια διαγραφή, θα πρέπει να ζητείται επιβεβαίωση. Ειδικότερα για την υποστήριξη της λειτουργίας 'εισαγωγής', πρέπει να ξεφύγουμε από την παρουσίαση δεδομένων σε μορφή πλέγματος: πηγαίνετε στην υπάρχουσα σελίδα 'Contact.aspx', σβήστε το υπάρχον περιεχόμενο και μέσω ενός εργαλείου FormView δώστε δυνατότητες εισαγωγής/ενημέρωσης/διαγραφής των δεδομένων στον πίνακα Company (δείτε την τελευταία εικόνα της παραγράφου).

Υποδειγματική λύση:

- Δημιουργία νέου έτοιμου Ιστότοπου (ASP.NET Web Forms site) με όνομα "Lab6" και γλώσσα C#.
- Επεξεργασία του Master Page
  - Αλλαγή της κεφαλίδας σε "Διαχείριση Database Site"
- Δημιουργία και εμπλουτισμός ΒΔ
  - Επιλέγουμε το φάκελο App_Data (Application Data) του Ιστότοπου, κάνουμε δεξί κλικ, Add New Item, και προσθέτουμε SQL Server Database με το όνομα Companies.mdf.
  - Στο Server Explorer (View-> Server Explorer) αρχικά με δεξί κλικ πάνω στο Companies.mdf ζητάμε Refresh ώστε να μην υπάρχει πάνω του η ένδειξη του κόκκινου x (που δηλώνει μη σύνδεση)
  - Επιλέγουμε Tables, δεξί κλικ, και Add New Table.
  - Προσοχή: για να αποθηκευτεί ο πίνακας με το όνομα 'Company', πρέπει από αυτό το σημείο να πάμε στο κάτω παράθυρο (Design) και να αλλάξουμε την CREATE TABLE σε:

    CREATE TABLE [dbo].[Company]

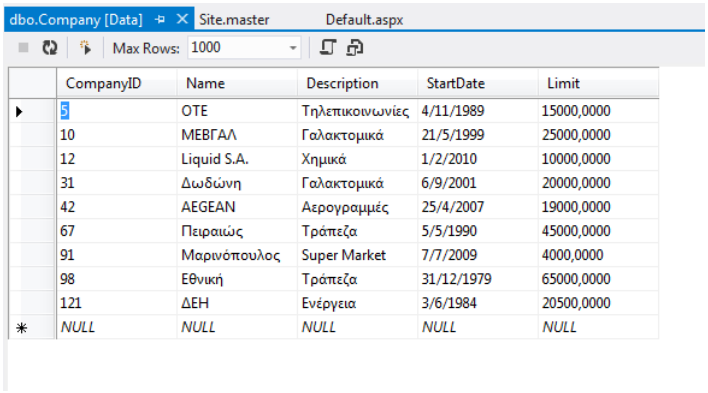
  - Εισάγουμε τα στοιχεία του πίνακα όπως φαίνονται στην παρακάτω εικόνα. Και ελέγχουμε ότι το CompanyID είναι primary key. Προσέξτε ότι προτιμούμε τον τύπο nvarchar για να έχουμε και υποστήριξη Ελληνικών χαρακτήρων στα δεδομένα.



Εικόνα 5.14 *Τροποποίηση πεδίων σε πίνακα*

  - Για την αποθήκευση, πατάμε Update και στη συνέχεια Update Database.

- Κάνουμε δεξί κλικ στον πίνακα Company από τον Server Explorer και επιλέγουμε Show Table Data. Γεμίζουμε τον πίνακα με δεδομένα. Πχ. με τις ακόλουθες 9 γραμμές.



CompanyID	Name	Description	StartDate	Limit
5	OTE	Τηλεπικοινωνίες	4/11/1989	15000,0000
10	ΜΕΒΓΑΛ	Γαλακτομικά	21/5/1999	25000,0000
12	Liquid S.A.	Χημικά	1/2/2010	10000,0000
31	Δωδώνη	Γαλακτομικά	6/9/2001	20000,0000
42	AEGEAN	Αερογραμμές	25/4/2007	19000,0000
67	Πειραιώς	Τράπεζα	5/5/1990	45000,0000
91	Μαρινόπουλος	Super Market	7/7/2009	4000,0000
98	Εθνική	Τράπεζα	31/12/1979	65000,0000
121	ΔΕΗ	Ενέργεια	3/6/1984	20500,0000
NULL	NULL	NULL	NULL	NULL

Εικόνα 5.15 *Εισαγωγή δεδομένων σε πίνακα*

- Εισαγωγή νέας σελίδας βασισμένη στο πρότυπο με το όνομα 'Companies.aspx'
- Προσθήκη της σελίδας 'Companies' στο menu του προτύπου, με κείμενο 'Εταιρίες' και εκτέλεση

- Στην σελίδα Companies.aspx
  - Εισάγουμε από το toolbox – Data το στοιχείο GridView. Επιλέγουμε New Data Source στο Choose data Source
    - επιλέγουμε SQL Database και διατηρούμε το προτεινόμενο ID για το control 'πηγής δεδομένων' (SqlDataSource1) και OK.
    - επιλέγουμε τη βάση μας Companies.mdf (επόμενο)
    - αποθηκεύουμε τη σύνδεση μας π.χ. με το όνομα «ConnectionStringComp» (επόμενο) – σημειώστε ότι η αποθήκευση θα γίνει στο application configuration file (Web.config)
    - επιλέγουμε την ανάκτηση των στηλών του πίνακα 'CompanyID', 'Name', 'Description' και
      (επειδή θέλουμε αυτή τη φορά να δώσουμε στους χρήστες του Ιστότοπου αυξημένες δυνατότητες – εισαγωγής/ενημέρωσης/διαγραφής των δεδομένων στη βάση)
    - Πατάμε το κουμπί Advanced και κάνουμε tick στις δυο επιλογές, Insert/update/delete – use opt. concurrency (επόμενο)
    - κάνουμε test query. Αφού δούμε τα δεδομένα μας πατάμε το Finish.
  - Στο GridView control κάνουμε tick to Enable Paging και επιλέγουμε μια αυτόματη μορφοποίηση (πχ. Simple). Εκτελούμε

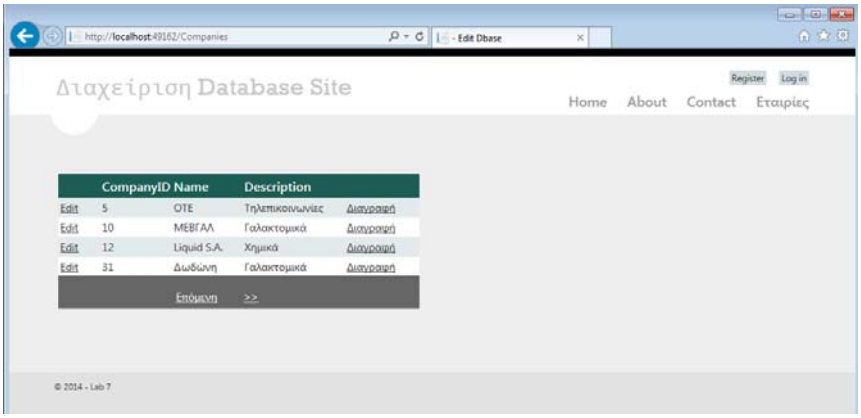
  - Στις ιδιότητες του GridView control, στην κατηγορία Paging
    - αλλάζουμε το Pagesize σε 4
    - αλλάζουμε το NextPageText (PagerSettigs) σε Επόμενη και το PreviusPageText σε Προηγούμενη
    - αλλάζουμε το mode σε NextPrevious
    - εκτελούμε και βλέπουμε τις αλλαγές
    - αλλάζουμε το mode σε NextPreviousFirstLast
    - εκτελούμε και βλέπουμε τις αλλαγές
  - Στο GridView control
    - κάνουμε tick to Enable Sorting και εκτελούμε και βλέπουμε τις αλλαγές

- Δοκιμάζουμε και ταξινομούμε τις εγγραφές (γραμμές) ανάλογα με τα διάφορα πεδία του πίνακα, πατώντας στους τίτλους των πεδίων (κατά αύξουσα και κατά φθίνουσα σειρά)
- κάνουμε Edit Columns
  - Τοποθετούμε το Description πάνω από το Name
  - Αλλάζουμε το HeaderText του Description σε 'Περιγραφή' και του Name σε 'Επωνυμία'
- κάνουμε tick στο Enable Editing. Εκτελούμε και κάνουμε αλλαγές στη Βάση μας
- κάνουμε tick στο Enable Deleting. Εκτελούμε και κάνουμε μια διαγραφή στη Βάση μας
- Παρατηρήστε τον κώδικα του GridView1
- Βάλτε τον ακόλουθο κώδικα ακριβώς πριν το </Columns> του GridView1:

```
<asp:TemplateField>
    <ItemTemplate>
                <asp:LinkButton ID="LinkButton1" Runat="server"
                OnClientClick="return confirm('Είσαι σίγουρος;');"
        CommandName="Delete">Διαγραφή</asp:LinkButton>
    </ItemTemplate>
</asp:TemplateField>
</Columns>
```

Δημιουργήσαμε μια πιο ασφαλή έκδοση για την εντολή Delete, αφού πριν την όποια διαγραφή, μας ζητείται να την επιβεβαιώσουμε.

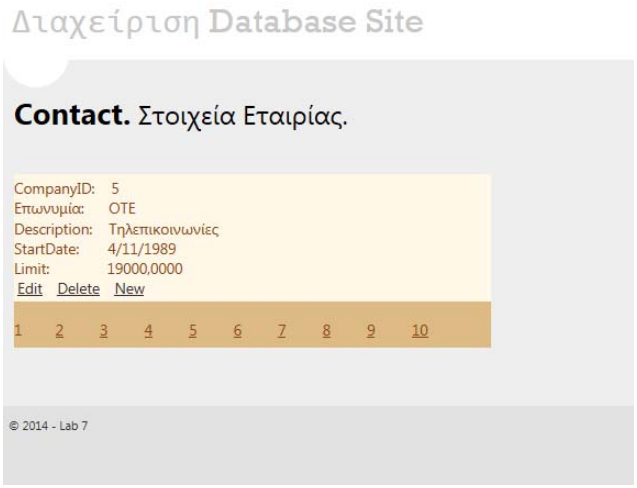
Μπορούμε τώρα να απενεργοποιήσουμε το εξ' ορισμού link του Delete, βάζοντας False στην αντίστοιχη ιδιότητα: ShowDeleteButton="False"



Εικόνα 5.16 *Ασφαλής 'Διαγραφή' στη βάση δεδομένων*

- Πηγαίνουμε στην υπάρχουσα σελίδα 'Contact.aspx' και αλλάζουμε το δεύτερο μέρος της επικεφαλίδας του από 'Your contact page' σε 'Στοιχεία Εταιρίας'.
- Προσθέτουμε το στοιχείο FormView, από το toolbox-Data. Επιλέγουμε New Data Source στο Choose data Source του FormView.
  - επιλέγουμε SQL Database (επόμενο)
  - επιλέγουμε το ConnectionString που έχουμε ήδη δημιουργήσει και σώσει προηγουμένως (επόμενο)
  - επιλέγουμε την ανάκτηση όλων των στηλών του πίνακα (tick στο *) (επόμενο)
  - πατάμε το κουμπί Advanced και κάνουμε tick στις δυο επιλογές (επόμενο)
  - κάνουμε test query. Αφού δούμε τα δεδομένα μας πατάμε το Finish.
  - Κάνουμε tick στο Enable Paging, αλλάζουμε τη μορφοποίησή του (Auto Format) και εκτέλεση

- Μπορούμε εκτός από Edit και Delete να εισάγουμε και νέα δεδομένα. Κάντε εισαγωγή μιας νέας εταιρίας.
- Επιλέγουμε Edit Templates και βλέπουμε πως μπορούμε να αλλάξουμε την εμφάνιση του FormView. Π.χ., επιλέγουμε ItemTemplate και αλλάζουμε το Name σε Επωνυμία και στο StartDateLabel επιλέγουμε Edit DataBindings και Format → Short date
- Επιλέγουμε EndTemplateEditing για να κλείσουμε τη διόρθωση των templates
- Εκτελούμε



Εικόνα 5.17 *Εισαγωγή ('New') εγγραφής μέσω Ιστοσελίδας*

2.7. Ηλεκτρονικό κατάστημα με ASP.NET web forms

Εκφώνηση:
Στην άσκηση αυτή δημιουργούμε έναν Ιστότοπο (με κεφαλίδα 'Ηλεκτρονικό Κατάστημα') που ξεκινώντας από την παρεχόμενη λειτουργικότητα των συστατικών που αυτόματα δημιουργεί το περιβάλλον ανάπτυξης VS 2012, προχωρούμε στην υποστήριξη βασικών λειτουργιών ηλεκτρονικού καταστήματος με ένα στοιχειώδες καλάθι αγορών. Υπάρχουν αρκετά στοιχεία προηγούμενων ασκήσεων που ενσωματώνονται εδώ, όπως και αρκετά νέα.

Sound 5.3.mp3	Ηχητικό (audio)
Εισαγωγή δεδομένων σε φόρμα	

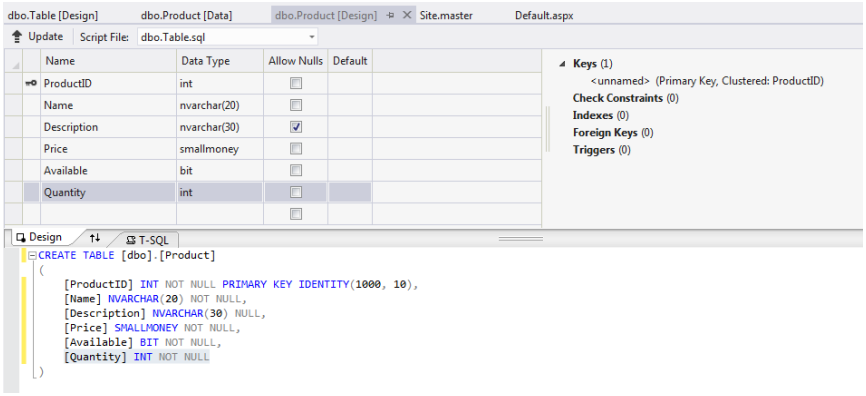
Sound 5.4.mp3	Ηχητικό (audio)
Παράδειγμα χρησιμοποίησης φόρμας	

Υποδειγματική λύση:

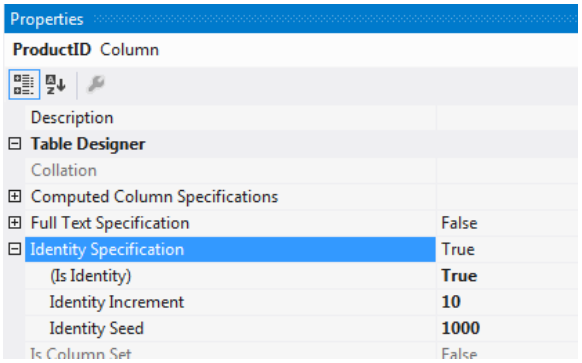
- Δημιουργία νέου έτοιμου Ιστότοπου (ASP.NET Web Forms site) με όνομα "Lab7" και γλώσσα C#.
- Επεξεργασία του Master Page
  - Αλλαγή της επικεφαλίδας σε "Ηλεκτρονικό Κατάστημα "
- Δημιουργία και εμπλουτισμός ΒΔ:
  - Επιλέγουμε το φάκελο App_Data (Application Data) του Ιστότοπου, κάνουμε δεξί κλικ, Add New Item, και προσθέτουμε SQL Server Database με το όνομα Products.mdf.
  - Στο Server/Database Explorer (είναι μαζί με τον solution explorer σε tabs) επιλέγουμε Tables, δεξί κλικ, και Add New Table.
  - Προσοχή: για να αποθηκευτεί ο πίνακας με το όνομα Product, πρέπει από αυτό το σημείο να πάμε στο κάτω παράθυρο (Design) και να αλλάξουμε την CREATE TABLE σε:

CREATE TABLE [dbo].[Product]

- Εισάγουμε τα στοιχεία του πίνακα όπως φαίνονται στην παρακάτω εικόνα. Ελέγχουμε ότι το ProductID είναι primary key. Επίσης, στις ιδιότητες του ProductID, στο πεδίο Identity Specification επιλεγούμε True στο Is Identity, και βάζουμε Identity Seed: 1000 (Αρχική τιμή) και Identity Increment: 10 (βήμα αύξησης).



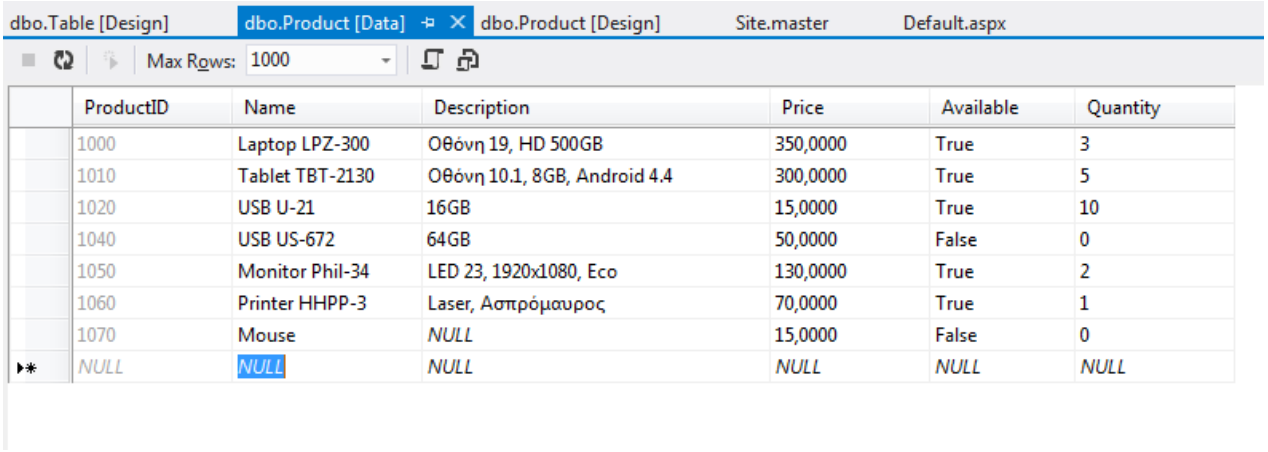
Εικόνα 5.18 *Εισαγωγή κλειδιού σε πίνακα της βάσης δεδομένων*



Εικόνα 5.19 *Εισαγωγή ιδιοτήτων κλειδιού σε πίνακα της βάσης δεδομένων*

- Για τη δημιουργία του πίνακα, πατάμε Update και στη συνέχεια Update Database.

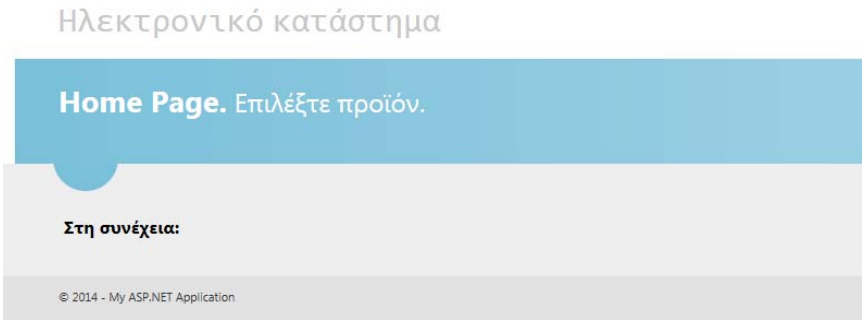
- Κάνουμε δεξί κλικ στον πίνακα Product από τον Server Explorer και επιλέγουμε Show Table Data. Γεμίζουμε τον πίνακα με δεδομένα. Πχ. με τις ακόλουθες 7 γραμμές.



ProductID	Name	Description	Price	Available	Quantity
1000	Laptop LPZ-300	Οθόνη 19, HD 500GB	350,0000	True	3
1010	Tablet TBT-2130	Οθόνη 10.1, 8GB, Android 4.4	300,0000	True	5
1020	USB U-21	16GB	15,0000	True	10
1040	USB US-672	64GB	50,0000	False	0
1050	Monitor Phil-34	LED 23, 1920x1080, Eco	130,0000	True	2
1060	Printer HHPP-3	Laser, Ασπρόμαυρος	70,0000	True	1
1070	Mouse	*NULL*	15,0000	False	0
NULL	*NULL*	*NULL*	*NULL*	*NULL*	*NULL*

Εικόνα 5.20 *Εισαγωγή δεδομένων σε πίνακα*

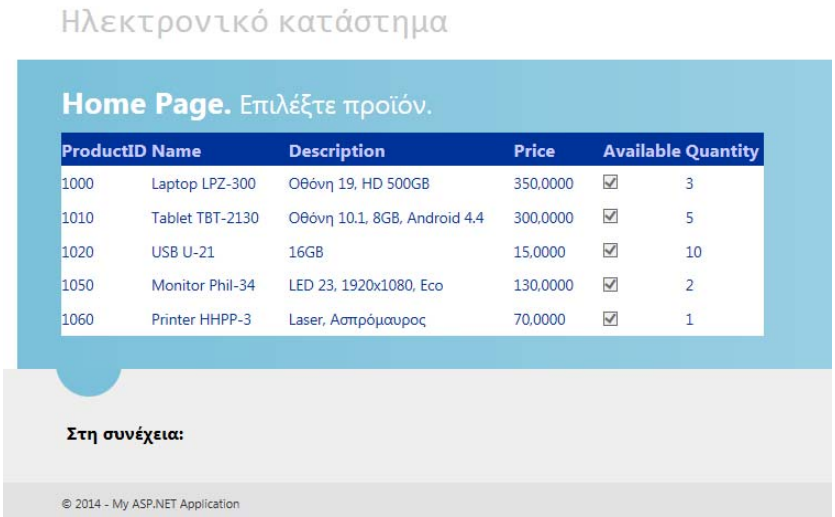
- Στην σελίδα Default.aspx
  - Αλλάζουμε τις επικεφαλίδες h2 και h3, και σβήνουμε το υπόλοιπο περιεχόμενο (από το Design view) ώστε να έχουμε την παρακάτω εμφάνιση:



Εικόνα 5.21 *Προβολή αρχικής σελίδας του ηλεκτρονικού καταστήματος*

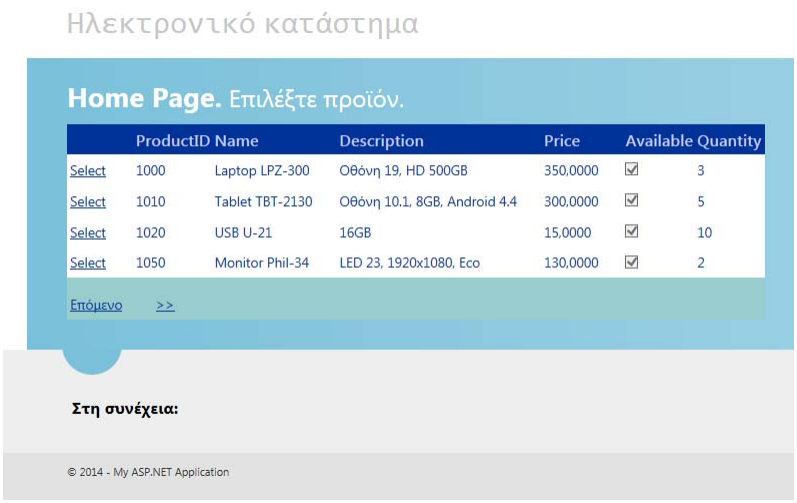
  - Εισάγουμε από το toolbox – Data το στοιχείο GridView. Επιλέγουμε New data source στο Choose Data Source
    - επιλέγουμε SQL Database (επόμενο)
    - επιλέγουμε τη βάση μας (επόμενο)
    - αποθηκεύουμε τη σύνδεση μας στο application configuration file όπως μας προτείνει, με το όνομα ConnStringProd (επόμενο)
    - επιλέγουμε την ανάκτηση όλων των στηλών του πίνακα (tick στο *)
    - Πατάμε το κουμπί WHERE και επιλέγουμε
      - Column: Available
      - Operator: =
      - Source: None
      - Value: True
      - Πατάμε το κουμπί Add και το OK
        Βλέπουμε ότι σχηματίστηκε το παρακάτω query:

        SELECT * FROM [Product] WHERE ([Available] = @Available)
    - Πατάμε το επόμενο και κάνουμε test query. Αφού δούμε τα δεδομένα μας (μόνο όσα είναι AVAILABLE, δηλαδή 5 από τις 7 γραμμές), πατάμε το Finish.
  - Στο GridView control κάνουμε tick to Enable Paging και επιλέγουμε μια μορφοποίηση. Εκτελούμε



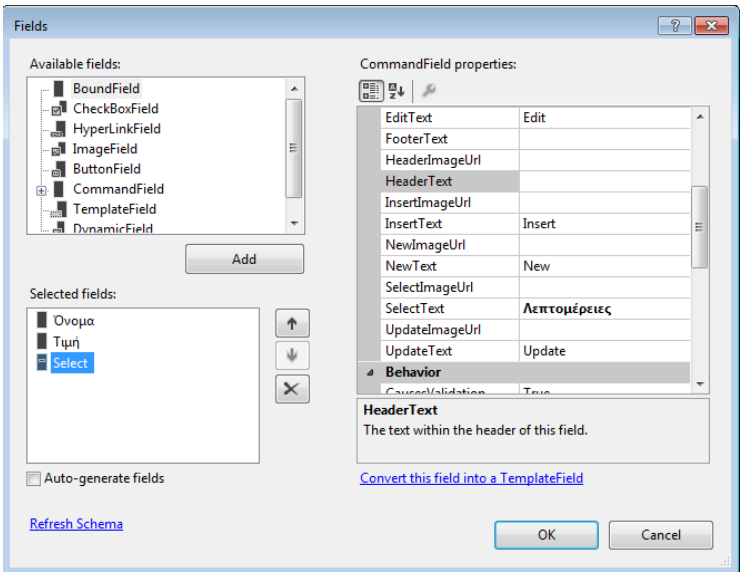
Εικόνα 5.22 *Προβολή προϊόντων*

- Στις ιδιότητες του GridView control, στην κατηγορία Paging
  - αλλάζουμε το Pagesize σε 4
  - αλλάζουμε το mode σε NextPreviousFirstLast
  - αλλάζουμε το NextPageText (PagerSettigs) σε 'Επόμενο' και το PreviusPageText σε 'Προηγούμενο'
  - εκτελούμε και βλέπουμε τις αλλαγές
- Στο GridView control
  - κάνουμε tick to Enable Sorting και Enable Selection. Επιλέγουμε μια μορφοποίηση, εκτελούμε και βλέπουμε τις αλλαγές



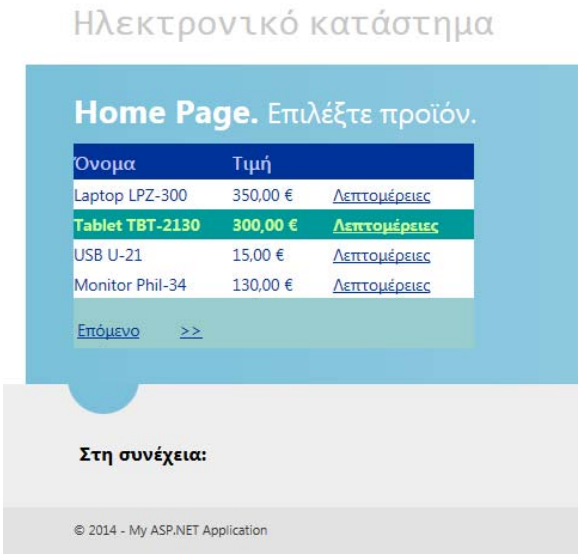
Εικόνα 5.23 *Προβολή προϊόντων με δυνατότητα επιλογής τους*

- κάνουμε Edit Columns
  - Αφαιρούμε τα πεδία ProductID , Description, Available και Quantity
  - Αλλάζουμε στα πεδία που έμειναν τα HeaderText στα Ελληνικά (Όνομα, Τιμή)
  - Δίνουμε στο πεδίο Price το {0:c} ως DataFormatString
  - Στο Select Δίνουμε ως SelectText το 'Λεπτομέρειες' και μετακινούμε το πεδίο τελευταίο.



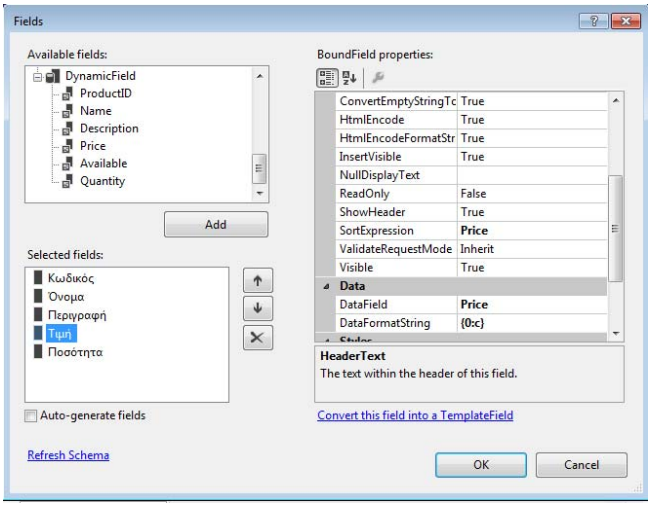
Εικόνα 5.24 *Αλλαγή ιδιοτήτων προβολής*

- Εκτελούμε και βλέπουμε το αποτέλεσμα



Εικόνα 5.25 *Προσαρμοσμένη προβολή προϊόντων*

- Αλλάζουμε (από το Design view) το h3-μορφοποίησης μήνυμα 'Στη συνέχεια:' με το 'Ζητήσατε λεπτομέρειες για το προϊόν:'
- Θέλουμε σε έναν μηχανισμό DetailsView, να βλέπουμε αναλυτικά το προϊόν που επιλέγουμε στο GridView.
- Προσθέτουμε ένα στοιχείο DetailsView από κάτω. Επειδή θα κάνουμε διαφορετική επιλογή πεδίων σε αυτόν τον μηχανισμό, καλύτερα στο Choose Data Source του DetailsView να επιλέξουμε New data source, και στα επόμενα βήματα να αξιοποιήσουμε το υπάρχον ConnStringProd. Σημειώστε ότι χρειάζεται και εδώ να βάλουμε τον ίδιο περιορισμό με το WHERE που βάλαμε και στο GridView. Δηλαδή στο Test Query πρέπει να φαίνονται τα 5 διαθέσιμα προϊόντα.
  - αλλάζουμε το μέγεθος του στοιχείου και τη μορφοποίησή του (Auto Format)
  - κάνουμε click το Edit Fields
  - κρατούμε όλα τα πεδία εκτός από το Available και αλλάζουμε τα HeaderText στα Ελληνικά (Κωδικός, Όνομα, Περιγραφή, Τιμή, Ποσότητα)
  - δίνουμε στο πεδίο Price το {0:c} ως DataFormatString

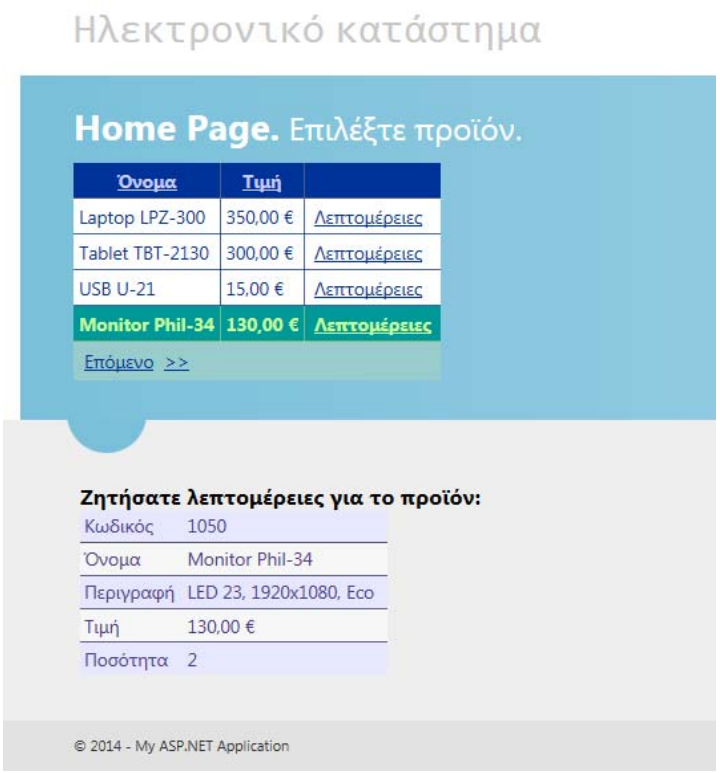


Εικόνα 5.26 *Αλλαγή ιδιοτήτων προβολής*

- Συνδέουμε τα δύο control μεταξύ τους, ώστε όταν πατάμε ‘Λεπτομέρειες’ στο GridView να βλέπουμε το προϊόν αναλυτικά στο DetailsView

```
protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
    {
      DetailsView1.PageIndex = GridView1.SelectedIndex + (GridView1.PageIndex * GridView1.PageSize);
    }
```

- Εκτελούμε και βλέπουμε το αποτέλεσμα



Εικόνα 5.27 *Προβολή σελίδας με δύο συνδεδεμένους μηχανισμούς*

- Στην συνέχεια θέτουμε την ιδιότητα Visible του DetailsView ως false και προσθέτουμε μια επιπλέον εντολή στο SelectedIndexChanged event του Gridview ως εξής

```
protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
    {
      DetailsView1.PageIndex = GridView1.SelectedIndex + (GridView1.PageIndex * GridView1.PageSize);
      DetailsView1.Visible = true;
    }
```

- Εκτελούμε και βλέπουμε το αποτέλεσμα
- Προσθέτουμε ένα κουμπί κάτω από το DetailsView με Text Απόκρυψη και θέτουμε την ιδιότητα Visible ως false. Ως Button Click event δίνουμε το παρακάτω

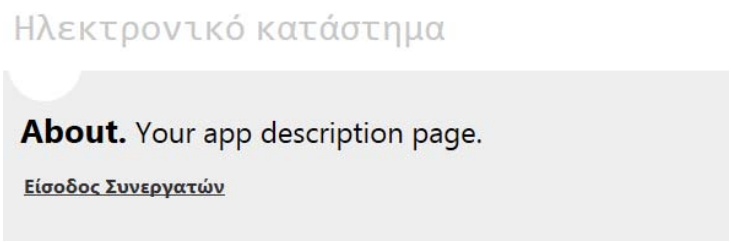
```
protected void Button1_Click(object sender, EventArgs e)
    {
      DetailsView1.Visible = false;
      Button1.Visible = false;
    }
```

- Ενημερώνουμε και πάλι το SelectedIndexChanged event του GridView με μια επιπλέον εντολή ως εξής

```
protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
    {
        DetailsView1.PageIndex = GridView1.SelectedIndex + (GridView1.PageIndex * GridView1.PageSize);
        DetailsView1.Visible = true;
        Button1.Visible = true;
}
```

- Εκτελούμε και βλέπουμε το αποτέλεσμα

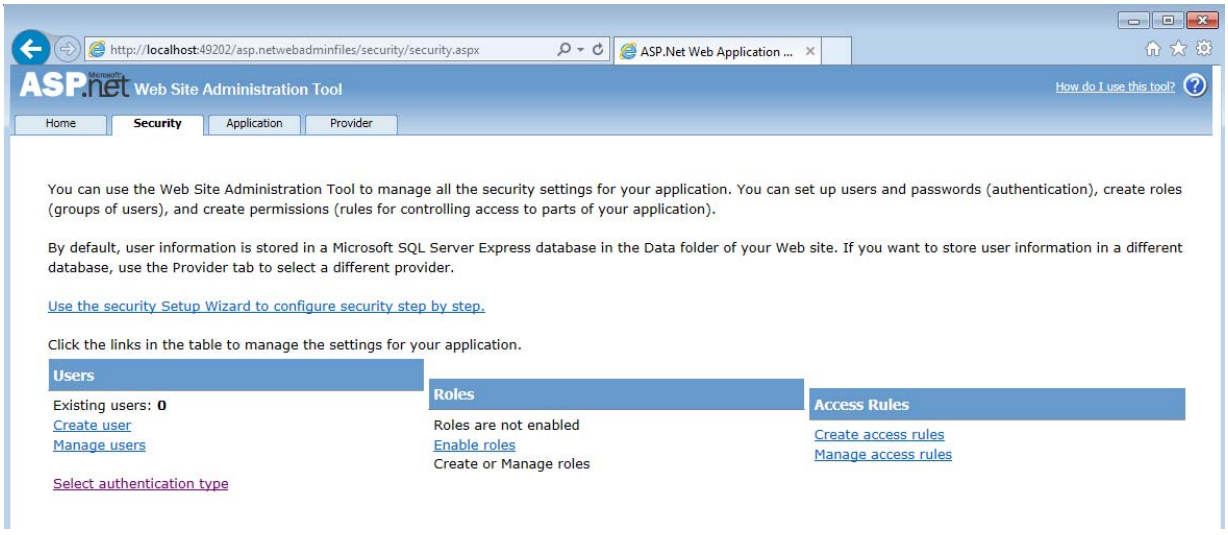
- Δημιουργούμε έναν νέο φάκελο με το όνομα Administration και μια νέα σελίδα μέσα σε αυτόν τον φάκελο (φροντίζοντας κατά τη δημιουργία να επιλέξουμε να ακολουθεί την Master Page) με το όνομα AdminSelida.aspx. Ως περιεχόμενό της βάλτε το κείμενο «Σελίδα Διαχείρισης».
- Προσθέτουμε ένα hyperlink σε αυτήν τη σελίδα από τη σελίδα About.aspx, με text '**Είσοδος Συνεργατών'**



Εικόνα 5.28 *Σχετικά με το κατάστημα (About Page)*

Δημιουργία Ρόλων και Χρηστών (security issues)

- Στο ASP.NET Configuration (στην γραμμή εργαλείων επιλέγουμε Website -> ASP.NET Configuration
  - Επιλέγουμε την ταμπέλα Security και κάνουμε click στο Select authentication type. Φροντίζουμε να είναι επιλεγμένη η επιλογή From the internet και πατάμε το Done
  - Είμαστε πίσω στην ταμπέλα Security και κάνουμε click στο Enable Roles

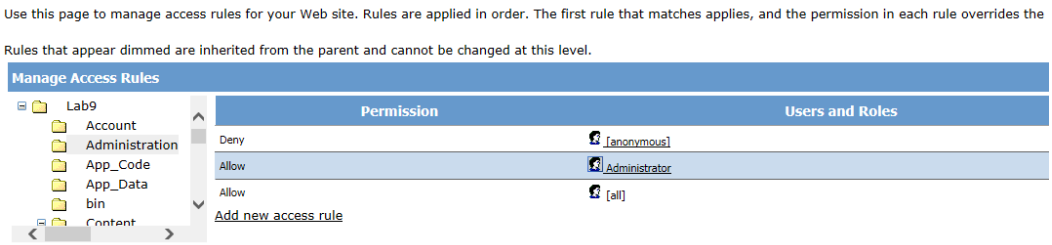


Εικόνα 5.29 *Σελίδα διαχείρισης (ASP.NET Configuration)*

- Κάνουμε επίσης click και στο Create or Manage roles.
  - Στο textbox που εμφανίζεται γράφουμε Customer και πατάμε το Add Role
  - Δημιουργούμε με τον ίδιο τρόπο και τον ρόλο Administrator
- Πηγαίνουμε πίσω με το κουμπί back και κάνουμε click στο Create User
  - Φτιάχνουμε δύο χρήστες User1 και User2 με ρόλο Customer
  - Φτιάχνουμε έναν χρήστη Admin με ρόλο Administrator

  (ως password βάλτε User1User1, User2User2 και AdminAdmin αντίστοιχα)

- Θέλουμε να εφαρμόσουμε κανόνες ελέγχου προσπέλασης επιπέδου σύμφωνα με τους οποίους:
  - όλοι οι ανώνυμοι χρήστες (όσοι δεν έχουν κάνει login) ΔΕΝ θα έχουν πρόσβαση σε ολόκληρο το φάκελο Administration (οπότε θα ανακατευθύνονται στη φόρμα σύνδεσης Login.aspx για να συνδεθούν ως χρήστες και να ελεγχθεί έτσι το επίπεδο δικαιωμάτων που έχουν) και
  - μόνο οι χρήστες με ρόλο Administrator θα μπορούν να βλέπουν τη σελίδα διαχείρισης AdminSelida. Όλοι οι χρήστες με άλλους ρόλους θα ανακατευθύνονται σε μια σελίδα OxiProsvasi.aspx που θα περιέχει κατάλληλο μήνυμα.
- Στο ASP.NET Configuration (στην γραμμή εργαλείων επιλέγουμε Website -> ASP.NET Configuration
  - Επιλέγουμε την ταμπέλα Security και κάνουμε click στο Create access rules.
    - Κάνουμε click στον φάκελο Administration και επιλέγουμε Anonymous users και deny
    - Κάνουμε click στον Manage Access Rules και στο Add new access rule και με τον ίδιο τρόπο (πάλι για τον φάκελο Administration)
      - Administrator, Allow



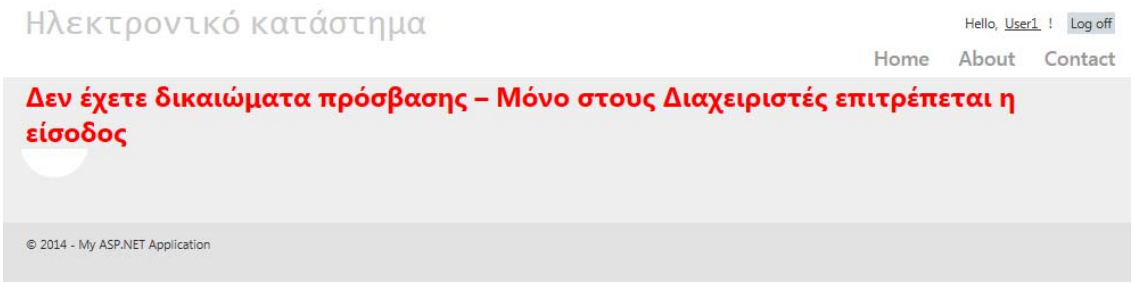
Εικόνα 5.30 *Δικαιώματα πρόσβασης (access rules)*

- δημιουργούμε μια νέα σελίδα OxiProsvasi.aspx στον γενικό φάκελο. Γράφουμε στην καινούργια σελίδα το κόκκινο h2-κείμενο "Δεν έχετε δικαιώματα πρόσβασης – Μόνο στους Διαχειριστές επιτρέπεται η είσοδος"

- Στην κώδικα της σελίδας AdminSelida.aspx προσθέτουμε στο event Page Load τον παρακάτω κώδικα:

```
protected void Page_Load(object sender, EventArgs e)
  {
    if (!User.IsInRole("Administrator"))
      Server.Transfer("../OxiProsvasi.aspx");
  }
```

- Βάλτε ως αρχική σελίδα (Set As Start Page) τη σελίδα Default.aspx
- Εκτελούμε και βλέπουμε πως υλοποιούνται οι κανόνες που θέσαμε με τις αλλαγές που κάναμε. Δοκιμάστε είσοδο ως μη-συνδεδεμένος χρήστης, ως user1 και ως Admin.



Εικόνα 5.31 *Μήνυμα προς μη εξουσιοδοτημένους χρήστες*

Β' μέρος (στοιχειώδες καλάθι αγορών):

- Δημιουργούμε έναν νέο φάκελο με το όνομα Registered και μια νέα σελίδα μέσα σε αυτόν τον φάκελο (φροντίζοντας κατά τη δημιουργία να επιλέξουμε να ακολουθεί την MasterPage) με το όνομα Cart.aspx

- Στο ASP.NET Configuration (στην γραμμή εργαλείων επιλέγουμε Website ->ASP.NET Configuration
    - Επιλέγουμε την ταμπέλα Security και κάνουμε click στο Create access rules.
        - Κάνουμε click στον φάκελο Registered και επιλέγουμε Anonymous users και Deny
        - Κάνουμε click στο Manage Access Rules και στο Add new access rule και με τον ίδιο τρόπο
            - Administrator, Allow
            - Customer, Allow
        - Προσθέτουμε τη σελίδα Cart στο menu του προτύπου (Καλάθι Αγορών) και εκτελούμε και ελέγχουμε αν οι κανόνες που θέσαμε λειτουργούν σωστά
- Στην σελίδα Default.aspx
    - Στα Tasks του GridView control επιλέγουμε Add New Column.
        - Επιλέγουμε TemplateField ως Field Type
        - Στο Header Text γράφουμε No. (δηλαδή Νούμερο) και πατάμε OK.
        - Μεταφέρουμε την καινούργια στήλη τέρμα αριστερά (από το Edit Columns)
        - Στον κώδικα του GridView control βρίσκουμε την ετικέτα <Columns> και μέσα σε αυτή εντοπίζουμε την ετικέτα

```
<asp:TemplateField HeaderText="No."></asp:TemplateField>
```

        - Είναι η ετικέτα που μόλις δημιουργήσαμε. Στο περιεχόμενο της ετικέτας αυτής και μέσα σε ετικέτα <ItemTemplate>, γράφουμε τον κώδικα που θα μετράει τον αριθμό των προϊόντων που εμφανίζονται:

```
<asp:TemplateFieldHeaderText="No.">
<ItemTemplate><%#Container.DataItemIndex + 1 %></ItemTemplate>
</asp:TemplateField>
```

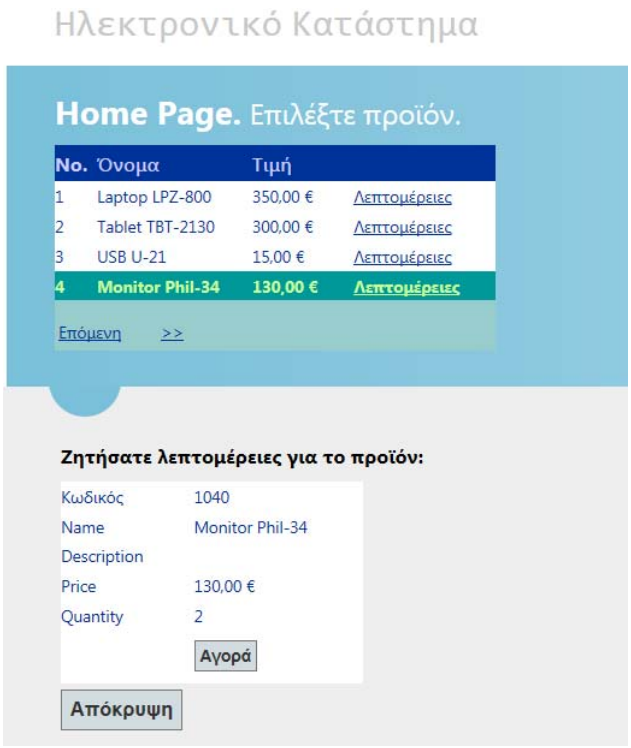
        - Εκτελούμε και βλέπουμε το αποτέλεσμα

    - Στα Tasks του DetailsView control επιλέγουμε Add New Field.
        - Επιλέγουμε TemplateField ως Field Type
        - Πατάμε OK (αφήνοντας κενό το HeaderText, γιατί στο νέο πεδίο δε θέλουμε κάποια περιγραφή να εμφανίζεται).
        - Στα Tasks του DetailsView επιλέγουμε Edit Templates.
        - Προσθέτουμε ένα Button στο ItemTemplate με Text "Αγορά"
        - Κάνουμε διπλό click στο κουμπί για να οδηγηθούμε στο Button2_Click event.

- Σε αυτό γράφουμε τον παρακάτω κώδικα

```
Protected void Button2_Click(object sender, EventArgs e)
  {
   Session["ProductID"] = DetailsView1.DataKey.Value;
   Response.Redirect("~/Registered/Cart.aspx");
   }
```

- Επιλέγουμε End Template Editing στο DetailsView
- Εκτελούμε και βλέπουμε το αποτέλεσμα



Εικόνα 5.32 *Στοιχειώδες καλάθι αγορών*

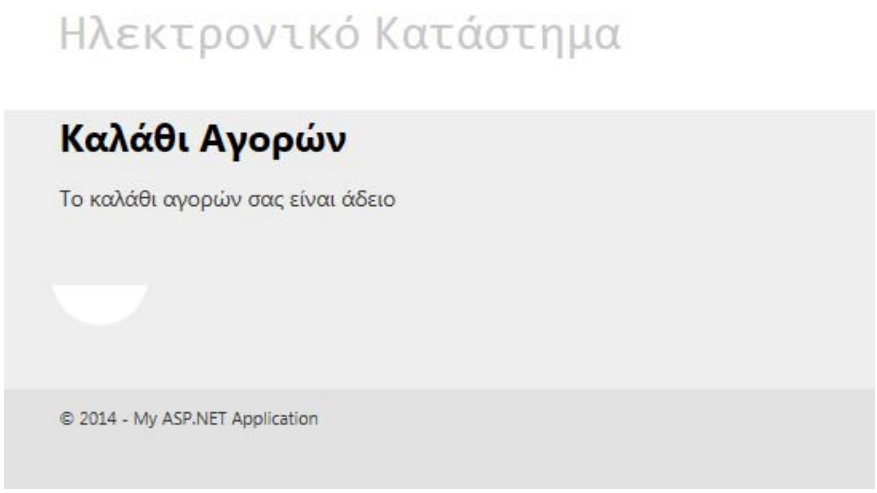
- Στην σελίδα Cart.aspx
  - Προσθέτουμε στο δεύτερο τμήμα της (Main content) μια επικεφαλίδα Καλάθι Αγορών (h2), και
  - Προσθέτουμε ένα Label με κενό text
  - Προσοχή: αν θελήσουμε να τα βάλουμε στο πρώτο τμήμα της (featured content), τότε για να έχουμε σε εσοχή την εμφάνιση όσων θέλουμε, θα πρέπει στον κώδικα που παράχθηκε, να περικλείσουμε με κατάλληλο τα <h2> και <asp:Label>, για να έχουμε:

```
<div class="content-wrapper">
   <h2>Καλάθι Αγορών</h2>
   <p>
      <asp:Label ID="Label1" runat="server"></asp:Label>
   </p>
</div>
```

  - Στο Page_Load event της σελίδας προσθέτουμε τον παρακάτω κώδικα

```
Protected void Page_Load(object sender, EventArgs e)
   {
    if (Session["ProductID"] == null)
    {
```

```
            Label1.Visible = true;
            Label1.Text = "Το καλάθι αγορών σας είναι άδειο";
          }
        else
          Label1.Visible = false;
    }
```



Εικόνα 5.33 *Άδειο καλάθι αγορών*

- Στην σελίδα Cart.aspx εισάγουμε ένα DetailsView. Στα Tasks επιλέγουμε New Data Source στο Choose Data Source
  - επιλέγουμε SQL Database (επόμενο)
  - επιλέγουμε το connectionString που έχουμε ήδη δημιουργήσει για τον Ιστότοπό μας (επόμενο)
  - επιλέγουμε την ανάκτηση όλων των στηλών του πίνακα (tick στο *)
  - Πατάμε το κουμπί WHERE και επιλέγουμε
    - Column: ProductID
    - Operator: =
    - Source: Session
    - Session Field: ProductID
    - Πατάμε το κουμπί Add και το OK
    - Βλέπουμε ότι σχηματίστηκε το παρακάτω query:

    SELECT * FROM [Product] WHERE ([ProductID] = @ProductID)
  - Πατάμε το επόμενο και κάνουμε Test Query (Δίνουμε μια τιμή στο ProductID για το test, πχ. 1020). Αφού δούμε τα δεδομένα μας πατάμε το Finish.
  - Ο κώδικας που σχηματίστηκε και είναι υπεύθυνος για την ερώτηση στη βάση με κριτήριο μεταβλητή Session είναι ο παρακάτω.

```
<asp:SqlDataSourceID="SqlDataSource1"runat="server"
  ConnectionString="<%$ConnectionStrings:ConnectionString%>"
  SelectCommand="SELECT * FROM [Product] WHERE ([ProductID] =  @ProductID)">
    <SelectParameters>
      <asp:SessionParameterName = "ProductID" SessionField =
        "ProductID" Type="Int32"/>
  </SelectParameters>
</asp:SqlDataSource>
```

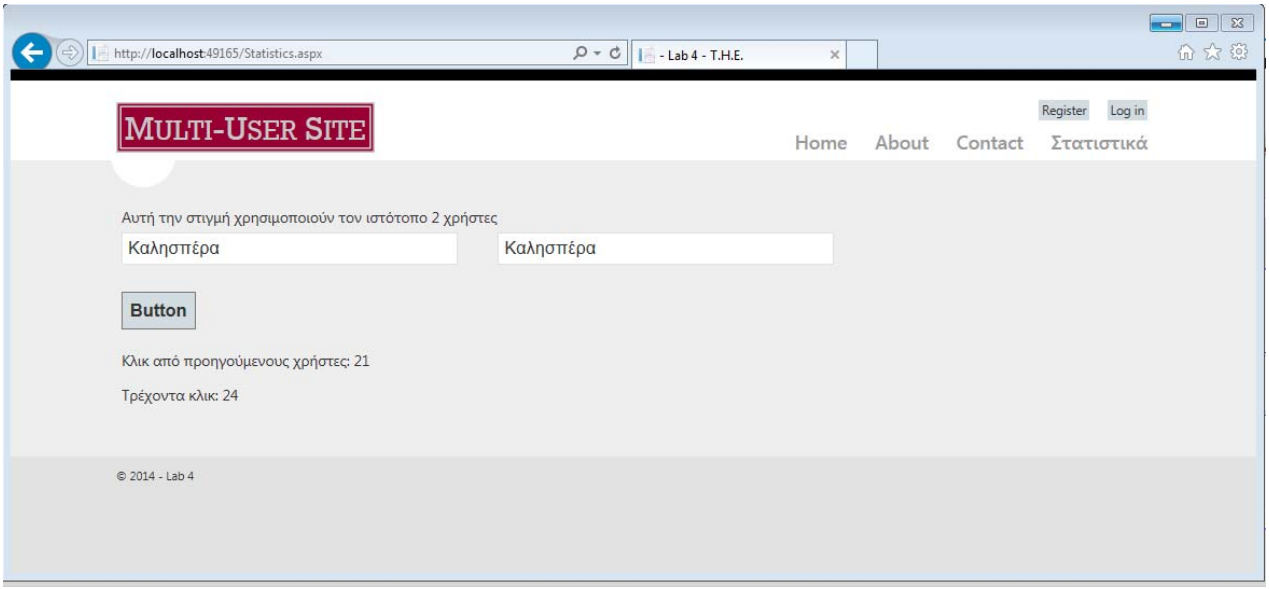
  - Στα Tasks του DetailsView επιλεγούμε μια μορφοποίηση (AutoFormat)

- Επιλέγουμε Edit Fields
  - Αφαιρούμε τα πεδία ProductID, Quantity και Available και αλλάζουμε τα HeaderText στα Ελληνικά (Όνομα, Περιγραφή, Τιμή)
  - Δίνουμε στο Πεδίο Price το {0:c} ως Data Format String
- Μεγαλώνουμε κατάλληλα το DetailsView στο μήκος και εκτελούμε, κάνουμε δοκιμές και βλέπουμε το αποτέλεσμα

3. Ασκήσεις Αυτοαξιολόγησης

3.1. Μετρήσεις συμβάντων

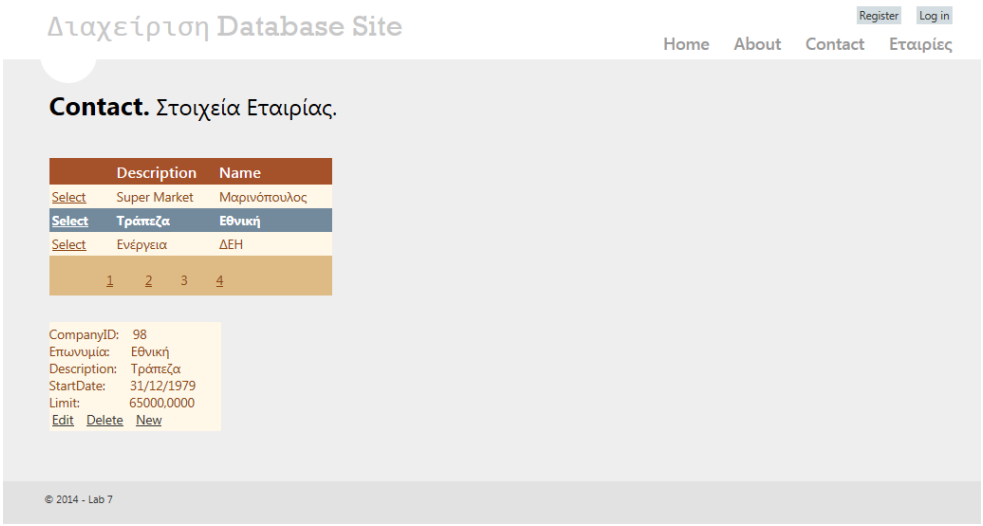
Εκφώνηση:
Στην σελίδα 'Statistics' του Ιστότοπου της άσκησης 2.3, προσθέστε δυο Label κάτω από το κουμπί (όπως στην παρακάτω εικόνα), με ID: Label2 και Label3. Γράψτε τον κατάλληλο κώδικα ώστε το Label2 να περιέχει μήνυμα με τον αριθμό των clicks που είχαν κάνει οι προηγούμενοι χρήστες πριν προσπελάσει τον Ιστότοπο ο τρέχων χρήστης, και στο Label3 μήνυμα με το τρέχον πλήθος των clicks όλων των χρηστών (προηγούμενων + συνδεδεμένων αυτή τη στιγμή).



Εικόνα 5.34 *Μετρήσεις συμβάντων*

3.2. Σύνδεση μηχανισμών

Εκφώνηση:
Στην σελίδα 'Contact' του Ιστότοπου της άσκησης 2.6 προσθέστε ένα GridView control, το όποιο θα δείχνει μόνο τα πεδία 'Description' και 'Name' και θα έχει επιπλέον δυνατότητα επιλογής γραμμής (Select), καθώς και δυνατότητα Sorting και Paging με PageSize 3.. Η επιλογή κάποιας Εταιρίας στο GridView θα προκαλεί την εμφάνιση του στο FormView (το οποίο όμως δε θα έχει paging).

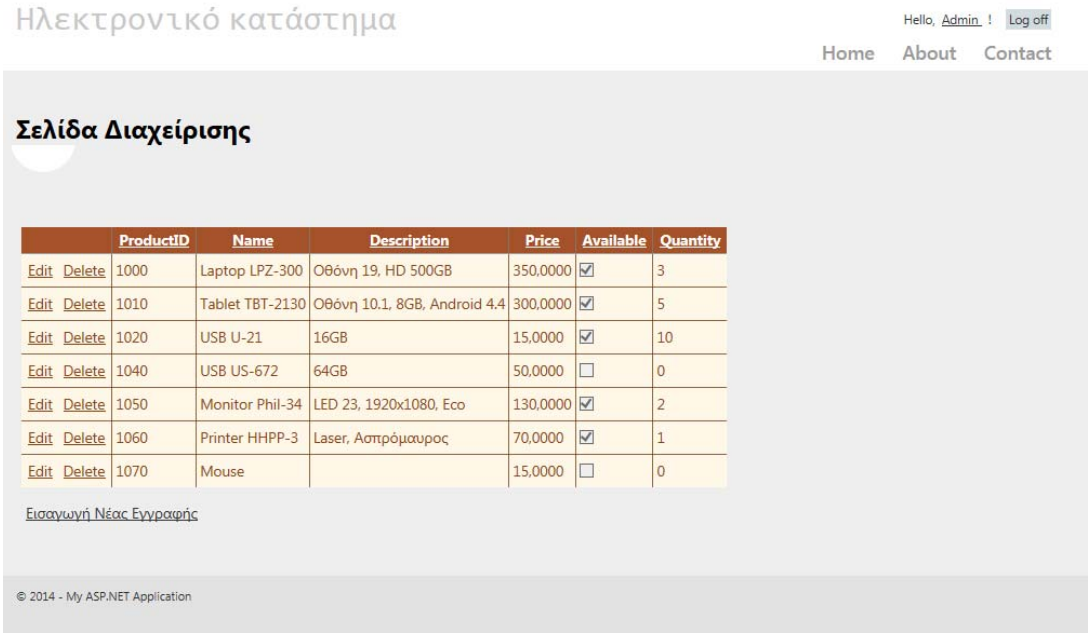


Εικόνα 5.35 *Σύνδεση μηχανισμών*

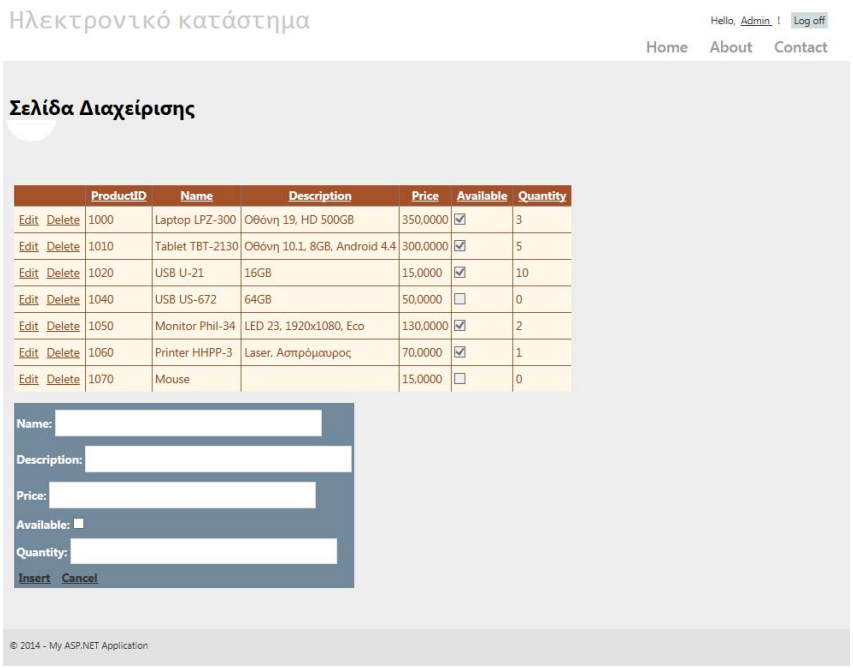
3.3. Σελίδα διαχείρισης

Εκφώνηση:
Στην σελίδα 'AdminSelida' του Ιστότοπου της άσκησης 2.7, να κάνετε τις απαραίτητες ενέργειες, ώστε ο διαχειριστής να μπορεί να ενημερώνει τη βάση. Προσοχή: η Διαγραφή να γίνεται με ασφαλή τρόπο (δηλαδή να ζητείται επιβεβαίωση, ενώ πρέπει να μπορεί να γίνεται και Εισαγωγή νέας εγγραφής εκτός από Διόρθωση.

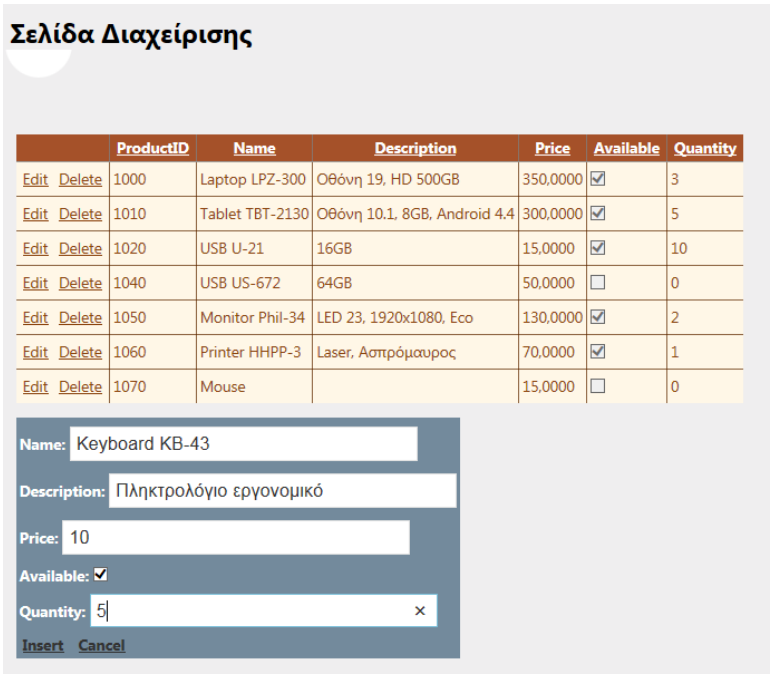
Sound 5.5.mp3	Ηχητικό (audio)
Περιγραφή σελίδας διαχείρισης	



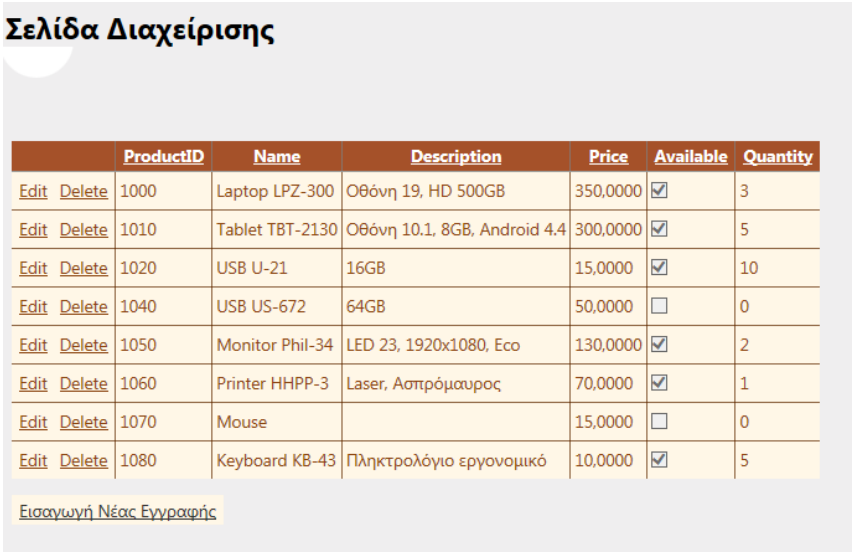
Εικόνα 5.36 *Σελίδα διαχείρισης*



Εικόνα 5.37 *Σελίδα διαχείρισης – δυνατότητα εισαγωγής νέας εγγραφής*



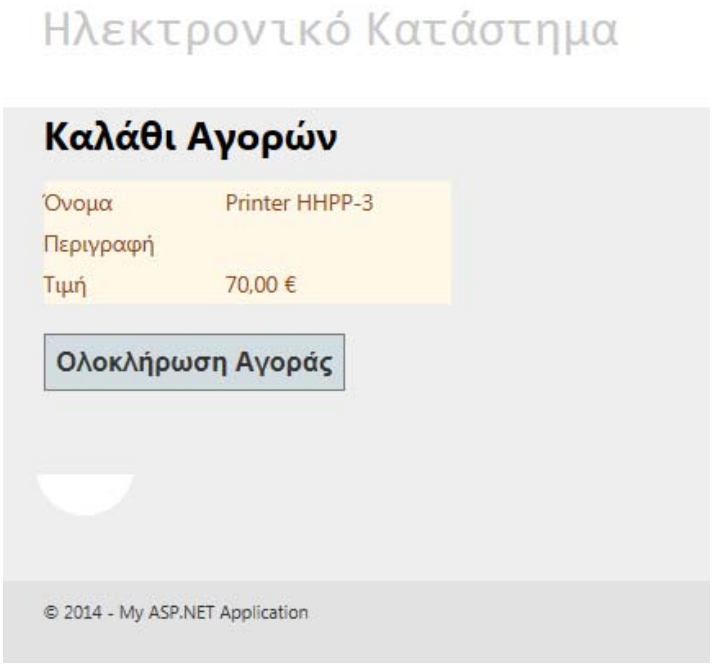
Εικόνα 5.38 *Εισαγωγή νέου προϊόντος*



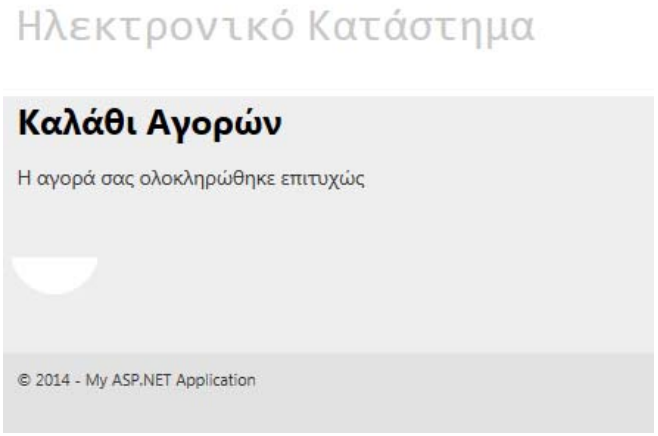
Εικόνα 5.39 *Σελίδα διαχείρισης - ενημερωμένη λίστα προϊόντων*

3.4. Ολοκλήρωση αγοράς

Εκφώνηση:
Στην σελίδα 'Cart' του Ιστότοπου της άσκησης 2.7, προσθέστε ένα κουμπί με το κείμενο "Ολοκλήρωση αγοράς" το οποίο θα κρύβει το προϊόν και θα εμφανίζεται το μήνυμα "Η αγορά σας ολοκληρώθηκε επιτυχώς". Το κουμπί αυτό δε θα πρέπει να φαίνεται όταν το καλάθι είναι άδειο.



Εικόνα 5.40 *Καλάθι αγορών με καταχωρημένο προϊόν*



Εικόνα 5.41 *Ολοκληρωμένη αγορά*

Σημείωση: Στον Ιστότοπο του συγγράμματος (http://ec-tech.uom.gr/WT-ECOM), θα βρείτε τον πηγαίο κώδικα για όλες τις υποδειγματικά λυμένες ασκήσεις, καθώς και για τις ασκήσεις αυτοαξιολόγησης.

4. Συμπεράσματα

Στο κεφάλαιο αυτό παρουσιάστηκε ένα σύνολο εργαστηριακών ασκήσεων, με σκοπό την κατανόηση τεχνικών προγραμματισμού από την πλευρά του διακομιστή (server-side scripting), με χρήση της τεχνολογίας ASP.NET (Active Server Pages). Οι ασκήσεις αυτές έχουν υποδειγματικά επιλυθεί στο περιβάλλον Visual Studio και χρησιμοποιούν ως βασική γλώσσα προγραμματισμού στον Ιστό τη γλώσσα C#. Μέσω των εργαστηριακών ασκήσεων παρουσιάζονται τεχνικές ανάπτυξης εφαρμογών Ιστού, ενώ ιδιαίτερη έμφαση δίνεται σε εφαρμογές που υποστηρίζουν συναλλαγές Ηλεκτρονικού Εμπορίου.

Βιβλιογραφία/Αναφορές

Gaylord, J. N., Wenz, C., Rastogi, P., Miranda, T. & Hanselman, S. (2013). *Professional Asp. net 4.5 in C# and VB*. John Wiley & Sons.

Spaanjaars, I. (2014). *Beginning ASP.NET 4.5.1: in C# and VB*, Wrox Programmer to Programmer, Wrox.

Wojcieszyn, F. (2014). *ASP. NET Web API 2 Recipes: A Problem-Solution Approach*. Apress.

Χρήσιμοι δικτυακοί τόποι:

Developing Web Applications with ASP.NET:
https://msdn.microsoft.com/en-us/library/bb400852%28v=vs.110%29.aspx

C# Web Development:

http://www.bellevuecollege.edu/ce/category/c-web-development/

Web application development resources:

https://msdn.microsoft.com/en-us/web-app-development-msdn.aspx

ASP.NET tutorial:

http://www.w3schools.com/aspnet/

Visual C# Resources:

https://msdn.microsoft.com/en-us/vstudio/hh341490.aspx

Quiz5.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Η CSS είναι μια γλώσσα:

Α) Ανάπτυξης ιστοσελίδων

Β) Μορφοποίησης ιστοσελίδων

Γ) Που χρησιμοποιείται από μια βάση δεδομένων

Απάντηση/Λύση

Β) Μορφοποίησης ιστοσελίδων

Κριτήριο αξιολόγησης 2

[*] Ένα ηλεκτρονικό κατάστημα θα πρέπει να:

Α) Χρησιμοποιεί μία μορφή οργάνωσης (πχ. μια βάση δεδομένων) για τα στοιχεία συναλλαγών του

Β) Έχει αναπτυχθεί χρησιμοποιώντας το περιβάλλον Visual Studio

Γ) Πουλάει αποκλειστικά προϊόντα και όχι υπηρεσίες

Απάντηση/Λύση

Α) Χρησιμοποιεί μία μορφή οργάνωσης (πχ. μια βάση δεδομένων) για τα στοιχεία συναλλαγών του

Κριτήριο αξιολόγησης 3

[*] Μια διαχειριστική ιστοσελίδα παρέχει πρόσβαση:

Α) Σε όλους τους χρήστες ενός συστήματος

Β) Μόνο σε παλιούς χρήστες

Γ) Σε πιστοποιημένους διαχειριστές, που έχουν τον ανάλογο ρόλο

Απάντηση/Λύση

Γ) Σε πιστοποιημένους διαχειριστές, που έχουν τον ανάλογο ρόλο

Κριτήριο αξιολόγησης 4

[*] Το Server-Side Scripting είναι μια μορφή εκτέλεσης κώδικα που αξιοποιεί τις δυνατότητες των σύγχρονων προγραμμάτων περιήγησης:

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 5
[*] Για να συνδεθεί μια Ιστοσελίδα με ένα αρχείο Stylesheet, η ετικέτα <link href="StyleSheet.css" rel="stylesheet" type="text/css" />:

Α) Μπαίνει ως ένθετη ετικέτα στο τμήμα <head>

Β) Μπαίνει ως ένθετη ετικέτα στο τμήμα <body>

Γ) Δεν έχει σημασία που θα τοποθετηθεί

Απάντηση/Λύση
Α) Μπαίνει ως ένθετη ετικέτα στο τμήμα <head>

Κριτήριο αξιολόγησης 6
[*] Για να απαγορεύσουμε την πρόσβαση κάποιων χρηστών σε μια Ιστοσελίδα, μπορούμε να το πετύχουμε:

Α) Μόνο εάν την τοποθετήσουμε σε έναν φάκελο ο οποίος έχει προστατευθεί μέσω του ASP.net Configuration

Β) Μόνο εάν ελέγξουμε, μέσω κώδικα (συμβάν Page_Load), ποιος χρήστης τη ζητάει

Γ) Με οποιονδήποτε από τους δύο προαναφερόμενους τρόπους

Απάντηση/Λύση
Γ) Με οποιονδήποτε από τους δύο προαναφερόμενους τρόπους

Κριτήριο αξιολόγησης 7
[*] Η ιδιότητα IsPostBack, χρησιμοποιείται για να:

Α) Θέσει την τιμή Back στη μεταβλητή Post

Β) Εξετάσει αν η μεταβλητή Back έχει τιμή ίση με Post

Γ) Υποστηρίξει τον έλεγχο για διατήρηση ενός session

Απάντηση/Λύση
Γ) Υποστηρίξει τον έλεγχο για διατήρηση ενός session

Κριτήριο αξιολόγησης 8
[**] Για ποιόν από τους ακόλουθους validators είναι απαραίτητο να γραφεί κώδικας για τη σωστή λειτουργία του:

Α) RequiredField Validator

Β) Custom Validator

Γ) RegularExpression Validator

Δ) Compare Validator

Απάντηση/Λύση
Β) Custom Validator

Κριτήριο αξιολόγησης 9
[**] Η έκφραση SELECT * FROM Table 1 WHERE ([Pedio2] = @XXX), δείχνει όλες τις εγγραφές του Table1 όπου:

Α) Το Pedio2 δεν είναι κενό

Β) Το Pedio2 είναι ίσο με "XXX"

Γ) Το Pedio2 είναι ίσο με την είσοδο που δίνει ο χρήστης κατά την εκτέλεση του ερωτήματος

Απάντηση/Λύση
Γ) Το Pedio2 είναι ίσο με με την είσοδο που δίνει ο χρήστης κατά την εκτέλεση του ερωτήματος

Κριτήριο αξιολόγησης 10
[**] Ποια από τις ακόλουθες προτάσεις είναι σωστή;

Α) Σε έναν ιστότοπο, είναι δυνατόν να υπάρχουν πολλά πρότυπα (.master) και πολλά αρχεία κανόνων μορφοποίησης (.css): κάθε πρότυπο μπορεί να χρησιμοποιεί κανόνες από ΠΟΛΛΑ αρχεία κανόνων, αλλά και κάθε αρχείο κανόνων μπορεί να χρησιμοποιείται από ΠΟΛΛΑ πρότυπα

Β) Σε έναν ιστότοπο, είναι δυνατόν να υπάρχουν πολλά πρότυπα (.master) και πολλά αρχεία κανόνων μορφοποίησης (.css): κάθε πρότυπο μπορεί να χρησιμοποιεί κανόνες από ΠΟΛΛΑ αρχεία κανόνων, αλλά κάθε αρχείο κανόνων μπορεί να χρησιμοποιείται ΜΟΝΟ ΑΠΟ ΕΝΑ πρότυπο

Γ) Σε έναν ιστότοπο, είναι δυνατόν να υπάρχουν πολλά πρότυπα (.master) και πολλά αρχεία κανόνων μορφοποίησης (.css): κάθε αρχείο κανόνων μπορεί να χρησιμοποιείται από ΠΟΛΛΑ πρότυπα, αλλά κάθε πρότυπο μπορεί να χρησιμοποιεί κανόνες ΜΟΝΟ ΑΠΟ ΕΝΑ αρχείο κανόνων

Δ) Σε έναν ιστότοπο, είναι δυνατόν να υπάρχουν πολλά πρότυπα (.master) και πολλά αρχεία κανόνων μορφοποίησης (.css): κάθε αρχείο κανόνων μπορεί να χρησιμοποιεί ΜΟΝΟ ΕΝΑ πρότυπο, και κάθε πρότυπο μπορεί να χρησιμοποιεί κανόνες ΜΟΝΟ ΑΠΟ ΕΝΑ αρχείο κανόνων

Ε) Καμία

Απάντηση/Λύση
Α) Σε έναν ιστότοπο, είναι δυνατόν να υπάρχουν πολλά πρότυπα (.master) και πολλά αρχεία κανόνων μορφοποίησης (.css): κάθε πρότυπο μπορεί να χρησιμοποιεί κανόνες από πολλά αρχεία κανόνων, αλλά και κάθε αρχείο κανόνων μπορεί να χρησιμοποιείται από πολλά πρότυπα

Κεφάλαιο 6: Κινητό Εμπόριο και Συναλλαγές μέσω Φορητών/Ασύρματων Συσκευών

Σύνοψη
Το κεφάλαιο αυτό παρέχει μια εισαγωγή σε σημαντικές έννοιες του κινητού ηλεκτρονικού εμπορίου (m-commerce): κινητό περιβάλλον, κινητός Ιστός, αξιοποίηση της τοποθεσίας και του περιβάλλοντος πλαισίου από τις εφαρμογές κινητού ηλεκτρονικού εμπορίου και είδη υπηρεσιών που παρέχονται. Επίσης γίνεται αναφορά στις ιδιαιτερότητες της αλληλεπίδρασης χρηστών και κινητών συσκευών. Τέλος, ένα μεγάλο τμήμα του κεφαλαίου, αφορά την πολύ σημαντική ενότητα των συστημάτων πληρωμών μέσω κινητών συσκευών (m-payment systems).

Προαπαιτούμενη γνώση
Το κεφάλαιο 1 του παρόντος συγγράμματος

1. Εισαγωγή

Το κινητό ηλεκτρονικό εμπόριο (m-commerce) δημιουργεί νέες προοπτικές στο ηλεκτρονικό εμπόριο (e-commerce), διότι αξιοποιεί καινοτόμες δυνατότητες που παρέχουν οι σύγχρονες κινητές συσκευές (όπως smartphones και tablets), για την παροχή υπηρεσιών ηλεκτρονικού εμπορίου. Η εκτέλεση οικονομικών αλλά και άλλων διάφορων συναλλαγών μέσω ενός ασύρματου δικτύου και μέσω μιας φορητής κινητής συσκευής, μπορεί να καταστήσει την όλη διαδικασία των συναλλαγών πιο ευχάριστη στον καταναλωτή και τελικά πιο αποδοτική στην επιχείρηση. Κινητό ηλεκτρονικό εμπόριο, είναι η εκτέλεση οποιασδήποτε οικονομική συναλλαγής μέσω κινητής συσκευής και αξιοποίησης της ασύρματης δικτύωσης αυτής. Το κινητό ηλεκτρονικό εμπόριο έχει εξελιχθεί αρκετά και είναι αρκετά διακριτό πλέον από το ηλεκτρονικό εμπόριο, με τις σημαντικές διαφορές αυτών να εστιάζονται στις ιδιαιτερότητες που διέπουν τις κινητές συσκευές, τα ασύρματα δίκτυα, αλλά και το τι απαιτήσεις και ποια συμπεριφορά μπορεί να έχει ο κινητός χρήστης.

Οι νέες τεχνολογίες, όπως οι κινητές συσκευές και το Διαδίκτυο, εξελίσσονται συνεχώς. Ταυτόχρονα, ο αριθμός των χρηστών αυτών των τεχνολογιών αυξάνεται συνεχώς. Οι υπηρεσίες που παρέχονται στο κινητό ηλεκτρονικό εμπόριο μπορούν να υποστηριχθούν σε διάφορες κινητές συσκευές αλλά και σε διάφορα ασύρματα δίκτυα. Αυτοί οι δύο παράγοντες πρέπει να λαμβάνονται υπόψιν σε κάθε νέο σχεδιασμό νέων υπηρεσιών ή εφαρμογών κινητού ηλεκτρονικού εμπορίου: είναι προφανές ότι οι κινητές συσκευές διαφέρουν έναντι των σταθερών ηλεκτρονικών υπολογιστών αλλά και τα ασύρματα δίκτυα διαφέρουν σε σχέση με τα ενσύρματα.

Κατά τη διάρκεια της σχεδίασης μίας υπηρεσίας κινητού ηλεκτρονικού εμπορίου θα πρέπει να γίνει προσδιορισμός των απαιτήσεων των χρηστών και θα πρέπει να διασφαλιστεί με κάποιο τρόπο ότι η ανάπτυξη της εφαρμογής θα γίνει με βάση αυτές τις απαιτήσεις, διότι σε άλλη περίπτωση η χρήση της εφαρμογής μπορεί να μην έχει τα αναμενόμενα αποτελέσματα. Οι υπηρεσίες του κινητού ηλεκτρονικού εμπορίου μπορούν να υποστηριχθούν από διάφορα ασύρματα δίκτυα επικοινωνιών όπως το GSM, GPRS, Wi-Fi, Bluetooth. Επίσης, μια εφαρμογή ηλεκτρονικού εμπορίου μπορεί να λειτουργήσει σε ένα κινητό περιβάλλον, αφού εξεταστούν οι περιορισμοί που υπάρχουν στις κινητές συσκευές και στα ασύρματα δίκτυα. Ο πίνακας 6.1 παρουσιάζει τις τεχνολογικές διαφορές ανάμεσα στις κινητές συσκευές και στους σταθερούς ηλεκτρονικούς υπολογιστές.

Κατηγορία	Κινητή συσκευή (Smartphone/Tablet)	Ηλεκτρονικός υπολογιστής
Μέγεθος οθόνης	Μικρό	Μεγάλο
Πληκτρολόγιο	Μικρό	Μεγάλο
Ισχύς επεξεργαστή	Μικρή	Μεγάλη
Μνήμη RAM	Μικρή	Μεγάλη
Διάρκεια Μπαταρίας	Περιορισμένη	-
Ευελιξία	Μικρή	Μεγάλη
Ανάλυση οθόνης	Χαμηλή	Υψηλή
Περιβάλλον	Αφιλόξενο	Φιλόξενο

Γραφικά	Περιορισμένα	Απεριόριστα
Πολυπλοκότητα εισαγωγής κειμένου	Μεγάλη	Μικρή
Ασφάλεια	Χαμηλή	Μέτρια

Πίνακας 6.1. *Διαφορές μεταξύ κινητών συσκευών και σταθερών Η/Υ*

Οι υπηρεσίες και οι εφαρμογές του κινητού ηλεκτρονικού εμπορίου βασίζονται κυρίως στα διαθέσιμα ασύρματα δίκτυα. Υπάρχουν ωστόσο περιορισμοί στα ασύρματα δίκτυα, οι οποίοι βέβαια πρέπει να ληφθούν υπόψιν. Κάποιοι σοβαροί περιορισμοί είναι οι εξής:

- **Παρεμβολές**. Υπάρχουν απώλειες κατά τη μετάδοση δεδομένων διότι υπάρχουν διάφορες παρεμβολές ανάμεσα στον χρήστη και τον πάροχο.
- **Χαμηλό εύρος**. Η ταχύτητα μετάδοσης των δεδομένων είναι ακόμη χαμηλότερη αυτής των ενσύρματων δικτύων.
- **Καθυστερήσεις**. Υπάρχει η πιθανότητα να υπάρχει καθυστέρηση στη μετάδοση των δεδομένων από τον πάροχο και η οποία μπορεί να φτάσει μέχρι αρκετά δευτερόλεπτα.
- **Ασφάλεια**. Η επικοινωνία μέσω ασύρματων δικτύων είναι πιο επιρρεπής σε επιθέσεις. Συνεπώς θα πρέπει να χρησιμοποιούνται σύγχρονες μεθόδους προστασίας.
- **Αποσυνδέσεις**. Οι παρεμβολές ή η αδυναμία κάλυψης του δικτύου μπορούν να οδηγήσουν σε συχνές αποσυνδέσεις.

2. Υπηρεσίες/εφαρμογές του κινητού ηλεκτρονικού εμπορίου

Οι χρήστες κινητών συσκευών έχουν τη δυνατότητα να απολαμβάνουν υπηρεσίες κινητού εμπορίου μέσω διαφόρων τεχνολογιών (μέσω κάποιας κινητής εφαρμογής, μέσω ενός κινητού ιστότοπου, η μέσω ακόμη μηνυμάτων (SMS/MMS). Ένα κλασσικό παράδειγμα είναι οι πληροφορίες καταλόγου, που είναι κάτι το οποίο μπορεί να δώσει στον χρήστη τη δυνατότητα να βρει προϊόντα ή υπηρεσίες με βάση την περιοχή που βρίσκεται. Χρησιμοποιώντας τον ταχυδρομικό κώδικα της περιοχής (ή ενναλακτικά την αυτόματη εύρεση τοποθεσίας), μπορούν να εντοπισθούν κοντινά βενζινάδικα, φαρμακεία, νοσοκομεία, αλλά και άλλα είδη καταστημάτων ή υπηρεσιών.

Οι εφαρμογές κινητών αγορών, δηλαδή η υποστήριξη υπηρεσιών πραγματοποίησης αγοράς ενός προϊόντος ή μίας υπηρεσίας χρησιμοποιώντας αποκλειστικά μια κινητή συσκευή, αποτελεί μια σημαντική κατηγορία υπηρεσιών κινητού ηλεκτρονικού εμπορίου. Η κινητή διασκέδαση είναι ένα ακόμη είδος εφαρμογών κινητού εμπορίου, οι οποίες στοχεύουν στην παροχή ψυχαγωγικών υπηρεσιών στους χρήστες (πχ. παιχνίδια, μουσική κλπ.). Άλλες κατηγορίες εφαρμογών κινητού εμπορίου είναι η κινητή έκδοση εισιτηρίων (οι οποίες μας βοηθούν να αποφύγουμε τις ουρές σε ένα ταμείο), η κινητή τραπεζική, το κινητό μάρκετινγκ και οι κινητές υπηρεσίες πληροφοριών. Στον πίνακα 6.2 βλέπουμε μια επισκόπηση των κατηγοριών των εφαρμογών κινητού εμπορίου με αντίστοιχα παραδείγματα προσφερόμενων υπηρεσιών.

Εφαρμογές	***Παραδείγματα προσφερόμενων υπηρεσιών***
Κινητές αγορές (Mobile Shopping)	• Κινητή αγορά αγαθών και υπηρεσιών
Κινητή τραπεζική (Mobile Banking)	• Κινητή Λογιστική • Κινητή Χρηματιστηριακή • Κινητές Οικονομικές Πληροφορίες
Κινητή διασκέδαση (Mobile Entertainment)	• Κινητά παιχνίδια • Λήψη της μουσικής και των ήχων κλήσης • Λήψη βίντεο και ψηφιακών εικόνων • Υπηρεσίες ψυχαγωγίας βασισμένες στη γεωγραφική τοποθεσία

Κινητές υπηρεσίες πληροφοριών (Mobile Information Services)	• Επικαιρότητα (Πολιτική, Αθλητισμός και άλλα Νέα) • Ταξιδιωτικές Πληροφορίες • Υπηρεσίες Εντοπισμού (πρόσωπα και αντικείμενα) • Κινητές μηχανές αναζήτησης και εφαρμογές κινητού γραφείου (αξιοποιώντας 'νεκρό' χρόνο, π.χ. στη διάρκεια κυκλοφοριακής συμφόρησης)
Κινητό μάρκετινγκ (Mobile Marketing)	• Κινητό Κουπόνι (εκπτώσεις) • Άμεσο (βασισμένο σε ευαίσθητο περιεχόμενο) Μάρκετινγκ • Οργάνωση Κινητών Εκδηλώσεων • Κινητά Ενημερωτικά Δελτία (Newsletters)
Κινητή έκδοση εισιτηρίων (Mobile Ticketing)	• Δημόσιες συγκοινωνίες • Αθλητισμός και Πολιτιστικές Εκδηλώσεις • Κινητή στάθμευση

Πίνακας 6.2. *Σύνοψη υπηρεσιών κινητού ηλεκτρονικού εμπορίου*

3. Κατανοώντας το κινητό περιβάλλον

3.1 Κινητά λειτουργικά συστήματα

Οι κινητές συσκευές για να λειτουργήσουν είναι απαραίτητη η χρήση ενός λειτουργικού συστήματος σχεδιασμένο ειδικά για αυτές τις συσκευές. Τα πιο διαδεδομένα συστήματα είναι το Android (με ποσοστό χρήσης το 2015 της τάξης του 80%), το iOS (με ποσοστό χρήσης το 2015 της τάξης του 15%) και το Windows Phone (με ποσοστό χρήσης το 2015 της τάξης του 4%).

3.1.1 Android

Το Android είναι ένα λειτουργικό σύστημα το οποίο έχει αναπτυχθεί για να τρέχει σε κινητές συσκευές και βασίζεται στο λειτουργικό σύστημα Linux. Το λειτουργικό σύστημα Android αναπτύχθηκε από τη Google και είναι λογισμικό ανοιχτού κώδικα. Η γλώσσα προγραμματισμού Java χρησιμοποιείται για την ανάπτυξη εφαρμογών.

3.1.2 iOS

Το iOS, γνωστό και παλαιότερα ως iPhone OS, είναι ένα λειτουργικό σύστημα κλειστού κώδικα που έχει αναπτυχθεί από την εταιρεία Apple και έχει άδεια για να τρέχει μόνο σε κινητές συσκευές κατασκευασμένες από την Apple. Πιο συγκεκριμένα, λειτουργεί σε συσκευές iPhone, iPad και iPod.

3.1.3 Windows Phone

Το Windows Phone είναι ένα λειτουργικό σύστημα το οποίο έχει αναπτυχθεί από την εταιρεία Microsoft και είναι ο κύριος ανταγωνιστής του Android και του iOS. Η ανάπτυξή του ξεκίνησε στις αρχές της δεκαετίας του 2000 με το όνομα Windows Mobile.

Sound 6.1.mp3	Ηχητικό απόσπασμα (audio)
Κινητά λειτουργικά συστήματα	

Εικόνα 6.1. *Συσκευές με ΛΣ Android 5.0, iOS 7 και Windows Phone 8*

3.2. Κινητός Ιστός (mobile Web)

Ως κινητός Ιστός αναφέρεται η πρόσβαση στον παγκόσμιο Ιστό μέσω του περιηγητή ενός κινητού λειτουργικού συστήματος. Παραδοσιακά, η πρόσβαση στον παγκόσμιο Ιστό γινόταν μέσω σταθερών ηλεκτρονικών υπολογιστών και μέσω ενσύρματων δικτύων. Στις μέρες μας οι σχετικές τεχνολογίες έχουν εξελιχθεί, έτσι ώστε η πρόσβαση στο Διαδίκτυο να είναι διαθέσιμη ανά πάσα στιγμή μέσω μιας κινητής συσκευής και μέσω ενός ασυρμάτου δικτύου. Επιπλέον, ο κινητός Ιστός δίνει τη δυνατότητα άμεσης πρόσβασης σε υπηρεσίες που παρέχονται από τη συσκευή και το λειτουργικό σύστημα (πχ. το GPS). Όμως υπάρχουν και αρκετοί περιορισμοί που συναντώνται σε περιβάλλοντα κινητού Ιστού. Στον πίνακα 6.3 αναφέρονται οι πιο σημαντικοί από αυτούς.

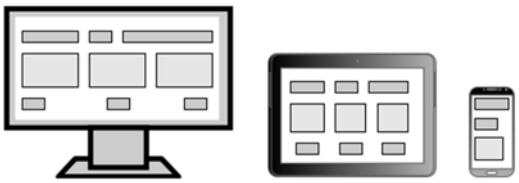
Περιορισμοί	Πληροφορίες
Μέγεθος οθόνης	• Το μικρό μέγεθος της οθόνη καθιστά αδύνατη την προβολή γραφικών όπως αυτά θα εμφανίζονταν σε ένα σταθερό Η/Υ.
Παράθυρα	• Σε ένα σταθερό Η/Υ η ταυτόχρονη περιήγηση σε πολλά διαφορετικά παράθυρα είναι εύκολη. Αυτό δεν ισχύει για τις κινητές συσκευές.
Πλοήγηση	• Η πλοήγηση είναι δύσκολη σε ιστοσελίδες που δεν έχουν σχεδιαστεί για κινητά περιβάλλοντα.
Διαθέσιμες ιστοσελίδες	• Πολλές ιστοσελίδες που είναι διαθέσιμες σε ένα σταθερό Η/Υ δεν είναι διαθέσιμες σε μια κινητή συσκευή.
Ταχύτητα	• Σε πολλές περιπτώσεις η ταχύτητα φόρτωσης είναι πολύ αργή.

Πίνακας 6.3. *Περιορισμοί του κινητού Ιστού*

Υπάρχουν τρεις τρόποι για να αναπτυχθούν ιστοσελίδες για κινητές συσκευές: σμίκρυνση ιστοσελίδων αναπτυγμένων για σταθερούς Η/Υ (miniaturization), ανάπτυξη ιστοσελίδων ειδικά για κινητές συσκευές (mobile specific) και ανάπτυξη προσαρμοστικού (responsive) ιστότοπου, για την οποία τεχνολογία θα αναφερθούμε στην επόμενη ενότητα.

3.2.1 Σχεδιασμός προσαρμοστικών ιστότοπων (responsive Web design)

Η είσοδος των κινητών συσκευών στην αγορά έχει διαφοροποιήσει τις συνήθειες των χρηστών και τον τρόπο με τον οποίο ο παγκόσμιος Ιστός χρησιμοποιείται στις μέρες μας. Οι κινητές συσκευές κατέχουν ένα τεράστιο μερίδιο της αγοράς το οποίο αυξάνεται συνεχώς. Οι συσκευές αυτές ποικίλουν σε χαρακτηριστικά όπως η ανάλυση οθόνης και το μέγεθος τους (ξεκινώντας από τη σχετικά μικρότερη οθόνη των έξυπνων κινητών, εν συνεχεία έρχονται οι ταμπλέτες και τέλος οι οθόνες Η/Υ με τη μεγαλύτερη ανάλυση). Η εικόνα 6.2 αποτυπώνει την τεχνολογία προσαρμοστικών ιστότοπων (responsive Websites), όπου κάθε προσφερόμενη ιστοσελίδα προσαρμόζεται αυτόματα στα χαρακτηριστικά της συσκευής (Jehl & Marcotte, 2014).



Εικόνα 6.2. *Responsive Web*

Η ανάγκη της περιήγησης στο Διαδίκτυο με κάθε συσκευή, ακόμη και με αυτές μικρής ή λίγο μεγαλύτερης ανάλυσης, γέννησε τον όρο 'responsive Web design'. Το 2014, το responsive Web design βρισκόταν στη δεύτερη θέση με τα Top Web Design Trends, σύμφωνα με τις σημαντικότερες κοινότητες και ηλεκτρονικά περιοδικά που έχουν σχέση με την ανάπτυξη ιστοσελίδων και Web εφαρμογών. Το πλεονέκτημα της τεχνολογίας σχεδίασης προσαρμοστικών ιστότοπων είναι ότι δεν απαιτείται η σχεδίαση μιας ξεχωριστής ιστοσελίδας ή εφαρμογής για κάθε τύπο συσκευής: ένας ιστότοπος, μια σχεδίαση προσφέρει ιστοσελίδες και εφαρμογές Διαδικτύου ικανές να προσαρμόζονται χωρίς ασυμβατότητες σε οποιαδήποτε συσκευή αλλά και με οποιαδήποτε ανάλυση οθόνης (Jehl & Marcotte, 2014).

Βασικό στοιχείο που επιτρέπει τη δυναμική αυτή μέθοδο ανάπτυξης ιστοσελίδων είναι τα media queries. Αυτά, εκτελούνται κατά τη διάρκεια φόρτωσης μια ιστοσελίδας και επιτρέπουν στα χαρακτηριστικά της οθόνης του χρήστη να καθορίσουν τον CSS κώδικα που θα εκτελεστεί για την εκάστοτε οθόνη μέσω των παραμέτρων που διαθέτουν. Έτσι, με τη χρήση του ίδιου HTML κώδικα γίνεται χρήση διαφορετικών κανόνων CSS σύμφωνα με τα χαρακτηριστικά της οθόνης, την ανάλυση αυτής και τις διαστάσεις της συσκευής. Παρακάτω, στον πίνακα 6.4, παρουσιάζονται οι πιο σημαντικές παράμετροι που χρησιμοποιούνται για τον σκοπό αυτό.

Παράμετρος	**Τιμή**	**Περιγραφή**
Πλάτος	Μέγεθος	Πλάτος της οθόνης
Ύψος	Μέγεθος	Ύψος της οθόνης
Πλάτος συσκευής	Μέγεθος	Πλάτος της συσκευής
Ύψος συσκευής	Μέγεθος	Ύψος συσκευής
Κατεύθυνση	Κάθετα ή οριζόντια	Προσανατολισμός συσκευής
Αναλογία (aspect-ratio)	Τιμές	Αναλογία μεταξύ πλάτους και ύψους οθόνης (π.χ. 16:9 ή 4:3)
Χρώμα	Ακέραιος	Ο αριθμός των bits ανά τμήμα χρώματος
Ανάλυση	Ανάλυση	Πυκνότητα των pixels στη συσκευή

Πίνακας 6.4. *Παράμετροι των media queries (responsive Web design)*

Η ύπαρξη της τεχνολογίας προσαρμοστικής ανάπτυξης ιστότοπων δε σημαίνει ότι όλα γίνονται αυτόματα με τον ορθό τρόπο, χωρίς προηγούμενη μελέτη και σχεδίαση. Το παράδειγμα που ακολουθεί, δείχνει ενδεικτικά τη φύση των ζητημάτων που προκύπτουν και λαμβάνονται υπόψη: στις κινητές συσκευές γίνεται αυτόματη κλιμάκωση των ιστοσελίδων για να χωρέσουν στην εκάστοτε οθόνη κάθε συσκευής με

αποτέλεσμα όμως (αν δε γίνει η κατάλληλη χρήση κανόνων) να παρουσιάζονται πολύ πιο ευρείες από ότι σε έναν Η/Υ. Αυτό συμβαίνει επειδή οι περιηγητές αντιμετωπίζουν τις mobile οθόνες σαν desktop οθόνες από προεπιλογή. Έτσι, σε ένα smartphone με πλάτος οθόνης 320 pixels, μια ιστοσελίδα πλάτους 960 pixels συρρικνώνεται αυτόματα για να αποδοθεί στην κινητή οθόνη. Για να λυθεί με αισθητικά καλύτερο αποτέλεσμα αυτό το πρόβλημα, μπορεί να γίνει χρήση του κανόνα viewport (και των παραμέτρων βεβαίως αυτού) της responsive Web σχεδίασης, τον οποίο επινόησε η Apple και έκτοτε χρησιμοποιήθηκε ευρέως από όλες τις εταιρείες. Οι συνηθέστεροι παράμετροι που χρησιμοποιούνται είναι ο προσδιορισμός αρχικού ποσοστού εστίασης σε συνδυασμό με τη δυνατότητα στο πρόγραμμα περιήγησης να ορίζει αυτό σαν πλάτος της ιστοσελίδας το πλάτος της εκάστοτε συσκευής. Έτσι, στο παράδειγμά μας, αν η συσκευή έχει πλάτος 320 τότε και το πλάτος της ιστοσελίδας θα είναι 320 και όχι 960 όπως σε έναν Η/Υ.

3.3 Επίγνωση θέσης και πλαισίου (location and context awareness)

Οι κινητές υπηρεσίες επωφελούνται από τις δυνατότητες των κινητών συσκευών (μέσω των κατάλληλων αισθητήρων που διαθέτουν) να γνωρίζουν τη θέση των χρηστών τους και την περιβάλλουσα κατάσταση (πλαίσιο) λειτουργίας αυτών. Με την ευρεία χρήση των κινητών τηλεφώνων έχουν αναπτυχθεί εφαρμογές οι οποίες αξιοποιούν τη θέση του χρήστη και γενικότερα το πλαίσιο (context) ώστε να του κάνουν συστάσεις για προϊόντα και υπηρεσίες σχετικά με την τοποθεσία ή τις διάφορες άλλες παραμέτρους πλαισίου, όπως η παρέα, ο καιρός, ή η απόσταση από συγκεκριμένα σημεία αναφοράς. Να σημειωθεί εδώ ότι για στον όρο context (αποδίδεται ως ‘πλαίσιο’, ‘συναφείς πληροφορίες’, ‘περιβάλλουσα κατάσταση’) αναφερόμαστε και σε άλλα κεφάλαια: για το πώς αξιοποιείται σε ζητήματα σχετικά με πολιτικές ασφάλειας και ελέγχου προσπέλασης, αλλά και πώς επηρεάζει τους μηχανισμούς εξατομίκευσης/παραγωγής συστάσεων.

Στις μέρες μας οι κινητές συσκευές έχουν γίνει μέρος της καθημερινότητας των περισσότερων ανθρώπων. Οι σύγχρονες κινητές συσκευές επιτρέπουν στους χρήστες τους να χρησιμοποιούν διάφορες υπηρεσίες, όπως οι υπηρεσίες που βασίζονται στην τοποθεσία του χρήστη (Location Based Services, LBS). Οι υπηρεσίες αυτές, χρησιμοποιούν την τοποθεσία του χρήστη της κινητής συσκευής, ώστε να του παρέχουν πληροφορίες και συστάσεις σχετικά με τα σημεία ενδιαφέροντός του (e Silva, 2013). Όπως για παράδειγμα εστιατόρια, βενζινάδικα, φαρμακεία. Η επίγνωση της θέσης του χρήστη συνεισφέρει στον προσδιορισμό του πλαισίου του χρήστη. Εκτός όμως από τη θέση του χρήστη υπάρχουν και άλλοι παράγοντες (όπως ο καιρός, η διάθεση ή η ώρα) που μπορούν να ληφθούν υπόψη για να σχηματιστεί το πλαίσιο του χρήστη ώστε ένα σύστημα συστάσεων σε περιβάλλον κινητού εμπορίου να μπορεί να παρέχει πιο ακριβείς εξατομικευμένες συστάσεις. Να σημειωθεί ότι το μικρό μέγεθος της οθόνης των κινητών συσκευών σε συνδυασμό με τις περιορισμένες δυνατότητες εισαγωγής κειμένου, δυσκολεύουν τους χρήστες να φιλτράρουν τις πληροφορίες που παρέχονται από ένα σύστημα συστάσεων και να επιλέξουν ανάμεσα σε αυτές. Για τον λόγο αυτό μία εφαρμογή με επίγνωση πλαισίου, θα πρέπει να είναι σε θέση να κάνει αυτή το όποιο φιλτράρισμα, ώστε να παρέχει στον κινητό χρήστη περιορισμένου όγκου περιεχόμενο.

3.3.1 Τι είναι πλαίσιο

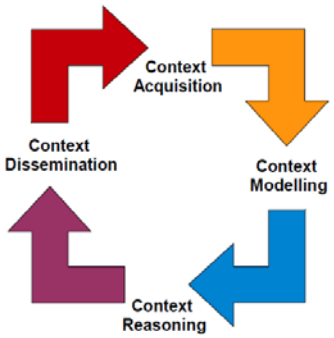
Ένας ευρέως αποδεκτός ορισμός (Abowd et al., 1999) περιγράφει ως πλαίσιο την «οποιαδήποτε πληροφορία που μπορεί να χρησιμοποιηθεί για να χαρακτηρίσει την κατάσταση μιας οντότητας. Μια οντότητα είναι ένα πρόσωπο, ένας χώρος, ή ένα αντικείμενο που θεωρείται ότι σχετίζεται με την αλληλεπίδραση μεταξύ ενός χρήστη και μιας εφαρμογής, συμπεριλαμβανομένων του χρήστη και της ίδιας της εφαρμογής».

Sound 6.2.mp3	Ηχητικό απόσπασμα (audio)
Ο κύκλος ζωής του πλαισίου	

Ο κύκλος ζωής του πλαισίου αποτελείται από τέσσερα στάδια (Perera et al., 2014), τα οποία περιγράφονται παρακάτω αλλά και στην εικόνα 6.3.

1. **Context Acquisition**: Σε αυτό το στάδιο τα δεδομένα που καθορίζουν το πλαίσιο εξάγονται από διάφορες πηγές που περιλαμβάνουν φυσικούς ή εικονικούς αισθητήρες.

2. **Context Modelling**: Στο στάδιο αυτό μοντελοποιούνται τα δεδομένα που συλλέχθηκαν σύμφωνα με κάποιες τεχνικές μοντελοποίησης. Τα μοντέλα αυτά μπορεί να είναι δυναμικά ή και στατικά. Τα στατικά βασίζονται σε ένα προκαθορισμένο σύνολο δεδομένων
3. **Context Reasoning**: Εδώ γίνεται η επεξεργασία των δεδομένων για την εξαγωγή πληροφοριών πλαισίου από τα δεδομένα των αισθητήρων. Αυτό το στάδιο μπορεί να οριστεί και σαν μία μέθοδος παραγωγής γνώσης και καλύτερης κατανόησης του πλαισίου.
4. **Context Dissemination**: Εδώ τα δεδομένα διανέμονται στους ενδιαφερόμενους χρήστες.



Εικόνα 6.3. *Κύκλος ζωής πλαισίου*

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 6.1.gif	Κινούμενη εικόνα (interactive)
Κύκλος ζωής πλαισίου	

3.3.2 Ποιες εφαρμογές κινητού εμπορίου είναι εφαρμογές πλαισίου;

Μία εφαρμογή κινητού εμπορίου η οποία έχει επίγνωση της κατάστασης και του περιβάλλοντος του χρήστη ή διαφορετικά μία εφαρμογή που διαχειρίζεται το πλαίσιο, είναι μια εφαρμογή πλαισίου. Χρησιμοποιείται και ο όρος «ευαίσθητη ως προς το πλαίσιο» (context-aware). Ένα σύστημα είναι σύστημα πλαισίου αν χρησιμοποιεί παραμέτρους πλαισίου για την παροχή σχετικών πληροφοριών ή υπηρεσιών στον χρήστη, όπου η σχετικότητα εξαρτάται από την εφαρμογή του χρήστη αλλά και από το προφίλ χαρακτηριστικών του χρήστη. Πολλοί ερευνητές περιγράφουν τις εφαρμογές αυτές ως εφαρμογές που αλλάζουν δυναμικά ή προσαρμόζουν τη συμπεριφορά τους ανάλογα με τις παραμέτρους πλαισίου. Το πλαίσιο μπορεί να αφορά είτε πληροφορίες σχετικές με τη συσκευή του χρήστη, είτε εκτός της συσκευής, όπως η φυσική θέση του χρήστη, η φυσική του κατάσταση, τα καθημερινά μοτίβα συμπεριφοράς του κ.α. Οι κινητές εφαρμογές χρειάζονται συνεχή ενημέρωση για τις αλλαγές που συμβαίνουν στο περιβάλλον πλαίσιο ώστε να παρέχουν εξατομικευμένες υπηρεσίες στους χρήστες. Τα περισσότερα από τα σημερινά συστήματα εστιάζουν στο πώς να παρέχουν στους χρήστες την πιο κατάλληλη υπηρεσία που ταιριάζει στις ανάγκες τους, ενώ συχνά παραμελούν το γεγονός ότι η δυναμική που έχει το πλαίσιο θα έχει μεγάλη επίδραση στην τελική απόδοση των υπηρεσιών προς τους χρήστες. Ένα σύστημα που βασίζεται στο πλαίσιο θα πρέπει να γνωρίζει εκείνες τις πληροφορίες του περιβάλλοντος του χρήστη που είναι σχετικές με την εφαρμογή. Για παράδειγμα, παράγοντες όπως η θερμοκρασία ή ο καιρός είναι σημαντικές όταν ο χρήστης ψάχνει να παρακολουθήσει μία πολιτιστική εκδήλωση ενώ δεν έχουν ενδιαφέρον όταν ψάχνει για εστιατόρια.

3.3.3 Καθοριστικοί παράγοντες

Υπάρχουν πολλοί παράγοντες που μπορούν να συμπεριληφθούν στο πλαίσιο του χρήστη και στην ιδανική περίπτωση θα πρέπει να λαμβάνονται υπόψιν όλες οι διαθέσιμες πληροφορίες κατά τη στιγμή της αλληλεπίδρασης με την εφαρμογή. Το μεγαλύτερο ενδιαφέρον των συστάσεων δίνεται σε παράγοντες που έχουν άμεση σχέση με το περιβάλλον λόγω του γεγονότος ότι είναι εύκολα μετρήσιμοι με αισθητήρες που είναι ενσωματωμένοι στις περισσότερες κινητές συσκευές. Ενδεικτικά, μπορούν να αξιοποιηθούν παράμετροι

όπως η τοποθεσία, ο προσανατολισμός, η ώρα, ο καιρός, ακόμη και η κίνηση στους δρόμους (Biancalana et al., 2013). Οι πληροφορίες αυτές μπορούν να συνδυαστούν και να εξαχθούν επιπλέον χρήσιμες πληροφορίες που απαιτούνται για τον καθορισμό του πλαισίου.

Για παράδειγμα, μια κινητή εφαρμογή μέσω της οποίας ο χρήστης αναζητά ένα εστιατόριο, μπορεί να συνδυάσει τις πληροφορίες σχετικά με τη θέση του χρήστη, την τρέχουσα ώρα καθώς και την ώρα που κλείνει ένα εστιατόριο ώστε να μπορέσει να προτείνει την καλύτερη σύσταση σύμφωνα με την τρέχουσα κατάσταση του πλαισίου. Ένας άλλος σημαντικός παράγοντας που καθορίζει το πλαίσιο είναι η ανθρώπινη δραστηριότητα του χρήστη. Μία εφαρμογή μπορεί για παράδειγμα να αναγνωρίζει αν ο χρήστης βρίσκεται κοντά στον χώρο εργασίας ή αν ταξιδεύει και έτσι να καταγραφεί στο πλαίσιο του. Όπως έχει ήδη αναφερθεί πληροφορίες που σχηματίζουν το πλαίσιο θα πρέπει να συνδυαστούν. Η αξιολόγηση κάθε πιθανής συσχέτισης μεταξύ των παραγόντων που καθορίζουν το πλαίσιο του χρήστη για ένα σημείο ενδιαφέροντος του, είναι μία δραστηριότητα που απαιτεί τη συλλογή σχετικά μεγάλου όγκου πληροφοριών. Για να βρεθούν οι παράγοντες που καθορίζουν το πλαίσιο θα πρέπει να γίνει μια αρχική μοντελοποίηση των παραγόντων του πλαισίου και του τρόπου που αλληλοεπιδρούν μεταξύ τους, να εκτιμηθούν οι παράγοντες του πλαισίου και η επίδρασή τους, να αναλυθούν οι συσχετίσεις μεταξύ των παραγόντων και να καθοριστούν κανόνες που να καθορίζουν ποιες δράσεις το σύστημα συστάσεων θα πρέπει να χρησιμοποιεί.

3.3.4 Χαρακτηριστικά της ποιότητας του πλαισίου

Για να αξιολογηθεί η ποιότητα του πλαισίου μιας εφαρμογής θα πρέπει να γίνουν μετρήσεις στα χαρακτηριστικά που καθορίζουν το πλαίσιο. Ορισμένα από αυτά είναι (Manzoor et al., 2014):

- Η αξιοπιστία των πληροφοριών.
- Η πληρότητα των πληροφοριών.
- Η επικαιρότητα των πληροφοριών.
- Η χρησιμότητα των πληροφοριών.

Επίσης υπάρχουν ορισμένα χαρακτηριστικά των αισθητήρων τα οποία επηρεάζουν την ποιότητα του πλαισίου όπως:

- Η ακρίβεια του αισθητήρα.
- Η διακριτότητα, η οποία έχει σχέση με το πως ένας αισθητήρας μπορεί να συλλέξει πληροφορίες για το πλαίσιο.
- Αν η πηγή των πληροφοριών είναι δυναμική ή στατική.
- Η μέγιστη απόσταση που μπορεί να καλύψει ένας αισθητήρας για να συλλέξει πληροφορίες του περιβάλλοντος.

3.3.5 Επιπτώσεις χρήσης των πληροφοριών πλαισίου

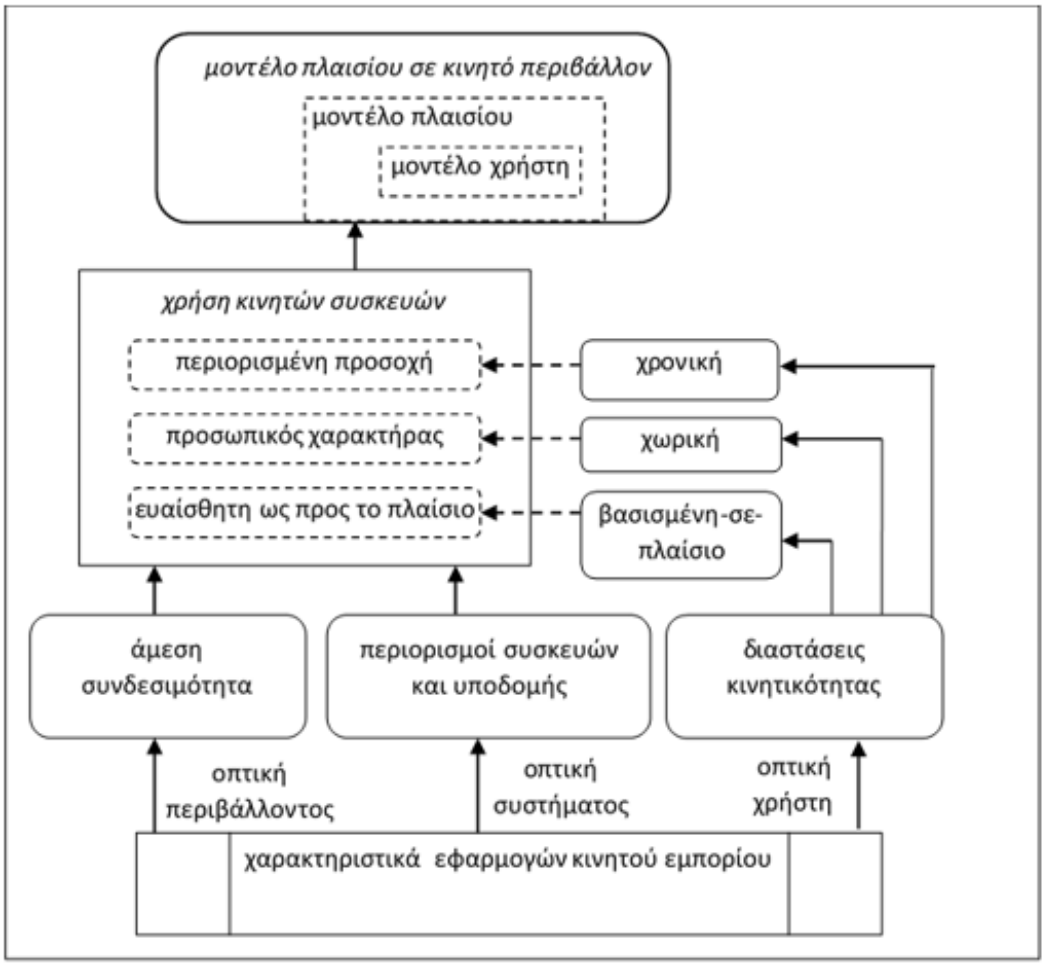
Η μεγάλη χρήση εφαρμογών οι οποίες έχουν σχέση με τη θέση του χρήστη και γενικότερα το πλαίσιό του, φέρνει στο προσκήνιο ζητήματα που σχετίζονται με τις κοινωνικές επιπτώσεις. Τα ζητήματα αυτά αφορούν κυρίως την προστασία της ιδιωτικής ζωής και την κοινωνικότητα των χρηστών γενικότερα. Από σχετικές έρευνες προκύπτει πως όχι η πλειοψηφία, αλλά ένα σημαντικό τμήμα χρηστών κινητών συσκευών, ανησυχούν περισσότερο για την παρακολούθησή τους από κυβερνήσεις καθώς και από διαφημιστικές εταιρίες (λόγω της διαπίστωσης μιας αυξανόμενης δημοτικότητας σε εξατομικευμένες διαφημίσεις με βάση παραμέτρους πλαισίου). Οι χρήστες των κινητών συσκευών, συχνά αισθάνονται ευάλωτοι κυρίως στα θέματα ιδιωτικότητας, και αυτό εξηγείται είτε από την έλλειψη ευαισθητοποίησης σε ζητήματα ασφάλειας που παρατηρείται σε πολλούς κινητούς χρήστες, είτε από το ότι μια σωστή πληροφόρηση στα ζητήματα αυτά απαιτεί προσπάθεια και χρόνο από τους κινητούς χρήστες.

Μια άλλη πλευρά, αναφορικά με τις επιπτώσεις χρήσης των πληροφοριών πλαισίου, είναι σχετική με τη συζήτηση κατά πόσο οι κινητές συσκευές και οι δυνατότητες συναλλαγών που αυτές προσφέρουν, συνηγορούν προς την απομόνωση του χρήστη από το περιβάλλον του. Η χρήση όμως εφαρμογών ευαίσθητων ως προς την τοποθεσία, παρατηρείται ότι αλλάζει τον τρόπο με τον οποίο οι άνθρωποι προγραμματίζουν τις

συναντήσεις τους, υποβοηθώντας κατά πολλούς ερευνητές τελικά την κοινωνικότητά τους. Και αυτό συμβαίνει γενικότερα με τις παραμέτρους πλαισίου: η αξιοποίησή τους βοηθά την επικοινωνία των χρηστών με την υποστήριξη δημιουργίας κοινοτήτων χρηστών με παρόμοια ενδιαφέροντα. Τέλος, ένα ζήτημα που αξίζει εδώ να επισημανθεί, είναι το πως η επίγνωση της θέσης των άλλων χρηστών (η οποία και μπορεί να αποτελέσει παράμετρο πλαισίου), μπορεί να επηρεάζει σαφώς την ιδιωτική τους ζωή, και για αυτό διάφορες τεχνολογίες αναπτύσσονται και εξελίσσονται για την προστασία αυτής (και όχι μόνο) της παραμέτρου με σκοπό τη διασφάλιση της ιδιωτικότητας των χρηστών.

3.4 Ιδιαιτερότητες του κινητού περιβάλλοντος συναλλαγών

Το κινητό περιβάλλον συναλλαγών έχει σημαντικές διαφορές από το αντίστοιχο περιβάλλον του ενσύρματου Διαδικτύου. Οι τυπικές συσκευές σύνδεσης των χρηστών (κινητά τηλέφωνα, ταμπλέτες) αλλά και η ασύρματη δικτύωση παρουσιάζουν ένα κατώτερο επίπεδο πρόσβασης (μικρές οθόνες, χαμηλές ταχύτητες, ασταθείς συνδέσεις κλπ.) σε υπολογιστικούς και δικτυακούς πόρους. Όμως, αυτό το μειονέκτημα έρχεται μαζί με πλεονεκτήματα: οι χρήστες έχουν πρόσβαση στο περιεχόμενο του κινητού Ιστού, αλλά και γενικότερα σε κινητές υπηρεσίες και εφαρμογές, οπουδήποτε και οποτεδήποτε. Από την οπτική γωνία λοιπόν καθαρά των πόρων συστήματος (system perspective), υπάρχουν δεδομένοι περιορισμοί, ενώ από την ευρύτερη οπτική του περιβάλλοντος (environment perspective), είναι εμφανή τα θετικά στοιχεία της άμεσης συνδεσιμότητας (instant connectivity), τη στιγμή ακριβώς που τη χρειάζεσαι.



Εικόνα 6.4. *Ιδιαιτερότητες κινητού περιβάλλοντος για συναλλαγές*

Υπάρχει όμως ακόμη μια πλευρά θεώρησης, αυτή του χρήστη (user perspective). Αυτή βεβαίως εξαρτάται από τα προαναφερόμενα χαρακτηριστικά συστήματος και περιβάλλοντος, όμως επιπρόσθετα

επηρεάζεται από την κινητικότητα των χρηστών (user mobility). Πιο συγκεκριμένα, διακρίνουμε τις ακόλουθες τρεις διαστάσεις κινητικότητας, στις διαδράσεις των χρηστών:

- χωρικής κινητικότητας (spatial mobility): η εκτεταμένη γεωγραφική μετακίνηση του χρήστη (με τη συσκευή του).
- χρονικής κινητικότητας (temporal mobility): οι χρήστες μπορούν να ασχολούνται και με κάποια άλλη εργασία παράλληλα με την ενασχόληση τους με την κινητή συσκευή
- βασισμένη-στο-πλαίσιο κινητικότητας (contextual mobility): η χρήση των κινητών συσκευών από τον χρήστη γίνεται κάτω από δυναμικές συνθήκες που επηρεάζουν τις ενέργειές του.

Μέσα από αυτή την προσέγγιση, προβάλλουν τρία αξιοσημείωτα χαρακτηριστικά στη χρήση των κινητών συσκευών (Georgiadis, 2010):

- οι χρήστες έχουν την τάση να χρησιμοποιούν με προσωπικό και συναισθηματικό τρόπο τα κινητά τους, κυρίως λόγω της χωρικής κινητικότητας, και της καθημερινής ενασχόλησής τους με αυτά
- οι χρήστες έχουν πολύ συχνά μειωμένη προσοχή όταν εκτελούν κάποια λειτουργία με τη συσκευή τους, κυρίως λόγω της χρονικής κινητικότητας. Πολλές φορές την ίδια στιγμή, ασχολούνται παράλληλα και με άλλες ασχολίες (παρακολούθηση τηλεόρασης, συμμετοχή σε συζήτηση, κλπ.), και έτσι η ενασχόλησή τους με μια κινητή λειτουργία εξελίσσεται ως μια δευτερεύουσα δραστηριότητα.
- οι χρήστες εκτιμούν όλες τις διευκολύνσεις που προσφέρει η συσκευή τους (αναγνώρισης τρέχουσας κατάστασης και προτάσεων κατάλληλων λειτουργιών (ή πληροφοριών, ενεργειών), κυρίως λόγω της βασισμένης-στο-πλαίσιο κινητικότητας και της χρήσης της συσκευής τους συχνά σε μια ποικιλία δραστηριοτήτων, σε πολλά διαφορετικά περιβάλλοντα (εικόνα 6.4).

Οι προτιμήσεις των χρηστών για υπηρεσίες σε όλες τις σημαντικές περιοχές κινητού εμπορίου επηρεάζονται προφανώς από όλα αυτά, όπως πολυάριθμες έρευνες έχουν δείξει. Σε πολλές περιπτώσεις, οι κινητοί χρήστες προτιμούν να αποκτήσουν προϊόντα και υπηρεσίες χαμηλού κινδύνου, πιο πρόθυμα από ότι τα υψηλού κινδύνου προϊόντα ή τις υπηρεσίες. Ο κίνδυνος θεωρείται υψηλός όταν οι κατηγορίες προϊόντων ή υπηρεσιών είναι τεχνικά σύνθετες, υψηλής δαπάνης, και σημαντικής διαφοροποίησης ανάμεσα στις διάφορες εμπορικές μάρκες, σε αντίθεση με τα τυποποιημένα και λιγότερο ακριβά προϊόντα (πχ. βιβλία, CDs). Ο κύριος λόγος είναι ότι δεν υπάρχει καμία ανάγκη για πρόσθετες πληροφορίες όταν κάποιος παραγγέλνει ένα απλό προϊόν. Αυτό οδηγεί στο ελάχιστο κόστος αναζήτησης για τους χρήστες. Αφ' ετέρου, κατά την απόκτηση των υψηλού κινδύνου προϊόντων, οι απαιτήσεις των ουσιαστικών σχετικών πληροφοριών και συνεπώς ενός επαρκούς χρόνου αναζήτησης (για τη σύγκριση των τιμών, των χαρακτηριστικών κλπ.) είναι μεγαλύτερες.

Επίσης, με εξαίρεση το πολυμεσικό υλικό (ταινίες, βίντεο, τραγούδια), οι χρήστες κινητών συσκευών προτιμούν το περιεχόμενο που δεν απαιτεί τη διαβίβαση σημαντικής ποσότητας πληροφοριών. Κατά συνέπεια, το χαμηλό σε ένταση περιεχόμενο (πχ. τόνοι κουδουνίσματος, εκθέσεις καιρού και εικονίδια οθόνης) αποδείχθηκε πολύ δημοφιλές. Αυτό συμβαίνει όχι μόνο λόγω της χαμηλής σε πόρους διαθεσιμότητας των κινητών συσκευών (η οποία καθιστά δύσκολα τα υψηλά επίπεδα επεξεργασίας πληροφοριών): οι κατεβασμένες μελωδίες, τα θέματα οθόνης, τα εικονίδια ή ακόμη και η επιλογή των εφαρμογών χρησιμοποιούνται κυρίως για να αντιπροσωπεύσουν τις απόψεις της ταυτότητάς κάποιου σε άλλους, και πραγματικά μπορούν να θεωρηθούν ως περιεχόμενο άμεσα σχετικό με την ιδιωτικότητα του κινητού χρήστη.

Οι χρήστες απαιτούν περαιτέρω προσαρμοσμένο περιεχόμενο στο κινητό περιβάλλον (ανά μεμονωμένο χρήστη), επειδή το επίπεδο της εν δυνάμει εξατομίκευσής του είναι πιο υψηλό από αυτό του ενσύρματου Διαδικτύου. Άλλωστε, πέρα από όλα, το κινητό τηλέφωνο είναι μια φορητή, εκτεθειμένη σε όλους συσκευή, ικανή να δηλώσει τις αισθητικές προτιμήσεις και την προσωπικότητα του χρήστη. Παρατηρείται λοιπόν μια συνεχώς αυξανόμενη τάση για χρήση προσωπικού ή εξατομικευμένου περιεχομένου από το οποίο οι χρήστες εκφράζονται και με το οποίο ‘δένονται’ συναισθηματικά. Να σημειωθεί τέλος, ότι η ανάλυση αυτών των συνθηκών οδηγεί σε προσεγγίσεις επέκτασης των τυπικών μοντέλων χρήστη (user model) σε μοντέλα πλαισίου (context model) ικανά να αποτυπώσουν τις ιδιαιτερότητες του κινητού περιβάλλοντος.

4. Κινητοί χρήστες και απαιτήσεις διάδρασης

4.1 Γραφικές διεπαφές χρήστη

Οι γραφικές διεπαφές χρήστη (Graphical User Interface, GUI) είναι ένα από τα πιο σημαντικά συστατικά στις κινητές συσκευές, αλλά και γενικότερα στα σύγχρονα υπολογιστικά συστήματα. Στην ουσία τους, οι γραφικές διεπαφές χρήστη είναι ιεραρχικές, λόγω της ομαδοποίησης που προσφέρουν για τα στοιχεία εμφάνισης αλλά και για τη συμπεριφορά αυτών (δηλαδή των γεγονότων στα οποία ανταποκρίνονται) παρέχοντας γραφικά οπτικά συστατικά, όπως τα παράθυρα, τα μενού κλπ. Αυτά τα οπτικά συστατικά (τα προσφερόμενα από τη GUI), τα αποκαλούμενα αλλιώς και γραφικά αντικείμενα, καθιστούν το λογισμικό εύκολο σε χρήση.

Τα χαρακτηριστικά μιας γραφικής διεπαφής χρήστη είναι ο γραφικός προσανατολισμός του και η ιεραρχική δομή των γραφικών συστατικών του. Ένα γραφικό περιβάλλον μπορεί να ορισθεί (Memon et al., 2001) ως ένα «ιεραρχικό, γραφικό front-end για ένα λογισμικό που δέχεται ως είσοδο στοιχεία που παράγονται από τον χρήστη και συμβάντα του συστήματος που δημιουργούνται από ένα σταθερό σύνολο γεγονότων και παράγει μια ντετερμινιστική γραφική έξοδο. Ένα γραφικό περιβάλλον περιέχει γραφικά αντικείμενα και κάθε αντικείμενο έχει ένα σταθερό σύνολο ιδιοτήτων. Σε οποιαδήποτε στιγμή κατά τη διάρκεια της εκτέλεσης του GUI, αυτές οι ιδιότητες έχουν διακριτές τιμές, το σύνολο των οποίων αποτελεί την κατάσταση του GUI». Χρησιμοποιείται και ο όρος widgets για τα γραφικά αντικείμενα (όπως τα εικονίδια, η γραμμή προόδου ή ένα μενού). Μαζί, πολλαπλά widgets αποτελούν ουσιαστικά τη γραφική διεπαφή χρήστη.

4.2 Σχεδίαση διάδρασης

Η σχεδίαση διάδρασης (Interaction Design), γνωστή και με τη συντομογραφία IxD, είναι ένα πεδίο σχεδίασης (design field) που αφορά τη μελέτη και σχεδίαση αλληλεπιδραστικών προϊόντων και υπηρεσιών με σκοπό την υποστήριξη του τρόπου με τον οποίο οι άνθρωποι επικοινωνούν στην καθημερινή τους ζωή (Sharp et al., 2007). Είναι κεντρικό σημείο ενδιαφέροντος στη διερεύνηση και την εξέλιξη της έρευνας στην περιοχή του πεδίου της επικοινωνία ανθρώπου-υπολογιστή (Human-Computer Interaction, HCI). Το φαινόμενο της ‘διάδρασης’ υφίσταται όταν ο χρήστης έρχεται σε επαφή – επικοινωνεί πχ. με μια συσκευή, και αφορά τον σχεδιασμό της αλληλεπίδρασης συσκευής - χρήστη με έναν ‘πειθαρχημένο’ τρόπο. Η σχεδίαση διάδρασης αφορά τη διαμόρφωση (shaping) ψηφιακών πραγμάτων με τέτοιο τρόπο ώστε να μπορούν να χρησιμοποιηθούν από ανθρώπους. Όπως και σε πολλούς άλλους τομείς του σχεδιασμού, ο σχεδιασμός διάδρασης ενδιαφέρεται για τη μορφή, αλλά ο κύριος στόχος του είναι η συμπεριφορά. Η σχεδίαση κινητής διάδρασης (mobile interaction design) είναι μια περιοχή της σχεδίασης διάδρασης που ενδιαφέρεται αποκλειστικά για τη δημιουργία εμπειριών χρήστη (user experiences) μέσω διαδραστικών προϊόντων, συσκευών και υπηρεσιών που δεν είναι σταθερής θέσης, αλλά τα οποία οι άνθρωποι μπορούν να κουβαλούν μαζί τους (Kjeldskov, 2013).

Σε μια ‘γλώσσα’ σχεδιασμού διάδρασης μπορούν να εντοπισθούν 5 διαστάσεις:

1. **Λέξεις** (words): Οι λέξεις είναι οι εκφράσεις που οι χρήστες χρησιμοποιούν για να επικοινωνήσουν.
2. **Οπτικές αναπαραστάσεις** (visual representations): Οι οπτικές αναπαραστάσεις είναι τα πράγματα που ο χρήστης αλληλοεπιδρά μέσω της διασύνδεσης.
3. **Φυσικά αντικείμενα ή χώρος** (Physical objects or space): Ο χώρος με τον οποίο ο χρήστης αλληλοεπιδρά είναι η τρίτη διάσταση του σχεδιασμού αλληλεπίδρασης.
4. **Χρόνος** (Time): Ο χρόνος μέσα στον οποίο ο χρήστης αλληλοεπιδρά με τη διεπαφή. Μερικά παραδείγματα είναι περιεχόμενο που αλλάζει με την πάροδο του χρόνου, όπως ήχος, βίντεο ή κινούμενα σχέδια.
5. **Συμπεριφορά** (Behavior): Η συμπεριφορά καθορίζει την αντίδραση των ενεργειών των χρηστών στη διεπαφή - το πώς θα απαντήσουν σε αυτή.

Σημαντικές έννοιες στη διαδικασία του σχεδιασμού διάδρασης:

6. Επαναληπτικός Σχεδιασμός και Αξιολόγηση (Iterative Design and Evaluation)
7. Εμπειρία χρήστη (User experience)
8. Ευχρηστία (Usability)

4.2.1 Επαναληπτικός σχεδιασμός και αξιολόγηση

Συνήθως ένα σύστημα διάδρασης έχει σχεδιαστεί μέσω επαναληπτικών διαδικασιών που αφορούν τον σχεδιασμό, την αξιολόγηση και επανασχεδιασμό. Η διαδικασία σχεδιασμού μπορεί να χωριστεί σε τρεις μεγάλες φάσεις:

1. **Αρχικό σχέδιο** (initial design): σε αυτό στάδιο του σχεδιασμού οι τελικές προδιαγραφές σχεδιασμού δημιουργούνται με προσέγγιση των στόχων και διερεύνηση των κατευθυντήριων γραμμών. Αυτό οδηγεί στην επόμενη φάση της διαδικασίας σχεδιασμού: το πρωτότυπο σχέδιο.
2. **Πρωτότυπο σχέδιο** (prototype design): Κατά τη φάση του σχεδιασμού πρωτοτύπου διεξάγεται μια αξιολόγηση της διεπαφής που υπάρχει ως πρωτότυπο. Η αξιολόγηση σε αυτό το επίπεδο έχει στόχο να εντοπιστούν προβλήματα χρηστικότητας πριν η διεπαφή διατεθεί ως τελικό σχέδιο για να δοκιμαστεί με τελικούς χρήστες (πχ. διόρθωση λαθών στον κώδικα που προκαλούν δυσλειτουργίες).
3. **Τελικό Σχέδιο** (final design): στο τελικό στάδιο του σχεδιασμού δημιουργείται ένα επιχειρησιακό περιβάλλον και γίνεται η τελική αξιολόγηση για να επικυρώσει την αποτελεσματικότητα του τελικού σχεδιασμού.

4.2.2 Εμπειρία χρήστη (user experience)

Η εμπειρία χρήστη (User eXperience, UX) είναι ένας ευρέως χρησιμοποιούμενος όρος για να περιγράψει τις πτυχές της συναισθηματικής κατάστασης του χρήστη κατά την αλληλεπίδρασή του με ένα σύστημα λογισμικού. Η εμπειρία χρήστη ως όρος έχει δει μια αύξηση της χρήσης της κατά τα τελευταία 15 χρόνια. Στις περισσότερες περιπτώσεις η εμπειρία του χρήστη επηρεάζεται από τον λόγο για τον οποίο σχεδιάστηκε η διεπαφή χρήστη και ασχολείται με το γενικό συναίσθημα αλλά και πτυχές ενός προϊόντος/υπηρεσία, όπως οι χρήστες το/την εισπράττουν (Chittaro, 2011). Σήμερα η βιομηχανία έχει στραφεί προς νέα πρότυπα, όπου η εμπειρία του χρήστη βρίσκεται στο επίκεντρο όταν σχεδιάζεται ένα νέο προϊόν ή μια υπηρεσία. Προγραμματιστές εφαρμογών και εταιρίες δίνουν μεγάλη βαρύτητα στην εμπειρία που έχει ο χρήστης με το προϊόν γιατί αυτή η εμπειρία το καθιστά ανταγωνιστικό και βιώσιμο.

4.2.3 Ευχρηστία

Η ευχρηστία είναι ένα χαρακτηριστικό που χρησιμοποιείται για να εκτιμηθεί η ευκολία χρήσης από τον άνθρωπο για προϊόντα, όπως λογισμικό εφαρμογών, ιστοσελίδες ή εργαλεία. Αποτελεί ένα θεμελιώδες χαρακτηριστικό που κρίνει την ποιότητα των συστημάτων λογισμικού. Με βάση το Διεθνές Πρότυπο για την ευχρηστία ISO 9241-11, αυτή ορίζεται ως «ο βαθμός στον οποίο ένα προϊόν μπορεί να χρησιμοποιηθεί από χρήστες, ο οποίος καθορίζεται από την επίτευξη ειδικών στόχων με αποτελεσματικότητα, αποδοτικότητα και ικανοποίηση σε ένα συγκεκριμένο πλαίσιο χρήσης». Για να αξιολογήσουμε την ευχρηστία μιας διεπαφής μπορούμε να διεξάγουμε δοκιμές ευχρηστίες (usability tests) με πραγματικούς χρήστες. Στόχος της διεξαγωγής μιας δοκιμής ευχρηστίας είναι να μετρήσει πόσο καλά και γρήγορα οι χρήστες μπορούν να επιτύχουν συγκεκριμένες τυποποιημένες εργασίες/βήματα και τι προβλήματα μπορεί να αντιμετωπίσουν κατά τη διάρκεια της χρήσης.

Οι δοκιμές ευχρηστίας μιας διεπαφής χρήστη είναι πιο αποτελεσματικές στον προσδιορισμό των ακόλουθων χαρακτηριστικών:

1. **Ονομασία** (naming): αν τα κουμπιά και οι ετικέτες που υπάρχουν είναι κατανοητά από τον τελικό χρήστη ή πρέπει κάτι να αλλάξει για να είναι κατανοητό.

2. **Οργάνωση** (organization): έχουν κατηγοριοποιηθεί οι πληροφορίες σε ομάδες κατανοητές από τον χρήστη; Έχει τοποθετηθεί το περιεχόμενο εκεί που θα περίμεναν οι χρήστες να το βρουν;
3. **Πρώτη χρήση και δυνατότητα εντοπισμού** (first-time use and discoverability): Είναι εύκολο για νέους χρήστες (που χρησιμοποιούν πρώτη φορά την εφαρμογή) να βρουν χωρίς να ψάξουν πολύ, τις πληροφορίες που θέλουν;
4. **Αποτελεσματικότητα** (effectiveness): Υπάρχουν λάθη; Είναι εύκολο οι χρήστες να βρουν εύκολα και γρήγορα συγκεκριμένα πράγματα που τους ζητήθηκαν;

4.3 Ζητήματα επικοινωνίας ανθρώπου-υπολογιστή σε κινητό περιβάλλον (mobile HCI)

4.3.1 Αρχές ευχρηστίας σε κινητές συσκευές

Οι αρχές ευχρηστίας είναι ισχύοντες ομαδοποιήσεις κανόνων που έχουν αναπτυχθεί για να βοηθήσουν στη φάση της ανάπτυξης ενός προϊόντος και να επικυρώσουν την ευχρηστία του. Μπορούμε να διακρίνουμε τις ακόλουθες κατηγορίες αρχών ευχρηστίας σε κινητές συσκευές (Chittaro, 2010):

1. **Υποστήριξη Γνώσης** (Cognition Support): σχετίζεται με γνωστικές πτυχές του χρήστη.
2. **Υποστήριξη Πληροφοριών** (Information Support): σχετίζεται με τα χαρακτηριστικά της οθόνης της κινητής συσκευής και των πληροφοριών που αυτή προβάλλει.
3. **Υποστήριξη Αλληλεπίδρασης** (Interaction Support): σχετίζεται με την αλληλεπίδραση μεταξύ του χρήστη και της κινητής συσκευής.
4. **Υποστήριξη Απόδοσης** (Performance Support): σχετίζεται με την απόδοση της αποστολής που έχει να διεκπεραιώσει ο χρήστης της κινητής συσκευής.
5. **Υποστήριξη Χρηστών** (User Support): αφορά τον βαθμό υποστήριξης του κινητού χρήστη.

4.3.2 Αξιολόγηση διεπαφής χρήστη

Για να πραγματοποιήσουμε την αξιολόγηση της διεπαφής χρήστη βασιζόμαστε σε χαρακτηριστικά που είναι κλειδιά (key-features) για μια επιτυχημένη κινητή εφαρμογή. Αυτά τα χαρακτηριστικά μπορεί να είναι ένα σύνολο κοινών χαρακτηριστικών που βασίζονται σε ήδη επιτυχημένες εφαρμογές οι οποίες έχουν μεγάλο μερίδιο στην αγορά. Οι πηγές που αντλούμε τα key-features για την αξιολόγηση της διεπαφής μας πρέπει να είναι αξιόπιστες και να έχουν χρησιμοποιηθεί από εταιρίες που τα προϊόντα τους κατέχουν μεγάλο μερίδιο αγοράς. Παράδειγμα: ένα κουμπί μπορεί να έχει διαφορετικές λειτουργίες, όμως καθώς αυτό συχνά αναφέρεται στα key-features, μπορούμε να αποφύγουμε λάθος τρόπο χρήσης του κουμπιού στη διεπαφή.

Για να μετρηθεί πρακτικά η αποτελεσματικότητα της σχεδίασης της διεπαφής και των βελτιώσεων που υπήρξαν στα προηγούμενα στάδια, χρησιμοποιούνται διαφορετικές συσκευές για να δούμε την απόκριση της διεπαφής σε διαφορετικές οθόνες και λειτουργικά. Στην εφαρμογές που έχουμε επιλέξει να διεξάγουμε τα τεστ μας αξιολογείται ο κατάλογος (checklist) που δημιουργήσαμε σε προηγούμενο βήμα για να προσδιοριστούν ζητήματα ευχρηστίας. Πραγματικοί χρήστες δοκιμάζουν σε πραγματικό περιβάλλον τη διεπαφή και την αξιολογούν με βάση την εμπειρία τους.

Το ζητούμενο: οι σχεδιαστές διεπαφών χρηστών καταβάλουν προσπάθειες για την ανάπτυξη διαδραστικών προϊόντων κινητής τηλεφωνίας που είναι:

- Εύκολα να τα μάθει ο χρήστης
- Αποτελεσματικότερα στη χρήση
- Ικανά να παρέχουν απολαυστική εμπειρία χρήστη

Μερικά ερωτήματα που προκύπτουν στο επίπεδο των κινητών περιβαλλόντων είναι:

- Ποιες είναι οι μεγάλες προκλήσεις του σχεδιασμού διαδραστικών προϊόντων για κινητές συσκευές;

- Ποιες είναι οι πιθανές λύσεις σε τέτοιου είδους προβλήματα στην ανάπτυξη μιας καλής σχεδίασης με επίκεντρο τον χρήστη για κινητές συσκευές;
- Ποιες είναι οι αρχές του σχεδιασμού διεπαφής χρήστη για κινητές συσκευές;
- Ποιες θα είναι οι νέες τάσεις στη βιομηχανία κινητών στο εγγύς μέλλον;

4.3.3 Προκλήσεις στην υποστήριξη χαρακτηριστικών HCI για κινητές εφαρμογές

Προκλήσεις υλικού:

1. Περιορισμένη δυνατότητα εισόδου
2. Περιορισμένη δυνατότητα εξόδου
3. Σχεδιάζοντας για την κινητικότητα

Ας δούμε αναλυτικότερα τις προκλήσεις σχετικά με την περιορισμένη δυνατότητα εισόδου.

Πληκτρολόγιο

- Χώρος για εγκατάσταση πλήκτρων
- Μικρή αύξηση του πληκτρολογίου μπορεί να συμβάλλει στη μείωση του ρυθμού των λαθών που γίνονται σε πιο μικρό πληκτρολόγιο
- Προσπάθεια για να μάθουν οι χρήστες νέα μέθοδο πληκτρολόγησης

Το στυλ και η οθόνη αφής

- Το μέγεθος της οθόνης του κινητού είναι μικρό

Οθόνη

- Η πιο συχνά χρησιμοποιούμενη ευκολία διεξόδου χρήσης

Ήχος

- Καλή εγκατάσταση παραγωγής για το μήνυμα ανάδρασης προς χρήστη
- Μπορεί να χρησιμοποιηθεί σε συνδυασμό με τα γραφικά και το κείμενο του μηνύματος

Διαχείριση ενέργειας για την κινητή συσκευή

- Ένδειξη (display), μονάδα διαχείρισης ισχύος που συλλέγει πληροφορίες, έτσι ώστε η απόδοση του συστήματος να μην υποβαθμίζεται

Προκλήσεις λογισμικού:

1. Ιεραρχικά μενού (Hierarchical Menus)
2. Πλοήγηση και Περιήγηση (Navigating and Browsing)
3. Εικόνες και Εικονίδια (Images and Icons)

Ένα από τα πιο σημαντικά προβλήματα στη σχεδίαση του λογισμικού για κινητές συσκευές είναι η διαχείριση της εμφάνισης των ιεραρχικών μενού στις περιορισμένων διαστάσεων οθόνες των κινητών συσκευών. Άλλο σημαντικό στοιχείο είναι ότι ο σχεδιασμός της διεπαφής χρήστη πρέπει να είναι ευαίσθητος σε βιώματα και εμπειρίες που έχει κάθε κινητός χρήστης. Στόχος: ένας σχεδιασμός διεπαφής χρήστη που μπορεί να χρησιμοποιηθεί από ένα ετερογενές σύνολο χρηστών. Η φορητότητα (portability) είναι ο κύριος πυρήνας του σχεδιασμού για να ανταποκριθεί ο σχεδιασμός στην απαίτηση της κινητικότητας (Wroblewski,

2011). Τέλος, αξίζει να σημειωθεί ότι ανάλογα με το λειτουργικό που επιλέγουμε για να σχεδιάσουμε την εφαρμογή μας (πχ. Android, iOS, Windows κλπ.) παρέχονται επίσημοι οδηγοί για τη σχεδίαση, που μπορούν να κατευθύνουν προς τη μέγιστη αξιοποίηση των ειδικότερων πόρων που παρέχει το κάθε λειτουργικό σύστημα.

4.3.4 Διάδραση και Ταμπλέτες

Τα τελευταία χρόνια, έχει υπάρξει μια μαζική αύξηση των πωλήσεων σε παγκόσμιο επίπεδο των συσκευών αφής με μεγαλύτερες οθόνες από αυτές των παλαιότερων κινητών συσκευών. Οι ρυθμοί υιοθέτησης αυτών των συσκευών είναι συνεχώς αυξανόμενοι. Η νέα αυτή μορφή κινητών συσκευών, οι ταμπλέτες (tablets), εισήγαγε στον κόσμο ένα νέο τρόπο για την περιήγηση σε ιστότοπους μέσων ενημέρωσης, για την παρακολούθηση ταινιών και τηλεοπτικού προγράμματος, και πολλές άλλες δραστηριότητες. Ένα κοινό χαρακτηριστικό μεταξύ ταμπλέτας και έξυπνου κινητού (smartphone) είναι η τεχνολογία πολλαπλής αφής. Η πολλαπλή αφή επιτρέπει στο σύστημα να ανιχνεύσει ταυτόχρονα τουλάχιστον τρία σημεία επαφής. Λόγω αυτής της τεχνολογίας, η εμπειρία αφής έχει βελτιωθεί δραματικά. Αναγνωρίζεται πλέον ως μια δημοφιλής και ευρέως αποτελεσματική μέθοδος εισόδου μιας γραφικής διεπαφής, χάρη στην ταχύτητα και την 'καθηλωτική' εμπειρία χρήσης που παρέχει.

Ένας υπολογιστής ταμπλέτα θεωρείται συχνά ως μια φορητή έκδοση ενός προσωπικού υπολογιστή και ως ένας σύντροφος όπως το smartphone. Αρκετές φορές μάλιστα μοιράζονται παρόμοιο λειτουργικό σύστημα και μεθόδους εισαγωγής (πχ. προσωπικός υπολογιστής και ταμπλέτα με λειτουργικό σύστημα Windows). Ακόμη: ένα smartphone μπορεί να το έχει παντού κανείς μαζί του, το ίδιο ισχύει και με την ταμπλέτα.

Η διαδικασία της μεθόδου εισαγωγής/εισόδου στοιχείων από έναν χρήστη ταμπλέτας γίνεται μέσω της οθόνης αφής του η οποία δημιουργεί ένα περιβάλλον όπου τα δάκτυλα του χρήστη έχουν την ίδια λειτουργία με δείκτες ποντικιού. Αυτή η νέα μέθοδος εισαγωγής απομάκρυνε την ανάγκη χρησιμοποίησης φυσικών εξαρτημάτων, όπως τα πληκτρολόγια ή το ποντίκι. Οι ταμπλέτες λειτουργούν μέσω της χρήσης διαφόρων λειτουργικών συστημάτων και παράγονται και πωλούνται σε διάφορα μεγέθη (ακόμη και οι μικρότερες ταμπλέτες είναι πολύ μεγαλύτερες από τα smartphones και τους προσωπικούς ψηφιακούς βοηθούς PDA). Να σημειωθεί τέλος ότι οι ταμπλέτες μπορούν να συνδεθούν ασύρματα με άλλα υπολογιστικά εξαρτήματα όπως πληκτρολόγια, κάμερες κλπ.

5. Κινητές πληρωμές

Σε γενικές γραμμές, ο όρος κινητή πληρωμή (mobile payment), ή σύστημα πληρωμών μέσω κινητών συσκευών, αναφέρεται σε πληρωμές για αγαθά, υπηρεσίες και λογαριασμούς μέσω μιας φορητής συσκευής, όπως κινητό τηλέφωνο, έξυπνο τηλέφωνο, ή προσωπικός ψηφιακός βοηθός, με την αξιοποίηση των ασύρματων κυρίως αλλά και άλλων τεχνολογιών επικοινωνιών (Dahlberg et al., 2008). Πραγματικά, η εξέλιξη της τεχνολογίας τα τελευταία χρόνια έχει δώσει αυτή τη δυνατότητα, δηλαδή τη χρήση μιας κινητής συσκευής ως μέσο πληρωμής. Ένας εναλλακτικός, αλλά επίσης καίριος ορισμός της κινητής πληρωμής ακολουθεί: είναι η μεταφορά χρηματικών μονάδων ως αντάλλαγμα για την απόκτηση ενός αγαθού ή την παροχή μίας υπηρεσίας όπου η χρήση της κινητής συσκευής εμπλέκεται τόσο στην έναρξη, όσο και στην εκτέλεση αλλά και στην επιβεβαίωση της πληρωμής (Alliance, 2011).

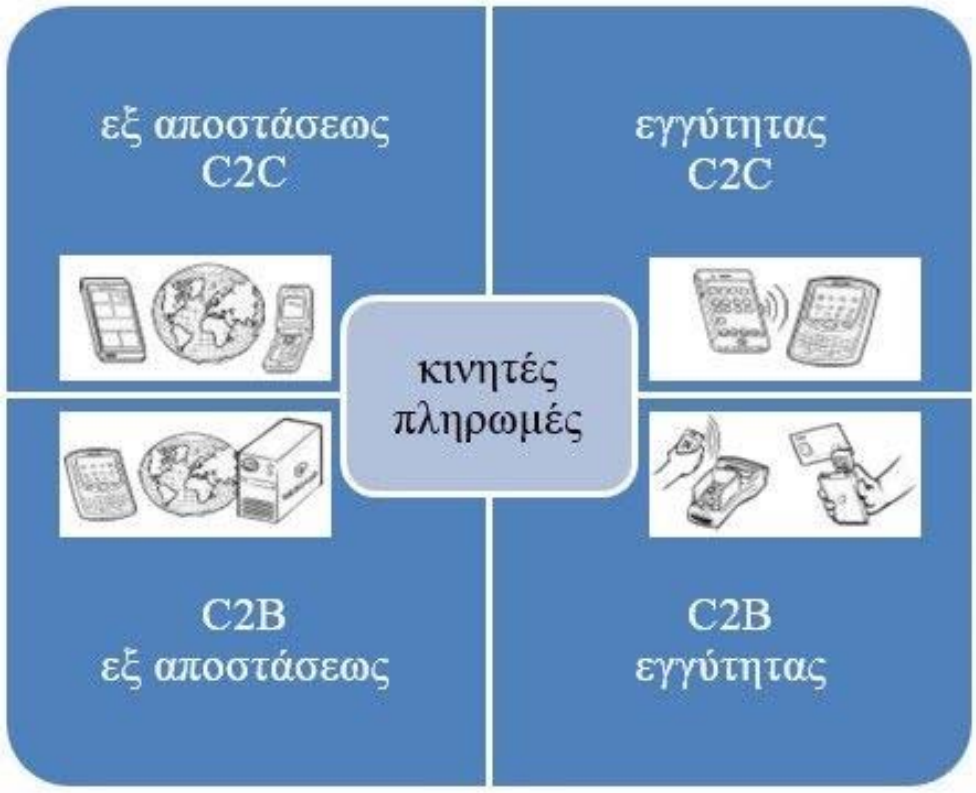
Η κινητή συσκευή στο πλαίσιο μιας οικονομικής συναλλαγής μπορεί να χρησιμοποιηθεί για πολλαπλές διαδικασίες, οι οποίες είναι όλες στενά συνδεδεμένες. Υπάρχει συχνά σύγχυση και επικάλυψη μεταξύ της κινητής πληρωμής, της κινητής τραπεζικής (mobile banking), στο περιβάλλον της οποίας βέβαια μπορεί να εκτελεστεί μια κινητή πληρωμή, της κινητής αυθεντικοποίησης (mobile authentication), της κινητής 'πιστότητας' (mobile loyalty) – διαχείρισης πόντων επιβράβευσης, της κινητής απλά αναζήτησης πληροφοριών (mobile RoPo - Research Online, Purchase Offline), καθώς και της χρήσης του κινητού απλά για παραγγελία (mobile order) ή για παραλαβή (mobile delivery) αγαθών/υπηρεσιών, ενώ η πληρωμή γίνεται με άλλα μέσα (Longini & Gâza, 2013). Η σωστή όμως καταγραφή και μελέτη των ειδικών αναγκών που έχει η διαδικασία της κινητής πληρωμής, επιβάλλει τη διάκρισή της από τις υπόλοιπες προαναφερόμενες διαδικασίες, με τις οποίες βέβαια μοιράζεται κοινά στοιχεία.

Οι κινητές πληρωμές μπορεί να στηρίζονται σε πολλαπλούς μηχανισμούς χρηματοδότησης. Οι συναλλαγές μπορούν να περιλαμβάνονται σε ένα τυπικό μηνιαίο λογαριασμό τηλεφώνου ή να

χρηματοδοτούνται από ένα προπληρωμένο λογαριασμό που συνδέεται με το τηλέφωνο. Αυτό συνηθίζεται για πληρωμές που βασίζονται σε μηνύματα κειμένου (SMS/MMS-based payments). Εναλλακτικά, μετρητά μπορούν να φορτωθούν σε έναν εικονικό λογαριασμό, που χρησιμοποιείται στη συνέχεια για την πληρωμή. Μια άλλη πηγή χρηματοδότησης είναι ένας παραδοσιακός τραπεζικός λογαριασμός ή μια πιστωτική, χρεωστική ή προπληρωμένη κάρτα, στα οποία δίνεται πρόσβαση μέσα από ένα εικονικό πορτοφόλι. Είναι στην ουσία ένα κινητό πορτοφόλι (m-wallet), δηλαδή ένα πορτοφόλι που διαχειριζόμαστε χρησιμοποιώντας είτε το πρόγραμμα περιήγησης της κινητής συσκευής (mobile Web λειτουργία) είτε μια κινητή εφαρμογή (mobile app λειτουργία). Το πορτοφόλι μπορεί να παρέχει πρόσβαση σε μία ή περισσότερες από τις παραπάνω πηγές χρηματοδότησης, οι οποίες και μπορούν να το τροφοδοτούν με χρηματικές μονάδες (Alliance, 2011).

5.1 Κατηγορίες κινητών πληρωμών με βάση το πλαίσιο συναλλαγής

Υπάρχουν διάφορες προσεγγίσεις κατηγοριοποίησης των κινητών πληρωμών. Σε αυτήν την ενότητα θα ακολουθήσουμε την προσέγγιση που βασίζεται στο Μοντέλο Πλαισίου Συναλλαγής (Transaction Context Model). Σύμφωνα με αυτό (Longini & Gâza, 2013), το πλαίσιο συναλλαγής καθορίζεται με δύο όρους: ποια είναι τα μέρη που συναλλάσσονται και ποια η σχετική θέση αυτών. Οι συναλλαγές ως προς τα μέρη που συναλλάσσονται, διακρίνονται σε συναλλαγές μεταξύ καταναλωτών (Consumer-to-Consumer, C2C) και σε συναλλαγές μεταξύ καταναλωτών και επιχειρήσεων (Consumer-to-Business, C2B). Ως προς τη σχετική θέση τους, διακρίνουμε τις κινητές πληρωμές σε εξ αποστάσεως κινητές πληρωμές (remote m-payments) και κινητές πληρωμές εγγύτητας (proximity m-payments). Με τη λογική αυτή, συνδυάζοντας τις συνθήκες, μπορούμε να διακρίνουμε τέσσερα διαφορετικά πλαίσια συναλλαγών, όπου και μπορούν να ενταχθούν όλα τα διαφορετικά συστήματα κινητών πληρωμών.



Εικόνα 6.5. *Κατηγορίες κινητών πληρωμών*

5.1.1 Ομότιμες κινητές πληρωμές (P2P mobile payments)

Ως μια διακριτή προσέγγιση (λόγω ιδιαιτεροτήτων που έχει αλλά και λόγω της μεγάλης δημοτικότητάς της), θα αναφερθούμε εδώ στις **ομότιμες κινητές πληρωμές** (P2P mobile payments) ή αλλιώς **κινητές πληρωμές αποθηκευμένης αξίας**. Άλλωστε, στο 3ο κεφάλαιο στην ενότητα των ηλεκτρονικών πληρωμών έχουμε κάνει μια πρώτη αναφορά στα ομότιμα συστήματα πληρωμών, τύπου PayPal. Οι Person-to-Person ή Peer-to-Peer (P2P) κινητές πληρωμές επιτρέπουν τους ιδιώτες να πληρώνει ο ένας τον άλλο μέσω ενός έμπιστου τρίτου μέρους. Δεν είναι όμως πληρωμές αποκλειστικά για υποστήριξη C2C συναλλαγών: οι υπηρεσίες πληρωμής P2P, επιτρέπουν πχ. σε επιχειρηματίες να μεταφέρουν χρήματα σε ένα λογαριασμό πελάτη ή προμηθευτή (και το αντίστροφο), κάνοντας χρήση μιας διεύθυνσης e-mail ή ενός αριθμού κινητού τηλεφώνου. Οι χρήστες κάνουν συναλλαγές χρησιμοποιώντας χρήματα από έναν τραπεζικό λογαριασμό ή από έναν λογαριασμό κάρτας (πιστωτικής, χρεωστικής ή προπληρωμένης) ή η πληρωμή πραγματοποιείται μέσω της μηνιαίας κατάστασης εξόδων που καλείται να εξοφλήσει ο κάτοχος της κινητής συσκευής (Alliance, 2011). Στην κατηγορία των P2P κινητών πληρωμών το PayPal είναι ένα πολύ διαδεδομένο σύστημα, ικανό τόσο για εξ αποστάσεως πληρωμές (προσφέρει κινητή εφαρμογή για μεταφορά χρημάτων στηριζόμενη σε διεύθυνση e-mail ή αριθμού τηλεφώνου, και επιπλέον υποστηρίζει μεταφορές χρημάτων με χρήση SMS), όσο και για πληρωμές εγγύτητας (μέσω της αξιοποίησης της NFC τεχνολογίας των κινητών συσκευών). Άλλα παραδείγματα ανάλογων υπηρεσιών κινητών πληρωμών προσφέρονται από τις Visa και MasterCard.

5.1.2 Εξ αποστάσεως κινητές πληρωμές

Οι εξ αποστάσεως κινητές πληρωμές είναι πληρωμές στις οποίες η κινητή συσκευή αυτού που πληρώνει δεν έρχεται σε άμεση αλληλεπίδραση με τη συσκευή αυτού που δέχεται την πληρωμή. Στην περίπτωση συναλλαγών C2B, στη μεριά της επιχείρησης βρίσκεται το τερματικό πωλήσεων, ή αλλιώς αναφερόμενο και ως ‘σημείο πώλησης’ (Point of Sale, POS). Η κινητή συσκευή λοιπόν, στις εξ αποστάσεως πληρωμές, δε βρίσκεται κοντά (με τη γεωγραφική έννοια) στη φυσική θέση του συστήματος/συσκευής που υποστηρίζει το σημείο πώλησης για την αποδοχή των πληρωμών (payment acceptance device).

Κινητή μεταφορά χρημάτων (mobile money transfers)
Πρόκειται για υπηρεσίες που επιτρέπουν τη μεταφορά χρηματικών ποσών από ένα λογαριασμό (account) σε έναν άλλο. Οι λογαριασμοί μπορεί να είναι τραπεζικοί, ή λογαριασμοί εικονικών πορτοφολιών (virtual wallets), όπως πχ. χρήματα αποθηκευμένα σε προπληρωμένη κάρτα SIM κινητής συσκευής). Δεν υπάρχει γεωγραφικός περιορισμός: και τοπικής/εθνικής εμβέλειας χρηματικές μεταφορές διεκπεραιώνονται, αλλά και εμβάσματα, δηλαδή διεθνείς μεταφορές χρημάτων. Αξίζει να σημειωθεί ότι η κινητή μεταφορά χρημάτων ως τρόπος πληρωμής αναφέρεται στις κινητές πληρωμές μεταξύ καταναλωτών (C2C). Οι προαναφερόμενες P2P κινητές πληρωμές είναι ένας τρόπος υποστήριξης κινητής μεταφοράς χρημάτων. Ένας άλλο τρόπος είναι οι αντίστοιχες λειτουργίες μεταφοράς χρημάτων ενός ιστότοπου κινητής τραπεζικής (mobile banking). Να σημειωθεί ότι με τη στενή ερμηνεία του όρου, πρέπει και τα δύο συναλλασσόμενα μέρη να χρησιμοποιούν κινητές συσκευές.

Κινητές online πληρωμές (mobile online payments, m-commerce payments)
Οι κινητές online πληρωμές αποτελούν τον κατ’ εξοχήν όρο για τις εξ αποστάσεως πληρωμές για συναλλαγές μεταξύ καταναλωτών και επιχειρήσεων (C2B). Είναι πληρωμές που η εκκίνησή τους γίνεται μέσω της κινητής συσκευής του καταναλωτή-χρήστη, ενώ από τη μεριά της επιχείρησης υπάρχει συνήθως ένα μηχάνημα που παίζει το ρόλο του εικονικού σημείου πώλησης. Οι σύγχρονες κινητές συσκευές είναι εξοπλισμένες με τη λειτουργικότητα να υποστηρίξουν εξ αποστάσεως κινητές πληρωμές μέσω μηνυμάτων SMS, ασφαλών συνόδων του κινητού προγράμματος περιήγησης (secure mobile browser sessions) και εξειδικευμένων κινητών εφαρμογών. Οι εξ αποστάσεως C2B πληρωμές μπορούν να υλοποιηθούν χρησιμοποιώντας (αν υπάρχει) την υφιστάμενη υποδομή (πχ. υποδομή πληρωμής σε Web περιβάλλον) ή ένα ανεξάρτητο κλειστό σύστημα για πληρωμές μέσω κινητών συσκευών. Εφαρμογές που διευκολύνουν τη χρήση τραπεζικών καρτών (πχ. λόγω των περιορισμών που έχουν οι κινητές συσκευές σε μέγεθος οθόνης, και των εξειδικευμένων τρόπων αλληλεπίδρασης που αυτές θέτουν) εντάσσονται σε αυτή την κατηγορία, όπως επίσης και οι P2P πληρωμές για συναλλαγές μεταξύ καταναλωτών και επιχειρήσεων.

5.1.3 Κινητές πληρωμές εγγύτητας (proximity payments)

Είναι οι συναλλαγές όπου βρίσκονται στον ίδιο φυσικό χώρο ο καταναλωτής με τη συσκευή του και η συσκευή αποδοχής πληρωμών (το σημείο πώλησης, POS). Ο καταναλωτής χρησιμοποιεί την κινητή συσκευή του για να αλληλοεπιδράσει φυσικά με το σημείο πώλησης και να μεταφερθούν έτσι οι περιεχόμενες σ' αυτήν, απαιτούμενες πληροφορίες του λογαριασμού πληρωμής.

Πληρωμές χωρίς επαφή (Contactless)
Οι πληρωμές χωρίς επαφή, αποτελούν την κατηγορία όπου οι πληρωμές πραγματοποιούνται χωρίς την επαφή των φορητών συσκευών και αφορούν πληρωμές τόσο μεταξύ των καταναλωτών (C2C) όσο και μεταξύ καταναλωτών και επιχειρήσεων (C2B). Η πληρωμή σε ένα εμπορικό κατάστημα πλησιάζοντας τη φορητή συσκευή στο τερματικό πωλήσεων (POS) και η μεταφορά χρηματικών μονάδων μεταξύ δύο ατόμων πλησιάζοντας τις φορητές τους συσκευές αποτελούν παραδείγματα πληρωμών χωρίς επαφή. Η επίτευξη των πληρωμών χωρίς επαφή επιτυγχάνονται με ποικίλους τεχνολογικά τρόπους (είτε με τη χρήση της τεχνολογίας bluetooth της φορητής συσκευής, είτε μέσω της σύνδεσης στο Διαδίκτυο της φορητής συσκευής, είτε με τη χρήση της τεχνολογίας Near Field Communication (NFC).

Πληρωμές όπου η φορητή συσκευή λειτουργεί ως τερματικό πωλήσεων (mobile PoS)
Συνήθως οι πληρωμές που πραγματοποιούνται με τον τρόπο αυτό αφορούν πληρωμή με κάρτα μεταξύ καταναλωτών και επιχειρήσεων (C2B). Οι πληρωμές πραγματοποιούνται με τη σύμπραξη μιας επιπρόσθετης συσκευής (συσκευή αναγνώρισης καρτών, card reader) στη φορητή συσκευή, καθώς και κατάλληλου λογισμικού για τη χρήση της.

5.2 Τεχνολογικά πλαίσια κινητών πληρωμών (mobile payment framework types)

Ακολουθώντας μια τεχνολογική προσέγγιση, διακρίνουμε τρία κύρια οικοσυστήματα κινητών πληρωμών, στα οποία ένα κινητό πορτοφόλι ή μια κινητή εφαρμογή πληρωμής μπορεί να αναπτυχθεί και λειτουργήσει (Allums, 2014):

1. Επικοινωνία κοντινού πεδίου (Near Field Communication, NFC)
2. Υπολογιστικό νέφος (Cloud)
3. Σύστημα κλειστού βρόχου (Closed Loop)

Κάθε κατηγορία έχει τα υπέρ και τα κατά της. Στη επικοινωνία κοντινού πεδίου υπάρχει καλύτερη ασφάλεια (διότι τα δεδομένα είναι αποθηκευμένα με ασφαλή τρόπο στην κινητή συσκευή), ταχύτητα στη συναλλαγή, αλλά η υλοποίηση είναι σχετικά δύσκολη. Στα πορτοφόλια που βασίζονται στο υπολογιστικό νέφος, η υλοποίηση είναι ευκολότερη, αλλά το κόστος είναι μεγαλύτερο. Η τεχνολογία κλειστού βρόχου συνήθως λειτουργεί συνεργατικά με τα υπάρχοντα συστήματα πληρωμών για υποστήριξη προγραμμάτων ανταμοιβής, είναι μικρού κόστους, αλλά μπορεί να αποτελέσει ρήγμα ασφαλείας αν επιχειρηθεί να συνδεθεί με συστήματα πιστωτικών/χρεωστικών καρτών. Να σημειωθεί ότι στο ίδιο πορτοφόλι θα μπορούσαν να συνδυαστούν οι παραπάνω τεχνολογίες προσφέροντας ένα ευέλικτο κινητό πορτοφόλι, ικανό να διανεμηθεί σε διαφορετικές πλατφόρμες συσκευών.

5.2.1 Τεχνολογία NFC

Η τεχνολογία NFC αποτελεί εξέλιξη της τεχνολογίας RFID (Radio Frequency Identification) και έχει αναπτυχθεί για ασύρματη επικοινωνία σε πολύ κοντινές αποστάσεις. Είναι μια αμφίδρομη τεχνολογία σύζευξης. Η μέθοδος της επαγωγής και της δημιουργίας ενός ηλεκτρομαγνητικού πεδίου χρησιμοποιείται ανάμεσα σε δύο συσκευές. Το ότι το μοντέλο αυτό είναι απλό σε συνδυασμό με τον μικρό χρονικό διάστημα που χρειάζεται για την εγκαθίδρυση της επικοινωνίας, καθιστά την τεχνολογία αυτή ελκυστική για διάφορες κατηγορίες υπηρεσιών και πέραν αυτής των κινητών πληρωμών. Η επικοινωνία χρησιμοποιεί τη συχνότητα των 13.56 MHz και τα όρια της απόστασης έχουν οριστεί στα δέκα εκατοστά.

Σε ένα NFC σύστημα (όπως και στα RFID συστήματα) διακρίνονται δύο συσκευές: η συσκευή ανάγνωσης και η ετικέτα. Η συσκευή ανάγνωσης εκπέμπει ραδιοκύματα στον χώρο για να ενεργοποιήσει τη

μετάδοση της ετικέτας και να ‘διαβάσει’ τα δεδομένα της. Η μεγαλύτερη διαφορά ανάμεσα στην RFID και NFC τεχνολογία είναι ότι στα NFC συστήματα η συσκευή με ενσωματωμένη την NFC τεχνολογία (συνήθως ένα μικρό chip) μπορεί να λειτουργήσει ταυτόχρονα σαν συσκευή ανάγνωσης ή σαν ετικέτα, μέσω δύο διαφορετικών ρυθμίσεων. Αντίθετα, στα RFID συστήματα οι δύο αυτές συσκευές είναι ξεχωριστές, με διαφορετικό μέγεθος αλλά και διαφορετικές δυνατότητες. Η τεχνολογία NFC διαθέτει λοιπόν δύο λειτουργίες: παθητική (passive) και ενεργητική (active). Στην παθητική της ρύθμιση, η μία συσκευή NFC λειτουργεί σαν συσκευή ανάγνωσης ενώ η άλλη σαν ετικέτα, επιτρέποντας στην πρώτη να την ‘διαβάσει’ (σάρωση). Αντίθετα, στην ενεργητική της μορφή οι δύο συσκευές NFC επικοινωνούν σε ‘Peer-to-Peer’ κατάσταση (απευθείας) ανοιγοκλείνοντας εναλλάξ τα ραδιοκύματα τους (Coskun et al., 2013; Vermaas et al., 2013).

Ας το εξετάσουμε και αλλιώς; μπορούν να υπάρξουν τριών ειδών συσκευές που ενσωματώνουν NFC: οι NFC ετικέτες (NFC tags), οι NFC αναγνώστες (NFC readers) και οι NFC κινητές συσκευές. Οι NFC συσκευές, ανάλογα με το εάν προκαλούν τη διάδραση (δηλαδή αποτελούν τους εκκινητές της διάδρασης) ή αν αποτελούν τους αποδέκτες, αντιστοιχούν στα ενεργά ή παθητικά μέρη της διαδικασίας επικοινωνίας. Ο διαχωρισμός αυτός συμβαίνει και αφορά το κατά πόσο μια NFC συσκευή χρησιμοποιεί τη δική της πηγή ενέργειας για τη δημιουργία RF πεδίου. Κατά τον ενεργό τρόπο λειτουργίας και οι δύο NFC συσκευές παράγουν πεδίο για να ανταλλάξουν δεδομένα. Αντίστοιχα, κατά τον παθητικό τρόπο λειτουργίας ο εκκινητής είναι αυτός που παράγει RF πεδίο ενώ ο αποδέκτης χρησιμοποιεί την ενέργεια από το πεδίο του εκκινητή (Coskun et al., 2013; Coskun et al., 2013b). Συνεπώς, ένα NFC tag το οποίο δε διαθέτει δική του εσωτερική πηγή ενέργειας λειτουργεί αποκλειστικά ως παθητικό μέρος στη διάδραση (όπως συμβαίνει και στην περίπτωση του RFID) και επαφίεται στην ενέργεια από το πεδίο του NFC κινητού ή του NFC reader για την ανταλλαγή δεδομένων.

Sound 6.3.mp3	Ηχητικό απόσπασμα (audio)
Επεξήγηση της NFC	

Στη παθητική λειτουργία η NFC συσκευή μπορεί να αναγνωστεί λοιπόν από άλλες συσκευές (ακόμη και RFID συσκευές ανάγνωσης) και μπορεί να προσομοιώσει τις παλαιότερες τεχνολογίες για επικοινωνία. Όταν βρίσκεται στην ενεργή κατάσταση μια NFC συσκευή, υπάρχει η δυνατότητα να ‘συνομιλήσει’ με άλλη NFC συσκευή σε απόσταση μέχρι δέκα εκατοστά. Η τεχνολογία NDEF (NFC Data Exchange Format) χρησιμοποιείται για την ανταλλαγή μηνυμάτων μεταξύ των συσκευών. Επίσης, θα πρέπει να σημειωθεί ότι το εύρος της ζώνης για την επικοινωνία βρίσκεται ανάμεσα στα 126 και 424 kbps, το οποίο είναι αρκετά μικρότερο σε σχέση με τις ασύρματες τεχνολογίες Wi-Fi αλλά και Bluetooth. Ωστόσο το εύρος αυτό είναι αρκετό για τον λόγο για τον οποίο χρησιμοποιείται η μέθοδος αυτή, δηλαδή τη μεταφορά ενός μικρού όγκου πληροφοριών για πληρωμή.

Η τεχνολογία NFC βασίζεται σε έναν αριθμό προτύπων ISO (Nelson et al., 2013):

1. ISO 18092. Near Field Communication Interface and Protocol (NFCIP-1). Σε αυτό το πρότυπο υπάρχει η βάση όπου καθορίζονται παράγοντες όπως είναι η ταχύτητα και η κωδικοποίηση. Επίσης περιγράφονται οι λειτουργίες χαμηλού επιπέδου όπως είναι η εγκαθίδρυση για να επικοινωνήσουν οι δύο συσκευές, ό έλεγχος συγκρούσεων των δεδομένων και η passive ή active λειτουργία.
2. ISO 21481. Near Field Communication Interface and Protocol (NFCIP-2). Αυτό το πρότυπο είναι υπεύθυνο για τον καθορισμό του τρόπου ανίχνευσης των συσκευών RFID.
3. ISO 28361. Near Field Communication Wired Interface (NFC-WI). Αυτό το πρότυπο είναι υπεύθυνο για την επεξεργασία του σήματος μεταξύ της κεραίας και του ελεγκτή της επικοινωνίας.
4. ISO 16353. Front-end Configuration Command for NFC-WI (NFC-FEC). Αυτό το πρότυπο υπάρχει για την ανταλλαγή πληροφοριών μεταξύ NFC-WI και NFC front-end.

5.2.2 Εφαρμογές NFC

Η NFC τεχνολογία έχει ωριμάσει σε σημείο που να υπάρχει σήμερα ένας ικανοποιητικός αριθμός εφαρμογών, εκτός των εφαρμογών υποστήριξης κινητών πληρωμών.

Πληρωμές (payments)
Η πιο διαδεδομένη εφαρμογή της τεχνολογίας NFC είναι οι πληρωμές μέσω κινητής συσκευής. Στη μέθοδο πληρωμής η κινητή συσκευή μετατρέπεται σε ένα είδος πορτοφολιού που αντικαθιστά τις πιστωτικές κάρτες ή και τα μετρητά. Ο επεξεργαστής που υπάρχει στην κινητή συσκευή χρησιμοποιεί την RFID τεχνολογία για να επικοινωνήσει με το τερματικό πωλήσεων (POS), καθιστώντας έτσι την κινητή συσκευή σε μία κινητή πιστωτική ή χρεωστική κάρτα (Allums, 2014). Ό χρήστης το μόνο που έχει να κάνει είναι να ξεκινήσει την εφαρμογή και να τοποθετήσει την κινητή συσκευή του κοντά στο τερματικό πωλήσεων. Μια από τις τελευταίες εφαρμογές στο πεδίο των πληρωμών μέσω τεχνολογίας NFC είναι το Google Wallet (Google, 2015). Η εφαρμογή μπορεί να χρησιμοποιηθεί σε όλα τα κινητά που διαθέτουν NFC hardware και τρέχουν κάτω από το λειτουργικό σύστημα Android. Το σύστημα αναπτύχθηκε σε συνεργασία με τη MasterCard. Η εφαρμογή επιτρέπει στους χρήστες να αποθηκεύουν τα στοιχεία των πιστωτικών καρτών στο τηλέφωνο και να χρησιμοποιούν αυτό στις συναλλαγές αντί των καρτών, με μια κοινή διαδικασία ενεργοποίησης για όλες τις κάρτες

Ηλεκτρονικά κλειδιά (electronic keys)
Η τεχνολογία ηλεκτρονικών κλειδιών μετατρέπει μία κινητή συσκευή σε ένα είδος πολλαπλών κλειδιών τα οποία μπορούν να χρησιμοποιηθούν αντί για τα κανονικά κλειδιά (Allums, 2014). Ο επεξεργαστής που υπάρχει στην κινητή συσκευή χρησιμοποιεί την RFID τεχνολογία για να μετατρέψει την κινητή συσκευή σε κλειδί. Μπορούμε να αντικαταστήσουμε τα κλειδιά του σπιτιού μας, του γραφείου μας ή του αυτοκινήτου μας με ένα σετ ηλεκτρονικών κλειδιών. Στόχος της εφαρμογής αυτής στο μέλλον είναι να δίνει, πέρα από πρόσβαση, και πληροφορίες για το αντικείμενο που ξεκλειδώνει (πχ. για το αυτοκίνητο να δείχνει τα διαθέσιμα καύσιμα του).

Εισιτήρια (tickets)
Ο χρήστης μιας NFC κινητής συσκευής, περνώντας από ένα NFC τερματικό εισιτήριων κλείνει το εισιτήριο του, πληρώνοντας για αυτό, ή ακυρώνει τα εισιτήρια που έχει ήδη 'αποθηκεύσει' (αγοράσει) σε αυτή, με μόλις ένα πέρασμα ή ένα άγγιγμα της κινητής συσκευής του πάνω από το τερματικό εισιτήριων. Ο χρήστης θα έχει επιλέξει από πριν τη διαδρομή στο κινητό του τηλέφωνο (μέσω μιας εφαρμογής) ή θα την επιλέγει εκείνη τη στιγμή στην οθόνη του τερματικού εισιτηρίων. Η 'προσομοίωση κάρτας' μετατρέπει την NFC κινητή συσκευή σε 'έξυπνο' ανέπαφο εισιτήριο (παθητική ετικέτα). Η κινητή συσκευή, πέρα από ηλεκτρονικό πορτοφόλι, λειτουργεί και σαν ηλεκτρονικό εισιτήριο για την είσοδο σε διάφορους χώρους ή για τον έλεγχο μετά την είσοδο. Η ιδέα αυτή έχει πολλές εφαρμογές σε χώρους όπως οι μεταφορές (λεωφορεία), η διασκέδαση (πχ. συναυλίες), και η στάθμευση (Coskun et al., 2013b; Vermaas et al., 2013).

Έξυπνα αντικείμενα (smart objects)
Αντικείμενα τα οποία έχουν ενσωματωμένη την τεχνολογία NFC, σε μορφή ετικέτας, μπορούν να μετατραπούν σε έξυπνα αντικείμενα διευκολύνοντας έτσι τις καθημερινές δραστηριότητες. Ουσιαστικά η ετικέτα παίζει τον ρόλο της πύλης (portal) που μεταφέρει τον χρήστη της NFC κινητής συσκευής στον ψηφιακό κόσμο παρέχοντας του χρήσιμες πληροφορίες για το αντικείμενο που 'ανιχνεύει'. Και εδώ, οι δυνατότητες της NFC τεχνολογίας είναι άπειρες αφού μπορεί να χρησιμοποιηθεί σχεδόν σε οποιοδήποτε φυσικό αντικείμενο τόσο για ενημερωτικούς όσο και για διαφημιστικούς σκοπούς. Ιδιαίτερα διαδεδομένη είναι η χρήση της ιδέας αυτής σε αφίσες (Coskun et al., 2013b).

5.2.3 Αρχιτεκτονική μιας NFC κινητής συσκευής

Υπάρχει ένας γενικός πλαίσιο που λειτουργεί σαν πρότυπο ανάμεσα στους κατασκευαστές. Αυτός ο οδηγός είναι διαθέσιμος από το NFC forum και περιλαμβάνει τα τρία παρακάτω μέρη (Vermaas et al., 2013; Nokia Forum, 2011).

1. Λειτουργίες Παρασκηνίου της NFC Κινητής Συσκευής (NFC Mobile Back End Server System Functionalities
2. Λειτουργίες NFC Κινητής Συσκευής (NFC Mobile Phone Functionalities)
3. NFC στόχος NFC (NFC Target)

Εικόνα 6.6. *Πλαίσιο NFC mobile*

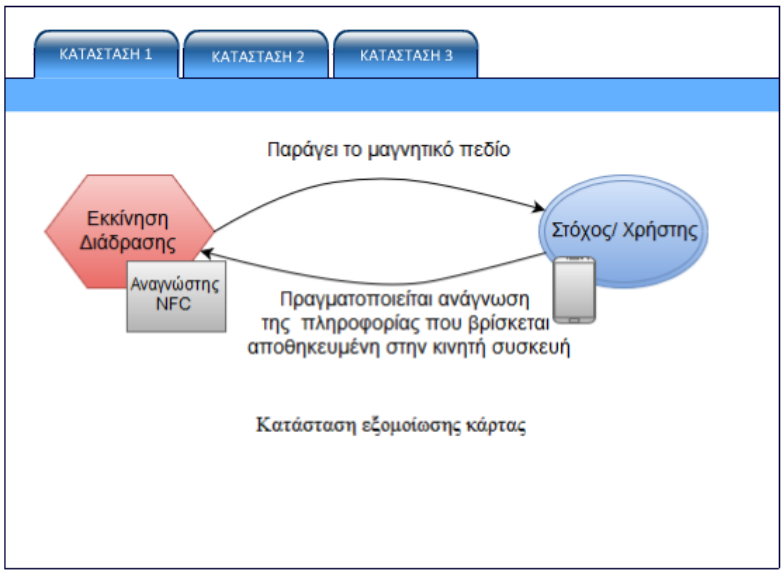
Η εικόνα 6.6 μας δείχνει μια απεικόνιση των τριών μερών. Το πλαίσιο αυτό μπορεί να χρησιμοποιηθεί σαν υπόδειγμα για προτυποποίηση χωρίς όμως να λειτουργεί περιοριστικά.

5.2.4 Τρόποι λειτουργίας NFC συσκευών

Η συσκευή NFC μπορεί να βρίσκεται σε μια από τις ακόλουθες τρείς καταστάσεις:

- Κατάσταση εξομοίωσης κάρτας (card emulation mode), όπου συμπεριφέρεται σαν πιστωτική κάρτα με δυνατότητα πραγματοποίησης συναλλαγών
- Κατάσταση ανάγνωσης/εγγραφής (reader/writer mode), οπότε μπορεί να ανιχνεύσει διαθέσιμες υπηρεσίες, για χρήση στην έξυπνη διαφήμιση σε POS ή στην υποστήριξη έξυπνων posters
- Κατάσταση ομότιμης σύνδεσης (peer-to-peer mode), για ανταλλαγή πληροφοριών μεταξύ δύο συσκευών μέσω NDEF.

Flash 6.1.swf	Αρχείο flash (interactive)
Καταστάσεις συσκευών NFC	



Εικόνα 6.7 *Κατάσταση εξομοίωσης κάρτας*

5.2.5 Απειλές εναντίον της τεχνολογίας NFC

Το μεγαλύτερο, ίσως, πλεονέκτημα που έχει η τεχνολογία NFC είναι η ασφάλεια (Vermaas, 2013). Λόγω της στιγμιαίας σύνδεσης και πολύ κοντινής απόστασης ανάμεσα στην κινητή συσκευή και το τερματικό, είναι σχεδόν αδύνατο να σημειωθεί κάποιου είδους επίθεση. Παρόλα αυτά εάν υπάρχει ο σχετικός εξοπλισμός, τότε υπάρχει η δυνατότητα να σημειωθούν επιθέσεις, οι οποίες περιγράφονται στη συνέχεια (Coskun et al., 2013; Vermaas, 2013):

Επίθεση τύπου αναμετάδοσης (relay)
Στις επιθέσεις αυτού του τύπου μπορούν να υπάρξουν δύο ενδεχομένως δράστες: ο πρώτος που τοποθετεί μια κινητή συσκευή με τεχνολογία NFC κοντά στο τερματικό του καταστήματος και ο δεύτερος που τοποθετεί πάλι μια κινητή συσκευή κοντά σε μια RFID ετικέτα από την οποία αντιγράφει πληροφορίες τις οποίες αναμεταδίδει.

Επίθεση τύπου Gateway
Σε αυτό το είδος επίθεσης ο ανύποπτος χρήστης χρησιμοποιώντας τη συσκευή του για να σκανάρει μια RFID ετικέτα μεταφέρεται σε μια κακόβουλη ιστοσελίδα ή κατεβάζει κάποιου είδους κακόβουλο λογισμικό το οποίο θα πειράξει τη συσκευή, θα υποκλέψει δεδομένα ή θα αποκτήσει τον πλήρη έλεγχο αυτής.

Υποκλοπή (eavesdropping)
Στις συσκευές NFC υπάρχει η δυνατότητα υποκλοπής δεδομένων, διότι χρησιμοποιούνται ραδιοκύματα για την επικοινωνία. Κάποιος πιθανός υποκλοπέας χρησιμοποιώντας τον κατάλληλο εξοπλισμό μπορεί να κρυφακούσει και να αποκτήσει πρόσβαση στα δεδομένα, ειδικά στην περίπτωση όπου τεχνολογίες ισχυρής κρυπτογράφησης δε χρησιμοποιούνται (Moloney, 2014).

Ένα άλλο είδος παρόμοιας επίθεσης γίνεται με τη χρήση κακόβουλου λογισμικού που ο επιτιθέμενος εγκαθιστά με κάποιο τρόπο στο τερματικό (POS) της επιχείρησης. Σε αυτήν την περίπτωση όλα τα προσωπικά στοιχεία των χρηστών που σκανάρονται από αυτό το τερματικό αποστέλλονται σε κάποιο προεπιλεγμένο σημείο. Παρόμοια επίθεση μπορεί να γίνει με τη χρήση κακόβουλου λογισμικού που ο επιτιθέμενος εγκαθιστά με κάποιο τρόπο στην κινητή συσκευή του χρήστη (αναφερόμαστε αναλυτικά στο κακόβουλο λογισμικό για κινητές συσκευές σε άλλο κεφάλαιο).

Τροποποίηση και φθορά δεδομένων (data modification, data corruption)
Σε αυτό το είδος επίθεσης ο δράστης μπορεί, εκτός από το να κρυφακούσει τη μετάδοση των δεδομένων, να μπλοκάρει ή τροποποιήσει το περιεχόμενο της, παραπλανώντας έτσι τις δύο συσκευές, την κινητή συσκευή και το τερματικό. Αυτό μπορεί εύκολα να γίνει τοποθετώντας ένα κακόβουλο κώδικα στην NFC κινητή συσκευή ή στο τερματικό (POS). Τότε ο κακόβουλος χρήστης θα μπορεί να τη χρησιμοποιήσει για παράνομες συναλλαγές προς όφελος του. Επίσης, κάποιος κακόβουλος χρήστης θα μπορούσε να βρεθεί ανάμεσα στην κινητή συσκευή του χρήστη και το τερματικό, κάνοντας και τους δύο να πιστεύουν ότι επικοινωνούν μεταξύ τους (man-in-the-middle attack).

Video 6.1.mp4	Βίντεο (video)
Απειλές εναντίον της τεχνολογίας NFC	

5.2.6 Κρίσιμοι παράγοντες σχετικές με τις επιθέσεις

Οι παρακάτω είναι οι πιο κρίσιμοι παράγοντες που μπορούν να καθορίσουν αν μια επίθεση μπορεί να επιτύχει.

1. Οι εξειδικευμένες γνώσεις του δράστη
2. Ο διαθέσιμος χρόνος για την επίθεση
3. Η πιθανή ευκαιρία για επίθεση
4. Η γνώση του συστήματος που θα γίνει η επίθεση
5. Ο διαθέσιμος τεχνολογικός εξοπλισμός

5.3 Τεχνολογία υπολογιστικού νέφους για κινητές πληρωμές

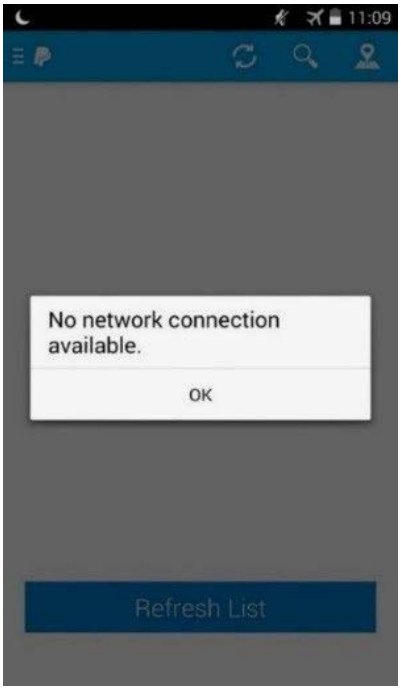
Η φράση αποθήκευση «στο νέφος» (in the cloud) είναι μια φράση που συζητιέται τεχνολογικά και που έχει βρει τον δρόμο της στον κόσμο των πληρωμών. Τα δεδομένα του χρήστη είναι πλέον άμεσα διαθέσιμα από οποιαδήποτε συνδεδεμένη συσκευή στο Διαδίκτυο, για να χρησιμοποιηθεί όταν και όπου τα χρειαστεί. Χάρη στις κινητές συσκευές ο κάθε χρήστης θα μπορεί να έχει πρόσβαση στα χρήματά του από οπουδήποτε (Allums, 2014). Θεωρείται από πολλούς η πιο αποτελεσματική τεχνολογία για το κινητό οικοσύστημα και αυτό διότι απαιτεί πολύ λιγότερη χρήση υποδομών σε σχέση με την τεχνολογία NFC. Σε ένα απλό παράδειγμα χρήσης, ο χρήστης καταχωρεί την κάρτα του σε ένα διαδικτυακό σύστημα όπως είναι το PayPal και οι πληρωμές γίνονται μέσω της κινητής συσκευής του. Τα ευαίσθητα δεδομένα της κάρτας δεν αποθηκεύονται στην κινητή συσκευή και η συναλλαγή διενεργείται διαβάζοντας κάποιο είδος κωδικού όπως είναι το barcode ή το QR code που βρίσκεται σε ένα εμφανές σημείο στην οθόνη, αφού πρώτα ο χρήστης έχει ανοίξει τη σχετική εφαρμογή. Στην εικόνα 8 βλέπουμε ένα παράδειγμα QR code.

Εικόνα 6.8. *Παράδειγμα QR Code*

Τα οφέλη της τεχνολογίας cloud είναι κυρίως η εύκολη υλοποίηση σε σχέση με την NFC. Για παράδειγμα μια νέα εταιρεία που θέλει να χρησιμοποιήσει αυτήν την τεχνολογία θα πρέπει μόνο να συμβαδίζει με τους κανόνες ιδιωτικότητας των εταιρειών πιστωτικών και χρεωστικών καρτών. Ακόμη, ο χρήστης δε θα χρειαστεί να αναβαθμίσει τη συσκευή του, όπως στην τεχνολογία NFC και αυτό διότι σχεδόν όλες οι συσκευές τελευταίας τεχνολογίας μπορούν να υποστηρίξουν την προβολή ενός barcode η QR code στην οθόνη τους.

Υπάρχουν βεβαίως και προκλήσεις: ένα ζήτημα είναι η αποθήκευση των πληροφοριών της κάρτας στον εξυπηρετητή της εταιρείας που παρέχει την υπηρεσία νέφους. Η εταιρεία θα πρέπει να παρέχει τη σχετική κρυπτογραφία για την προστασία αυτών των δεδομένων και επίσης το κανάλι επικοινωνίας θα πρέπει να είναι ασφαλές. Ένας άλλος παράγοντας είναι ο κωδικός PIN που είναι απαραίτητος από τη μεριά του χρήστη για να μπορέσει να γίνει η πληρωμή και το πρόβλημα σε αυτήν την περίπτωση είναι ο ίδιος ο κωδικός που επιλέγει ο χρήστης: με τους πιο διαδεδομένους κωδικούς να είναι ο 1234 και 0000, είναι δεδομένο ότι κάποιος κακόβουλος χρήστης μπορεί να τους μαντέψει εύκολα.

Άλλο σημείο προσοχής είναι ότι έχει μεγαλύτερες χρεώσεις ανά συναλλαγή για τις επιχειρήσεις, και αυτό βέβαια είναι ένας ανασταλτικός παράγοντας για την υιοθέτηση της τεχνολογίας νέφους στη διεκπεραίωση κινητών πληρωμών. Οι μεγαλύτερες χρεώσεις προκύπτουν από το ότι οι πληρωμές θεωρούνται (και είναι) συναλλαγές χωρίς την παρουσία της κάρτας (card-not-present transactions) και συνεπώς υψηλότερου ρίσκου σε σχέση με τις συναλλαγές με παρούσα την κάρτα (card-present transactions), στην οποία κατηγορία εμπίπτουν οι NFC πληρωμές. Τέλος, πρέπει να αναφερθεί το σημαντικό ζήτημα διαθεσιμότητας: σε περίπτωση που δεν υπάρχει ασύρματη πρόσβαση στο Διαδίκτυο δεν υπάρχει η δυνατότητα χρήσης της υπηρεσίας νέφους, και άρα και της ολοκλήρωσης της πληρωμής. Σε αυτήν την περίπτωση ο χρήστης θα έρθει αντιμέτωπος με ένα μήνυμα παρόμοιο με αυτό της εικόνας 6.9.



Εικόνα 6.9. *Μη εύρεση δικτύου*

5.4 Τεχνολογία κλειστού βρόχου

Στην τεχνολογία κλειστού βρόχου (closed loop) ο χρήστης αγοράζει από ένα συγκεκριμένο κατάστημα ένα ποσό, το οποίο αποθηκεύεται στη σχετική εφαρμογή της κινητής συσκευής του. Αργότερα όταν θελήσει να πραγματοποιήσει κάποια αγορά τότε ανοίγει την εφαρμογή αυτή και με τη χρήση πχ. της τεχνολογίας QR code το κατάστημα διαβάζει από την οθόνη του χρήστη τον κωδικό που αντιστοιχεί στο ποσό της αγοράς. Ο χρήστης μπορεί να ανεβάζει το ποσό εφόσον το επιθυμεί και όσες φορές επιθυμεί (Allums, 2014).

Τα συστήματα κλειστού βρόχου είναι πολύ εύκολα στην υλοποίηση τους και αυτό συμβαίνει κυρίως διότι η τεχνολογία είναι διαθέσιμη και θέματα σοβαρά ασφάλειας δεν υπάρχουν, αν δε συνδεθούν με κάποιο λογαριασμό κάρτας ώστε να τίθενται θεωρητικά σε κίνδυνο μεγαλύτερα χρηματικά ποσά. Επίσης, οι περισσότεροι έμποροι/επιχειρήσεις δεν έχουν πρόβλημα σχετικά με την υλοποίηση, λόγω της ευκολίας και του χαμηλού κόστους. Στις προκλήσεις: για πιο εύχρηστη λειτουργία (έλεγχος υπολοίπου πριν τη συναλλαγή) χρειάζεται διαθεσιμότητα ασύρματης σύνδεσης. Και βέβαια σημαντικός περιορισμός είναι ακριβώς η κλειστότητα αυτών των συστημάτων: κάθε ένα από αυτά μπορεί να χρησιμοποιηθεί μόνο για πληρωμές που αφορούν συγκεκριμένη συνεργαζόμενη επιχείρηση.

5.5 Παράγοντες υιοθέτησης των κινητών πληρωμών

Σε αυτό το σημείο, είναι σημαντικό να σημειώσουμε τους παράγοντες που αποδεδειγμένα επηρεάζουν την υιοθέτηση των συστημάτων κινητών πληρωμών (Tan et al., 2014), είναι:

- Η προσλαμβανόμενη χρησιμότητα που σχετίζεται με έννοιες όπως η παραγωγικότητα, η αποτελεσματικότητα και η ευκολία που προκύπτουν από τη χρήση.
- Η προσλαμβανόμενη ευκολία χρήσης που αφορά την εκμάθηση, την απομνημόνευση, την προσπάθεια και τις ικανότητες που απαιτούνται.
- Η κοινωνική επιρροή που σχετίζεται με τη σύσταση χρήσης από την οικογένεια, συγγενείς, συνεργάτες, μέλη της κοινότητας, δημοφιλείς προσωπικότητες και την απήχηση στην κοινωνική αποδοχή και καταξίωση (status).
- Η προσωπική έφεση στην καινοτομία και τη χρήση τεχνολογίας που σχετίζεται με την περιέργεια, τον πειραματισμό και την ανάληψη ρίσκου του χρήστη.

Άλλοι παράγοντες που ενδεχομένως να επηρεάζουν αλλά απαιτούν περεταίρω διερεύνηση είναι:

- Το προσλαμβανόμενο οικονομικό κόστος που περιλαμβάνει το κόστος της συσκευής, το ετήσιο κόστος χρήσης και το κόστος ανά συναλλαγή.
- Ο προσλαμβανόμενος κίνδυνος από τη χρήση που σχετίζεται με μη εγκεκριμένη χρήση, υποκλοπή πληροφορίας, και υποκλοπή τραπεζικών δεδομένων του χρήστη.
- Γενικότερα δημογραφικά στοιχεία των χρηστών, όπως ηλικία, οικονομική κατάσταση, επίπεδο εκπαίδευσης κλπ.

5.5.1 Δημιουργία εμπιστοσύνης

Η δημιουργία εμπιστοσύνης (trust) είναι ίσως ο πιο καθοριστικός παράγοντας για το αν ο τελικός χρήστης θα αποφασίσει να χρησιμοποιήσει ένα σύστημα κινητών πληρωμών. Βέβαια υπάρχουν ανασταλτικοί παράγοντες στη υιοθέτηση συστημάτων κινητών πληρωμών: ένας σημαντικός είναι ότι το υπάρχον σύστημα με τη χρήση κάρτας δουλεύει αρκετά καλά. Άλλο σημείο προβληματισμού είναι, ειδικά στην περίπτωση της NFC τεχνολογίας, ότι θα πρέπει οι περισσότεροι χρήστες να προμηθευτούν συμβατές κινητές συσκευές. Και μάλιστα θα πρέπει να υπάρχει και επαγγελματική συνεργασία ανάμεσα στον πάροχο κινητής τηλεφωνίας και την τράπεζα του χρήστη. Από την άλλη μεριά: ακόμη και αν ένας χρήστης έχει μια NFC συμβατή συσκευή και είναι σε ισχύ συμφωνία συνεργασίας παρόχου και τράπεζας, αυτό δε σημαίνει αυτόματα ότι ο έμπορος που ο χρήστης θέλει να κάνει μια συναλλαγή, έχει υιοθετήσει την τεχνολογία αυτή. Αυτό δυσκολεύει έναν χρήστη να ξεκινήσει να χρησιμοποιεί την κινητή συσκευή του ως μέσο πληρωμής (Allums, 2014).

Ένας άλλος σημαντικός παράγοντας που έχει άμεση σχέση με την εμπιστοσύνη του χρήστη είναι το πώς θα μπορέσει μια νέα μέθοδος πληρωμής να πείσει τον χρήστη ότι η διαδικασία είναι εύκολη και ικανοποιητική και να τον πείσει στο ότι αυτή η καινούρια μέθοδος είναι καλύτερη. Είναι αξιοσημείωτο το ότι ενώ οι πληρωμές μέσω κινητών συσκευών είναι πιο ασφαλείς σε σχέση με τη χρήση πιστωτικής ή χρεωστικής κάρτας, ο μέσος χρήστης δυσκολεύεται να το πιστέψει.

Οι χρήστες όταν συναλλάσσονται, ανησυχούν για τη χρήση, την απώλεια και την υποκλοπή δεδομένων τους. Οι ανησυχίες τους είναι πολύπλευρες και εμπίπτουν στις ακόλουθες κατηγορίες θεμάτων εμπιστοσύνης (Allums, 2014): α) η ασφάλεια των πληροφοριών των τραπεζικών λογαριασμών τους, β) η έκθεση των προσωπικών δεδομένων τους (και τα συνακόλουθα ζητήματα πλαστοπροσωπίας αλλά και ιδιωτικότητας), γ) ο έλεγχος πότε και πώς γίνονται οι πληρωμές και δ) το ενδεχόμενο κλοπής ή απώλειας της κινητής τους συσκευής (με ότι αυτό συνεπάγεται). Ένα κινητό σύστημα πληρωμών για να κερδίσει την εμπιστοσύνη των χρηστών θα πρέπει να ακολουθήσει μια ολιστική προσέγγιση στα ζητήματα αυτά, προσπαθώντας να καλύψει τις ανησυχίες και των τεσσάρων κατηγοριών.

6. Συμπεράσματα

Το κεφάλαιο αυτό παρέχει μια επισκόπηση σημαντικών εννοιών και τεχνολογιών που αφορούν το κινητό ηλεκτρονικό εμπόριο. Και βεβαίως αξίζει να σημειωθεί ότι οι προοπτικές εξέλιξης των μεγεθών του είναι πολύ ελπιδοφόρες. Το κινητό ηλεκτρονικό εμπόριο μπορεί πράγματι να αναπτυχθεί ακόμη περισσότερο εάν συνεχιστεί η διαρκής διαδικασία διερεύνησης ζητημάτων όπως η σχεδίαση πιο κατάλληλων μορφών αλληλεπίδρασης του χρήστη με την κινητή συσκευή, η αξιοποίηση με καινοτόμους τρόπους της επίγνωσης θέσης και επίγνωσης πλαισίου που προσφέρουν οι κινητές συσκευές και η διασφάλιση των συναλλαγών/κινητών πληρωμών αλλά και της προστασίας της ιδιωτικότητας των κινητών χρηστών, παραγόντων που βοηθούν τη δημιουργία κλίματος εμπιστοσύνης.

Βιβλιογραφία / Αναφορές

Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., & Steggles, P. (1999). Towards a better understanding of context and context-awareness. In *Handheld and ubiquitous computing* (pp. 304-307). Springer Berlin Heidelberg.

Alliance, S. C. (2011). The Mobile Payments and NFC Landscape: A US Perspective. Smart Card Alliance.

Allums, S. (2014). *Designing Mobile Payment Experiences: Principles and Best Practices for Mobile Commerce*. O'Reilly Media, Inc.

Biancalana, C., Gasparetti, F., Micarelli, A., & Sansonetti, G. (2013). An approach to social recommendation for context-aware mobile services. *ACM Transactions on Intelligent Systems and Technology* (TIST), 4(1), 10.

Chittaro, L. (2010). Distinctive aspects of mobile interaction and their implications for the design of multimodal interfaces. *Journal on Multimodal User Interfaces* 3, 3, 157-165.

Chittaro, L. (2011). Designing visual user interfaces for mobile applications. In *Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems* (pp. 331-332). ACM.

Coskun, V., Ozdenizci, B., & Ok, K. (2013). A survey on near field communication (NFC) technology. *Wireless personal communications*, 71(3), 2259-2294.

Coskun, V., Kerem, O., & Ozdenizci, B. (2013b). *Professional NFC Application Development for Android, 1st ed*. Birmingham, UK: Wrox Press Ltd.

Dahlberg, T., Mallat, N., Ondrus, J., & Zmijewska, A. (2008). Past, present and future of mobile payments research: A literature review. *Electronic Commerce Research and Applications*, 7(2), 165-181.

e Silva, A. D. S. (2013). Location-aware mobile technologies: Historical, social and spatial approaches. *Mobile Media & Communication*, 1(1), 116-121.

Georgiadis, C. K. (2010). Developing Personalized Information Services for Mobile Commerce Location-Aware Applications. *International Journal on Advances in Internet Technology,* Volume 3, Number 3 & 4.

Google (2015). Google Wallet. Δικτυακός τόπος, διεύθυνση https://www.google.com/wallet

Jehl, S. & Marcotte, E. (2014). *Responsible Responsive Design.* A Book Apart.

Kjeldskov, J. (2013). Mobile Computing. In: Soegaard, M. and Dam, R.F. (eds.). *The Encyclopedia of Human-Computer Interaction, 2nd Ed*.

Longini, A. & Gâza, M. (2013). Mobile Payments 2013 - Changing checkout. Report, Innopay BV.

Manzoor, A., Truong, H. L., & Dustdar, S. (2014). Quality of context: models and applications for context-aware systems in pervasive environments. *The Knowledge Engineering Review*, 29(02), 154-170.

Memon, A. M., Pollack, M. E., & Soffa, M. L. (2001). Hierarchical GUI test case generation using automated planning. *Software Engineering, IEEE Transactions on*, 27(2), 144-155.

Moloney, M. (2014). State of the Art for Near Field Communication: Security and Privacy Within the Field. *Security, Privacy, Trust, and Resource Management in Mobile and Wireless Communications*, IGI.

Nokia Forum, (2011). Introduction to NFC, Version 1.0, White Paper.

Nelson, D., Qiao, M., & Carpenter, A. (2013). Security of the Near Field Communication Protocol: Overview, *Journal of Computing Sciences in Colleges*, ACM, Volume 29, Issue 2.

Perera, C., Zaslavsky, A., Christen, P., & Georgakopoulos, D. (2014). Context aware computing for the internet of things: A survey. *Communications Surveys & Tutorials*, IEEE, 16(1), 414-454.

Sharp, H., Rogers, Y., Preece, J.J. (2007). *Interaction Design: Beyond Human-Computer Interaction*, John Wiley and Sons

Tan, G. W. H., Ooi, K. B., Chong, S. C., & Hew, T. S. (2014). NFC mobile credit card: the next frontier of mobile payment?, *Telematics and Informatics*, 31(2), 292-307.

Vermaas, R., Tervonen, T., Zhang, Y., & Siljee, J. (2013). The Security Risks of Mobile Payment Applications Using Near-Field Communication. Master Thesis, Erasmus University of Rotterdam.

Wroblewski, L. (2011). *Mobile First*. A Book Apart.

Χρήσιμοι δικτυακοί τόποι:

https://developer.android.com/design/index.html
https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/index.html
https://dev.windows.com/en-us/design
http://timkadlec.com/2012/04/media-query-asset-downloading-results

Quiz6.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Κριτήριο αξιολόγησης 1

[*] Απαραίτητα στοιχεία σε μια συναλλαγή κινητού εμπορίου είναι:

Α) Η χρήση κινητών συσκευών όπως Smartphones ή Tablets

Β) Η χρήση ασύρματων δικτύων

Γ) Η χρήση της τεχνολογίας NFC

Δ) Όλα τα παραπάνω

Ε) Ο συνδυασμός των α) και β)

ΣΤ) Ο συνδυασμός των β) και γ)

Απάντηση/Λύση

Ε) Ο συνδυασμός των Α) και Β)

Κριτήριο αξιολόγησης 2

[*] Η τεχνολογία προσαρμοστικών ιστότοπων (Responsive Web) μετατρέπει αυτόματα τον κώδικα μίας ιστοσελίδας έτσι ώστε αυτή να είναι συμβατή με το λειτουργικό σύστημα της κινητής συσκευής

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 3

[*] Το κινητό ηλεκτρονικό εμπόριο επωφελείται από:

Α) Την τεχνολογία της εξατομίκευσης

Β) Την επίγνωση θέσης

Γ) Την επίγνωση της περιβάλλουσας κατάστασης

Δ) Όλα τα παραπάνω

Απάντηση/Λύση

Δ) Όλα τα παραπάνω

Κριτήριο αξιολόγησης 4

[*] Ποιο από τα παρακάτω δεν αποτελεί κριτήριο ποιότητα του πλαισίου (context) μιας κινητής εφαρμογής:

Α) Η επικαιρότητα των πληροφοριών

Β) Η χρησιμότητα των πληροφοριών

Γ) Ο όγκος των πληροφοριών

Δ) Η αξιοπιστία των πληροφοριών

Απάντηση/Λύση

Γ) Ο όγκος των πληροφοριών

Κριτήριο αξιολόγησης 5

[*] Οι εξ αποστάσεως κινητές πληρωμές στηρίζονται:

Α) στην τεχνολογία NFC

Β) στις τεχνολογίες ασφάλειας των προγραμμάτων περιήγησης

Γ) στην τεχνολογία των SMS/MMS

Δ) σε εξειδικευμένες κινητές εφαρμογές

Ε) σε όλα τα παραπάνω

ΣΤ) στα Α), Β) και Γ)

Ζ) στα Β), Γ) και Δ)

Απάντηση/Λύση

Ζ) στα Β), Γ) και Δ)

Κριτήριο αξιολόγησης 6

[*] Η τεχνολογία NFC χρησιμοποιείται για:

Α) Μεταφορά δεδομένων σε κοντινή απόσταση

Β) Σύνδεση στο Διαδίκτυο

Γ) Όλα τα παραπάνω

Απάντηση/Λύση

Α) Μεταφορά δεδομένων σε κοντινή απόσταση

Κριτήριο αξιολόγησης 7

[*] Ένα σύστημα κινητών πληρωμών μπορεί να χρησιμοποιεί:

Α) Μόνο NFC

Β) Μόνο Cloud

Γ) Οποιαδήποτε από τις λύσεις Α) ή Β)

Απάντηση/Λύση
Γ) Οποιαδήποτε από τις λύσεις Α) ή Β)

Κριτήριο αξιολόγησης 8
[**] Υπάρχουν οι εξής ακόλουθοι τρεις τρόποι για να αναπτυχθούν ιστοσελίδες για κινητές συσκευές:
Α) σμίκρυνση (ή μεγέθυνση αν χρειαστεί) ιστοσελίδων του παραδοσιακού Ιστού, ανάπτυξη νέων ιστοσελίδων βασισμένων στο κινητό λειτουργικό σύστημα, και σχεδίαση προοδευτικών ιστοσελίδων

Β) σμίκρυνση ιστοσελίδων του παραδοσιακού Ιστού, ανάπτυξη νέων ιστοσελίδων ειδικά για κινητές συσκευές και σχεδίαση προσαρμοστικών ιστοσελίδων

Γ) αξιοποίηση εγγενών κινητών εφαρμογών, σχεδίαση προσαρμοστικών ιστοσελίδων και επεξεργασία ιστοσελίδων του παραδοσιακού Ιστού

Απάντηση/Λύση
Β) σμίκρυνση ιστοσελίδων του παραδοσιακού Ιστού, ανάπτυξη νέων ιστοσελίδων ειδικά για κινητές συσκευές και σχεδίαση προσαρμοστικών ιστοσελίδων

Κριτήριο αξιολόγησης 9
[**] Ποια από τα παρακάτω αποτελούν χαρακτηριστικά μιας γραφικής διεπαφής χρήστη;

Α) Η ιεραρχική δομή των γραφικών συστατικών της

Β) Η παραγωγή στοχαστικής γραφικής εξόδου

Γ) Η δενδροειδής δομή των γραφικών συστατικών της

Δ) Ο γραφικός προσανατολισμός της

Ε) Όλα τα παραπάνω

ΣΤ) Το Α) και Β)

Ζ) Το Α) και Δ)

Η) Το Γ) και Δ)

Απάντηση/Λύση
Ζ) Το Α) και Δ)

Κριτήριο αξιολόγησης 10
[**] Συγκρίνοντας τα συστήματα κινητών πληρωμών, ποιο από τα παρακάτω ισχύει;
Α) Τα συστήματα βασισμένα στην NFC τεχνολογία είναι πιο εύκολα υλοποιήσιμα αλλά με χειρότερη ασφάλεια σε σχέση με αυτά που βασίζονται στο υπολογιστικό νέφος

Β) Τα πιο ασφαλή αλλά και πιο ακριβά είναι τα συστήματα κλειστού βρόχου

Γ) Τα συστήματα που βασίζονται στο υπολογιστικό νέφος είναι μεγαλύτερου κόστους αλλά ευκολότερα στην υλοποίηση από τα συστήματα τα βασισμένα στην NFC τεχνολογία

Δ) Κανένα δεν ισχύει

Απάντηση/Λύση

Γ) Τα συστήματα που βασίζονται στο υπολογιστικό νέφος είναι μεγαλύτερου κόστους αλλά ευκολότερα στην υλοποίηση από τα συστήματα τα βασισμένα στην NFC τεχνολογία

Κεφάλαιο 7: Ασφαλείς Υπηρεσίες και Συναλλαγές σε Περιβάλλοντα Κινητού Εμπορίου

Σύνοψη
Το κεφάλαιο αυτό αρχικά ασχολείται με τις απειλές που καλείται να αντιμετωπίσει ο χρήστης μίας έξυπνης συσκευής σε περιβάλλοντα κινητού εμπορίου. Παρουσιάζεται μια κατηγοριοποίηση των απειλών και επιθέσεων και αναλύεται η συμπεριφορά του κακόβουλου και ανεπιθύμητου λογισμικού στις κινητές συσκευές. Επεξηγείται η λειτουργία των συστημάτων ανίχνευσης εισβολών σε έξυπνες συσκευές, ενώ ιδιαίτερη συζήτηση γίνεται στα στοιχεία ασφάλειας του κινητού λειτουργικού συστήματος Android. Στη συνέχεια, το κεφάλαιο διερευνά ζητήματα ιδιωτικότητας και εμπιστοσύνης σε περιβάλλοντα κινητού εμπορίου. Εστιάζοντας στην επαγγελματική χρήση της κινητής συσκευής, παρουσιάζονται τα οφέλη και οι προκλήσεις που προκύπτουν. Τέλος, ένα σημαντικό τμήμα του κεφαλαίου αφορά τα ζητήματα ιδιωτικότητας και εμπιστοσύνης σε κινητά συστήματα συστάσεων. Επεξηγούνται οι κύριοι τρόποι εξόρυξης δεδομένων χρήσης των κινητών χρηστών οι οποίοι είναι ικανοί να διαφυλάξουν την ιδιωτικότητα αυτών. Το κεφάλαιο κλείνει με την αναλυτική παρουσίαση μιας προσέγγισης διαφύλαξης της ιδιωτικότητας στα κινητά συστήματα συστάσεων.

Προαπαιτούμενη γνώση
Τα κεφάλαια 2,3,4 και 6 του παρόντος συγγράμματος

1. Απειλές ασφαλείας στις έξυπνες κινητές συσκευές

Οι έξυπνες κινητές συσκευές (smartphones) αποτελούν πλέον μια πύλη προς τον εξωτερικό κόσμο, συνδυάζοντας δυνατότητες παραδοσιακής τηλεφωνίας με πλατφόρμες διασκέδασης, πλατφόρμες γεωγραφικού εντοπισμού (GPS), ηλεκτρονικές αγορές και γενικότερα πάσης φύσης συναλλαγές (μέσω προγράμματος περιήγησης αλλά και ειδικών εφαρμογών). Οι χρήστες δε χρησιμοποιούν τις κινητές συσκευές πια μόνο για τηλεφωνικές κλήσεις και γραπτά μηνύματα. Αντιθέτως, ένα πλήθος καινοτόμων δυνατοτήτων ανοίγεται μπροστά τους: διασκέδαση, τραπεζικές συναλλαγές, ηλεκτρονικές πληρωμές, αλλά και κοινωνική δικτύωση προσφέρονται στις κινητές συσκευές. Ο χρήστης έχει αντικαταστήσει την τηλεόραση, το ραδιόφωνο, τη φωτογραφική μηχανή, την κάμερα λήψης βίντεο, τον επιτραπέζιο υπολογιστή, τα ογκώδη εργασιακά και εκπαιδευτικά αρχεία αλλά και άλλα πολλά αντικείμενα και υπηρεσίες με μόνο μια κινητή συσκευή και πληθώρα εφαρμογών μέσα σε αυτή. Οι έξυπνες κινητές συσκευές έχουν σημαντικό ρόλο στην καθημερινή ζωή καθώς μέσω αυτών παρέχεται ψυχαγωγία, επικοινωνία, δικτύωση, συναλλαγές, αγορές. Έχουν κερδίσει ακόμα και τον επιχειρηματικό κόσμο, καθώς μέσω αυτών οι εργαζόμενοι οργανώνουν καλύτερα τη δουλειά τους αλλά και οι ίδιες οι επιχειρήσεις προωθούν ευκολότερα τα προϊόντα στους καταναλωτές (Juniper, 2011).

Οι εξελίξεις στον κλάδο της κινητής τηλεφωνίας προχωρούν ταχύτατα όπως αποδεικνύει η τεράστια αύξηση στον αριθμό και την ποικιλία των εφαρμογών smartphones σε αγορές όπως το Apple AppStore, το Android Market και το Amazon AppStore. Ωστόσο υπάρχει ένα σημείο το οποίο έχει σημειώσει λίγα σχετικά βήματα προόδου. Πρόκειται για τον τομέα της ασφάλειας, στον οποίο οι κινητές συσκευές υστερούν αρκετά: σε αντίθεση με τους υπολογιστές και τα laptops που είναι εδώ και χρόνια εξοπλισμένοι με λογισμικό ασφάλειας, οι κινητές συσκευές, κατά τη συντριπτική τους πλειοψηφία παραμένουν δίχως προστασία και έτσι είναι ευπαθείς σε απειλές όπως υποκλοπές, κακόβουλο λογισμικό κλπ. Η πληθώρα καινοτομιών που παρέχουν οι κινητές συσκευές και εφαρμογές έχουν δημιουργήσει όντως πολλούς κινδύνους, οι οποίοι σε πληθώρα περιπτώσεων γίνονται πραγματικότητα κοστίζοντας μάλιστα ακριβά (La Polla et al., 2013).

Λόγω της αξίας των δεδομένων που αποθηκεύονται στις κινητές συσκευές αλλά και άλλους παράγοντες, όπως το μικρό μέγεθος των συσκευών, τις καθιστούν περιζήτητους στόχους επίθεσης. Αυτές οι επιθέσεις εκμεταλλεύονται τις αδυναμίες που σχετίζονται με τις συσκευές αυτές και χρησιμοποιούν τεχνολογίες όπως SMS, MMS, δίκτυα Wi-Fi κλπ. Υπάρχουν τρεις πρωταρχικοί στόχοι για τους επιτιθέμενους: δεδομένα, ταυτότητα κάτοχου/χρήστη και διαθεσιμότητα. Οι εισβολείς, μέσα από μια επίθεση, προσπαθούν να χρησιμοποιήσουν τα δεδομένα που έχει ο χρήστης και να προχωρήσουν σε άλλες κακόβουλες ενέργειες. Οι κακόβουλες αυτές ενέργειες μπορούν να γίνουν πράξη είτε μέσω υποκλοπής της ταυτότητας του

νόμιμου χρήστη, είτε μέσω τακτικών που περιορίζουν τη δυνατότητα πρόσβασης στον νόμιμο κάτοχο της συσκευής (διαθεσιμότητα). Ο όρος ασφάλεια κινητών/έξυπνων συσκευών (mobile/smartphone security) αναφέρεται ακριβώς στις προσπάθειες προστασίας τόσο των ίδιων των συσκευών όσο και των εφαρμογών που αυτές εκτελούν.

1.1 Δεδομένα κινητών συσκευών: τι αποθηκεύουν ως περιεχόμενο οι χρήστες;

Το τι επιλέγει ο κάθε χρήστης να αποθηκεύει στη συσκευή του συνδέεται άμεσα με το προφίλ του και το είδος των δραστηριοτήτων στις οποίες συμμετέχει. Ακολουθώντας μια γενική προσέγγιση κατηγοριοποίησης, εντοπίζουμε τέσσερις κύριες ομάδες: ευαίσθητα προσωπικά δεδομένα, πολυμεσικό υλικό, διάφορους κωδικούς και εμπιστευτικά ή απόρρητα έγγραφα. Η πλειοψηφία των χρηστών συνηθίζει να αποθηκεύει στο κινητό του ευαίσθητα προσωπικά δεδομένα (όπως στοιχεία της ταυτότητάς του, διευθύνσεις, πληροφορίες ιατρικού ιστορικού), αλλά και προσωπικά δεδομένα άλλων χρηστών που αποθηκεύει σαν διαθέσιμες επαφές προς επικοινωνία στο κινητό του. Η πρώτη αυτή κατηγορία αποθηκευμένων δεδομένων υπάρχει από τις πρώτες απλές κινητές συσκευές: οι χρήστες αρχικά οδηγήθηκαν στις κινητές συσκευές με σκοπό να κάνουν τη διαδικασία της επικοινωνίας γρηγορότερη, άμεση και ευκολότερη. Για να επιτευχθεί αυτό, χρειαζόταν η δυνατότητα αποθήκευσης επαφών και προσωπικών πληροφοριών, αρχικά στη μνήμη του κινητού, και αργότερα (καθώς η τεχνολογία εξελισσόταν) σε εφαρμογές.

Έρευνες έχουν δείξει ότι περίπου οι μισοί χρήστες έχουν πολυμεσικό υλικό αποθηκευμένο στις συσκευές τους. Στην κατηγορία αυτή ανήκουν διάφορα μουσικά αρχεία, προσωπικά βίντεο, ταινίες αλλά και φωτογραφίες. Οι χρήστες αποθηκεύουν τραγούδια που προτιμούν και τα χρησιμοποιούν όχι μόνο για ψυχαγωγία αλλά και για ενδεχόμενο ήχο κλήσης. Επίσης διατηρούν βίντεο αλλά και κάποιες ταινίες ίσως, εφόσον υποστηρίζεται από τη μνήμη και τις προδιαγραφές της συσκευής, για ψυχαγωγικούς κυρίως λόγους. Όλοι όμως διατηρούν φωτογραφίες είτε δικιές τους είτε αγαπημένων προσώπων ακόμα και από ιδιαίτερες προσωπικές στιγμές.

Η τρίτη κατηγορία περιλαμβάνει προσωπικούς κωδικούς, όχι μόνο σχετικούς με τη συσκευή στην οποία αποθηκεύονται αλλά και άλλοι κωδικοί (που σχετίζονται με άλλες συσκευές, άλλες υπηρεσίες ακόμα και κωδικοί άλλων δικαιούχων ορισμένες φορές). Σε αυτή την ομάδα ανήκουν προσωπικοί κωδικοί για πιστωτικές κάρτες, για διάφορες ηλεκτρονικές υπηρεσίες, ηλεκτρονικές διευθύνσεις, κοινωνικά δίκτυα και γενικά πλατφόρμες που περιλαμβάνουν μηχανισμούς κωδικών πρόσβασης με τους οποίους αυθεντικοποιείται ο χρήστης. Περίπου ένας στους τρεις χρήστες, σύμφωνα με έρευνες, επιλέγει να αποθηκεύει τραπεζικούς κωδικούς στις κινητές συσκευές που διαχειρίζεται. Επιπρόσθετα, αν αναλογιστούμε ότι οι ηλεκτρονικές πληρωμές (είτε μέσω τραπεζικών λογαριασμών είτε μέσω άλλων ειδικά σχεδιασμένων συστημάτων κινητών πληρωμών), αποτελούν καινούρια τάση, δεν αποτελεί γεγονός προς έκπληξη το ότι οι χρήστες αποθηκεύουν κωδικούς στις συσκευές τους.

Κατηγορία τύπου δεδομένων	**Σχετικό περιεχόμενο**
Ευαίσθητα προσωπικά δεδομένα	• Αριθμός δελτίου ταυτότητας • Διεύθυνση οικίας και ηλεκτρονικού ταχυδρομείου • Ιατρικός φάκελος • Προσωπικές πληροφορίες επαφών καταλόγου
Πολυμεσικό υλικό	• Βίντεο • Φωτογραφίες • Μουσική
Κωδικοί	• Αριθμός PIN πιστωτικής/χρεωστικής κάρτας • Κωδικοί διαφόρων ηλεκτρονικών υπηρεσιών • Κωδικοί σχετικοί με λογαριασμούς κοινωνικών δικτύων
Εμπιστευτικά έγγραφα	• Προσωπικά και επαγγελματικά έγγραφα

Πίνακας 7.1 *Κατηγορίες περιεχομένου στις κινητές συσκευές*

Τελευταία κατηγορία αποτελεί η αποθήκευση εμπιστευτικών ή απόρρητων αρχείων. Οι χρήστες επιθυμούν να αποθηκεύουν αρχεία στις συσκευές τους, τόσο αρχεία προσωπικά και ποικίλης θεματολογίας, όσο και αρχεία εμπιστευτικά του εργασιακού τους περιβάλλοντος. Με την αποθήκευση τέτοιων αρχείων οι

χρήστες έχουν τη δυνατότητα να είναι «εξοπλισμένοι» ανά πάσα στιγμή και οπουδήποτε, αφού μαζί με την προσωπική τους συσκευή έχουν και ό,τι έγγραφο μπορεί να τους φανεί χρήσιμο τόσο σε προσωπικό επίπεδο όσο και σε εργασιακό. Τα τελευταία χρόνια όλο και περισσότερο αυξάνεται η τάση να αποθηκεύουν οι χρήστες δεδομένα εργασιακού χαρακτήρα στις κινητές συσκευές τους (Dimensional Research, 2013). Οι τέσσερις παραπάνω κατηγορίες μαζί με το σχετικό περιεχόμενο της καθεμιάς μπορεί να οργανωθεί και να παρουσιαστεί στον πίνακα 7.1.

1.2 Είδη παραβιάσεων και τύποι επιθέσεων στις κινητές συσκευές

Μια ενδιαφέρουσα κατηγοριοποίηση των απειλών και επιθέσεων στις κινητές συσκευές διακρίνει οκτώ επιμέρους κατηγορίες (MSRA, 2013): α) φυσικές απειλές (physical threats), πχ. απώλεια συσκευής, β) απειλές βάσει λογισμικού (software-based threats), πχ. κινητός ιός, γ) απειλές βάσει του παγκόσμιου Ιστού (web-based threats), δ) εκμετάλλευση ευπαθούς κινητού λειτουργικού συστήματος (exploitation of vulnerable mobile OS), ε) απειλές βάσει δικτύου (network-based threats), στ) απειλές βάσει παρόχου υπηρεσιών (service provider-based threats), ζ) απειλές βάσει χρηστών (user-based threads), η) απειλές συσκευών για την επιχείρηση (mobile device threats to the enterprise).

Οι κυριότερες παραβιάσεις ασφαλείας σε κινητές συσκευές και εφαρμογές μπορούν για εποπτικούς λόγους να ομαδοποιηθούν σε δύο μεγάλες κατηγορίες: παραβιάσεις ασφαλείας μέσω κλοπής ή απώλειας της κινητής συσκευής και παραβιάσεις ασφαλείας από επιθέσεις κακόβουλου/ανεπιθύμητου λογισμικού τόσο στη συσκευή όσο και στις εφαρμογές που αυτή περιέχει. Σε μια τρίτη κατηγορία μπορούμε να εντάξουμε ορισμένους τύπους επιθέσεων που ενώ η εκδήλωσή τους μπορεί να συνδυαστεί είτε με την απώλεια συσκευής είτε με κακόβουλο/ανεπιθύμητο λογισμικό, μπορούν ωστόσο να συμβούν και ανεξάρτητα από την παρουσία αυτών των γεγονότων.

1.2.1 Παραβιάσεις λόγω απώλειας της συσκευής

Η φορητότητα των κινητών συσκευών επιτρέπει τη συνεχή πρόσβαση σε επιχειρηματικές και προσωπικές πληροφορίες, ανεξαρτήτως τοποθεσίας. Αυτή η φορητότητα έχει ως αποτέλεσμα πολύ συχνά την απώλεια ή την κλοπή των κινητών συσκευών, με σοβαρότατες συνέπειες. Αυτόματα τίθενται σε κίνδυνο τα προσωπικά δεδομένα του χρήστη, όπως επαφές με εικόνες και διευθύνσεις, τραπεζικοί λογαριασμοί με αυτόματη συμπλήρωση κωδικών στους browsers και στις αντίστοιχες εφαρμογές, προσωπικές φωτογραφίες, λογαριασμοί κοινωνικής δικτύωσης, emails και πολλά άλλα δεδομένα. Επίσης, επειδή οι χρήστες χρησιμοποιούν τις κινητές συσκευές για λειτουργίες που σχετίζονται με την εργασία τους, η απώλεια της συσκευής μπορεί να παρουσιάσει επιχειρηματικές επιπτώσεις, εκθέτοντας την πνευματική ιδιοκτησία, τις προσωπικές πληροφορίες πελατών και εργαζόμενων καθώς και άλλα εταιρικά περιουσιακά στοιχεία (Juniper, 2011). Όταν μια συσκευή χάνεται (ή ενδεχομένως όταν ο νόμιμος χρήστης πέσει θύμα κλοπής), πολλά θέματα ασφαλείας έρχονται στην επιφάνεια. Αρκεί να αναλογιστούμε ότι υπάρχει αρκετά μεγάλο ποσοστό κινητών χρηστών που δε χρησιμοποιούν στη συσκευή τους κάποια μέθοδο αυθεντικοποίησης χρήστη (όπως για παράδειγμα κωδικό PIN για είσοδο στο λειτουργικό σύστημα της συσκευής) κατά τη διαδικασία της ενεργοποίησης. Επιπλέον, ακόμη μεγαλύτερο ποσοστό κινητών χρηστών δε χρησιμοποιεί μεθόδους αυθεντικοποίησης με κωδικούς και σε άλλες λειτουργίες του κινητού. Οι χρήστες αποφεύγουν τη χρήση κωδικών για να επιταχύνουν τις διαδικασίες ξεκλειδώματος, ενεργοποίησης και απενεργοποίησης κατά ελάχιστα δευτερόλεπτα. Από την άλλη, προσπερνούν κινδύνους και απειλές που μπορούν να στοιχίσουν πολύ περισσότερα από λίγα δευτερόλεπτα, τόσο σε χρήματα όσο και σε άλλης μορφής απώλεια. Σαν αποτέλεσμα, η συσκευή μπορεί πολύ εύκολα να κλαπεί και να παραβιαστεί (Georgiadis et al., 2014).

Ο χρήστης όχι μόνο θα έχει χάσει μια πιθανόν ακριβή συσκευή αλλά θα έχει χάσει και όλα τα δεδομένα που διατηρούσε στη μνήμη του κινητού ή και στις εφαρμογές που αυτό είχε. Ο επίδοξος εισβολέας από την άλλη, μετά την εύκολη απόκτηση της πρόσβασης σε όλα τα τμήματα της συσκευής είναι σε θέση να προχωρήσει σε άλλες ενέργειες εις βάρος του νόμιμου χρήστη. Ο παράνομος χρήστης μπορεί να υποκλέψει προσωπικά δεδομένα, όπως στοιχεία ταυτότητας, και να υποκριθεί κάποιον άλλον ώστε να ολοκληρώσει διάφορες συναλλαγές οικονομικού περιεχομένου κυρίως. Τηλεφωνικές κλήσεις και μηνύματα μπορούν να επιβαρύνουν τον λογαριασμό του νόμιμου χρήστη. Ο εισβολέας επιπρόσθετα μπορεί και χρησιμοποιεί και τις διαθέσιμες εφαρμογές που υποστηρίζει η συσκευή μαζί με τους απαραίτητους κωδικούς. Έτσι, μπορεί εύκολα να αποκτήσει πρόσβαση σε κοινωνικά δίκτυα και προσωπικές σελίδες από τις οποίες μπορεί να αντλήσει

παράνομα πολυμεσικό υλικό από προσωπικές ή όχι στιγμές του νόμιμου χρήστη. Οι τελευταίες ενέργειες μπορούν να οδηγήσουν μέχρι και σε διαδικτυακό εκφοβισμό του νόμιμου χρήστη από τον εισβολέα. Ο διαδικτυακός εκφοβισμός έχει πολλές διαφορετικές μορφές, όμως πολύ συχνά ζητείται χρηματικό αντάλλαγμα για την παύση του εκφοβισμού.

Sound 7.1.mp3	Ηχητικό απόσπασμα (audio)
Περιγραφή του παράνομου χρήστη	

1.2.2 Παραβιάσεις λόγω κακόβουλου/ανεπιθύμητου λογισμικού

Στη δεύτερη κατηγορία παραβιάσεων ανήκουν επιθέσεις που λαμβάνουν χώρα μέσω μολυσμένων προγραμμάτων, συνδέσεων και εργαλείων. Ως κακόβουλο/ανεπιθύμητο λογισμικό θεωρείται κάθε εχθρικό και ενοχλητικό λογισμικό που είναι σχεδιασμένο για χρήση από μια συσκευή χωρίς τη συγκατάθεση του νόμιμου χρήστη ή στηριζόμενο στην 'εκμαίευση' της ανοχής του. Περιλαμβάνονται στην κατηγορία αυτή μια πληθώρα επιθέσεων, που λόγω τόσο του πλήθους τους όσο και της βαρύνουσας σημασίας που έχουν (όσον αφορά γενικότερα την επιστημονική περιοχή της ασφάλειας στις κινητές συσκευές), θα αναλυθούν με λεπτομέρειες σε επόμενη ειδική ενότητα.

1.2.3 Άλλες παραβιάσεις/απειλές

Υποκλοπή της επικοινωνίας (communication interception): είναι μια απειλή για κάθε συσκευή που συνδέεται στο Διαδίκτυο, και οι κινητές συσκευές δε θα μπορούσαν να αποτελούν εξαίρεση. Το πλεονέκτημα των έξυπνων συσκευών είναι ότι οι επικοινωνίες τους, συνήθως είναι κρυπτογραφημένες μέσω κυψελών, απαιτώντας από τους επίδοξους εισβολείς να έχουν εξειδικευμένο εξοπλισμό και εργαλεία προκειμένου να ακούσουν τις συνομιλίες μεταξύ της συσκευής και των κυψελών. Ωστόσο η κρυπτογράφηση αυτή μπορεί να σπάσει με την κατάλληλη μεθοδολογία (Juniper, 2011). Μία επιπρόσθετη απειλή υποκλοπής επικοινωνίας είναι η σύνδεση Wi-Fi των κινητών συσκευών, καθώς με την ασύρματη σύνεση δικτύου μεγαλώνει σημαντικά ο κίνδυνος να διεισδύσει και να παρακολουθήσει τη συσκευή ένας εισβολέας. Μελέτες έχουν δείξει ότι μια κινητή συσκευή που μεταβαίνει σε δίκτυο Wi-Fi, είναι επιρρεπής σε επιθέσεις Man-In-The-Middle (MITM). Η MITM είναι μια μέθοδος κατά την οποία οι εισβολείς αυτοσυστήνονται σε μία ροή επικοινωνίας. Στην ουσία ο εισβολέας, παίζει τον ρόλο του μεσολαβητή (middleman) για μία συζήτηση, καταγράφοντας όλες τις πληροφορίες μεταξύ των δύο επικοινωνούντων μερών (Juniper, 2011). Ανάλογα τώρα με το πώς μια κινητή συσκευή χειρίζεται τη μετάβαση δεδομένων, οι επικοινωνίες μεταξύ δύο μερών μπορούν να μεταδοθούν μέσω απλού κειμένου και έτσι να είναι ορατές σε έναν εισβολέα που χρησιμοποιεί MITM.

Εκμετάλλευση και κακή διαγωγή (exploitation and misconduct): είναι οι απειλές που βασίζονται στον ανθρώπινο παράγοντα (Juniper, 2011). Δύο είναι οι μεγάλες κατηγορίες κινητών χρηστών που έχουν συνήθως ασταμάτητη πρόσβαση σε φορητές συσκευές, και η οποία μπορεί να επιδεινώσει τις απειλές: οι έφηβοι και οι υπάλληλοι εταιριών που χρησιμοποιούν τις κινητές συσκευές ως μέσο για τη διεκπεραίωση της εργασίας τους. Και οι δύο αυτές κατηγορίες, για διαφορετικούς λόγους η καθεμία (ηλικία χρηστών και κρισιμότητα των δεδομένων αντίστοιχα), αποτελούν ιδανικούς στόχους εκμετάλλευσης. Για παράδειγμα, το cyber bulling αποτελεί μία τέτοια απειλή (σε σχέση με την κατηγορία των εφήβων κινητών χρηστών), βασιζόμενο στην πολύ συχνή χρήση των κινητών τηλεφώνων αλλά και των μέσων κοινωνικής δικτύωσης.

1.3 Το κακόβουλο λογισμικό στις κινητές συσκευές

Το κακόβουλο λογισμικό στις κινητές συσκευές (mobile malware), αποτελεί τη μεγαλύτερη απειλή ασφαλείας τους. Εξαπλώνεται ραγδαία εκμεταλλευόμενο τις αδυναμίες των μηχανισμών ασφάλειας και όλες οι μεγάλες πλατφόρμες σήμερα (Android, iOS, Windows Phone OS) αποτελούν στόχο του. Εμφανίστηκε για πρώτη φορά σε κινητές συσκευές το 2004, με τον ιό Cabir ο οποίος μόλυνε συσκευές που χρησιμοποιούσαν το λειτουργικό σύστημα Symbian (κυρίως στις τότε συσκευές Nokia) και μεταδιδόταν μέσω του Bluetooth. Έκτοτε το κακόβουλο λογισμικό και οι ιοί πολλαπλασιάστηκαν καθώς ακολούθησαν οι Qdial, Skulls, Pbstealer, Commwarrior (Trend, 2012).

Κατά αναλογία των κατηγοριών του κακόβουλου λογισμικού που περιγράψαμε στο 3ο Κεφάλαιο, και με βάση τα χαρακτηριστικά του και τον τρόπο λειτουργίας του στις κινητές συσκευές (αλλά και γενικότερα στα περιβάλλοντα κινητών ηλεκτρονικών συναλλαγών), διακρίνουμε πέντε επιμέρους υποκατηγορίες του κινητού κακόβουλου λογισμικού (La Polla et al., 2013):

- **Ιός** (virus): κομμάτι κώδικα που αυτό-αναπαράγεται και μολύνει άλλες εφαρμογές ή αρχεία. Ένας κινητός ιός είναι ένα κακόβουλο λογισμικό που στοχεύει κινητά τηλέφωνα ή συσκευές ασύρματης δυνατότητας (πχ. ταμπλέτες), προκαλώντας την κατάρρευση του συστήματος και την απώλεια ή διαρροή εμπιστευτικών πληροφοριών. Καθώς οι ασύρματες συσκευές έχουν γίνει όλο και πιο συχνές και έχουν αναπτυχθεί σε θέμα πολυπλοκότητας, έχει γίνει όλο και πιο δύσκολο να εξασφαλιστεί η ασφάλεια και η προστασία τους κατά τις ηλεκτρονικές επιθέσεις σε μορφή ιών ή άλλων κακόβουλων λογισμικών.
- **Σκουλήκι** (worm): μολυσμένα προγράμματα που κι αυτά με τη σειρά τους πολλαπλασιάζονται μολύνοντας άλλα προγράμματα. Ένα κινητό σκουλήκι μεταφέρεται από μία συσκευή σε μια άλλη με διάφορες τεχνολογίες μεταφοράς που εκμεταλλεύονται δικτυακές συνδέσεις (ασύρματες ή και ενσύρματες), πάντα χωρίς τη συμβολή του χρήστη. Οι κακόβουλες επιθέσεις αυτής της κατηγορίας συνήθως επικεντρώνονται στην ισοπέδωση της ασφάλειας της συσκευής και την κατανάλωση του εύρους ζώνης δικτύου (network bandwidth).
- **Δούρειος ίππος** (trojan): λογισμικό που παριστάνει ότι παρέχει κάποιες λειτουργίες αλλά αντί αυτού περιέχει ένα κακόβουλο πρόγραμμα. Πολύ συχνά, εγκαθιστά και άλλα κακόβουλα προγράμματα χωρίς να το αντιληφθεί ο χρήστης. Συνήθως, αποτέλεσμα της μόλυνσης από κινητό δούρειο ίππο είναι η εγκατάσταση κάποιας εφαρμογής στην κινητή συσκευή που επιτρέπει σε μη εξουσιοδοτημένους χρήστες να έχουν πρόσβαση στη μολυσμένη συσκευή και να τη χρησιμοποιούν για να ξεκινήσουν άλλες επιθέσεις προς άλλες συνδεδεμένες συσκευές (μέσω Διαδικτύου, μέσω Wi-Fi, κλπ.). Σε αντίθεση με τους ιούς, οι δούρειοι ίπποι δε μεταδίδονται μολύνοντας αρχεία.

Sound 7.2.mp3	Ηχητικό απόσπασμα (audio)
Περιγραφή του Δούρειου ίππου	

- **Rootkit**: λογισμικό που επιτρέπει τη συνεχή πρόσβαση σε έναν υπολογιστή με προνόμια διαχειριστή- υπερχρήστη (super user), ενώ κρύβει ενεργά την παρουσία του από τους διαχειριστές με το να ενσωματώνεται σε βασικά αρχεία του λειτουργικού συστήματος ή άλλων εφαρμογών. Τυπικά, ένας επίδοξος εισβολέας εγκαθιστά ένα κινητό rootkit σε μια συσκευή μόλις αποκτήσει πρόσβαση σε επίπεδο υπερχρήστη (είτε με την αξιοποίηση γνωστών κενών στην ασφάλεια του λειτουργικού, είτε με την απόκτηση ενός αντίστοιχου κωδικού πρόσβασης. Τα mobile rootkits κρύβουν κακόβουλες διαδικασίες, αρχεία αλλά και δούρειους ίππους. Έχουν τη δυνατότητα να απενεργοποιήσουν τοίχους προστασίας (firewalls) της εκάστοτε συσκευής αλλά ακόμα και τα ίδια τα προγράμματα εξυγίανσης (antivirus) της κινητής συσκευής.
- **Botnet**: σύνολο μολυσμένων συσκευών που δίνουν στον εισβολέα τη δυνατότητα της απομακρυσμένης διαχείρισής τους. Οι κακόβουλοι χειριστές ενός botnet είναι σε θέση να διευθύνουν τις δραστηριότητες αυτών των προσβεβλημένων συσκευών μέσα από τα κανάλια επικοινωνίας που σχηματίζονται από τα πρωτόκολλα που βασίζονται σε πρότυπα του δικτύου, όπως το Hypertext Transfer Protocol (HTTP). Αποτελούν σοβαρή απειλή με σκοπό το κέρδος μέσω επιθέσεων. Παράδειγμα τέτοιων επιθέσεων είναι η αποστολή μηνυμάτων spam, επιθέσεις τύπου άρνησης υπηρεσίας (Denial of Service, DoS), ή η συλλογή πληροφοριών που μπορούν να αξιοποιηθούν για παράνομους σκοπούς.

Για να μολύνει τις κινητές συσκευές το κακόβουλο λογισμικό, πρέπει να αποκτήσει δικαιώματα διαχειριστή. Η Apple, δεν επιτρέπει στους κατόχους συσκευών με λειτουργικό iOS να εγκαταστήσουν third party εφαρμογές που δεν έχουν υπογραφεί ψηφιακά και κατά συνέπεια υπάρχει σημαντική προστασία στο App Store. Ωστόσο, λίγο καιρό μετά την κυκλοφορία του πρώτου iPhone εμφανίστηκαν τα πρώτα εργαλεία

τα οποία καταργούν τον περιορισμό που έχει επιβάλλει η Apple και δίνοντας δικαιώματα διαχειριστή, οι χρήστες μπορούν να εγκαταστήσουν third party εφαρμογές και να επέμβουν στο λειτουργικό σύστημα. Η διαδικασία «ξεκλειδώματος» της συσκευής ονομάζεται Jailbreak (La Polla et al., 2013; Liu et al., 2014).

Κάτι αντίστοιχο ισχύει για τις συσκευές με λειτουργικό Android: η Google επιβάλει περιορισμούς στα δικαιώματα χρηστών με σκοπό να αποτρέψει την πρόσβαση του κακόβουλου λογισμικού στο λειτουργικό σύστημα. Σε αντίθεση με το App Store, το Android Market καθιστά δυνατή τη δημιουργία και διανομή εφαρμογών ακόμα και σε αρχάριους προγραμματιστές, και έτσι αποτελεί έναν ιδανικό μηχανισμό για την παροχή κακόβουλου λογισμικού σε μεγάλο αριθμό κινητών συσκευών. Το 2010 ανακαλύφθηκε μία εφαρμογή τύπου bank phising για κινητές συσκευές, ενώ την ίδια χρονιά ένας κατασκευαστής συσκευών βρέθηκε να προμηθεύει συσκευές με ενσωματωμένη SD card και προ-εγκατεστημένο σε αυτή το Mariposa botnet το οποίο μόλυνε τους υπολογιστές των χρηστών κατά την usb-σύνδεση της συσκευής με τον υπολογιστή. Η διαδικασία που δίνει τη δυνατότητα στον χρήστη να «ξεκλειδώσει» τη συσκευή του και να αποκτήσει δικαιώματα διαχειριστή ονομάζεται (και) root. Οι διαδικασίες root και Jailbreak είναι υπεύθυνες για αρκετά κενά ασφάλειας στις συσκευές τα οποία μπορούν να δημιουργήσουν πολλά προβλήματα στους χρήστες. Στους παρακάτω πίνακες, φαίνονται οι κατηγορίες και οι αντίστοιχες απειλές για τις πιο γνωστές πλατφόρμες.

Το ANDROIDOS_DROIDSMS.A αποτελεί το πρώτο trojan για Android συσκευές και ανακαλύφθηκε το 2010. Ο συγκεκριμένος δούρειος ίππος ήταν κρυμμένος σε μία εφαρμογή που ονομαζόταν Media Player και ζητούσε το δικαίωμα να στείλει SMS μηνύματα σε αριθμούς υψηλής χρέωσης χωρίς ο χρήστης να μπορεί να το αντιληφθεί (La Polla et al., 2013). Το ίδιο διάστημα ανακαλύφθηκαν δύο ακόμα κακόβουλα λογισμικά. Αρχικά ένας ακόμα δούρειος ίππος, μεταμφιεσμένος σε εφαρμογή παιχνιδιού το οποίο χρησιμοποιούσε μία εφαρμογή παρακολούθησης γεωγραφικής τοποθεσίας (GPS Spy), καθώς και ένα σκουλήκι, το Ikee, το πρώτο malware για συσκευές iOS, το οποίο και μόλυνε μόνο Jailbroken συσκευές. Το συγκεκριμένο, εκμεταλλεύεται το Secure Shell (SSH) του iOS, το πιο διαδεδομένο σημείο ευπάθειάς του. Στις δυνατότητές του περιλάμβανε και τον εντοπισμό γειτονικών Jailbroken συσκευών, στα οποία είχε τη δυνατότητα να αντιγραφεί με την προϋπόθεση να μην έχει αλλάξει ο χρήστης τις προκαθορισμένες ρυθμίσεις (default user name και password) του SSH Server. Το αποτέλεσμα της επίθεσης ήταν η αλλαγή της εικόνας φόντου συνοδευόμενη από ένα κείμενο που δήλωνε ότι η συσκευή έχει παραβιαστεί (La Polla et al., 2013) (Trend, 2012).

Το 2011 εμφανίστηκε το DroidDream, ένας δούρειος ίππος που αποτέλεσε τεράστιο πλήγμα στην αξιοπιστία του Android Market. Ενσωματωνόταν σε νόμιμες και δημοφιλείς εφαρμογές και είχε τη δυνατότητα, λειτουργώντας στο παρασκήνιο, να κλέβει προσωπικά δεδομένα και πληροφορίες της συσκευής αποστέλλοντάς τα σε ένα απομακρυσμένο διακομιστή. Μια άλλη αξιοσημείωτη παρουσία ήταν το ZeuS, ένα trojan/botnet με ειδικότερο στόχο τις συναλλαγές.

Χαρακτηριστικά παραδείγματα κακόβουλης χρήσης είναι οι εφαρμογές κλήσης που λειτουργούν στο παρασκήνιο και χρεώνουν τον συνδρομητή υπέρογκα ποσά ανάλογα με την απόσταση του καλούντος αριθμού, καθώς και οι keylogging εφαρμογές που μπορούν να θέσουν σε κίνδυνο τους κωδικούς πρόσβασης του χρήστη. Τηλεφωνικές κλήσεις και μηνύματα μπορούν βεβαίως να επιβαρύνουν τον λογαριασμό του νόμιμου χρήστη και να σημειωθεί ότι ο νόμιμος χρήστης συνήθως το συνειδητοποιεί αργά (αφού λίγοι χρήστες ελέγχουν τον λογαριασμό του κινητού τους για τυχόν ύποπτες χρεώσεις). Οι απειλές αυτές έχουν εξελιχτεί, χρησιμοποιώντας πολυμορφικές επιθέσεις (κακόβουλο λογισμικό ικανό να αλλάζει χαρακτηριστικά κατά τη διάδοσή του για να αποφύγει τον εντοπισμό) (Juniper, 2011).

1.4 Ανεπιθύμητο λογισμικό (mobile spyware και mobile grayware) στις κινητές συσκευές

Μια ιδιαιτερότητα στο κινητό περιβάλλον, είναι η (κατ' ανάγκη) μεγαλύτερη ανοχή σε λογισμικό διαφημιστικών μηνυμάτων (mobile adware) ή και παρακολούθησης δεδομένων της συσκευής (τύπου spyware), με αντάλλαγμα τη χωρίς χρέωση παροχή περιεχομένου (πχ. έκδοση παιχνιδιού που είναι μεν δωρεάν, αλλά συνεχώς εμφανίζει διαφημίσεις σε τμήματα της οθόνης). Οι εφαρμογές mobile spyware είναι αρκετά διαδεδομένες, και γίνονται επικίνδυνες όταν εμπλέκονται κακόβουλοι χρήστες. Ο ρόλος του spyware τότε είναι η παρακολούθηση των επικοινωνιών της συσκευής, συνήθως με απομακρυσμένη πρόσβαση. FlexiSpy, MobileSpy, MobiStealth είναι μερικές από τις εμπορικές εφαρμογές spyware οι οποίες είναι ιδιαίτερα αποτελεσματικές στην απόκρυψη της παρουσίας τους από τον χρήστη σε μία μολυσμένη συσκευή και μπορούν να ανιχνευθούν μόνο με ειδικά anti-spyware προϊόντα λογισμικού. Αυτό που κάνουν είναι να

επιτρέπουν στον επίδοξο εισβολέα την παρακολούθηση των αρχείων καταγραφής κλήσεων, SMS, MMS, e-mail, την τοποθεσία του χρήστη ακόμα και την υποκλοπή συνομιλιών (Juniper, 2011; Felt et al., 2011).

Ορισμένες εφαρμογές από αυτή την κατηγορία, είναι νόμιμες και συλλέγουν τα δεδομένα των χρηστών με σκοπό το marketing ή τη δημιουργία και συντήρηση προφίλ χρηστών (user profiling) με στόχο την παροχή εξατομικευμένου περιεχομένου. Όπως είδαμε και στο 3ο κεφάλαιο, κατ' αναλογία, ο όρος mobile grayware χρησιμοποιείται για το λογισμικό που κατασκοπεύει τους κινητούς χρήστες, αλλά οι εταιρείες που διανέμουν το grayware δεν το χρησιμοποιούν για να βλάψουν τους χρήστες. Μάλιστα, αρκετά λογισμικά τύπου grayware παρέχουν πραγματική λειτουργικότητα και αξία στους χρήστες.

1.5 Μεθοδολογίες επιθέσεων σε κινητές συσκευές

Κακόβουλα προγράμματα σε κινητές συσκευές μπορούν να εξαπλωθούν ποικιλοτρόπως μέσω διαφόρων φορέων, όπως μέσω SMS που περιέχει ένα σύνδεσμο σε μια μολυσμένη ιστοσελίδα (όπου ο χρήστης μπορεί να κατεβάσει τον κακόβουλο κώδικα ή την κακόβουλη εφαρμογή), μέσω MMS με μολυσμένα συνημμένα, μέσω ασύρματων μολυσμένων δικτύων στα οποία συνδέονται οι συσκευές και νοσούν κι αυτές με τη σειρά τους, μέσω της τεχνολογίας κινητών ηλεκτρονικών πληρωμών NFC (Near Field Communication), καθώς και μέσω κακόβουλου περιεχομένου που λαμβάνεται μέσω πλοήγησης στο Διαδίκτυο ή μέσω τεχνολογίας Bluetooth (Georgiadis et al., 2014).

Είναι πολλές οι επιθέσεις που προέρχονται από αδυναμίες στη διαχείριση των SMS και των MMS. Ορισμένα μοντέλα κινητών τηλεφώνων έχουν προβλήματα στη διαχείριση των σύντομων μηνυμάτων SMS. Είναι μάλιστα δυνατόν, με την αποστολή ενός κακόβουλου μηνύματος, να πραγματοποιηθεί επανεκκίνηση της λειτουργίας του τηλεφώνου, οδηγώντας σε επιθέσεις άρνησης παροχής υπηρεσίας. Μια άλλη πιθανή επίθεση θα μπορούσε να ξεκινήσει με ένα τηλέφωνο που στέλνει ένα MMS σε άλλα τηλέφωνα, το οποίο περιλαμβάνει ένα συνημμένο αρχείο, το οποίο έχει μολυνθεί με έναν ιό. Μετά την παραλαβή των MMS, ο χρήστης μπορεί να επιλέξει να ανοίξει το συνημμένο. Αν γίνει αυτό, το τηλέφωνο έχει μολυνθεί και ο ιός στέλνει ένα MMS με το ίδιο μολυσμένο συνημμένο σε όλες τις επαφές που είναι αποθηκευμένες στο βιβλίο διευθύνσεων του προσβεβλημένου χρήστη. Ένας επίδοξος εισβολέας όμως, πέραν των SMS/MMS, μπορεί να επιχειρήσει να παρακολουθήσει τις συνδέσεις μιας συσκευής μέσω ασύρματου δικτύου (Wi-Fi). Μέσω αυτής της παρακολούθησης μπορεί να αποκτήσει πληροφορίες πρόσβασης και ταυτότητας του χρήστη, τον οποίο παρακολουθεί ώστε να κερδίσει πρόσβαση στη συσκευή αλλά επίσης και για να στείλει κάποιο κακόβουλο πρόγραμμα χρησιμοποιώντας το ίδιο κανάλι επικοινωνίας που χρησιμοποιείται στη σύνδεση με το ασύρματο δίκτυο. Οι έξυπνες κινητές συσκευές είναι αρκετά ευάλωτες σε τέτοιες επιθέσεις, καθώς πολύ συχνά το Wi-Fi είναι το μόνο μέσο επικοινωνίας με το οποίο μπορούν οι χρήστες να αποκτήσουν πρόσβαση στο Διαδίκτυο.

Μια άλλη οδός είναι η ασύρματη τεχνολογία NFC, που χρησιμοποιείται για την αποστολή πληροφοριών ή πραγματοποίηση πληρωμών. Λόγω ακριβώς της διευκόλυνσης της μεταφοράς πληροφοριών που προσφέρει, είναι εκτεθειμένη σε ορισμένους κινδύνους: υπάρχει ο κίνδυνος από κακόβουλο κώδικα που λαμβάνουν οι συσκευές NFC. Ο κώδικας αυτός μπορεί να υποκλέψει διάφορες πληροφορίες από τη συσκευή NFC και να τις στείλει στη συνέχεια στον εισβολέα, ο οποίος έπειτα μπορεί να αποκτήσει πρόσβαση στη συγκεκριμένη συσκευή και να διαφθείρει ή να μεταποιήσει τα δεδομένα αυτής. Παρόμοια οδός επιθέσεων αποτελεί η τεχνολογία Bluetooth: υπάρχει σε όλες τις κινητές συσκευές και χρησιμοποιείται ευρέως από τους χρήστες για μεταφορά δεδομένων και αρχείων. Η τεχνολογία Bluetooth, επιτρέπει πχ. σε ορισμένα worms να εξαπλωθούν μεταξύ των συνδεδεμένων συσκευών, στις οποίες δεν απαιτείται αυθεντικοποίηση χρηστών, σε απόσταση 10 μέτρων.

Και βέβαια, δεν πρέπει να ξεχνάμε ότι η πλοήγηση στο Διαδίκτυο δεν είναι πάντα ασφαλής. Οι χρήστες μπορούν να ανακατευθυνθούν (δίχως να το καταλάβουν) σε ιστοσελίδες κακόβουλου περιεχομένου με προσφερόμενες μολυσμένες εφαρμογές που δελεάζουν τον χρήστη να τις χρησιμοποιήσει. Το πρόγραμμα περιήγησης που υπάρχει στις κινητές συσκευές είναι ένας ιδανικός φορέας επίθεσης για κινητές συσκευές. Η εικόνα των επιθέσεων είναι ανάλογη με αυτή που δέχονται οι υπολογιστές χρησιμοποιώντας τα κοινά προγράμματα περιήγησης στον παγκόσμιο Ιστό: σύνθετα γραφικά και πρόσθετα οδηγούν σε κλασσικές κακόβουλες επιθέσεις που σχετίζονται με το Διαδίκτυο (όπως phishing, DoS, κλπ.).

Πιο συστηματικά, οι ξεχωριστές μεθοδολογίες για την πραγματοποίηση επίθεσης σε κινητές συσκευές είναι οι εξής (La Polla et al., 2013):

- **Ασύρματες επιθέσεις** (wireless attacks): υπάρχουν πολλά και διαφορετικά είδη ασύρματων επιθέσεων. Η πιο διαδεδομένη είναι η υποκλοπή (eavesdropping) των ασύρματων μεταδόσεων για την απόσπαση εμπιστευτικών δεδομένων, όπως το όνομα χρήστη και ο αντίστοιχος κωδικός. Οι ασύρματες επιθέσεις μπορούν επίσης να καταχραστούν τα μοναδικά αναγνωριστικά υλικών διατάξεων (πχ. την ασύρματη LAN MAC διεύθυνση) για την παρακολούθηση του ιδιοκτήτη της συσκευής. Να σημειωθεί ότι το κακόβουλο λογισμικό πολλές φορές εκμεταλλεύεται το Bluetooth ως μέσο για να επιταχύνει τη διάδοσή του (πχ. το σκουλήκι Cabir).
- **Επιθέσεις διάρρηξης** (break-in attacks): επιτρέπουν στον επιτιθέμενο να αποκτήσει τον έλεγχο της συσκευής-στόχο με την εκμετάλλευση είτε προγραμματιστικών σφαλμάτων (πχ. πρόκληση υπερχείλισης ενδιάμεσης μνήμης, buffer overflow), είτε ευπαθειών στον χειρισμό αλφαριθμητικών (πχ. οι επιθέσεις XSS, που αναφέρθηκαν στο 3ο κεφάλαιο). Συνήθως, οι επιθέσεις αυτές χρησιμοποιούνται ως σκαλοπάτι για την εκτέλεση περαιτέρω επιθέσεων, όπως επιθέσεις υπερχρέωσης (overbilling) ή κλοπή δεδομένων/ταυτότητας (πχ. ο δούρειος ίππος Doomboot.A).
- **Επιθέσεις μέσω υποδομής** (infrastructure based attacks): δεδομένου ότι οι υπηρεσίες που παρέχονται από την υποδομή αποτελούν τη βάση για τις κύριες λειτουργίες στις κινητές συσκευές (όπως η κλήση και λήψη τηλεφωνημάτων, μηνυμάτων SMS και ηλεκτρονικού ταχυδρομείου), η οικονομική και κοινωνική επίπτωση των επιθέσεων αυτών μπορεί να είναι πολύ μεγάλη. Ειδικότερα:
    - Όσον αφορά τις επιθέσεις εναντίον *GPRS* δικτύων, αυτές στοχεύουν τα ευαίσθητα ως προς την ασφάλεια σημεία της GPRS τεχνολογίας, όπως η συσκευή (με την SIM κάρτα της), το δίκτυο μετάδοσης (radio access network), το δίκτυο κορμού (backbone network) και οι διασυνδέσεις των GPRS δικτύων μεταξύ τους ή με το Διαδίκτυο.
    - Και στα δίκτυα όμως *UMTS* υπάρχουν προβλήματα: αν και η αρχιτεκτονική ασφαλείας του UMTS καθορίζει ένα σύνολο διαδικασιών για την επίτευξη αυξημένης εμπιστευτικότητας και ακεραιότητας μηνυμάτων, αρκετά τρωτά σημεία υπάρχουν που μπορεί να αξιοποιηθούν από κακόβουλους που επιθυμούν να εξαπολύσουν επιθέσεις DoS. Συνήθως, ένας εισβολέας προσπαθεί να αποκτήσει πρόσβαση σε απροστάτευτα μηνύματα ελέγχου (control messages), προκειμένου να χειραγωγήσουν ειδικές διαδικασίες. Τα αναμενόμενα αποτελέσματα ποικίλουν ανάμεσα σε υπηρεσίες χαμηλότερης ποιότητας υπηρεσίας μέχρι και τελικά σε άρνηση υπηρεσίας (DoS).
- **Επιθέσεις μέσω σκουληκιού** (worm-based attacks): τα κύρια στοιχεία που χαρακτηρίζουν αυτές τις επιθέσεις είναι:
    - *Κανάλι μετάδοσης* (transmission channel): οι κινητές συσκευές είναι εξοπλισμένες με πολλούς τρόπους συνδεσιμότητας και έτσι υπάρχουν εξίσου πολλοί πιθανοί τρόποι μόλυνσης (όπως κατέβασμα μολυσμένων αρχείων καθώς ο χρήστης περιηγείται στον παγκόσμιο Ιστό, μεταφορά μέσω Bluetooth κακόβουλων αρχείων από συσκευή σε συσκευή, συγχρονισμός μιας κινητής συσκευής με έναν μολυσμένο υπολογιστή, πρόσβαση σε μια μολυσμένη κάρτα μνήμης, άνοιγμα μολυσμένων αρχείων που είναι συνημμένα σε ένα MMS μήνυμα),
    - *Παράμετροι διάδοσης* (spreading parameters): εκτός από τη μόλυνση της συσκευής, τα worms μπορούν να επιτεθούν στο ίδιο το δίκτυο επικοινωνίας. Σε αυτό το σενάριο, δεν τίθεται σε κίνδυνο μόνο η δυνατότητα των χρηστών να χρησιμοποιούν τις συσκευές τους, αλλά και η λειτουργία των δικτύων.
    - *Μοντέλα κινητικότητας χρήστη* (user mobility models): σε σύγκριση με το Διαδίκτυο, τα δίκτυα κινητής τηλεφωνίας έχουν πολύ διαφορετικά χαρακτηριστικά όσον αφορά τις τοπολογίες, τις υπηρεσίες, την παροχή, τη χωρητικότητα, τις συσκευές και τα μοτίβα επικοινωνίας. Αυτά τα στοιχεία επίσης χαρακτηρίζουν τον τρόπο που μεταδίδονται οι νέοι τύποι mobile worms: δεν απαιτούν σύνδεση στο Internet για τον πολλαπλασιασμό τους και, ως εκ τούτου, μπορούν να εξαπλωθούν χωρίς να εντοπίζονται από τα υπάρχοντα συστήματα ασφαλείας. Οπότε, τα κινητά σκουλήκια

μπορούν να μολύνουν πολλές συσκευές χρησιμοποιώντας επιθέσεις εγγύτητας (*proximity attacks*) κατά ευάλωτων συσκευών που είναι γεωγραφικά πολύ κοντά. Για τη μοντελοποίηση της διάδοσης αυτών των σκουληκιών, απαιτούνται δύο βήματα: 1) να οικοδομήσουμε ένα μοντέλο που περιγράφει ακριβώς πώς η μια συσκευή 'συναντά' μια άλλη, και 2) να κατανοήσουμε πώς ο κακόβουλος κώδικας εκμεταλλεύεται τόσο την κινητικότητα των χρηστών όσο και τις ικανότητες/χωρητικότητες (capacities) των δικτύων.

- **Επιθέσεις απομακρυσμένης διαχείρισης κινητών συσκευών** (mobile-based botnets): με την ενοποίηση των δικτύων κινητής τηλεφωνίας και Διαδικτύου, οι απειλές που μέχρι τώρα υπήρχαν μόνο στους υπολογιστές περνάνε και στις κινητές συσκευές (και αντίστροφα). Έτσι, μπορούν εύκολα οι κινητές συσκευές να μετατραπούν σε botnet client, δηλαδή (όπως εξηγήσαμε στην αντίστοιχη υποκατηγορία του κακόβουλου λογισμικού) ως μέλη ενός δικτύου υπολογιστών που ελέγχεται εξ αποστάσεως από τρίτους. Το δίκτυο διοίκησης-και-ελέγχου (command-and-control, C&C), που χρησιμοποιείται για να μεταδώσει εξ' αποστάσεως μηνύματα και εργασίες μεταξύ των ελεγχόμενων συσκευών (bots) και των ελεγκτών τους (botmasters), και αντίστροφα, μπορεί να δημιουργηθεί χρησιμοποιώντας Bluetooth, SMS μηνύματα, το Διαδίκτυο (πχ., το πρωτόκολλο HTTP), ομότιμη δικτύωση (Peer-to-Peer, P2P) ή οποιοδήποτε συνδυασμό αυτών.
- **Επιθέσεις που βασίζονται στους χρήστες** (user-based attacks): οι επιθέσεις αυτού του τύπου αναφέρονται σε μη-τεχνικής φύσεως ευπάθειες: ο στόχος είναι ο χρήστης. Προσπαθούν να τον ξεγελάσουν για να παρακάμψει τους μηχανισμούς ασφάλειας. Αυτό συμβαίνει διότι ο μέσος χρήστης δεν αντιλαμβάνεται πλήρως τη λειτουργία των μηχανισμών ασφάλειας, ούτε αντιλαμβάνεται τη σημασία των απειλών και έτσι μπορεί εύκολα να πέσει θύμα εκμετάλλευσης. Χρησιμοποιείται και ο όρος 'επιθέσεις κοινωνικής μηχανικής' (social engineering attacks). Ενδεικτική περίπτωση: η κατάχρηση των σχέσεων αξιοπιστίας, η οποία μπορεί να συμβεί όταν ένα κακόβουλο λογισμικό αποκτήσει πρόσβαση στο βιβλίο διευθύνσεων του θύματος και αποστείλει τον εαυτό του στις επαφές που εμπιστεύονται τον μολυσμένο χρήστη. Ή η περίπτωση όπου ο χρήστης δεν μπορεί να διακρίνει αν ένα χαρακτηριστικό (feature) λειτουργίας της κινητής εφαρμογής είναι μια νόμιμη λειτουργία ή μια παραπλάνηση (όπως πχ. η λήψη μέσω Bluetooth ενός μηνύματος με κακόβουλο περιεχόμενο.

Video 7.1.mp4	Βίντεο (video)
Μεθοδολογίες επιθέσεων σε κινητές συσκευές	

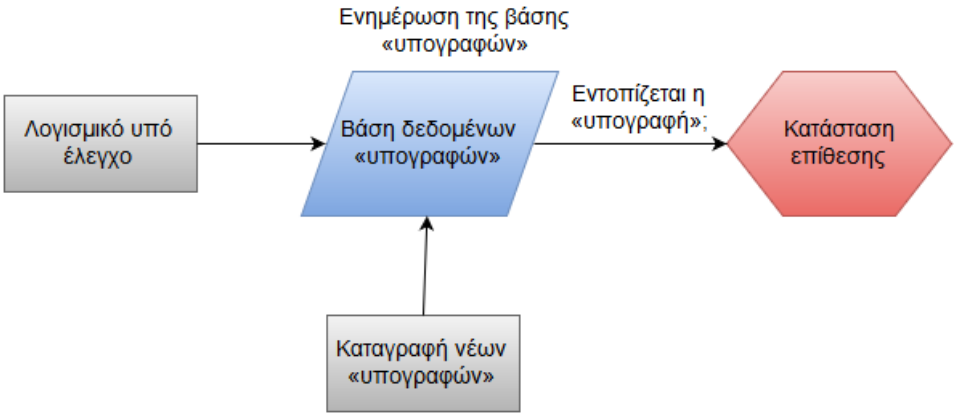
1.6 Συστήματα ανίχνευσης εισβολών

Από τη στιγμή που οι έξυπνες κινητές συσκευές είναι συσκευές που αλλάζουν τη θέση του δικτύου τους, και βεβαίως ανάλογα με την κίνηση των χρηστών τους σε διάφορους τόπους, ως προς την ανίχνευση εισβολών μπορούν να χρησιμοποιηθούν τεχνικές συστημάτων ανίχνευσης εισβολών (Intrusion Detection Systems, IDS), και ιδιαίτερα οι εξειδικευμένες προσεγγίσεις αυτών για κινητά ad-hoc δίκτυα (Halilovic & Subasi, 2012). Στην πρώτη γραμμή άμυνας ενός κινητού δικτύου, είναι ένα τοίχος προστασίας (firewall), φυσικά εφόσον αυτός ρυθμιστεί σωστά. Αλλά υπάρχουν και άλλα μέτρα ασφαλείας, όπως οι μηχανισμοί ελέγχου προσπέλασης (πχ. με χρήση Access Control Lists ή δικαιωμάτων/εξουσιοδοτήσεων βασισμένων σε ρόλους), που ενισχύουν την πρώτη γραμμή άμυνας, σε περίπτωση αποτυχίας των firewalls. Συνήθως, η πρώτη γραμμή άμυνας (της πρόληψης) δεν είναι πάντα αποτελεσματική, για αυτό είναι απαραίτητη και μία επιπλέον γραμμή άμυνας: εκεί χρησιμοποιούνται τα συστήματα ανίχνευσης εισβολών, που θεωρούνται και μια καλή λύση για την αναγνώριση των εισβολών που εξαπάτησαν την πρώτη γραμμή άμυνας και εισήλθαν στο εσωτερικό (είτε της κινητής συσκευής, είτε του δικτύου υποστήριξης).

1.6.1 Τύποι ανίχνευσης/ανάλυσης εισβολών στις κινητές/έξυπνες συσκευές

Τα συστήματα ανίχνευσης εισβολών (IDS), είναι λογισμικό (software) ή υλικό (hardware) ή συνδυασμός αυτών. Ανιχνεύουν και αντιδρούν σε περίεργες κινήσεις στο δίκτυο ή στα διάφορα υπολογιστικά συστήματα, αλλά δεν κάνουν καμία ενέργεια για την αντιμετώπισή τους. Στην ουσία τα IDS είναι τα συστήματα που αναγνωρίζουν την εισβολή και ενημερώνουν τον διαχειριστή του δικτύου ή τον κάτοχο της κινητής συσκευής. Στην πιο απλή τους μορφή, αναλύουν τα αρχεία καταγραφής και ελέγχου (logging and audit) του συστήματος και προσπαθούν να εντοπίσουν «ίχνη» από γνωστές επιθέσεις εισβολής. Σε ορισμένες περιπτώσεις βέβαια, αυτά τα συστήματα συμβάλουν στην αντιμετώπιση κάποιας κακόβουλης απειλής, όπως με τη μη παροχή πρόσβασης σε κάποιον χρήστη, σε κάποια κινητή εφαρμογή ή σε κάποια IP διεύθυνση του δικτύου. Τα IDS για τις κινητές συσκευές χρησιμοποιούν κυρίως δύο ειδών τύπους ανίχνευσης εισβολών (La Polla et al., 2013; Halilovic & Subasi, 2012; Mitchell & Chen, 2014):

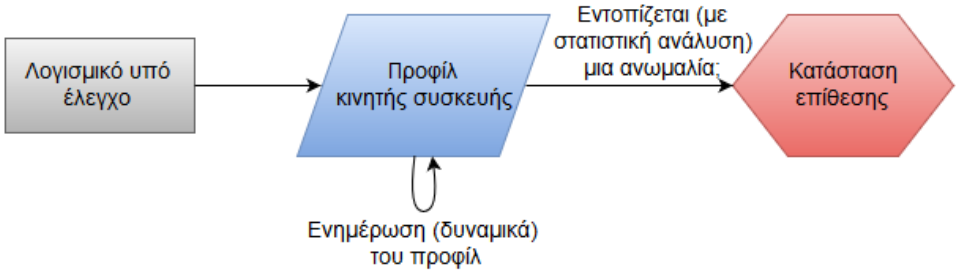
- **Ανίχνευση κακής χρήσης** (misuse-based): είναι και ο πιο συνηθισμένος τρόπος ανίχνευσης εισβολών. Άλλοι όροι που χρησιμοποιούνται για αυτή την κατηγορία είναι «ανίχνευση υπογραφής» (signature-based), «βασισμένη στη γνώση» (knowledge-based), «misuse detection», και «detection by appearance». Αναλύεται η πληροφορία που έχει συγκεντρωθεί (από τα αρχεία καταγραφής) και συγκρίνονται τα αποτελέσματα της ανάλυσης με ήδη γνωστές επιθέσεις, οι «υπογραφές» των οποίων είναι αποθηκευμένες σε μια βάση δεδομένων (Tang et al., 2014). Ανιχνεύονται λοιπόν κάποιες γνωστές υπογραφές κακόβουλων κινητών εφαρμογών, και έτσι το σύστημα γνωρίζει ότι απειλείται. Είναι μια λειτουργικότητα παρόμοια με την ανίχνευση «υπογραφών» ενός προγράμματος αντιμετώπισης ιών. Οι τεχνικές αυτές χρησιμοποιούν συνεπώς τη γνώση του τι είναι κακόβουλο για να αποφασίσουν αν μια κινητή εφαρμογή είναι επικίνδυνη ή όχι. Να σημειωθεί ότι ένας τέτοιος τρόπος ανίχνευσης δεν μπορεί να ανιχνεύσει επιθέσεις που δεν είναι γνωστές. Οι συνηθέστερες πρακτικές που ακολουθούνται είναι η μοντελοποίηση κατάστασης (state modeling), τα έμπειρα συστήματα (expert systems), το ταίριασμα αλφαριθμητικών (string matching) και η χρήση απλών κανόνων (simple rule-based). Οι προσεγγίσεις ανίχνευσης κακής χρήσης για τις έξυπνες κινητές συσκευές, μπορούν ειδικότερα να διακριθούν σε αυτές που η βάση δεδομένων των υπογραφών mobile malware παράγεται αυτόματα (automatically-defined) και σε αυτές που αυτό συμβαίνει χειροκίνητα (manual-defined), μέσω ανάλυσης του malware και εξαγωγής της υπογραφής αυτού.



Εικόνα 7.1: *Ανίχνευση κακής χρήσης*

- **Ανίχνευση ανωμαλίας** (anomaly-based): οι τεχνικές αυτές χρησιμοποιούν τη γνώση του τι αποτελεί κανονική συμπεριφορά μιας κινητής συσκευής για να αποφασίσουν αν μια κινητή εφαρμογή είναι κακόβουλη ή όχι. Στην ουσία ανιχνεύουν δραστηριότητες οι οποίες είναι ασυνήθιστες από το συνηθισμένο προφίλ. Ο τρόπος αυτός ανίχνευσης βασίζεται στο γεγονός ότι, όλες οι επιθετικές δραστηριότητες είναι ανωμαλίες και, πως, αν κάτι παρεκκλίνει από το σύνηθες προφίλ θεωρείται επίθεση (Tang et al., 2014). Άλλοι όροι που χρησιμοποιούνται για

αυτή την κατηγορία είναι «βασισμένη στη συμπεριφορά» (behavior-based) και «anomaly detection». Είναι μια λειτουργικότητα παρόμοια με την ευρετική (heuristic) λειτουργία ενός προγράμματος αντιμετώπισης ιών. Η ανίχνευση (με βάση τη στατιστική) μιας ανωμαλίας (statistical anomaly), απαιτεί την οριοθέτηση ενός συνηθισμένου προφίλ δραστηριότητας με στατιστικό τρόπο. Πιο συγκεκριμένα: ο διαχειριστής καθορίζει κάποιες παραμέτρους που αφορούν την ποσότητα και το είδος (πχ. είδη και μέγεθος πακέτων, τύποι πρωτοκόλλων, αριθμοί θυρών) της κίνησης που επιτρέπεται σε ένα δίκτυο, ή παραμέτρους που οριοθετούν την κανονική συμπεριφορά της κινητής συσκευής, ενώ το IDS αναλύει την πληροφορία που έχει συγκεντρώσει από τα αρχεία καταγραφής. Όταν οι στατιστικές της ανάλυσης παρεκκλίνουν από τα όρια (thresholds) που έχουν καθοριστεί, αυτό συνιστά ένδειξη ανωμαλίας και 'χαρακτηρίζεται' μια κατάσταση ως 'επίθεση'. Παραδείγματα ασυνήθιστων ενεργειών θα μπορούσαν να είναι: αυξημένος αριθμός αποτυχημένων αποπειρών εισόδου, αυξημένος αριθμός συνδέσεων ενός χρήστη στο σύστημα, αυξημένη κατανάλωση υπολογιστικών πόρων (πχ. μπαταρίας κινητού) ή ροής πακέτων προς ένα σύστημα, κλπ. (Μάγκος, 2013). Τα πρότυπα κανονικής συμπεριφοράς μπορούν να εισαχθούν στο IDS κατά το στάδιο ανάπτυξης του ή να πραγματοποιηθεί σταδιακή εκπαίδευση του συστήματος, αφού εγκατασταθεί. Η εκπαίδευση του συστήματος πραγματοποιείται κυρίως με μεθόδους μηχανικής μάθησης (στατιστικές μέθοδοι, κατηγοριοποίηση σε συστάδες, μετρικές θεωρίας πληροφορίας -πχ. εντροπία, δίκτυα Bayes, νευρωνικά δίκτυα, τεχνικές εξόρυξης δεδομένων, μηχανές διανυσμάτων, τεχνικές πλησιέστερου γείτονα). Αξίζει να σημειωθεί ότι τα συστήματα ανίχνευσης ανωμαλιών, λόγω του «ευρετικού» (heuristic) χαρακτήρα τους, είναι δυνατόν να καταλήγουν σε λάθος διαγνώσεις. Το ποσοστό των εσφαλμένων θετικών (false positives) ή των εσφαλμένων αρνητικών (false negatives) εξαρτάται (και) από τις ρυθμίσεις στις οποίες προβαίνει ο διαχειριστής (πχ. περισσότερη ευαισθησία = υψηλότερο ποσοστό εσφαλμένων θετικών). Τα ποσοστά αυτά, μαζί με άλλες μετρικές όπως ο χρόνος απόκρισης (response time) χρησιμοποιούνται ως μετρικές αποτελεσματικότητας των μηχανισμών αυτών ανίχνευσης. Οι προσεγγίσεις ανίχνευσης ανωμαλίας για τις έξυπνες κινητές συσκευές, μπορούν ειδικότερα να διακριθούν σε αυτές που βασίζονται σε τεχνικές μηχανικής μάθησης (machine learning techniques) και σε αυτές που παρακολουθούν την κατανάλωση ενέργειας (monitoring power consumption).

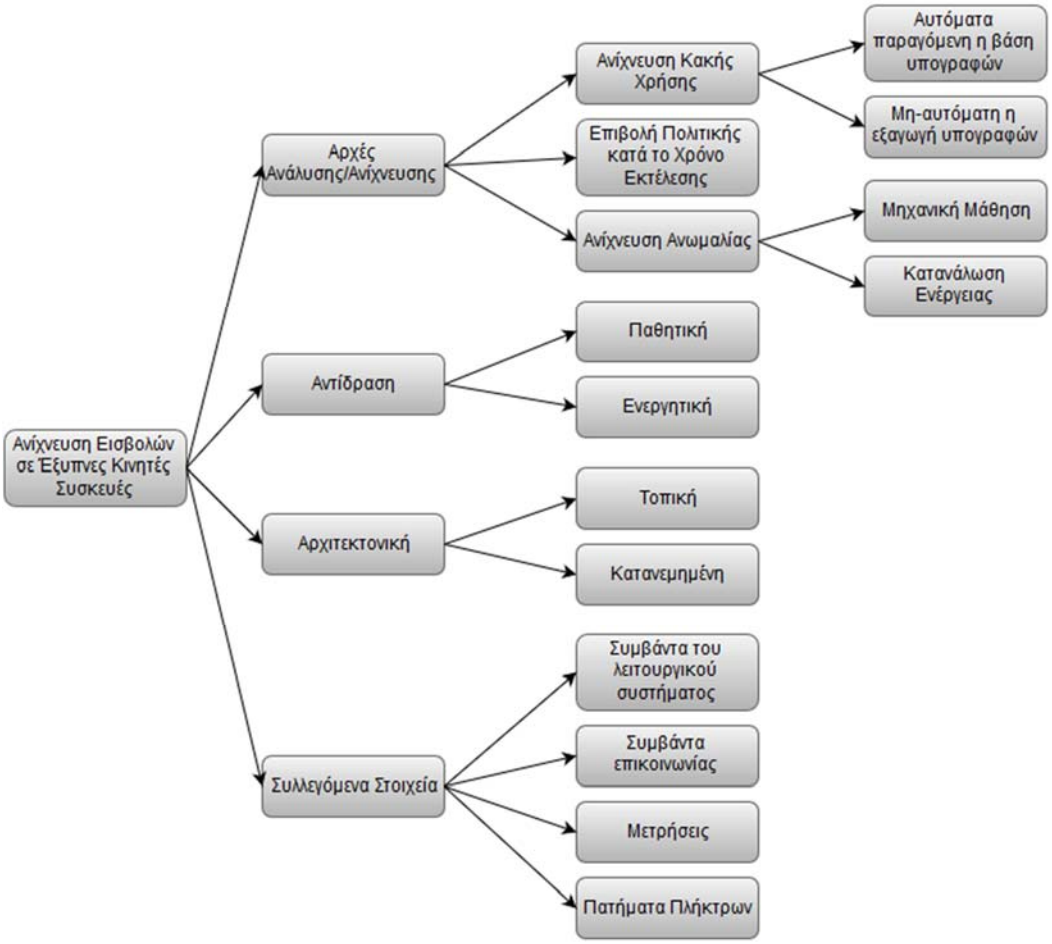


Εικόνα 7.2: *Ανίχνευση ανωμαλίας*

Υπάρχουν βεβαίως και υβριδικές προσεγγίσεις που συνδυάζουν τους δύο προηγούμενους τύπους ανίχνευσης εισβολών. Αξίζει τέλος να επισημανθεί ένας τρίτος τύπος ανίχνευσης εισβολής, που αποκαλείται «Επιβολή Πολιτικής κατά τον Χρόνο Εκτέλεσης» (run-time policy enforcement) που χρησιμοποιεί τους μηχανισμούς για τον έλεγχο των δικαιωμάτων των κινητών εφαρμογών. Η βασική ιδέα είναι ότι οι 'καταναλωτές' του κώδικα των κινητών εφαρμογών, ουσιαστικά αποδέχονται τον κώδικα 'ως έχει' και έτσι για να ανιχνεύσουν και να σταματήσουν ανωμαλίες μπορούν να αξιοποιούν τον υπάρχοντα υποστηρικτικό μηχανισμό για την επιβολή της πολιτικής που συνδέεται με τον κώδικα.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 7.1.gif	Κινούμενη εικόνα (interactive)
Κατηγορίες συστημάτων ανίχνευσης εισβολών	



Εικόνα 7.3: *Κατηγορίες συστημάτων ανίχνευσης εισβολών*

1.6.2 Πρόσθετη κατηγοριοποίηση των συστημάτων ανίχνευσης εισβολών

Εκτός από το κριτήριο των εφαρμοζόμενων αρχών ανάλυσης/ανίχνευσης, τα συστήματα ανίχνευσης εισβολών για τις κινητές συσκευές μπορούν να κατηγοριοποιηθούν και με βάση άλλα κριτήρια:

Κατηγοριοποίηση ως προς την αντίδραση (reaction).

- **Παθητική αντίδραση** (passive): συστήματα τα οποία ανιχνεύουν και καταγράφουν μια 'ύποπτη' ενέργεια, χωρίς να την αποτρέπουν. Μπορεί να ενημερώσουν μέσω ενός μηνύματος τον χρήστη συσκευής ή/και τον διαχειριστή του δικτύου, ενεργοποιώντας έναν συναγερμό, αλλά δεν προβαίνουν σε άλλες ενέργειες.
- **Ενεργητική αντίδραση** (active): συστήματα τα οποία αντιδρούν για την εξουδετέρωση του κακόβουλου λογισμικού (Mitchell & Chen, 2014). Έχουν ενσωματωμένες λειτουργίες αντιμετώπισης επιθέσεων: προσπαθούν να διακόψουν τη δικτυακή ροή/σύνδεση του εισβολέα. Πχ. χρησιμοποιούν κρυπτογραφικούς αλγόριθμους και ψηφιακές υπογραφές για να διασφαλίσουν την εμπιστευτικότητα και την ακεραιότητα των πληροφοριών της συσκευής ή ενεργοποιούν την υπηρεσία φιλτραρίσματος των προγραμμάτων προστασίας της συσκευής με τα οποία μπορεί να συνεργάζονται. Τα συστήματα ενεργητικής αντίδρασης πρέπει να λειτουργούν σε πραγματικό χρόνο, και αποκαλούνται 'Συστήματα Αποτροπής Εισβολών' (Intrusion Prevention Systems, IPS) ή και Intrusion Detection Prevention Systems (IDPS).

Κατηγοριοποίηση ως προς την αρχιτεκτονική (architecture).

- **Τοπική αρχιτεκτονική**: τόσο η φάση συλλογής στοιχείων όσο και η φάση ανάλυσης στοιχείων διενεργείται τοπικά στη συσκευή και δεν απαιτείται για την ολοκλήρωσή τους καμία αλληλεπίδραση με κάποιον εξωτερικό εξυπηρετητή (La Polla et al., 2013).

- **Κατανεμημένη αρχιτεκτονική**: η φάση συλλογής στοιχείων διενεργείται τοπικά στη συσκευή, αλλά η φάση ανάλυσης στοιχείων απαιτεί την ύπαρξη ενός εξυπηρετητή, ώστε να αποφευχθούν ζητήματα περιορισμένης διαθεσιμότητας πόρων (όπως η κατανάλωση ρεύματος, η μικρή οθόνη κλπ.).

Κατηγοριοποίηση ως προς τα στοιχεία που συλλέγονται (collected data).

- **Συμβάντα του λειτουργικού συστήματος**: τα συμβάντα αυτά αφορούν δραστηριότητες που σχετίζονται με την κανονική λειτουργία του λειτουργικού συστήματος και αφορούν κλήσεις του συστήματος (system calls), κλήσεις συναρτήσεων και λειτουργίες δικτύου.
- **Συμβάντα επικοινωνίας**: τα συμβάντα επικοινωνίας αποτελούν μια ιδιαίτερη κατηγορία συμβάντων που πραγματοποιούνται στο επίπεδο εφαρμογής μιας συσκευής. Περιλαμβάνονται λειτουργίες όπως αποστολή/λήψη SMS και MMS, και αποστολή/λήψη αρχείων.
- **Μετρήσεις**: διάφορες μετρήσεις, όπως η δραστηριότητα επεξεργαστή (CPU activity), η κατανάλωση μνήμης, η δραστηριότητα εισόδου/εξόδου στα αρχεία και στο δίκτυο (file I/O activity, network I/O activity) μπορούν να αποτελέσουν δείκτες απόδοσης μιας έξυπνης συσκευής. Υπολογισμοί, όπως πλήθος SMS που έχουν αποσταλεί, ποσότητα ελεύθερης μνήμης (RAM free), χρήση επεξεργαστή (CPU usage) αποτελούν πολύ χρήσιμα δεδομένα προς ανάλυση.
- **Πατήματα πλήκτρων** (keystrokes): κάποια συστήματα ανίχνευσης εισβολών χρησιμοποιούν τεχνικές καταγραφής του πατήματος πλήκτρων από τους χρήστες (keylogging) για να βρουν πιθανές ανεπιθύμητες ενέργειες. Πχ. ένα προφίλ χρήστη 'συνηθισμένης συμπεριφοράς' μπορεί να στηριχθεί σε μοτίβα πατήματος πλήκτρων που προέκυψαν από μακρά χρονική περίοδο (long-term keystrokes), ενώ μια τρέχουσα δραστηριότητα χρήστη καταγράφεται ως short-term keystrokes τα οποία και συγκρινόμενα με το προφίλ 'πληκτρολόγησης' μπορεί να οδηγήσουν σε ανίχνευση παράνομης χρήσης της συσκευής.

1.7 Στοιχεία ασφαλείας στο λειτουργικό σύστημα των συσκευών Android

Κάθε εφαρμογή Android εκτελείται ως ένας ξεχωριστός χρήστης και για να αλλάξει το μοντέλο αυτό λειτουργίας, πρέπει να επιχειρηθεί αλλαγή σε επίπεδο ασφάλειας. Συνεπώς οι hackers, κάνουν επιθέσεις στις κινητές εφαρμογές, λειτουργώντας 'κάτω' από τον 'μανδύα'-λογαριασμό των χρηστών, οι οποίοι είναι υπεύθυνοι, για την πρόσβαση σε αυτές κάθε φορά. Η κάθε εφαρμογή, 'τρέχει' σαν ένας ξεχωριστός λογαριασμός και δεν έχει πρόσβαση σε τίποτα άλλο παρά μόνο σε ότι απαιτείται για τη λειτουργία που εξυπηρετεί. Ο τρόπος που λειτουργεί ο συγκεκριμένος μηχανισμός ασφαλείας ονομάζεται «one-app/one-user model» και ξεκινά τη λειτουργία του από το κατώτερο στρώμα της αρχιτεκτονικής Android, τον πυρήνα (kernel). Το kernel τύπου Linux, είναι η καρδιά μιας συσκευής Android και χρησιμοποιεί τις ίδιες στη βάση τους μεθόδους ασφαλείας που προβλέπει το λειτουργικό σύστημα Linux. Δηλαδή, κάθε χρήστης έχει ένα ξεχωριστό αναγνωριστικό User ID, ο χρήστης μπορεί να ανήκει σε κάποια ομάδα χρηστών και έτσι θα έχει ένα αναγνωριστικό ομάδας Group ID, και πάνω σε αυτό το μοντέλο λειτουργίας μπορεί να εκτελέσει εντολές όπως Write (W), Read (R), και eXecute (X). Όταν κάποιος χρήστης φορτώσει δεδομένα σε μια συσκευή Android, τότε τα δεδομένα χαρακτηρίζονται δικά του, εφόσον έχει κάνει προηγουμένως σύνδεση (login) με το δικό του User ID. Σε κάθε άλλη περίπτωση, το σύστημα θα δώσει ένα νέο User ID στον χρήστη που θα χρησιμοποιήσει το σύστημα για την ίδια εφαρμογή.

Όσον αφορά την πλευρά των προγραμματιστών (developers), υπάρχουν κάποιοι τρόποι, με τους οποίους μπορούν να διασφαλίσουν τα δικαιώματα των εφαρμογών που δημιουργούν κάθε φορά. Αυτό σημαίνει ότι ο κάθε προγραμματιστής έχει μία ψηφιακή υπογραφή, ένα πιστοποιητικό (certificate), το οποίο τον ξεχωρίζει από τους υπόλοιπους. Σε θέματα ασφαλείας, οι εφαρμογές πρέπει να 'σφραγίζονται', έτσι ώστε να τρέξουν σε Android, αλλά κάτι τέτοιο δεν είναι απαραίτητα υποχρεωτικό. Για παράδειγμα ένας προγραμματιστής μπορεί μέσω του αρχείου 'AndroidManifest.xml' και της λειτουργίας sharedUserID, να δημιουργήσει πολλές εφαρμογές αλλά κάτω από το User ID του ίδιου προγραμματιστή κάθε φορά (Six, 2012). Αυτό είναι λειτουργικό, επειδή μέσω αυτής της μορφοποίησης στον xml φάκελο, κάθε προγραμματιστής μπορεί να δημιουργήσει όσες εφαρμογές θέλει χωρίς να είναι απαραίτητο να επικαλεστεί από την αρχή τα δικαιώματά του και έτσι το σύστημα κερδίζει, και σε χώρο αλλά και σε χρόνο από άποψη

λειτουργικότητας. Ένα άλλο πλεονέκτημα, είναι ότι κάθε δραστηριότητα μπορεί να τρέξει από μόνη της, μέσα στο ίδιο πακέτο δεδομένων και αυτό γίνεται αφενός μεν, για να διατηρηθεί η προέλευσή τους, δηλαδή από ποιον developer υλοποιήθηκε το πακέτο, αφετέρου δε, για να αφήσει τα στοιχεία που είναι μέρη της ίδιας εφαρμογής να τρέξουν σε διαφορετικές διεργασίες (Elenkov, 2014).

Το σύστημα ασφαλείας των συσκευών Android, ξεχωρίζει τις εφαρμογές μεταξύ τους κάτω από την ίδια διεργασία και αυτό οφείλεται στο γεγονός ότι η όλη λειτουργία του βασίζεται σε Linux πρότυπο. Οι εφαρμογές αυτές, μετά από παρέμβαση του προγραμματιστή στο σύστημα, μπορούν να είναι (Six, 2012): α) ιδιωτικές (MODE_PRIVATE), που σημαίνει ότι η εφαρμογή δεν έχει επικοινωνία με καμία άλλη εφαρμογή στο σύστημα, β) MODE_WORLD_WRITABLE, όπου επιτρέπεται οι άλλες εφαρμογές να κάνουν τροποποιήσεις σε αυτήν, και γ) MODE_WORLD_READABLE, που σημαίνει, ότι η εφαρμογή μπορεί να διαβαστεί, από τον οποιοδήποτε. Αυτός ο τρόπος λειτουργίας προσφέρει επαρκή ασφάλεια ώστε οι εφαρμογές να μην παραβιάζονται εύκολα. Η πλατφόρμα Android χρησιμοποιεί ένα μοντέλο σύμφωνα με το οποίο κάθε φορά που ο χρήστης επιθυμεί να εγκαταστήσει μία εφαρμογή, προβάλλονται οι λειτουργίες που απαιτούνται από την εφαρμογή όπως και οι πόροι του συστήματος. Το μοντέλο αυτό (install-time permission request model) έχει ως σκοπό να προστατέψει τον χρήστη από επικίνδυνες εφαρμογές που έχουν σαν σκοπό να υποκλέψουν δεδομένα και να χρησιμοποιήσουν παράνομα βασικές λειτουργίες της συσκευής. Για παράδειγμα, εάν κάποιος χρήστης αγνοήσει την προειδοποίηση του συστήματος κατά τη διάρκεια της εγκατάστασης μιας κακόβουλης εφαρμογής, τότε η παράνομη αυτή εφαρμογή μπορεί να αποκτήσει πρόσβαση στα SMS, στις κλήσεις, κλπ.

Από τη μεριά των προγραμματιστών, το 'πρόβλημα' έγκειται στο γεγονός ότι πρέπει, κατά τη διάρκεια σχεδιασμού μιας κινητής εφαρμογής, να μειώσουν όσο γίνεται τη λίστα που θα εμφανίζεται στον χρήστη, με τις κρίσιμες άδειες λειτουργίας, που πρέπει να χρησιμοποιήσει η εφαρμογή από τη συσκευή. Αυτό είναι απαραίτητο, επειδή κάθε φορά που κάποιος χρήστης εγκαθιστά μια εφαρμογή πρέπει να κάνει αποδοχή σε 10 σημεία τουλάχιστον κρίσιμων λειτουργιών, έτσι ώστε να λειτουργήσει η εφαρμογή. Υπάρχει λοιπόν ο κίνδυνος ο χρήστης μελλοντικά να νομίσει ότι κάποια παράνομη εφαρμογή είναι εξίσου ασφαλής και να ζημιωθεί παραλείποντας τις ερωτήσεις ασφαλείας (Elenkov, 2014). Πρέπει δηλαδή, να βοηθήσουν τον χρήστη να παρατηρεί όσο γίνεται (μέσω της μείωσης των αδειών χρήσης) και να μαθαίνει ποιες εφαρμογές να χρησιμοποιεί και ποιες όχι.

Υπάρχουν κάποια πρότυπα ασφαλείας τα οποία ένας προγραμματιστής μπορεί να τα επικαλεστεί κατά τη διάρκεια της λειτουργίας μιας εφαρμογής. Τα πρότυπα αυτά εμπεριέχονται στον φάκελο 'Manifest.xml' και χωρίζονται σε τρία είδη. Αρχικά, μία εφαρμογή μπορεί να χαρακτηριστεί 'Normal', όταν ο χρήστης δε χρειάζεται να κάνει επιβεβαίωση σε διάφορες φάσεις της λειτουργίας της. Για παράδειγμα, μία τέτοια εφαρμογή μπορεί να είναι η λήψη μιας φωτογραφίας και η χρήση της σαν υπόβαθρο στη συσκευή του χρήστη. Επίσης, μια εφαρμογή μπορεί να χαρακτηριστεί 'Dangerous' όταν επικαλείται κρίσιμες λειτουργίες της συσκευής, όπως τη συνδεσιμότητα του Internet, ή τη λειτουργία των κλήσεων και πρέπει με αυτό τον τρόπο, ο χρήστης να είναι σε θέση να επιλέξει, σε ποια σημεία να επιτρέψει την πρόσβασή της ή όχι μέσω των ερωτήσεων ασφαλείας. Μια άλλη περίπτωση είναι αυτή κατά την οποία ο χρήστης εγκαθιστά μία ή περισσότερες εφαρμογές από έναν developer. Σε αυτή την περίπτωση όταν για παράδειγμα ο χρήστης εγκαταστήσει μία αξιόπιστη εφαρμογή, θα υιοθετήσει και το πλάνο της ασφαλείας μέσω των πόρων που θα χρησιμοποιήσει η εφαρμογή κατά τη διάρκεια της εγκατάστασης. Τότε η εφαρμογή χαρακτηρίζεται ασφαλής και χρησιμοποιείται από τον χρήστη. Επιπλέον, ο προγραμματιστής που κατασκεύασε την εφαρμογή, καθόρισε και το πλάνο με τις λειτουργίες που πρέπει να επικαλεστεί η εφαρμογή από τη συσκευή, έτσι ώστε να λειτουργήσει. Σε περίπτωση που ο χρήστης εγκαταστήσει μία εφαρμογή από τον ίδιο προγραμματιστή, τότε αυτόματα εάν η εφαρμογή έχει κατοχυρωμένη 'υπογραφή' (signature), θα εγκατασταθεί χωρίς ιδιαίτερη δυσκολία καθώς το λειτουργικό σύστημα Android θα ξέρει ότι η εφαρμογή είναι ασφαλής από την ήδη εγκατεστημένη εφαρμογή.

Υπάρχουν πολλά συστατικά (components) που πρέπει να ληφθούν υπόψη κατά τη διάρκεια σχεδιασμού μιας κινητής εφαρμογής. Οι προγραμματιστές μαθαίνουν πρώτα από όλα το συστατικό Activity. Το συγκεκριμένο εμπεριέχεται στο στρώμα εντολών παρουσίασης (presentation layer) της εφαρμογής και έχει να κάνει με αυτό που βλέπει ο χρήστης, στην οθόνη της συσκευής του, όταν τρέχει την εφαρμογή. Λειτουργεί με τη βοήθεια κλάσεων και ο χρήστης καλεί τα αντικείμενα των κλάσεων, για διάφορες λειτουργίες όπως να διαβάσει ένα κείμενο, να πατήσει ένα κουμπί, κλπ. Τα components είναι τα επίπεδα ασφαλείας με τα οποία λειτουργεί μία κινητή εφαρμογή. Άλλα components φαίνονται και άλλα λειτουργούν χωρίς να είναι ορατά, δηλαδή ο χρήστης δεν τα αντιλαμβάνεται. Για παράδειγμα, το συστατικό 'Content Provider' έχει σαν σκοπό το μοίρασμα στοιχείων με άλλες εφαρμογές που έχει εγκατεστημένες ο χρήστης. Αυτό γίνεται για την

καλύτερη οργάνωση των λειτουργιών μιας εφαρμογής αλλά οπωσδήποτε και για λόγους ασφαλείας. Κάθε συστατικό μπορεί να διακριθεί σε δημόσια (public) ή ιδιωτική (private) λειτουργία (Six, 2012). Όταν τα δικαιώματα είναι public, τότε μπορεί να επικοινωνήσει και με άλλες εφαρμογές. Σε περίπτωση που τα δικαιώματα είναι σε private επίπεδο, τότε η εφαρμογή μπορεί να επικοινωνήσει μόνο με εφαρμογές που έχουν το ίδιο user ID. Πρέπει να σημειωθεί ότι τα συστατικά παίζουν σημαντικό ρόλο στη δημιουργία μιας εφαρμογής και πρέπει να είναι συγκεκριμένα και να μην επικαλούνται πληροφορίες που δε χρειάζονται για τη λειτουργία τους.

Όλα τα δεδομένα των εφαρμογών σε μία συσκευή Android αποθηκεύονται συνήθως σε κλασικές αποθηκευτικές διατάξεις. Αυτό βέβαια δεν παρέχει ασφάλεια καθώς τα δεδομένα είναι εύκολο να παραβιαστούν από hackers και όταν μιλάμε για ευαίσθητα δεδομένα τότε υπάρχει σοβαρός κίνδυνος. Για τον λόγο αυτό χρησιμοποιείται η τεχνική της κρυπτογράφησης των πληροφοριών έτσι ώστε να παρέχεται επιπλέον ασφάλεια στην κυκλοφορία των δεδομένων. Χρησιμοποιείται η τεχνική της συμμετρικής κρυπτογράφησης, πχ. ο αλγόριθμος Advanced Encryption Standard (AES). Οι συμμετρικοί αλγόριθμοι, όπως αναφέραμε σε προηγούμενο κεφάλαιο, βασίζονται στο κλειδί κρυπτογράφησης που χρησιμοποιούν κάθε φορά. Δηλαδή η δύναμη της συμμετρικής κρυπτογράφησης, βασίζεται στο μήκος της τιμής του κλειδιού που χρησιμοποιείται, και όσο μεγαλύτερο μήκος έχει το κλειδί, τόσο πιο δυνατός είναι ο αλγόριθμος. Χρησιμοποιούνται κλειδιά μήκους 256, 512 και 1024 bits. Παρόλο που είναι λειτουργικοί αλγόριθμοι, προκύπτουν προβλήματα στη μεταβίβαση του μηνύματος καθώς πρέπει ο αποδέκτης να χρησιμοποιεί το ίδιο κλειδί με το οποίο έγινε η κρυπτογράφηση, αλλιώς δε γίνεται να διαβάσει το αρχικό μήνυμα. Από την άλλη μεριά, χρησιμοποιείται και η ασύμμετρη τεχνική κρυπτογράφησης, όπου διαφορετικό είναι το κλειδί για την κρυπτογράφηση και διαφορετικό για την αποκρυπτογράφηση. Δηλαδή το κλειδί κρυπτογράφησης είναι δημόσιο και το κλειδί αποκρυπτογράφησης είναι ιδιωτικό (και διαφορετικό για κάθε χρήστη). Από τους πιο γνωστούς ασύμμετρους αλγόριθμους είναι ο RSA και όπως στους συμμετρικούς αλγόριθμους η δύναμή του βασίζεται στο μέγεθος του κλειδιού που χρησιμοποιείται. Κλασικό μήκος κλειδιού για ασύμμετρο αλγόριθμο είναι τα 2048 bits και η ασφάλεια που προσφέρει μπορεί να συγκριθεί με το κλειδί του συμμετρικού αλγορίθμου των 256 bits. Αυτό σημαίνει ότι η συγκεκριμένη κατηγορία κρυπτογράφησης είναι σχετικά αργή καθώς απαιτεί πολύ μνήμη και δεδομένα από το σύστημα, για την ασφαλή μεταβίβαση του μηνύματος.

2. Ιδιωτικότητα και εμπιστοσύνη σε περιβάλλοντα κινητού εμπορίου

2.1 Προσωπική και επαγγελματική χρήση κινητής συσκευής

Τα γενικότερα ζητήματα ασφαλείας, αλλά και τα ειδικότερα όσον αφορά την ιδιωτικότητα των κινητών χρηστών, αποκτούν άλλη επιπλέον διάσταση όταν αναφερόμαστε σε επαγγελματική χρήση (business use) των κινητών συσκευών. Η πρωτοβουλία 'Φέρτε τη δική σας συσκευή' (‘Bring Your Own Device, BYOD’) αναφέρεται στην πολιτική ενός οργανισμού ή μιας επιχείρησης να επιτρέπει στους εργαζομένους να χρησιμοποιούν προσωπικές κινητές συσκευές (φορητοί υπολογιστές, ταμπλέτες και έξυπνα τηλέφωνα) στον χώρο εργασίας τους, και με αυτές τις συσκευές να έχουν πρόσβαση σε προνομιακές πληροφορίες καθώς και στις εφαρμογές της εταιρείας (Keyes, 2013). Ο όρος χρησιμοποιείται επίσης για να περιγράψει την ίδια πρακτική που εφαρμόστηκε σε σχολεία με μαθητές που χρησιμοποιούν προσωπικές συσκευές σε χώρους εκπαίδευσης (Lennon, 2012).

Η στρατηγική BYOD ουσιαστικά γεννήθηκε στον χώρο των επιχειρήσεων το 2009 από την Intel, όταν η τελευταία αναγνώρισε μια αυξανόμενη τάση μεταξύ των εργαζομένων της για να χρησιμοποιούν τις δικές τους συσκευές για να εργαστούν και να συνδέονται στο εταιρικό δίκτυο (Astani et al., 2013). Αντί να απορριφθεί η τάση, όπως πολλές άλλες επιχειρήσεις αρχικά προσπάθησαν, τα ανώτερα ηγετικά στελέχη της Intel έσπευσαν να αγκαλιάσουν την καινούρια αυτή τακτική για τη μείωση του κόστους της επιχείρησης και τη βελτίωση της παραγωγικότητας των εργαζομένων. Πιο συγκεκριμένα, από τον Ιανουάριο του 2010, ο αριθμός των εργαζομένων που χρησιμοποιούσαν κινητές συσκευές στο εργασιακό περιβάλλον έχει τριπλασιαστεί (από 10.000 έως 30.000), και από το 2014 αναμένεται ότι το 70% των συνολικά 80.000 εργαζομένων της Intel θα χρησιμοποιούν τα δικά τους συστήματα για τουλάχιστον ένα μέρος των καθημερινών εργασιακών καθηκόντων.

Η πολιτική BYOD είναι μια πραγματικότητα για τα περισσότερα τμήματα πληροφορικής επιχειρήσεων σήμερα. Οι χρήστες, οι οποίοι γνωρίζουν πώς να λειτουργούν τις δικές τους κινητές συσκευές, είναι βεβαίως πιο παραγωγικοί με μικρό ή καθόλου κόστος για τον οργανισμό. Οι χρήστες επίσης

ενθαρρύνονται να αγοράσουν συσκευές με το κίνητρο ότι μπορούν να τις χρησιμοποιούν σε όλες τις πτυχές της ζωής τους, συμπεριλαμβανομένης και της επαγγελματικής (Bestmann et al, 2015). Έτσι δεν αποτελεί έκπληξη, ότι όλο και περισσότερο οι εργαζόμενοι (και ιδίως τα ανώτερα στελέχη) απαιτούν πρόσβαση σε πόρους της εταιρείας, email και άλλα δεδομένα εργασιακού περιεχομένου.

2.1.1 Οφέλη

Η υιοθέτηση της στρατηγικής BYOD οδηγεί σε πολλά οφέλη είτε προς τους εργαζόμενους είτε προς την επιχείρηση. Τα οφέλη αυτά μπορούν να ταξινομηθούν σε τρεις μεγάλες κατηγορίες: αύξηση παραγωγικότητας, ικανοποίηση των εργαζομένων και μείωση λειτουργικών εξόδων (Pillay et al., 2013). Μέσα από μια BYOD πολική οι εργαζόμενοι γίνονται πιο παραγωγικοί στα εργασιακά τους καθήκοντα αλλά και αρκετές φορές καινοτόμοι. Νοιώθουν πιο άνετα με μια προσωπική συσκευή και μαθαίνουν να τη χειρίζονται σαν ειδικοί, γεγονός που τους καθιστά πιο ικανούς. Η χρήση αυτής της πρωτοποριακής τεχνολογίας στον χώρο εργασίας έχει αυξηθεί σημαντικά κατά τα τελευταία χρόνια. Οι τελικοί χρήστες θέτουν τις προσωπικές τους συσκευές σε λειτουργία και επιλέγουν από μόνοι τους πότε, πώς και με ποια εργαλεία στις συσκευής τους θα ολοκληρώσουν τα εργασιακά τους καθήκοντα. Η ελευθερία αυτή είναι ο κύριος λόγος της αύξησης της παραγωγικότητας αλλά και της καινοτομικής δράσης των υπαλλήλων (Georgiadis et al., 2014).

Οι κινητές συσκευές BYOD επέτρεψαν στους εργαζομένους να έχουν πρόσβαση στα δεδομένα της εταιρείας, χωρίς περιορισμό τοποθεσίας. Η βελτίωση της επικοινωνίας και πρόσβασης στα δεδομένα επιτρέπει στους οργανισμούς να βελτιώσουν τα προϊόντα και τις υπηρεσίες που προσφέρονται και να αυξήσουν την αξία τους στους πελάτες. Σε BYOD εργασιακά περιβάλλοντα οι εργαζόμενοι αρκετές φορές χρησιμοποιούν τις συσκευές που οι ίδιοι από μόνοι τους έχουν επιλέξει και επενδύσει και όχι συσκευές που επιβλήθηκαν από τις επιχειρήσεις. Αν και υπάρχουν περιπτώσεις που οι επιχειρήσεις χωρίς να επιβάλλουν τη χρήση συγκεκριμένων συσκευών, απαιτούν την επιλογή συσκευών με κάποιες συγκεκριμένες προδιαγραφές. Επιτρέποντας στους υπαλλήλους να χρησιμοποιούν προσωπικές συσκευές αποφεύγεται η χρήση πολλαπλών συσκευών κάτι που διευκολύνει τους εργαζόμενους και τους χαροποιεί (Morrow, 2012). Πολιτικές BYOD επιτρέπουν επίσης στους εργαζόμενους να χρησιμοποιούν την τεχνολογία με την οποία είναι άνετοι και προτιμούν, αντί για αυτό που η επιχείρηση συνήθιζε να τους υπαγορεύει. Πολλές φορές η ικανοποίηση των εργαζομένων προκύπτει και από συμφωνίες που γίνονται ανάμεσα σε αυτούς και τους εργοδότες τους, προκειμένου να εφαρμοστούν πολιτικές BYOD προς όφελος και των δύο. Για παράδειγμα, οι εργοδότες πληρώνουν ένα σταθερό ποσοστό από τους λογαριασμούς των κινητών συσκευών των εργαζομένων, ένα ποσό που καθορίζεται από τα χαρακτηριστικά του κάθε υπαλλήλου και κατατίθεται κάθε μήνα. Γενικά, οι χρήστες έχουν ελευθερία κινήσεων κάτι το οποίο τους ικανοποιεί και με τη σειρά τους ικανοποιούν και αυτοί τους εργοδότες τους.

Οι πολιτικές BYOD βοηθούν τον οικονομικό προϋπολογισμό των επιχειρήσεων με τη μετατόπιση του κόστους αγοράς τεχνολογικού εξοπλισμού στον χρήστη, καθώς οι εργαζόμενοι πληρώνουν για κινητές συσκευές και υπηρεσίες δεδομένων. Άλλωστε οι χρήστες μπορούν να αναβαθμίσουν τις συσκευές τους πιο συχνά από ό,τι η επιχείρηση. Έτσι λοιπόν, επιτρέποντας στους υπαλλήλους να φέρουν και να χρησιμοποιούν τις δικές τους συσκευές, μπορεί ένας οργανισμός να έχει πάντα νέα μοντέλα της τεχνολογίας μέσω των υπαλλήλων, χωρίς να αναλαμβάνει τουλάχιστον ολόκληρο το κόστος αυτό. Οι επιχειρήσεις εξοικονομούν χρήματα και μέσω της αυξημένης παραγωγικότητας των εργαζομένων. Όταν μια πολιτική BYOD εφαρμόζεται σε έναν οργανισμό, οι εργαζόμενοι τείνουν να προστατεύουν τις συσκευές τους σε μεγαλύτερο βαθμό, καθώς υπάρχει μια μεγαλύτερη αίσθηση προσωπικής ιδιοκτησίας, που οδηγεί σε μείωση του κόστους υλικού και συντήρησης (Mont, 2012). Σύμφωνα μάλιστα με μια έρευνα της CISCO (2012), η υιοθέτηση BYOD τακτικών οδηγεί τις επιχειρήσεις σε ετήσια κέρδη που κυμαίνονται από 300 έως 1300 δολάρια ανά εργαζόμενο.

2.1.2 Προκλήσεις

Οι προκλήσεις και οι κίνδυνοι που εμφανίζονται μετά από την υιοθέτηση μιας BYOD πολιτικής είναι ιδιαίτερα σημαντικοί. Οι βασικοί κίνδυνοι είναι η ενδεχόμενη απώλεια δεδομένων, η αδυναμία επιβολής ελεγκτικών διαδικασιών και τα υψηλά κόστη από τα απαιτούμενα μέτρα ασφαλείας (Pillay et al., 2013).

Ένας από τους πιο σημαντικούς κινδύνους που σχετίζονται με την πρωτοβουλία BYOD είναι η απώλεια δεδομένων από συσκευές που έχουν χαθεί ή κλαπεί εντός ή εκτός του χώρου εργασίας. Η ευκολία και η άνεση που προσφέρει το μέγεθος των έξυπνων συσκευών καθιστά εύκολο να χαθούν ή να κλαπούν (Calder, 2013). Επιπλέον, όταν οι εργαζόμενοι αντικαθιστούν τα κινητά τους τηλέφωνα με κάποια άλλα νεότερης τεχνολογίας, εμπιστευτικές και ευαίσθητες πληροφορίες μεταβιβάζονται σε μη εξουσιοδοτημένους χρήστες (McAfee, 2012).

Sound 7.3.mp3	Ηχητικό απόσπασμα (audio)
Προκλήσεις της BYOD πολιτικής	

Οι ειδικοί υποστηρίζουν ότι η απώλεια μιας BYOD συσκευής είναι ο υπ' αριθμόν ένα κίνδυνος όχι μόνο επειδή χάνονται δεδομένα που υπήρχαν σε αυτή, αλλά και επειδή τα δεδομένα μπορεί να περιέχουν εμπιστευτικές πληροφορίες που μπορούν να κοστίζουν σημαντικά στη φήμη μιας επιχείρησης. Επιπλέον, όταν ένας εργαζόμενος αποχωρεί από την επιχείρηση, δεν είναι υποχρεωμένος να παραδώσει τη συσκευή καθώς αυτός είναι ο νόμιμος ιδιοκτήτης. Ως αποτέλεσμα, οι εφαρμογές της εταιρείας και άλλα δεδομένα μπορούν και εξακολουθούν να υπάρχουν στη συσκευή, γεγονός που κρύβει επίσης πρόσθετους κινδύνους απώλειας πληροφοριών που η επιχείρηση δεν μπορεί να εκτιμήσει (Thomson, 2012). Άλλος κρίσιμος παράγοντας σχετικά με τη διαρροή δεδομένων που σχετίζονται με BYOD πολιτικές είναι η δυσκολία εντοπισμού της διαρροής πριν να είναι πολύ αργά. Αυτό αποτρέπει τον εμπλεκόμενο φορέα να λάβει άμεσα αντίμετρα για την ελαχιστοποίηση των επιπτώσεων, η οποία θα μπορούσε να είναι δυνατόν στο αρχικό στάδιο της διάγνωσης.

Η κακή χρήση των πόρων που προσφέρει μία επιχείρηση στους εργαζομένους αυξάνεται σε ένα περιβάλλον BYOD. Οι εργαζόμενοι παρακάμπτουν σκοπίμως τους περιορισμούς ασφαλείας (πχ. κωδικούς πρόσβασης, πολιτικές ασφαλείας IT) και θέτουν σε κίνδυνο την ασφάλεια της εμπλεκόμενης επιχείρησης. Οι οργανισμοί δεν έχουν κανέναν έλεγχο πάνω στο τι τύπους εφαρμογών υπάρχουν στην κάθε συσκευή, γεγονός το οποίο καθιστά πολύ δύσκολο να εφαρμοστούν πολιτικές ασφαλείας. Αν και οι εργαζόμενοι δεν κατεβάζουν παιχνίδια ή άλλες εφαρμογές ψυχαγωγίας στον υπολογιστή που χρησιμοποιούν στην εργασία τους (συνήθως …), στην περίπτωση BYOD, δεδομένου ότι η συσκευή είναι δικής τους κυριότητας είναι φυσικό να κατεβάζουν πολλές προσωπικές εφαρμογές στη συσκευή, έστω εκτός χρόνου δουλειάς. Οι εφαρμογές αυτές μπορούν να θέσουν σε κίνδυνο σημαντικές πληροφορίες της επιχείρησης. Από την άλλη πλευρά, οι επιχειρήσεις θεωρούν ότι είναι δύσκολο να ελέγχουν και να παρακολουθούν κατά πόσο ο εξουσιοδοτημένος υπάλληλος χρησιμοποιεί σωστά τη συσκευή (αλλά και τις πληροφορίες της) εκτός εργασιακού περιβάλλοντος. Οργανισμοί που έχουν υιοθετήσει μια BYOD πολιτική έχουν την 'επικρεμάμενη' απειλή της απώλειας του ελέγχου επί των δεδομένων τους, εφόσον υπάρχει δυνατότητα πρόσβασης εκτός της δικαιοδοσίας του οργανισμού ή του επαγγελματικού δικτύου. Αυτό γίνεται κατανοητό αν αναλογιστούμε ότι το 30% των εργαζομένων περίπου σε BYOD οργανισμούς σε όλο τον κόσμο (στοιχεία 2012, με τάση αυξητική …) κάνουν χρήση υπηρεσιών cloud (όπως το Dropbox, το iCloud, το OneDrive και το Google Docs) για την αποθήκευση και πρόσβαση σε δεδομένα χωρίς χρονικούς ή τοπικούς περιορισμούς. Επιπλέον, τα επαγγελματικά μηνύματα ηλεκτρονικού ταχυδρομείου που είναι πλέον προσβάσιμα μέσω του δικτύου κινητής τηλεφωνίας στην BYOD συσκευή, δεν είναι υπό τον έλεγχο του οργανισμού και είναι επιρρεπή σε κακόβουλες επιθέσεις (Phifer, 2013).

Μία τελευταία διάσταση που αποτελεί και φυσική κατάληξη των δύο προαναφερόμενων κινδύνων είναι το αυξανόμενο κόστος από τα μέτρα ασφαλείας που επιβάλλει η υιοθέτηση BYOD τακτικών. Οι επιχειρήσεις που υιοθετούν BYOD πολιτικές, προκειμένου να προσφέρουν υψηλά επίπεδα ασφαλείας οδηγούνται στη λήψη μέτρων με σημαντικό κόστος, όπως η δημιουργία ενός τμήματος IT-help-desk που παρέχει υποστήριξη προς τους εργαζόμενους. Επιπλέον, θα πρέπει να παρέχεται κατάλληλο προστατευτικό λογισμικό κατά των κακόβουλων προγραμμάτων που επίσης κοστίζουν.

2.2 Ευαισθητοποίηση σε ζητήματα ασφάλειας των χρηστών κινητών συσκευών

Αν και η χρήση των έξυπνων κινητών συσκευών είναι ευρέως διαδεδομένη, και ενώ καινοτόμες πολιτικές επιτρέπουν τη χρήση προσωπικών κινητών συσκευών στο εργασιακό περιβάλλον είναι ιδιαίτερα δημοφιλείς παγκοσμίως, οι κίνδυνοι που ελλοχεύουν είτε παραμένουν άγνωστοι, είτε αγνοούνται. Οι κινητές συσκευές,

ως επέκταση του εργασιακού περιβάλλοντος του χρήστη, επιτρέπουν περισσότερα σημεία εισόδου και εξόδου, με πολύ λιγότερο έλεγχο από την παραδοσιακή εταιρική υποδομή. Αναμφίβολα, οι κινητές συσκευές και οι εφαρμογές που αυτές προσφέρουν, έχουν μια σειρά από χαρακτηριστικά που τις ξεχωρίζουν από ένα παραδοσιακό υπολογιστικό περιβάλλον, γεγονός που δημιουργεί την ανάγκη ειδικού χειρισμού των απειλών. Μια αξιόπιστη μελέτη του 2011 διαπίστωσε ότι σε 14 χώρες και με 1.500 ερωτηθέντες, ενώ το 95% των επιχειρήσεων εφαρμόζουν πολιτικές χρήσης κινητών συσκευών στο εργασιακό περιβάλλον, **λιγότεροι από ένας στους τρεις** υπαλλήλους έχουν επίγνωση της πολιτικής στην πράξη. Η πλειοψηφία των χρηστών κινητής τηλεφωνίας αισθάνονται ότι έχουν μικρό ή καθόλου έλεγχο επί των προσωπικών πληροφοριών που είναι αποθηκευμένα στις συσκευές και δεν έχουν καμία γνώση πάνω στα δεδομένα που συλλέγουν οι εταιρείες, κατά την περιήγηση στο Διαδίκτυο ή κατά τη χρήση ηλεκτρονικών υπηρεσιών (Microsoft, 2013).

Υπάρχει λοιπόν έντονη χρήση των νέων τεχνολογιών και πρακτικών που οι κινητές συσκευές έχουν επιφέρει, αλλά ταυτόχρονα δεν υπάρχει γνώση σχετικά με το τι ακριβώς είναι και τι ακριβώς περιέχουν όλες αυτές οι νέες τεχνολογικές πρακτικές. Το ανθρώπινο στοιχείο είναι μία από τις μεγαλύτερες πηγές κινδύνου για το θέμα της ασφάλειας των πληροφοριών, καθώς και μία πηγή από τις πιο δύσκολες να ελεγχθούν. Εργαζόμενοι χωρίς επαρκή ευαισθητοποίηση σε θέματα ασφάλειας, μπορούν να θέσουν σε κίνδυνο την επιχείρηση στην οποία δουλεύουν, απαντώντας σε κακόβουλα μηνύματα, αφήνοντας μη εξουσιοδοτημένα άτομα να αποκτήσουν πρόσβαση στο εσωτερικό της επιχείρησης, ή ακόμα και προσφέροντας (έναντι υψηλής αμοιβής) δεδομένα πνευματικής ιδιοκτησίας της επιχείρησης σε άλλους ανταγωνιστικούς φορείς. Οι νέες τεχνολογίες επιτείνουν το πρόβλημα: παρά το γεγονός ότι παρέχουν ισχυρές νέες δυνατότητες που μπορούν να ωφελήσουν την επιχείρηση, μπορούν επίσης κάλλιστα να εισάγουν νέους κινδύνους ασφαλείας με ταχύτερο ρυθμό από ότι πολλές επιχειρήσεις μπορούν να χειριστούν.

Σε γενικές γραμμές οι χρήστες ενδιαφέρονται για την ασφάλεια των συσκευών τους, αλλά δεν υιοθετούν πρακτικές ασφαλείας μέχρι να βιώσουν ένα περιστατικό παραβίασης ασφαλείας. Σύμφωνα με πρόσφατη έρευνα (Dimensional Research, 2013), οι εργαζόμενοι όχι μόνο ενδιαφέρονται για την προστασία των συσκευών και για τις συνέπειες των παραβιάσεων ασφαλείας σε αυτές, αλλά επιπλέον αναγνωρίζουν ότι η μη επίγνωση των κινδύνων και οι αδιάφοροι χρήστες αποτελούν μεγαλύτερο κίνδυνο από τους ίδιους τους επιτιθέμενους. Η έλλειψη ουσιαστικής εκπαίδευσης των χρηστών σχετικά με ζητήματα ασφάλειας είναι ο κυριότερος λόγος που δημιουργούνται λανθασμένες αντιλήψεις σε αρκετές φορές υποτιθέμενα ενημερωμένους χρήστες. Κάποιες έρευνες δείχνουν, και αυτό είναι επίσης σημείο συζήτησης, ότι χρήστες που έχουν συσκευές όχι τόσο εξελιγμένες (μπορεί δηλαδή να μην ανήκουν στην κατηγορία των έξυπνων συσκευών), έχουν περισσότερη επίγνωση πάνω σε θέματα ασφαλείας και παραβιάσεων.

Προδιαγεγραμμένες πολιτικές ασφάλειας, ως μέτρα προστασίας, τίθενται σε ισχύ από τους κατασκευαστές συσκευών και τους παρόχους, αλλά αυτό που χρειάζεται να γίνει με επιτυχία είναι η σωστή εκπαίδευση των χρηστών κινητής τηλεφωνίας, προκειμένου να προστατεύσουν την ιδιωτική τους ζωή και τα δεδομένα που αποθηκεύουν σε συσκευές (είτε πρόκειται για προσωπικά δεδομένα ή για αρχεία που σχετίζονται με το περιβάλλον εργασίας). Στα πλαίσια της εκπαιδευτικής μεθόδου, θα πρέπει να υπάρχει μια σαφέστατη και πλήρως κατανοητή μέθοδος επικοινωνίας με τους εκπαιδευόμενους εργαζομένους. Οι εκπαιδευόμενοι θα πρέπει να κατανοήσουν ότι τυχόν αποκλίσεις από την καθιερωμένη διαδικασία και το κανάλι επικοινωνίας που αυτή εισάγει στην επιχείρηση προκαλεί ρήγματα εμπιστοσύνης. Οι περιορισμοί στο κανάλι επικοινωνίας μειώνουν τους κινδύνους των χρηστών που πέφτουν θύματα επιθέσεων. Οι χρήστες θα πρέπει επίσης να έχουν έναν καθιερωμένο τρόπο επικοινωνίας στη διάθεση τους αν έχουν τις οποιεσδήποτε υπόνοιες παραβίασης ασφαλείας στη συσκευή τους και να κατανοήσουν πώς η επιχείρηση θα επιληφθεί του θέματος και δε θα αποδώσει ευθύνες στους ίδιους. Τελικά, οι διαδικασίες αυτές θα οδηγήσουν τον εργαζόμενο να δηλώσει τις υπόνοιες αυτές, χωρίς να φοβάται τυχόν κυρώσεις (σε περίπτωση που πρόκειται για δικό του σφάλμα το οποίο οδήγησε άθελα του σε παραβίαση ασφαλείας).

Θα πρέπει να υπάρχει ένα βασικό πρόγραμμα εκπαίδευσης των εργαζομένων σε σχέση με την ασφάλεια, το οποίο θα εκπαιδεύει τους υπαλλήλους σχετικά με το τι ενέργειες θα πρέπει να ακολουθήσει σε περίπτωση που μια επίθεση λάβει χώρα και απαιτεί άμεση αντιμετώπιση. Η εκπαίδευση θα περιλαμβάνει την εκμάθηση τρόπων αντιμετώπισης επιθέσεων και χειρισμού ζητημάτων (όπως η σημασία χρήσης κωδικών πρόσβασης και η δομή των ισχυρών κωδικών πρόσβασης), εξασφαλίζοντας με αυτό τον τρόπο ότι οι χρήστες θα χρησιμοποιούν κωδικούς και θα προστατεύουν τη συσκευή τους. Ακόμα στους εργαζομένους θα διδάσκεται η σημασία της ενημέρωσης του λειτουργικού τους συστήματος και των εγκατεστημένων εφαρμογών αλλά και οι ενδεχόμενοι κίνδυνοι από μια παραβιασμένη συσκευή. Επιπρόσθετα, θα υπάρχει η δυνατότητα εκμάθησης κρυπτογραφικών μεθόδων αλλά και η χρήση λογισμικών προστασίας. Τέλος, οι εργαζόμενοι θα πρέπει να εκπαιδεύονται σχετικά με τη σημαντικότητα της λήψης λογισμικού, αρχείων και

προγραμμάτων από αξιόπιστους δικτυακούς τόπους, καθώς και να κατανοούν πως δουλεύει μια εφαρμογή και τι είδους απαιτήσεις έχει από τον χρήστη.

3. Ιδιωτικότητα και εμπιστοσύνη σε κινητά συστήματα συστάσεων

Τα συστήματα συστάσεων και εξατομίκευσης είναι αλγόριθμοι ή λογισμικό του υπολογιστή που έχουν σχεδιαστεί για να παρέχουν προτάσεις για προϊόντα ή υπηρεσίες που θα μπορούσαν να ενδιαφέρουν τον χρήστη του δικτυακού τόπου συναλλαγής (Bobadilla et al., 2013; Konstan & Riedl, 2012). Μιλήσαμε για αυτά αναλυτικά στο 4ο κεφάλαιο. Οι πρόσφατες εξελίξεις στον τομέα των κινητών συσκευών (όπως smartphones και tablets που κάνουν χρήση υψηλής ισχύος επεξεργασίας και εξελιγμένων λειτουργικών συστημάτων) και της κινητής υπολογιστικής (mobile computing), καθώς και η συνεχής ανάπτυξη του Διαδικτύου και των σχετικών ασύρματων υποδομών, οδήγησε στην ανάπτυξη του πεδίου των κινητών συστημάτων συστάσεων (mobile recommender systems) (Polatidis & Georgiadis, 2013; Polatidis & Georgiadis, 2014; Ricci, 2010; del Carmen & Ilarri, 2014). Η πρόσβαση σε ένα σύστημα συστάσεων σε κινητό περιβάλλον σε κάθε δεδομένη χρονική στιγμή και σε κάθε μέρος καλείται πανταχού παρουσία, και αυτός είναι ένας εναλλακτικός ορισμός για τα πανταχού παρόντα συστήματα συστάσεων (ubiquitous recommender systems) και εξατομίκευσης (Mettouris & Papadopoulos, 2014).

Η χρήση των δεδομένων θέσης από το παγκόσμιο σύστημα εντοπισμού θέσης (Global Positioning System, GPS) και η χρήση άλλων συναφών πληροφοριών, όπως είναι ο χρόνος, ο καιρός, οι φυσικές συνθήκες, κοινωνικά και άλλα δεδομένα, είναι κάτι φυσιολογικό στα κινητά συστήματα συστάσεων (Mettouris & Papadopoulos, 2014; Sun et al., 2015). Αυτά τα νέα είδη δεδομένων έχουν βοηθήσει προς την κατεύθυνση της προσφοράς περισσότερο εξατομικευμένων συστάσεων σε κινητά περιβάλλοντα (Jannach et al., 2010; Polatidis & Georgiadis, 2013; Polatidis & Georgiadis, 2014; Ricci, 2010). Υπάρχουν βεβαίως διαφορετικά πεδία εφαρμογών, με διαφορετικές απαιτήσεις (όπως τα περιβάλλοντα κινητού ηλεκτρονικού εμπορίου και του τουρισμού που σχετίζονται με τις υπηρεσίες (Jannach et al., 2010; Ricci, 2010), αλλά πέρα από τη φυσιολογική έως ένα βαθμό ανομοιογένεια του κάθε τομέα, μπορούμε να διακρίνουμε κοινά χαρακτηριστικά:

- Όλα παρέχουν κάποια μορφή σύστασης
- Όλα τρέχουν σε μια φορητή συσκευή, όπως ένα smartphone ή tablet
- Όλα αξιοποιούν κάποια μορφή πληροφορίας από το περιβάλλον (context-awareness)
- Όλα στηρίζονται σε κάποια ασύρματη σύνδεση, που θα μπορούσε πιθανότατα να είναι αργή

Μπορούμε πλέον (στα σύγχρονα κινητά συστήματα συστάσεων) να προσθέσουμε και την προσπάθεια για προστασία των προσωπικών δεδομένων των κινητών χρηστών, ως ένα ακόμη κοινό χαρακτηριστικό. Αποτελεί πράγματι ένα σημαντικό ζήτημα των κινητών συστημάτων συστάσεων που η διερεύνησή του είναι ανοικτό ερευνητικό θέμα. Αναφέρθηκε ότι τα συστήματα συστάσεων που λειτουργούν σε κινητά περιβάλλοντα στηρίζονται σε μεγάλο βαθμό στο περιβάλλον πλαίσιο (context) για την παροχή συστάσεων, έτσι ώστε να είναι όσο το δυνατόν πιο κοντά στα ενδιαφέροντα του χρήστη και την τρέχουσα τοποθεσία του. Ωστόσο χαρακτηριστικές τεχνικές προστασίας της ιδιωτικής ζωής, όπως η χρήση ψευδωνύμων ή της ανωνυμίας δεν μπορούν να εφαρμοστούν με τον κλασικό τρόπο, και αυτό οφείλεται στο γεγονός ότι τα συστήματα συστάσεων βασίζονται στη χρήση των προσωπικών δεδομένων (Scipioni, 2011). Για παράδειγμα στο (Kido et al., 2005) περιγράφεται μια προσέγγιση για την ανώνυμη επικοινωνία που χρησιμοποιούν υπηρεσίες τοποθεσίας (location-based) που βασίζονται στη χρήση ψεύτικων δεδομένων. Παρόμοιες μέθοδοι που έχουν χρησιμοποιηθεί για την προστασία της ιδιωτικής ζωής στον τομέα των υπηρεσιών με βάση τη θέση είναι οι τεχνικές διεύρυνσης της θέσης (Juniper, 2011). Υπάρχουν στη βιβλιογραφία (Boutet et al., 2015; Aïmeur et al., 2008; Polat & Du, 2005) διάφορες μέθοδοι παραγωγής συστάσεων που βασίζονται στο συνεργατικό φιλτράρισμα και οι οποίοι επιχειρούν να προστατέψουν (ως ένα βαθμό) την ιδιωτική ζωή των χρηστών.

3.1 Παράγοντες που επηρεάζουν τις συστάσεις στα κινητά περιβάλλοντα

Μια σειρά από παράγοντες υπάρχουν που μπορούν να επηρεάσουν τα κινητά συστήματα συστάσεων και την ικανότητά τους να παρέχουν ακριβείς εξατομικευμένες συστάσεις. Αυτά περιλαμβάνουν τη μέθοδο σύστασης,

το πλαίσιο (ή περιβάλλουσα κατάσταση ή συναφείς πληροφορίες) και τις ανησυχίες προστασίας της ιδιωτικής ζωής (Polatidis & Georgiadis, 2013; Polatidis & Georgiadis, 2014). Για τις μεθόδους/αλγορίθμους σύστασης, με κύριες προσεγγίσεις το συνεργατικό φιλτράρισμα και το φιλτράρισμα με βάση το περιεχόμενο (Bobadilla et al., 2013; Shi et al., 2014), έχουμε αναφερθεί αναλυτικά στο 4ο κεφάλαιο. Και στην περιβάλλουσα κατάσταση (context) έχουμε αναφερθεί, και σε ζητήματα ιδιωτικότητας, αλλά αξίζει στο κεφάλαιο αυτό να συμπληρώσουμε κάποια επιπλέον ενδιαφέροντα στοιχεία.

3.1.1 Πλαίσιο (context)

Το περιβάλλον πλαίσιο χρησιμοποιείται από τα κινητά συστήματα συστάσεων για να παρέχουν πιο ακριβείς και εξατομικευμένες συστάσεις. Είναι ένας σύνολο δεδομένων που είναι αναγκαίο για την περιγραφή των συνθηκών που έχουν οι χρήστες που κινούνται συνεχώς. Διαφορετικοί τύποι πλαισίου μπορεί να χρησιμοποιηθούν και περιλαμβάνουν, μεταξύ άλλων, τη θέση, την ώρα, τον καιρό και την κοινωνική παρουσία (Adomavicius & Tuzhilin, 2011; Ricci, 2010; Liu et al., 2013). Οι πληροφορίες μπορούν να συλλέγονται είτε ρητά, ζητώντας από τον χρήστη να παράσχει δεδομένα, ή σιωπηρά (συλλέγοντας δεδομένα από την κινητή συσκευή και τους αισθητήρες της, πχ. αξιοποιώντας το σύστημα εντοπισμού θέσης) (Adomavicius & Tuzhilin, 2011; Mettouris & Papadopoulos, 2014).

Sound 7.4.mp3	Ηχητικό απόσπασμα (audio)
Τι είναι το περιβάλλον πλαίσιο	

Το πλαίσιο μπορεί να εφαρμοστεί με χρήση τριών διαφορετικών τρόπων (Adomavicius & Tuzhilin, 2011):

- **Προ-φιλτράρισμα** (pre filtering): η διαδικασία που χρησιμοποιείται για να γίνει το φιλτράρισμα, λαμβάνει μέρος πριν την εφαρμογή του αλγόριθμου σύστασης
- **Μετά-φιλτράρισμα** (post filtering): η διαδικασία που χρησιμοποιείται για να γίνει το φιλτράρισμα, λαμβάνει μέρος μετά την εφαρμογή του αλγόριθμου σύστασης
- **Εντός του αλγόριθμου σύστασης**: η διαδικασία που χρησιμοποιείται για να γίνει το φιλτράρισμα, λαμβάνει μέρος κατά τη διάρκεια εκτέλεσης του αλγόριθμου σύστασης.

3.1.2 Ιδιωτικότητα

Τα κινητά συστήματα συστάσεων προσφέρουν το πλεονέκτημα της παροχής εξατομικευμένων συστάσεων σε χρήστες ενός περιβάλλοντος που αλλάζει συνεχώς. Από την άλλη πλευρά, οι τρόποι με τους οποίους τα δεδομένα του χρήστη θα μπορούσαν να υποβληθούν σε επεξεργασία, φέρνει άμεσα τους χρήστες προς μια αρνητική στάση, όταν πρόκειται για την παροχή προσωπικών πληροφοριών πλαισίου (Liu et al., 2013; Mettouris & Papadopoulos, 2014). Οι τεχνικές προστασίας της ιδιωτικής ζωής έχουν στηριχθεί κυρίως σε υπηρεσίες θέσης (Jensen et al., 2009; Scipioni, 2011) και δε λαμβάνουν υπόψη την όλη ιδέα του περιβάλλοντος πλαισίου. Η προστασία των προσωπικών δεδομένων είναι ένας σημαντικός παράγοντας που αν αντιμετωπιστεί σωστά, (χρησιμοποιώντας τις σωστές μεθόδους) τότε επιτρέπει στον χρήστη να παράσχει με εμπιστοσύνη τις απαιτούμενες πληροφορίες πλαισίου, καθιστώντας έτσι λειτουργικό το σύστημα και τη χρησιμοποίησή του για εξαιρετικά εύστοχες εξατομικευμένες συστάσεις (Pallapa et al., 2012).

3.2 Ιδιωτικότητα και εξόρυξη δεδομένων κινητών χρηστών

Είναι ενδιαφέρον να εξετάσουμε λίγο πιο αναλυτικά τους υπάρχοντες τρόπους υποστήριξης για την εξόρυξη των δεδομένων χρήσης των χρηστών κινητών συσκευών, οι οποίοι διατηρούν την ιδιωτικότητα των χρηστών. Οι περισσότερες μέθοδοι χρησιμοποιούν κάποια μορφή μετασχηματισμού στα εξαγόμενα δεδομένα, ώστε να επιτύχουν την προστασία της ιδιωτικότητας. Τυπικά, τέτοιες μέθοδοι μειώνουν τη διακριτότητα της αναπαράστασης των δεδομένων. Αυτή η μείωση της διακριτότητας οδηγεί βέβαια σε κάποια απώλεια της αποτελεσματικότητας της διαχείρισης των δεδομένων και των αλγορίθμων εξόρυξης (Aggarwal & Philip, 2008). Ορισμένες ενδιαφέρουσες ερευνητικές μέθοδοι και τεχνολογίες/αλγόριθμοι που ασχολούνται με τη

διαφύλαξη της ιδιωτικότητας κατά τη διαδικασία εξόρυξης των δεδομένων (privacy-preserving data mining), αναφέρονται στις επόμενες παραγράφους.

3.2.1 Τυχαιοποίηση

Η μέθοδος της τυχαιοποίησης (randomization method) είναι μια τεχνική στην οποία προστίθεται θόρυβος στα δεδομένα προκειμένου να γίνει απόκρυψη των πραγματικών τιμών των δεδομένων, των εγγραφών (records). Για κάθε εγγραφή προστίθεται ένα συστατικό θορύβου και έτσι η κάθε νέα εγγραφή ισούται με την παλιά, συν την προσθήκη του θορύβου. Ο θόρυβος που προστίθεται είναι βάσει μιας κατανομής πιθανοτήτων (probability distribution) και επιπλέον είναι αρκετά μεγάλος, ώστε η τιμή της επιμέρους εγγραφής να μη μπορεί να ανακτηθεί. Κατά συνέπεια, δεν μπορούν να ανακτηθούν οι αρχικές (πρωτότυπες) εγγραφές, αλλά μπορεί να ανακτηθεί η κατανομή αυτών, η οποία περιέχει τη συμπεριφορά όπως λέγεται αυτών. Κατάλληλες λοιπόν τεχνικές σχεδιάζονται με σκοπό να αντληθούν αθροιστικές κατανομές (aggregate distributions) από τις παραμορφωμένες (perturbed) εγγραφές.

Ένα βασικό πλεονέκτημα της μεθόδου τυχαιοποίησης είναι ότι είναι σχετικά απλή και δεν απαιτεί τη γνώση της κατανομής των άλλων εγγραφών στα δεδομένα. Αυτό δεν ισχύει για τις άλλες μεθόδους, όπως η k-ανωνυμία οι οποίες απαιτούν τη γνώση των άλλων εγγραφών στα δεδομένα. Συνεπώς, η μέθοδος τυχαιοποίησης μπορεί να εφαρμόζεται κατά τον χρόνο συλλογής των στοιχείων, και δεν απαιτεί τη χρήση ενός αξιόπιστου διακομιστή που περιέχει όλες τις πρωτότυπες εγγραφές για να εκτελέσει τη διαδικασία ανωνυμίας. Ενώ αυτό είναι ένα πλεονέκτημα, οδηγεί επίσης σε ορισμένες αδυναμίες, δεδομένου ότι αντιμετωπίζει όλες τις εγγραφές εξίσου, ανεξάρτητα από την τοπική τους πυκνότητα. Ως εκ τούτου, οι ‘ακραίες’ εγγραφές (outlier records) είναι πιο επιρρεπείς σε επιθέσεις σε σύγκριση με εγγραφές σε πιο πυκνές περιοχές των δεδομένων.

3.2.2 Ομαδική ανωνυμοποίηση

Η μέθοδος της τυχαιοποίησης είναι μια απλή τεχνική, η οποία μπορεί να υλοποιηθεί κατά τη διάρκεια της συλλογής των δεδομένων, γιατί ο θόρυβος που προστίθεται σε μία εγγραφή είναι ανεξάρτητος από τη συμπεριφορά των άλλων εγγραφών. Σε περιπτώσεις βεβαίως στις οποίες η διαδικασία διαφύλαξης της ιδιωτικότητας δε χρειάζεται να εφαρμόζεται κατά τη διάρκεια της συλλογής των δεδομένων, είναι επιθυμητό να υπάρχει μια τεχνική στην οποία το επίπεδο της ανακρίβειας (inaccuracy) εξαρτάται από τη συμπεριφορά της γειτονιάς-τοπικότητας (locality) της κάθε εγγραφής. Μια άλλη βασική αδυναμία της μεθόδου της τυχαιοποίησης είναι ότι δεν εξετάζει το ενδεχόμενο ότι δημοσίως διαθέσιμες εγγραφές μπορούν να χρησιμοποιηθούν για τον προσδιορισμό της ταυτότητας των ιδιοκτητών των εγγραφών. Αυτό ισχύει ιδιαίτερα στις ‘ακραίες’ εγγραφές, οι οποίες μπορούν εύκολα να διακριθούν από άλλες εγγραφές που ανήκουν στην ίδια περιοχή. Ως εκ τούτου, μια ευρύτερη προσέγγιση ικανή για πολλές μεταμορφώσεις της ιδιωτικότητας, είναι η ομαδική ανωνυμοποίηση (group based anonymization), δηλαδή η κατασκευή ομάδων από ανώνυμες εγγραφές που μετατρέπονται με ένα τρόπο ειδικό για την ομάδα (Βασιλακάκος, 2015).

Χαρακτηριστικές μέθοδοι είναι το μοντέλο k-ανωνυμίας και l-διαφορετικότητα. Στη μέθοδο k-ανωνυμία (k-anonymity), μειώνουμε τη διακριτότητα της αναπαράστασης δεδομένων με τη χρήση τεχνικών όπως η γενίκευση (generalization) και η καταστολή (suppression). Στη μέθοδο της γενίκευσης, οι τιμές των ιδιοτήτων γενικεύονται σε ένα εύρος, έτσι ώστε να επιτευχθεί μείωση της διακριτότητας της αναπαράστασης. Στη μέθοδο της καταστολής, η τιμή της ιδιότητας αφαιρείται ολοκληρωτικά. Είναι ξεκάθαρο ότι τέτοιες μέθοδοι μειώνουν το ρίσκο της αναγνώρισης από τη χρήση δημόσιων δεδομένων, καθώς μειώνουν την ακρίβεια των μετασχηματισμένων δεδομένων. Αυτή η μείωση διακριτότητας γίνεται επαρκώς έτσι ώστε οποιαδήποτε δεδομένη εγγραφή αντιστοιχεί σε τουλάχιστον k άλλες εγγραφές στα δεδομένα. Το μοντέλο της l-διαφορετικότητας (l-diversity) σχεδιάστηκε για να χειριστεί κάποιες αδυναμίες του μοντέλου k-ανωνυμία: η προστασία των ταυτοτήτων στο επίπεδο των k-ατόμων δεν είναι ταυτόσημη με την προστασία των αντίστοιχων ευαίσθητων τιμών, ειδικά όταν υπάρχει ομοιογένεια των ευαίσθητων τιμών μέσα σε μια ομάδα. Αυτό επιτυγχάνεται με την υποστήριξη της έννοιας της διαφορετικότητας μέσα στην ομάδα των ευαίσθητων τιμών (intra-group diversity) στο πλαίσιο του σχήματος ανωνυμοποίησης.

Μία ενδεικτική εφαρμογή της μεθόδου της k-ανωνυμίας σε περιβάλλοντα κινητών συσκευών, αποτελεί η δημιουργία ενός συστήματος πλέγματος απόκρυψης περιοχής για συνεχή ερωτήματα σε υπηρεσίες θέσης (location-based services) σε κατανεμημένα συστήματα (Kim et al., 2013). Το προτεινόμενο σύστημα

αποθηκεύει πληροφορίες και εκτελεί τις πράξεις με έναν κατανεμημένο τρόπο για να δημιουργήσει ένα χώρο απόκρυψης, και ταυτόχρονα να αποφύγει τη συμφόρηση στα ερωτήματα.

3.2.3 Κατανεμημένη διατήρηση της ιδιωτικότητας

Σε πολλές περιπτώσεις, τα δεδομένα είναι κατανεμημένα σε διάφορες οντότητες και μεμονωμένες οντότητες μπορεί να επιθυμούν να αποκομίσουν συγκεντρωτικά αποτελέσματα από σύνολα δεδομένων που είναι κατανεμημένα στις οντότητες αυτές. Μια τέτοια διαμέριση (partioning) μπορεί να είναι οριζόντια (όταν οι εγγραφές έχουν κατανεμηθεί σε πολλαπλές οντότητες, καθεμία από τις οποίες έχει το ίδιο σύνολο χαρακτηριστικών/πεδίων) ή κάθετη (όταν τα χαρακτηριστικά/πεδία των εγγραφών έχουν κατανεμηθεί σε πολλαπλές οντότητες). Ενώ οι μεμονωμένες οντότητες μπορεί να μην επιθυμούν να μοιραστούν ολόκληρα τα σύνολα δεδομένων τους, μπορεί να συναινούν σε περιορισμένη διαμοίραση πληροφοριών, βάσει μιας ποικιλίας από πρωτόκολλα. Το συνολικό αποτέλεσμα-στόχος των μεθόδων αυτών είναι να διατηρήσουν την ιδιωτικότητα για κάθε μεμονωμένη οντότητα, χωρίς να μειώνεται η δυνατότητά τους για παραγωγή συγκεντρωτικών αποτελεσμάτων πάνω στα συνολικά δεδομένα.

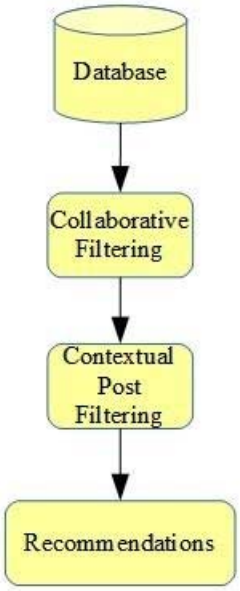
Το πρόβλημα της κατανεμημένης διατήρησης της ιδιωτικότητας (distributed privacy preservation) της εξόρυξης δεδομένων επικαλύπτεται αρκετά με πεδίο της κρυπτογραφίας που αφορά τον καθορισμό ασφαλών υπολογισμών πολλαπλών μερών. Ο σχεδιασμός του πρωτοκόλλου πρέπει να είναι τέτοιος, ώστε η συνάρτηση να υπολογίζεται χωρίς να διακυβεύεται η ιδιωτικότητα. Σημειώνουμε ότι η αξιοπιστία του εν ισχύ πρωτοκόλλου περιορισμένης διαμοίρασης πληροφοριών, εξαρτάται από το επίπεδο της εμπιστοσύνης που ο κάθε συμμετέχοντας είναι διατεθειμένος να δείξει. Αυτό συμβαίνει γιατί το πρωτόκολλο μπορεί να υπόκειται σε διάφορων ειδών αντίπαλες συμπεριφορές:

- **ημι-έντιμες συμπεριφορές**: οι συμμετέχοντες είναι περίεργοι και προσπαθούν να μάθουν από τις πληροφορίες που λαμβάνουν κατά τη διάρκεια ισχύος του πρωτοκόλλου, αλλά οι ίδιοι δεν παρεκκλίνουν από το πρωτόκολλο.
- **εχθρικές συμπεριφορές**: οι συμμετέχοντες αποκλίνουν από το πρωτόκολλο και μπορεί να στείλουν δεδομένα ο ένας στον άλλο, ώστε να μάθουν από τις παρεχόμενες πληροφορίες.

3.3 Μια προσέγγιση διαφύλαξης της ιδιωτικότητας στα κινητά συστήματα συστάσεων

Η προστασία των προσωπικών δεδομένων γίνεται ολοένα και πιο σημαντική για κινητά υπολογιστικά περιβάλλοντα. Έχουν γίνει αρκετές προσπάθειες προς την κατεύθυνση των υπηρεσιών θέσης, το οποίο όμως είναι μόνο μία από τις πολλές παραμέτρους του περιβάλλοντος που μπορεί να βρεθεί σε κινητά συστήματα συστάσεων. Σε επόμενες παραγράφους προτείνεται μια μέθοδος που προστατεύει τις μεταβλητές/παραμέτρους του πλαισίου. Η προστασία στη μέθοδο αυτή δεν κινείται μόνο προς την κατεύθυνση της μεταβλητής θέσης. Η διαδικασία, περιλαμβάνει ως στάδια το συνεργατικό φιλτράρισμα και το μετά-φιλτράρισμα πλαισίου (όπως φαίνεται στην εικόνα 7.4).

Dynamic 7.1.zip	Διαδραστική εικόνα (interactive)
Εικόνα 7.4 Διαδικασία δύο βημάτων για παραγωγή συστάσεων	

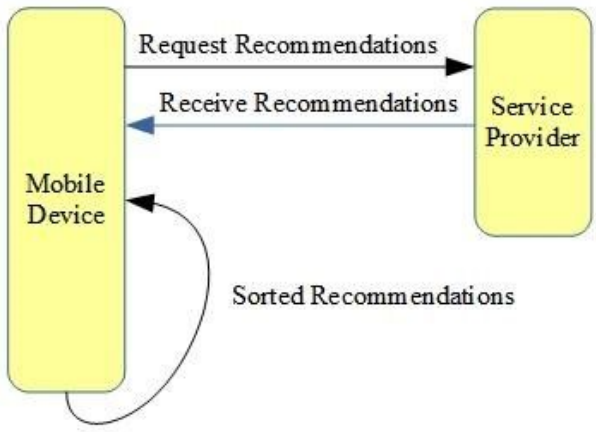


Εικόνα 7.4. *Διαδικασία δύο βημάτων για παραγωγή συστάσεων*

Το μοντέλο λειτουργίας είναι τύπου 'πελάτη-εξυπηρετητή': ο χρήστης υποβάλει ένα αίτημα για συστάσεις στον εξυπηρετητή, κατόπιν λαμβάνει μέρος το συνεργατικό φιλτράρισμα και μετά γίνεται το μετά-φιλτράρισμα. Η εικόνα 7.5 δείχνει εποπτικά τον τρόπο λειτουργίας.

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 7.2.gif	Κινούμενη εικόνα (interactive)
Αλληλεπίδραση πελάτη-εξυπηρετητή με στόχο την προστασία της ιδιωτικότητας	



Εικόνα 7.5. *Αλληλεπίδραση πελάτη-εξυπηρετητή*

3.3.1 Αλγόριθμοι διαφύλαξης ιδιωτικότητας

Ο ακόλουθος αλγόριθμος 1 παρουσιάζει τη λειτουργία από την πλευρά του πελάτη (κινητή συσκευή), ενώ ο αλγόριθμος 2 από την πλευρά του εξυπηρετητή. Είναι εμφανές ότι ο πελάτης εκτός από τις πραγματικές μεταβλητές που έχουν σχέση με το περιβάλλον πλαίσιο στέλνει ένα τμήμα με ψεύτικες μεταβλητές. Κατόπιν ο εξυπηρετητής παρέχει τις συστάσεις με βάση τις πραγματικές και τις ψεύτικες μεταβλητές χωρίς να γνωρίζει ποιες είναι οι πραγματικές. Στο επόμενο βήμα παρατηρούμε ότι όταν φτάσουν οι παρεχόμενες συστάσεις στον πελάτη, αυτομάτως οι ψεύτικες διαγράφονται και εμφανίζονται μόνο οι πραγματικές.

Algorithm 1: Recommendation request (Mobile client)

Input: User id, Context parameters
Output: Recommendations */* *List of recommendations* */*

```
Retrieve Location, Context_Parameters[n]
Generate Dummy_Location, Dummy_Context_Parameters[n]
/* one fake context parameter for each real context parameter */
Request /* to the service provider with parameters */ User id, Location, Dummy_Location,
Context_Parameters[n], Dummy_Context_Parameters[n]
/* the service provider receives the request makes the recommendation as shown in algorithm 2 and provides
the recommendations back to the client */
Receive Recommendations /* real and false from the service provider */
For (int i=0; i<Recommendations.size; i++)
        If
        i.hasParameter (Dummy_Location)
        Delete i;
        Else
        For (int j=1; j<n; j++)
                If
                i.hasParameter (Dummy_Context_Parameters[j]
                Delete i;
                End If
        End For
        End If
End For
Return Recommendations
```

Algorithm 2: Recommendation process (Service provider)

Input: User id, Context parameters
Output: Recommendations

```
/* Starts with collaborative filtering */
Load User ratings
Load Similarity measure /* Pearson correlation similarity */
Provide Recommendations
/* Contextual post filtering follows */
For (int i=0; i<Recommendations.size; i++)
  If
  i.hasParameter != (Location || Dummy_Location)
  Delete i;
  Else
  For (int j=1; j<n; j++)
        If
        i.hasParameter != (Context_Parameters[j] || Dummy_Context_Parameters[j])
        Delete i;
        End If
  End For
  End If
End For
Return Recommendations
```

3.3.2 Αξιολόγηση των αλγορίθμων προστασίας

Στην παράγραφο αυτή παρουσιάζονται αποτελέσματα μιας πρώτης αξιολόγησης των προηγουμένων αλγόριθμων προστασίας των ιδιωτικών δεδομένων. Να σημειωθεί ότι ο εξοπλισμός ήταν ένας απλός υπολογιστής Pentium i3 (2.13GHz, 4GB RAM, Windows 8.1), ενώ όλοι οι αλγόριθμοι υλοποιήθηκαν στη γλώσσα προγραμματισμού Java. Χρησιμοποιήθηκε ένα πραγματικό σετ δεδομένων το οποίο περιλαμβάνει ένα πλούσιο σύνολο μεταβλητών περιβάλλοντος πλαισίου. Το σετ ονομάζεται LDOS-CoMoDa (Košir et al., 2011). Το σετ περιλαμβάνει αξιολογήσεις ταινιών στην κλίμακα 1-5 και επίσης 12 μεταβλητές πλαισίου που περιγράφονται αναλυτικά στον πίνακα 7.2. Επίσης, ο πίνακας 7.3 μας δίνει στατιστικές πληροφορίες σχετικά με το χρησιμοποιούμενο σετ.

Παράμετρος	**Τιμές**	**Περιγραφή τιμών**
Ώρα	1 ως 4	1=Πρωί, 2=Μεσημέρι, 3=Απόγευμα, 4=Βράδυ
Τύπος ημέρας	1 ως 3	1=Καθημερινή, 2=Σάββατο/Κυριακή, 3=Αργία
Εποχή	1 ως 4	1=Άνοιξη, 2=Καλοκαίρι, 3=Φθινόπωρο, 4=Χειμώνας
Τοποθεσία	1 ως 3	1=Σπίτι, 2=Δημόσιος χώρος, 3=Σπίτι φίλου
Καιρός	1 ως 5	1=Ήλιος, 2=Βροχή, 3=Καταιγίδα, 4=Χιόνι, 5=Συννεφιά
Κοινωνικές επαφές	1 ως 7	1=Μόνος, 2=Με συνεργάτες, 3=Με φίλους, 4=Με συναδέλφους, 5=Με γονείς, 6=Με άλλους, 7=Με οικογένεια
Τελική διάθεση	1 ως 7	1=Στενάχωρος, 2=Χαρούμενος, 3=Φοβισμένος, 4=Έκπληκτος, 5=Θυμωμένος, 6=Αηδιασμένος, 7=Ουδέτερος
Κυρίαρχη διάθεση	1 ως 7	1=Στενάχωρος, 2=Χαρούμενος, 3=Φοβισμένος, 4=Έκπληκτος, 5=Θυμωμένος, 6=Αηδιασμένος, 7=Ουδέτερος
Γενική διάθεση	1 ως 3	1=Θετική, 2=Ουδέτερη, 3=Αρνητική
Φυσική κατάσταση	1 ως 2	1=Υγιής, 2=Άρρωστος
Απόφαση	1 ως 2	1=Από τον χρήστη, 2=Από άλλους
Αλληλεπίδραση	1 ως 2	1=Πρώτη, 2=Δεύτερη και πάνω

Πίνακας 7.2. *Περιγραφή μεταβλητών του σετ δεδομένων LDOS-CoMoDa*

Μεταβλητές	**Τιμές**
Χρήστες	95
Ταινίες	961
Αξιολογήσεις	1665

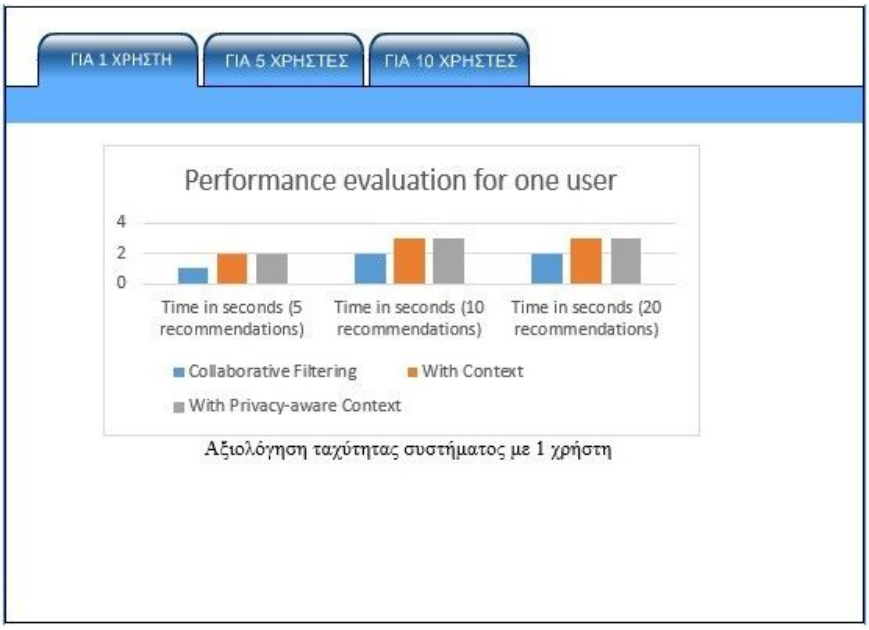
Μέση ηλικία των χρηστών	27
Χώρες καταγωγής	6
Πόλεις καταγωγής	18
Αριθμός περισσότερων αξιολογήσεων από ένα χρήστη	220
Αριθμός λιγότερων αξιολογήσεων από ένα χρήστη	1

Πίνακας 7.3. *Στατιστική περιγραφή μεταβλητών του σετ δεδομένων LDOS-CoMoDa*

Στο σενάριο χρήσης του αλγόριθμου έχουμε χρήστη ο οποίος ζητάει από το σύστημα να του παράσχει συστάσεις, με την προϋπόθεση ότι η ιδιωτικότητα του θα είναι προστατευμένη. Έτσι λοιπόν για κάθε μια πραγματική μεταβλητή του χρήστη το σύστημα, στο κινητό του τηλέφωνο, διαλέγει και μία ακόμη στην τύχη, από κάθε κατηγορία, εκτός από την ώρα. Μετά οι πληροφορίες στέλνονται στον εξυπηρετητή, όπου ο αλγόριθμος παρέχει τις συστάσεις με βάση όλες αυτές τις πληροφορίες του περιβάλλοντος, σωστές και μη. Τέλος, οι συστάσεις αποστέλλονται στο κινητό του χρήστη, όπου οι μη σωστές διαγράφονται αυτόματα. Η εικόνα 7.6 δείχνει συγκριτικά τις ταχύτητες παραγωγής συστάσεων όταν ο εξυπηρετητής παρέχει συστάσεις σε ένα χρήστη, όταν ο εξυπηρετητής παρέχει συστάσεις σε 5 χρήστες και όταν οι χρήστες είναι 10. Είναι εμφανές ότι όταν το πλήθος χρηστών είναι μικρό, τότε το 'κόστος' της προστασίας ιδιωτικότητας είναι πολύ μικρό. Όμως η εικόνα αλλάζει όταν αυξάνεται το πλήθος (περίπτωση 10 χρηστών).

Στο παρακάτω σχήμα επιλέξτε μια από τις τρεις καρτέλες

Flash 7.1.swf	Αρχείο flash (interactive)
Αξιολόγηση ταχύτητας συστήματος	



Εικόνα 7.6. *Αξιολόγηση ταχύτητας συστήματος*

4. Συμπεράσματα

Η ασφάλεια των συναλλαγών σε περιβάλλοντα κινητού ηλεκτρονικού εμπορίου είναι μια από τις πιο κρίσιμες παραμέτρους για την υιοθέτηση των διαδικασιών ηλεκτρονικού εμπορίου από τους χρήστες κινητών συσκευών. Το τοπίο των απειλών είναι αρκετά σύνθετο: κακόβουλο και ανεπιθύμητο λογισμικό μπορεί να θέσει σε κίνδυνο την εμπιστευτικότητα αλλά και την ακεραιότητα διάφορων ευαίσθητων προσωπικών δεδομένων ή κρίσιμων πληροφοριών που διατηρεί ο χρήστης στην κινητή συσκευή του. Η ευαισθητοποίηση σε ζητήματα ασφάλειας των συμμετεχόντων σε κινητές συναλλαγές, και η υπεύθυνη χρήση όλων των

προβλεπόμενων διαδικασιών ορθής χρήσης, αποτελεί ουσιαστική δικλείδα επιτυχίας στην αποτροπή οποιασδήποτε προσπάθειας παραβίασης των προβλεπόμενων λειτουργιών. Ιδιαίτερα τα ζητήματα διαφύλαξης της ιδιωτικότητας, αποτελούν πρόκληση κατά τη λειτουργία των μηχανισμών παραγωγής συστάσεων: σε περιβάλλοντα κινητού εμπορίου, η τεχνολογία της εξατομίκευσης θεωρείται άκρως απαραίτητη ως μέσο φιλτραρίσματος του μεγάλου όγκου πληροφοριών που καλείται ο κινητός χρήστης να 'αντιμετωπίσει'. Η ευστοχία των συστάσεων επηρεάζεται από την αξιοποίηση των πληροφοριών πλαισίου (context), όμως ακριβώς αυτό το πλαίσιο αποτελεί ένα σύνολο πληροφοριών που η μη-ορθή χρήση του μπορεί να παραβιάσει την ιδιωτικότητα των χρηστών. Συνεπώς, οι σύγχρονοι μηχανισμοί εξατομίκευσης, επηρεαζόμενοι από αλγορίθμους εξόρυξης δεδομένων χρήσης των κινητών χρηστών, προσπαθούν να προστατέψουν τις παραμέτρους πλαισίου (πχ. τη θέση του χρήστη), με σκοπό την οικοδόμηση κλίματος εμπιστοσύνης στους κινητούς χρήστες.

Βιβλιογραφία / Αναφορές

Adomavicius, G., & Tuzhilin A. (2011). Context-aware recommender systems. *Recommender systems handbook*. Springer US. 217-253.

Aggarwal, C. C., & Philip, S. Y. (2008). A general survey of privacy-preserving data mining models and algorithms. *Privacy-Preserving Data Mining*, Volume 34 of the series Advances in Database Systems, Springer US, pp. 11-52.

Aïmeur, E., Brassard, G., Fernandez, J. M., & Onana, F. S. M. (2008). Alambic: a privacy-preserving recommender system for electronic commerce. *International Journal of Information Security*, 7(5), 307-334.

Astani, M., Ready, K., & Tessema, M. (2013). BYOD issues and strategies in organizations. *Issues in Information Systems*, 14(2), 195-201.

Bestmann, M., Cartier, J., & Miltner, J. (2015). U.S. Patent No. 9,009,857. Washington, DC: U.S. Patent and Trademark Office.

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, *46*, 109-132.

Boutet, A., Frey, D., Guerraoui, R., Jégou, A., & Kermarrec, A. M. (2015). Privacy-preserving distributed collaborative filtering. In *Computing.* dx.doi.org/10.1007/s00607-015-0451-z

Cisco (2012). Survey Report: BYOD: A Global Perspective Harnessing Employee-Led Innovation. CISCO IBSG, διαθέσιμη στη διεύθυνση: https://www.cisco.com/web/about/ac79/docs/re/BYOD_Horizons-Global.pdf

del Carmen Rodríguez-Hernández, M., & Ilarri, S. (2014). Towards a Context-Aware Mobile Recommendation Architecture. In *Mobile Web Information Systems* (pp. 56-70). Springer International Publishing.

Dimensional Research (2013). The impact of mobile devices on information security: A survey of IT professionals, διαθέσιμη στη διεύθυνση: http://www.checkpoint.com/downloads/products/check-point-mobile-security-survey-report2013.pdf

Elenkov, N. (2014). *Android Security Internals: An In-depth Guide to Android's Security Architecture*. No Starch Press.

Felt, A. P., Finifter, M., Chin, E., Hanna, S., & Wagner, D. (2011). A survey of mobile malware in the wild. In *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices*, Chicago, USA.

Georgiadis, C., Stiakakis, E., & Andronoudi, A. (2014). The Significance of Mobile Security Breaches in Terms of Their Economic Impact on Users. In *Proc. of the Int. Conf. on Mobile Business* (ICMB 2014), AISeL, London, UK.

Halilovic, M., & Subasi, A. (2012). Intrusion Detection on Smartphones. arXiv preprint arXiv:1211.6610.

Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2010). *Recommender systems: an introduction*. Cambridge University Press.

Jensen, C. S., Lu, H., & Yiu, M. L. (2009). Location privacy techniques in client-server architectures. In *Privacy in location-based applications* (pp. 31-58). Springer Berlin Heidelberg.

Juniper Networks (2011), Mobile Device Security - Emerging Threats, Essential Strategies, White Paper

Keyes, J. (2013). *Bring Your Own Devices (BYOD) Survival Guide*. CRC Press, Boca Raton, FL.

Kido, H., Yanagisawa, Y., & Satoh, T. (2005, July). An anonymous communication technique using dummies for location-based services. In *Pervasive Services*, 2005. ICPS'05. Proceedings. International Conference on (pp. 88-97). IEEE.

Kim, H.-I., Kim, Y.-K., & Chang, J.-W. (2013). A Grid-based Cloaking Area Creation Scheme for Continuous LBS Queries in Distributed Systems. Future Technology Research Association International, IV (1), pp. 23-30.

Konstan, J. A., & Riedl, J. (2012). Recommender systems: from algorithms to user experience. *User Modeling and User-Adapted Interaction* 22.1-2 (2012): 101-123.

Košir, A., Odic, A., Kunaver, M., Tkalcic, M., & Tasic, J. F. (2011). Database for contextual personalization. *Elektrotehniški vestnik*, *78*(5), 270-274.

La Polla, M., Martinelli, F. & Sgandurra, D. (2013). A survey on security for mobile devices. *IEEE Communications Surveys & Tutorials*, 15 (1), 446-471.

Lennon, R.G., 2012. Bring your own device (BYOD) with cloud 4 education. In *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity*, ACM, pp. 171–180.

Liu, Q., Ma, H., Chen, E., & Xiong, H. (2013). A survey of context-aware mobile recommendations. *International Journal of Information Technology & Decision Making*, *12*(01), 139-172.

Liu, B., Lin, J., & Sadeh, N. (2014). Reconciling Mobile App Privacy and Usability on Smartphones, In the *ACM Proceedings of the 23rd International World Wide Web Conference* (WWW'14), Seoul, Korea.

McAfee (2012). Putting IT Back in Control of BYOD. Osterman Research Inc., USA, διαθέσιμη στη διεύθυνση: http://www.mcafee.com/us/resources/reports/

Mettouris, C. & Papadopoulos, G.A. (2014). Ubiquitous recommender systems. *Computing* 96.3: 223-257.

Microsoft (2013). Survey Shows People Need More Help Controlling Personal Info Online. Microsoft Corporation, διαθέσιμη στη διεύθυνση: http://news.microsoft.com/2013/01/23/survey-shows-people-need-more-help-controlling-personal-info-online/

Mitchell, R., & Chen, I. R. (2014). A survey of intrusion detection techniques for cyber-physical systems. *ACM Computing Surveys (CSUR)*, *46*(4), 55.

Mobile Security Reference Architecture (2013), Federal CIO Council and Department of Homeland Security National Protection and Program Directorate Office of Cybersecurity and Communications Federal Network Resilience, διαθέσιμη στη διεύθυνση: https://cio.gov/wpcontent/uploads/2013/05/Mobile-Security-Reference-Architecture.pdf

Mont, J. (2012). The Risks and Benefits of Employee-Owned Devices. *Compliance and Technology*, pp.48-52.

Morrow, B. (2012). BYOD security challenges: control and protect your most sensitive data. *Network Security*, 2012 (12), 5-8.

Pallapa, G., Di Francesco, M., & Das, S. K. (2012). Adaptive and context-aware privacy preservation schemes exploiting user interactions in pervasive environments. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a* (pp. 1-6). IEEE.

Phifer, L. (2013). Bring your own danger. *Information Security*, pp. 29-35.

Pillay, A., Diaki, H., Nham, E., Senanayake, S., Tan, G. & Deshpande, S. (2013). Does BYOD increase risks or drive benefits?, διαθέσιμη στη διεύθυνση: http://sitic.org/wp-content/uploads/Does-BYOD-increase-risks-or-drive-benefits.pdf

Polat, H., & Du, W. (2005). Privacy-preserving collaborative filtering. *International Journal of Electronic Commerce*, *9*(4), 9-35.

Polatidis, N., & Georgiadis, C. K. (2013). Mobile recommender systems: An overview of technologies and challenges. In *Informatics and Applications* (ICIA), 2013 Second International Conference on (pp. 282-287). IEEE.

Polatidis, N., & Georgiadis, C. K. (2014). Factors Influencing the Quality of the User Experience in Ubiquitous Recommender Systems. In Distributed, Ambient, and Pervasive Interactions (pp. 369-379). Springer International Publishing.

Ricci, F. (2010). Mobile recommender systems. *Information Technology & Tourism*, 12(3), 205-231.

Scipioni, M. P. (2011). Towards privacy-aware location-based recommender systems. IFIP Summer School.

Shi, Y., Larson, M., & Hanjalic, A. (2014). Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys* (CSUR), 47(1), 3.

Six, J. (2012). *Application Security for the Android Platform: Processes, Permissions, and Other Safeguards*. O' Reilly Media, Inc.

Sun, Y., Chong, W. K., Han, Y. S., Rho, S., & Man, K. L. (2015). Key Factors Affecting User Experience of Mobile Recommendation Systems. In *Proceedings of the International MultiConference of Engineers and Computer Scientists* (Vol. 2).

Tang, A., Sethumadhavan, S., & Stolfo, S. J. (2014). Unsupervised anomaly-based malware detection using hardware features. In *Research in Attacks, Intrusions and Defenses* (pp. 109-129). Springer International Publishing.

Thomson, G. (2012). BYOD: enabling the chaos. *Network Security*, 2012 (2), pp. 5–8.

Trend Micro (2012). A Brief History of Mobile Malware, White paper, διαθέσιμη στη διεύθυνση: http://countermeasures.trendmicro.eu/wp-content/uploads/2012/02/History-of-Mobile-Malware.pdf

Βασιλακάκος, Δ. (2015). Κινητικότητα, Εξόρυξη Δεδομένων και Ιδιωτικότητα, Διπλωματική Εργασία, ΜΠΣ Σχεδίαση και Ανάπτυξη Διάχυτων Συστημάτων Υπολογισμού, Ελληνικό Ανοικτό Πανεπιστήμιο.

Μάγκος, Ε. (2013). Ασφάλεια Υπολογιστών και Προστασία Δεδομένων, Πανεπιστημιακές Σημειώσεις, Ιόνιο Πανεπιστήμιο.

Χαλκίδης, Μ., Βαζιργιάννης, Μ. (2005). *Εξόρυξη Γνώσης από Βάσεις Δεδομένων και τον Παγκόσμιο Ιστό, 2η έκδοση*, εκδόσεις ΤΥΠΩΘΗΤΩ.

Quiz7.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Τα συστήματα συστάσεων σε κινητά περιβάλλοντα πρέπει να χρησιμοποιούν οπωσδήποτε τη μέθοδο συνεργατικού φιλτραρίσματος

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 2

[*] Η τεχνολογία NFC έχει αναπτυχθεί για την επικοινωνία σε πολύ κοντινές αποστάσεις

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Α) Σωστό

Κριτήριο αξιολόγησης 3

[*] Η περιβάλλουσα κατάσταση (context) βασίζεται

Α) Στην ώρα

Β) Στην τοποθεσία

Γ) Σε άλλες μεταβλητές όπως ο καιρός και οι άλλοι χρήστες

Δ) Σε συνδυασμό των παραπάνω

Απάντηση/Λύση

Δ) Σε συνδυασμό των παραπάνω

Κριτήριο αξιολόγησης 4

[*] Το περιβάλλον πλαίσιο (context) χρησιμοποιείται:

Α) Ως μέσο παράκαμψης των μηχανισμών προστασίας ιδιωτικότητας

Β) Ως μέσο παροχής εύστοχων εξατομικευμένων συστάσεων

Γ) Αποκλειστικά ως μέσο υποβοήθησης πολιτικών ελέγχου προσπέλασης

Απάντηση/Λύση

Β) Ως μέσο παροχής εύστοχων εξατομικευμένων συστάσεων

Κριτήριο αξιολόγησης 5

[*] Η τεχνολογία NFC χρησιμοποιείται αποκλειστικά για πραγματοποίηση κινητών πληρωμών

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 6

[**] Με τον όρο ανίχνευση κακής χρήσης αναφερόμαστε:

Α) Στην παροχή άστοχων συστάσεων σε σχέση με το περιβάλλον πλαίσιο

Β) Στην ανάλυση της πληροφορίας που έχει συγκεντρωθεί (από τα αρχεία καταγραφής) και τη σύγκριση των αποτελεσμάτων της ανάλυσης με ήδη γνωστές επιθέσεις

Γ) Στην ανίχνευση δραστηριοτήτων που δε συμβαδίζουν με το συνηθισμένο προφίλ

Απάντηση/Λύση

Β) Στην ανάλυση της πληροφορίας που έχει συγκεντρωθεί (από τα αρχεία καταγραφής) και τη σύγκριση των αποτελεσμάτων της ανάλυσης με ήδη γνωστές επιθέσεις

Κριτήριο αξιολόγησης 7

[**] Τα στοιχεία που συλλέγονται από τα συστήματα ανίχνευσης εισβολών:

Α) Στηρίζονται σε διάφορες μετρήσεις, όπως η δραστηριότητα επεξεργαστή (CPU activity)

Β) Στηρίζονται σε τεχνολογίες οι οποίες ανιχνεύουν και καταγράφουν μια 'ύποπτη' ενέργεια, χωρίς να την αποτρέπουν.

Γ) Στηρίζονται σε τεχνολογίες με ενσωματωμένες λειτουργίες αντιμετώπισης επιθέσεων, οι οποίες αντιδρούν για την εξουδετέρωση του κακόβουλου λογισμικού

Απάντηση/Λύση

Α) Στηρίζονται σε διάφορες μετρήσεις, όπως η δραστηριότητα επεξεργαστή (CPU activity)

Κριτήριο αξιολόγησης 8

[*] Ποια από τις ακόλουθες προτάσεις δεν είναι σωστή;

Α) Οι χρήστες επιθυμούν να αποθηκεύουν στις κινητές συσκευές τους εμπιστευτικά αρχεία του εργασιακού τους περιβάλλοντος

Β) Οι χρήστες δεν επιθυμούν να αποθηκεύουν τραπεζικούς κωδικούς στις κινητές συσκευές τους

Γ) Η πλειοψηφία των χρηστών συνηθίζει να αποθηκεύει στις συσκευές της προσωπικά δεδομένα άλλων χρηστών

Απάντηση/Λύση
Β) Σπάνια οι χρήστες αποθηκεύουν τραπεζικούς κωδικούς στις κινητές συσκευές τους

Κριτήριο αξιολόγησης 9
[**] Χαρακτηριστικές μέθοδοι προστασίας της ιδιωτικότητας στην κατηγορία ομαδικής ανωνυμοποίησης είναι:

Α) οι μέθοδοι απόκρυψης και παραμόρφωσης εγγραφών

Β) οι μέθοδοι τυχαιοποίησης και διαμέρισης

Γ) οι μέθοδοι k-ανωνυμίας και l-διαφορετικότητας

Απάντηση/Λύση
Γ) οι μέθοδοι k-ανωνυμίας και l-διαφορετικότητας

Κριτήριο αξιολόγησης 10
[**] Ποια από τις ακόλουθες προτάσεις είναι σωστή;

Α) Η μέθοδος τυχαιοποίησης δεν μπορεί να εφαρμόζεται κατά τον χρόνο συλλογής των στοιχείων

Β) Κατά την εφαρμογή της μεθόδου τυχαιοποίησης, το επίπεδο της ανακρίβειας εξαρτάται από τις τιμές της γειτονιάς (locality) της κάθε εγγραφής.

Γ) Η ομαδική ανωνυμοποίηση δεν μπορεί να εφαρμόζεται κατά τον χρόνο συλλογής των στοιχείων

Απάντηση/Λύση
Γ) Η ομαδική ανωνυμοποίηση δεν μπορεί να εφαρμόζεται κατά τον χρόνο συλλογής των στοιχείων

Κεφάλαιο 8: Ανάπτυξη Περιεχομένου για Κινητές Συσκευές: Εργαστηριακές Ασκήσεις

Σύνοψη
Στο κεφάλαιο αυτό αναλύονται αρχικά οι κατηγορίες ανάπτυξης περιεχομένου για κινητές συσκευές και στη συνέχεια παρουσιάζεται ένα σύνολο εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών), με σκοπό την κατανόηση τεχνικών προγραμματισμού ανάπτυξης περιεχομένου για κινητές συσκευές, κυρίως από την πλευρά του πελάτη (client-side scripting). Οι επιμέρους τεχνολογίες που αξιοποιούνται είναι η τεχνολογία JavaScript με έμφαση στο πλαίσιο mobile jQuery για κατασκευή ιστοσελίδων mobile Web, οι 'διευκολύνσεις' του περιβάλλοντος ανάπτυξης υβριδικών εφαρμογών PhoneGap και η γλώσσα προγραμματισμού Java για εγγενείς κινητές εφαρμογές περιβάλλοντος Android.

Προαπαιτούμενη γνώση
Τα κεφάλαια 1, 5 και 6 του παρόντος συγγράμματος, και επιπλέον θα είναι χρήσιμη κάποια προηγούμενη εμπειρία σε ζητήματα Προγραμματισμού Υπολογιστών.

1. Εισαγωγή

Μπορούμε να διακρίνουμε (Γαβαλάς και άλλοι, 2015; Firtman, 2013) τρεις μεγάλες κατηγορίες περιεχομένου για κινητές συσκευές, ως προς τον τρόπο ανάπτυξής του: α) οι *ιστοσελίδες κινητού Ιστού* (mobile Web), οι οποίες χρησιμοποιούν τις κυρίαρχες τεχνολογίες ανάπτυξης περιεχομένου για τον παγκόσμιο Ιστό (κυρίως τις τεχνολογίες HTML5, CSS3, JavaScript και jQuery) και οι οποίες «προσφέρονται» στον κινητό χρήστη μέσω του προγράμματος περιήγησης που παρέχει η κινητή συσκευή (mobile browser), β) οι *υβριδικές κινητές εφαρμογές* (hybrid mobile applications), οι οποίες χρησιμοποιούν και αυτές τις ίδιες τεχνολογίες ανάπτυξης περιεχομένου που χρησιμοποιούν οι κινητές ιστοσελίδες της προηγούμενης κατηγορίας, αλλά διαφοροποιούνται στο ότι η σχεδίασή της διάδρασής τους δε στοχεύει στη χρήση του browser από τον κινητό χρήστη, διότι μέσω κάποιων επιπλέον διαδικασιών ελέγχου τελικά οι υβριδικές αυτές εφαρμογές 'συσκευάζονται' και διατίθενται ως κινητές εφαρμογές μέσω του λειτουργικού συστήματος της κινητής συσκευής, και γ) οι *γνήσιες, εγγενείς κινητές εφαρμογές* (native mobile applications), για τις οποίες η ανάπτυξη του κώδικα βασίζεται στο συγκεκριμένο λειτουργικό σύστημα που διαθέτει η κινητή συσκευή, και οι οποίες βέβαια, όπως και οι υβριδικές εφαρμογές της προηγούμενης κατηγορίας, διατίθενται ως κινητές εφαρμογές μέσω του λειτουργικού συστήματος της κινητής συσκευής.

1.1. Ιστοσελίδες κινητού Ιστού (mobile Web development)

Πλεονεκτήματα-μειονεκτήματα:

- Δεν υπάρχουν πολλές διαδικασίες και έλεγχοι για να γίνει διαθέσιμη μια εφαρμογή/ιστοσελίδα κινητού Ιστού. όπως συμβαίνει στις άλλες κατηγορίες εφαρμογών που στοχεύουν στη διάθεση μέσω του app store του λειτουργικού συστήματος της εφαρμογής. Φτιάχνουμε την εφαρμογή, την ανεβάζουμε και τρέχει χωρίς περιττές και χρονοβόρες διαδικασίες έγκρισης.
- Μπορούμε να έχουμε έναν κοινό κώδικα που εξυπηρετεί πολλές διαφορετικές εφαρμογές. Επομένως το κόστος ανάπτυξης και συντήρησης είναι μικρότερο.
- Το τελικό προϊόν δε θα είναι ποτέ τέλειο όπως μια καθαρά native εφαρμογή. Δε θα υπάρχει ομαλότητα (smoothness) όπως συμβαίνει με τον native κώδικα που ταιριάζει κατάλληλα σε ένα κινητό με το αντίστοιχο λειτουργικό.

Τι είναι το mobile Web από την πλευρά του προγραμματιστή (developer);

Είναι ένα σύνολο βέλτιστων πρακτικών, πρότυπων σχεδίασης - ένα νέο ‘πλαίσιο’ που πρέπει να μάθει για να αναπτύσσει εφαρμογές για τον κινητό Ιστό (ή αλλιώς κινητές εφαρμογές Ιστού)

- Μερικές δυσκολίες που συναντάει ο προγραμματιστής που αναπτύσσει ιστοσελίδες για σταθερούς υπολογιστές και μετά πηγαίνει στο επόμενο βήμα της ανάπτυξης για τον κινητό Ιστό:
    - Πιο αργά δίκτυα με υψηλότερη λανθάνουσα κατάσταση
    - Πιο αργό υλικό και λιγότερη διαθέσιμη μνήμη
    - Διαφορετική εμπειρία περιήγησης
    - Διαφορετικά πλαίσια χρηστών
    - Διαφορετική συμπεριφορά περιηγητών
    - Πάρα πολλά κινητά στην αγορά με διαφορετικούς περιηγητές, με διαφορετικές εκδοχές κατά την ίδια στιγμή

Μύθοι για το Mobile Web:

- **Ο κινητός Ιστός είναι απλώς το Web:** ο κινητός Ιστός μπορεί να χρησιμοποιεί τα ίδια πρωτόκολλα δικτύου όπως το κλασικό Διαδίκτυο (HTTP, HTTPS, POP3, Wireless LAN ακόμα και TCP/IP). Μπορεί τα πρωτόκολλα υποδομής να είναι τα ίδια με το κλασικό Web. Όταν αναπτύσσουμε όμως κώδικα για εφαρμογές mobile Web στοχεύουμε σε πολύ διαφορετικές συσκευές σε σχέση με αυτές που χρησιμοποιούν το κλασικό Web. Ένα ευρέως γνωστό χαρακτηριστικό είναι το μήκος της οθόνης αλλά υπάρχουν πολλές άλλες όχι και τόσο εμφανείς διαφορές.
- **Δε χρειάζεται να κάνετε κάτι ιδιαίτερο στην επιτραπέζια ιστοσελίδα σας για τους κινητούς χρήστες:** πλέον αρκετές ιστοσελίδες είναι διαθέσιμες και στο mobile Web όπως είναι στο κλασικό Web για το οποίο έχουν φτιαχτεί. Ωστόσο είναι δύσχρηστες γιατί οι χρήστες θα πρέπει να κάνουν συνεχώς zoom in και zoom out και η πλοήγηση γίνεται πολύ δύσκολη. Έτσι υπάρχουν οι εφαρμογές που αναπτύσσονται καθαρά για mobile Web και διευκολύνουν την εμπειρία του χρήστη.
- **Μια ιστοσελίδα θα πρέπει να λειτουργεί για όλες τις συσκευές**: υπάρχουν τεχνικές που μας επιτρέπουν να δημιουργήσουμε μόνο ένα αρχείο, αλλά αυτό το αρχείο θα παρέχει διαφορετικές εμπειρίες σε μια ποικιλία συσκευών, συμπεριλαμβανομένων των επιτραπέζιων υπολογιστών, κινητών τηλέφωνων, ταμπλετών, τηλεοράσεων και κονσόλων παιχνιδιών. Το όραμα αυτό ονομάζεται “one Web" και υπάρχει η τεχνική που ονομάζεται Responsive Web Design που έχει ως στόχο να το επιτύχει. Ωστόσο μέχρι τώρα δεν έχει καταστεί πλήρως εφικτό και έτσι δεν έχουν εκλείψει εντελώς οι λόγοι που οδηγούν στο ότι χρειάζεται αρκετές φορές να αναπτύσσεται διαφορετικός κώδικας για κάθε πλατφόρμα.
- **Απλά να δημιουργήσετε ένα αρχείο HTML με πλάτος 320 pixels και έχετε μια κινητή ιστοσελίδα**: Αυτός είναι ο άλλος τρόπος fast-food για να σκεφτεί κανείς το mobile Web. Σήμερα, υπάρχουν περισσότερες από 3.000 κινητές συσκευές στην αγορά με σχεδόν 30 διαφορετικά προγράμματα περιήγησης (στην πραγματικότητα περισσότερα από 300 διαφορετικά προγράμματα περιήγησης αν τα χωρίσουμε σύμφωνα με τον αριθμό έκδοσης). Δημιουργώντας ένα αρχείο HTML ως κινητή ιστοσελίδα θα είναι ένα αποτυχημένο σχέδιο. Επιπλέον, το γεγονός αυτό συμβάλλει στην πεποίθηση ότι η κινητή περιήγηση στο Διαδίκτυο είναι οδυνηρή.
- **Οι εγγενείς κινητές (native mobile) εφαρμογές θα ‘σκοτώσουν’ το mobile Web**: Κάθε λύση έχει πλεονεκτήματα και μειονεκτήματα. Το κινητό Διαδίκτυο έχει πολλά να προσφέρει με τις εγγενείς εφαρμογές. Από την άλλη μεριά, το mobile Web (και η σχετικά νέα έννοια της υβριδικής εφαρμογής - native Web app σε αντιδιαστολή του όρου native mobile app), μας προσφέρει μια μεγάλη πλατφόρμα πολλαπλών εφαρμογών, συμπεριλαμβανομένων και τοπικών εφαρμογών που δεν απαιτούν πάντα σύνδεση στο Διαδίκτυο με διευθύνσεις URL και προγράμματα περιήγησης.
- **Οι άνθρωποι δε χρησιμοποιούν τους κινητούς περιηγητές τους**: Οι συνδέσεις στο Διαδίκτυο σε όλο τον κόσμο είναι: 3,035,749,340 (42% του παγκόσμιου πληθυσμού)

[τελευταία στοιχεία: 30/6/2014, http://www.internetworldstats.com/stats.htm]
Οι συνδρομές για κινητές συσκευές είναι: 7,000,000,000 (υπολογίζονται στο τέλος του 2014, U.N. Telecommunications Agency, http://www.un.org/apps/news/story.asp?NewsID=47729#.VPd5aC7Vqhk)
Προσοχή: 2, 300,000,000 mobile broadband συνδέσεις, 32% του παγκ. πληθυσμού, υπολογίζονται στο τέλος του 2014, http://www.itu.int/net/pressoffice/press_releases/2014/23.aspx#.VPeAZy7Vqhl **ΤΕΛΙΚΑ:** Οι mobile λογαριασμοί αποτελούν το 25% της όλης χρήσης του Web (2014) (http://www.smartinsights.com/digital-marketing-strategy/internet-trends-2014-mary-meeker/)

1.2. Υβριδικές κινητές εφαρμογές (hybrid ή cross - platform development)

Πλεονεκτήματα-μειονεκτήματα:

- Απλή λογική: γράφω σε HTML5 (και κατ' επέκταση σε CSS3 και JavaScript κλπ.) αλλά το 'συσκευάζω' ως native εφαρμογή.
- Η εφαρμογή πρέπει να περάσει από κάποιες διαδικασίες ελέγχου σε σύγκριση με τις εφαρμογές/ιστοσελίδες mobile Web της προηγούμενης κατηγορίας.
- Ένα μεγάλο πλεονέκτημα είναι ότι γράφουμε μια φορά τον κώδικά μας και μπορούμε να τον τρέξουμε σε πολλές πλατφόρμες. Αυτό συνεπάγεται μειωμένο κόστος ανάπτυξης.
- Η πρόσβαση σε εγγενείς δυνατότητες είναι καλύτερη από ότι αυτές που υπάρχουν στην κατηγορία των mobile Web ιστοσελίδων.
- Το τελικό προϊόν που παράγεται ως εφαρμογή δεν είναι πάλι τέλειο (σε σύγκριση με μια εγγενή κινητή εφαρμογή), αλλά είναι αισθητικά καλύτερο από ότι θα ήταν το αποτέλεσμα με χρήση mobile Web ιστοσελίδας.

1.3. Εγγενείς κινητές εφαρμογές (native mobile development)

Πλεονεκτήματα-μειονεκτήματα:

- Η πρόσβαση σε εγγενείς δυνατότητες είναι καλύτερη σε σύγκριση με τις άλλες κατηγορίες.
- Όταν αναφερόμαστε σε εγγενείς δυνατότητες σημαίνει ότι χρησιμοποιώντας native development αξιοποιούμε καθαρά την ανάπτυξη του κώδικα της εφαρμογής μας βασισμένη στο λειτουργικό σύστημα που τρέχει στο κινητό (π.χ. iPhone - iOS ανάπτυξη κώδικα με την ObjectiveC, Android ανάπτυξη κώδικα με Java, Windows ανάπτυξη κώδικα με .NET κλπ.). Τα πλεονεκτήματα είναι εμφανώς μεγαλύτερα γιατί το τελικό προϊόν είναι αισθητικά καλύτερο (τα χρώματα, σχεδίαση κουμπιών κλπ.) και αξιοποιεί όλες τις δυνατότητες της συσκευής.
- Το κόστος άδειας ανάπτυξης κώδικα είναι υψηλό (εξαίρεση το Android).
- Αν θέλουμε να κάνουμε την εφαρμογή μας διαθέσιμη σε περισσότερα από ένα λειτουργικά συστήματα ο κώδικας της εφαρμογής θα πρέπει να γραφτεί καθαρά για κάθε εφαρμογή στη γλώσσα ανάπτυξης που υπάρχει. Το κόστος είναι πολύ υψηλό, απαιτούνται ειδικευμένοι προγραμματιστές και το κόστος συντήρησης είναι επίσης πολύ υψηλό.
- Όταν ολοκληρωθεί η εφαρμογή θα πρέπει να περάσει όλες τις διαδικασίες και τους ελέγχους των app stores.

1.4. Σύγκριση

Ακολουθεί πίνακας με σύγκριση των τριών αυτών κατηγοριών περιεχομένου για κινητές συσκευές, βασισμένη στα χαρακτηριστικά αυτών:

	Mobile Web	Hybrid	Native
Γραφικά	HTML5, CSS3, JavaScript	HTML, CSS3, JavaScript	Native APIs
Εκτέλεση	Αργή	Όχι τόσο αργή	Γρήγορη
Native εμφάνιση και αίσθηση	Απομίμηση	Απομίμηση	Native
Διανομή	Διαδίκτυο	App Store	App Store
Συνδεσιμότητα	Κυρίως εντός σύνδεσης	Εντός και εκτός σύνδεσης	Εντός και εκτός σύνδεσης
Δεξιότητες Ανάπτυξης	HTML5, CSS3, JavaScript	HTML5, CSS3, JavaScript	Objective C, Java
Πρόσβαση συσκευής:			
Φωτογ. Μηχανή		√	√
Ειδοποιήσεις		√	√
Επαφές, Ημερολόγιο		√	√
Αποθήκευση χωρίς σύνδεση	Κοινόχρηστη SQL	Ασφαλής αποθήκευση αρχείων, Κοινόχρηστη SQL	Ασφαλής αποθήκευση αρχείων
Γεωγραφικοί χάρτες	√	√	√
Χειρονομίες:			
Άγγιγμα	√	√	√
Τσίμπημα, Διαδοχική επαφή		√	√

Πίνακας 8.1 *Διαφορές μεταξύ κατηγοριών περιεχομένου για κινητές συσκευές*

Sound 8.1.mp3	Ηχητικό απόσπασμα (audio)
Διαφορές Mobile Web, Hybrid και Native εφαρμογών	

2. Ανάπτυξη ιστοσελίδων κινητού Ιστού

Όπως αναφέραμε προηγουμένως, οι mobile εφαρμογές για διαφορετικές συσκευές αναπτύσσονται χρησιμοποιώντας τελείως διαφορετικές γλώσσες προγραμματισμού. Για παράδειγμα, μια Android εφαρμογή είναι γραμμένη σε Java, μια iOS εφαρμογή είναι γραμμένη σε Objective-C, και σε περιβάλλον .NET αναπτύσσεται μια Windows Phone εφαρμογή. Κάθε εταιρεία έχει το δικό της λειτουργικό σύστημα με αποτέλεσμα να μην υπάρχει φορητότητα για τη χρήση όλων των mobile εφαρμογών σε όλες τις πλατφόρμες και συσκευές. Με τις τεχνολογίες HTML5, CSS3 και JavaScript μπορούμε να δημιουργήσουμε mobile applications χωρίς να υπάρχει αυτός ο περιορισμός. Μια εφαρμογή η οποία είναι γραμμένη με τις τρεις παραπάνω γλώσσες είναι το ίδιο λειτουργική τόσο σε μια Android όσο και σε μια iOS ή οποιουδήποτε άλλου λειτουργικού συστήματος συσκευή. Αυτό συμβαίνει διότι οι Web browsers, οι οποίοι δουλεύουν με αυτές τις Web γλώσσες, είναι συμβατοί και χρησιμοποιούνται από κάθε συσκευή και λειτουργικό σύστημα.

2.1. JavaScript και jQuery

Η χρήση της JavaScript από τους web browsers επιτρέπει την αλληλεπίδραση μεταξύ των client-side scripts και των χρηστών. Επίσης ελέγχει τους browsers, και μέσω ασύγχρονης επικοινωνίας κρατάει ενημερωμένο το Document Object Model (DOM) των ιστοσελίδων. Το DOM είναι μια πλατφόρμα μέσω της οποίας αναπαριστάνονται όλα τα HTML (και γενικότερα όλα τα XML) αντικείμενα. Η αναπαράσταση των αντικειμένων αυτών αποτελείται από κόμβους οι οποίοι χωρίζονται σε δενδροειδή μορφή η οποία ονομάζεται DOM tree.

Η JavaScript είναι μια δυναμική γλώσσα προγραμματισμού η οποία έχει πολλές δυνατότητες κυρίως στην κατασκευή ιστοσελίδων και σε ότι έχει να κάνει με διαδικτυακές εφαρμογές. Θα λέγαμε ότι είναι η γλώσσα του Internet όσον αφορά την πλευρά του πελάτη (client-side scripting).

Αυτό που πετυχαίνει η JavaScript είναι ότι μπορεί να ελέγξει όλα τα στοιχεία του DOM μια ιστοσελίδας και να τα πυροδοτήσει προκαλώντας διάφορα συμβάντα. Η εξέλιξη του Διαδικτύου ανέβασε τις προσδοκίες για ένα πιο λειτουργικό Web, τόσο από την πλευρά του χρήστη όσο και από την πλευρά του developer. Αυτή τη στιγμή υπάρχουν πολλές βιβλιοθήκες οι οποίες είναι γραμμένες σε JavaScript και χρησιμοποιούνται αποκλειστικά για τη διευκόλυνση χρήσης των html στοιχείων μέσω του DOM. Μια τέτοια βιβλιοθήκη είναι η jQuery.

Η jQuery χρησιμοποιεί JavaScript κώδικα για να διευκολύνει τη διαχείριση των html στοιχείων, δημιουργεί πολύπλοκα animations, effect και ελέγχει events τα οποία χρησιμοποιούνται για τη δημιουργία δυναμικών ιστοσελίδων και web εφαρμογών. Επίσης μπορεί να χρησιμοποιηθεί για την ανάπτυξη AJAX εφαρμογών. Η JavaScript ελέγχει τα συμβάντα τα οποία ενεργοποιούνται από εξωτερικές συσκευές όπως είναι το ποντίκι και το πληκτρολόγιο, καθώς αναφερόμαστε σε desktop περιβάλλοντα. Βέβαια τα συμβάντα αυτά δεν περιορίζονται μόνο στις φυσικές ενέργειες του χρήστη αλλά και σε ενέργειες που αφορούν τη διαδραστικότητα του χρήστη σε μια ιστοσελίδα. Τα κυριότερα εξ' αυτών είναι τα Drag & Drop events, media, animation, frame και form events.

2.2. Mobile frameworks: mobile jQuery

Υπάρχουν πολλά frameworks τα οποία δίνουν τη δυνατότητα ανάπτυξης εφαρμογών χρησιμοποιώντας μόνο τις HTML5, CSS3 και JavaScript. Μια από αυτές είναι η mobile jQuery, η οποία είναι βασισμένη στην jQuery. Είναι ένα interface το οποίο διαχειρίζεται events των mobile συσκευών για τη δημιουργία mobile εφαρμογών (Matthews & Gliser, 2015; Firtman, 2012). Συμβάντα τέτοιου τύπου είναι τα tap, touch, scroll και orientation.

2.3. Ενδεικτικό παράδειγμα κώδικα σε mobile jQuery

Flash 8.1.swf	Αρχείο flash (interactive)
Εφαρμογή με χρήση εσωτερικών συνδέσμων	

Εκφώνηση:

Σε αυτή την εργαστηριακή άσκηση θα δούμε κάποια βασικά στοιχεία χρήσης του πλαισίου mobile jQuery για την κατασκευή mobile Web εφαρμογής. Στοιχεία, όπως η κεφαλίδα, το υποσέλιδο και οι σύνδεσμοι (προς μια εξωτερική σελίδα ή προς ένα άλλο τμήμα της ίδιας σελίδας, προς μια ‘σελίδα’ μέσα στη σελίδα).

Αρχείο *mJQuery1.html*

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Πολλαπλές Σελίδες</title>
```

```
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.css" />
    <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.js"></script>
  </head>

  <body>
    <div data-role="page" id="homePage">
          <div data-role="header"><h1>Καλημέρα</h1></div>
          <div data-role="content">
            <p>   Welcome to our first mobile web site. It's going to be the best site you've ever seen. Once we get some content. And a business plan. But the hard part is done!
            </p>
            <p>
                  Find out about our wonderful <a href="tmimata.html" data-prefetch="true" data-transition="pop" data-direction="reverse">departments</a>.
            </p>
            <p>
                  You can also <a href="#aboutPage">learn more</a> about Πανεπιστήμιο Μακεδονίας.
            </p>
          </div>

          <div data-role="footer">
             <h4>Copyright UoM 2015</h4>
          </div>
  </div>

  <div data-role="page" id="aboutPage" data-title="ΠΑΜΑΚ">
          <div data-role="header"><h1>Σχετικά με ΠΑΜΑΚ</h1></div>

          <div data-role="content">
            <p>
              This text talks about ΠΑΜΑΚ and how interesting it is. Most likely though you want to  <a href="#homePage">return</a> to the home page.
            </p>
          </div>

          <div data-role="footer">
             <h4>Copyright UoM 2015</h4>
          </div>
    </div>
  </body>
</html>
```

Αρχείο *tmimata.html*

```
<!DOCTYPE html>
<html>
   <head>
     <meta charset="utf-8"/>
     <meta name="viewport" content="width=device-width, initial-scale=1">
     <title>Τμήματα ΠΑΜΑΚ</title>
     <link rel="stylesheet" href="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.css" />
     <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
     <script src="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.js"></script>
```

```
  </head>

  <body>
     <div data-role="page" id="deptsPage">
        <div data-role="header"><h1>Departments</h1></div>
         <div data-role="content">
         <p>
                  Our departments include:
         </p>
           <ul>
                  <li>Εφαρμοσμένη Πληροφορική</li>
                  <li>Οργάνωση και Διοίκηση Επιχειρήσεων</li>
                  <li>Χρηματοοικονομικά και Λογιστική</li>
           </ul>
         </div>
      </div>
   </body>
</html>
```

2.4. Παράδειγμα φόρμας με κώδικα σε mobile jQuery

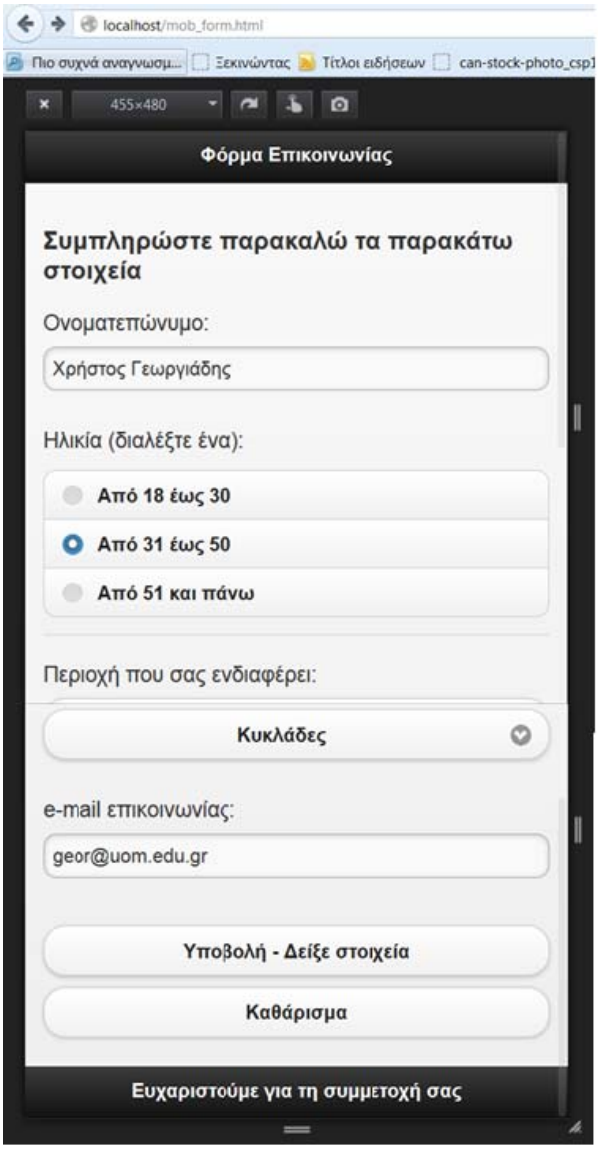
Εκφώνηση:

Δημιουργείστε μια φόρμα με το όνομα mob_form.html, που να έχει την ακόλουθη μορφή, κάνοντας χρήση του πλαισίου mobile jQuery. Στη φόρμα αυτή, τα πεδία ονοματεπώνυμο και email είναι απαραίτητα (κάντε τον έλεγχο αυτό καθώς και τον έλεγχο ορθότητας της εισαγωγής στο πεδίο email, με απλό τρόπο (μέσω ιδιοτήτων των ετικετών της φόρμας). Το πάτημα του πλήκτρου Υποβολή, όταν είναι εντάξει οι προηγούμενοι έλεγχοι, μας οδηγεί στη σελίδα mob_stoixeia.php, που εμφανίζει (κάνοντας χρήση php κώδικα) ότι έβαλε ο χρήστης στα πεδία της φόρμας. Δείτε την παρακάτω εικόνα. Σημειώστε ότι το δεύτερο κουμπί 'καθαρίζει' τις όποιες επιλογές/εισαγωγές στοιχείων έκανε ο χρήστης στη φόρμα.

Αρχείο *mob_form.html*

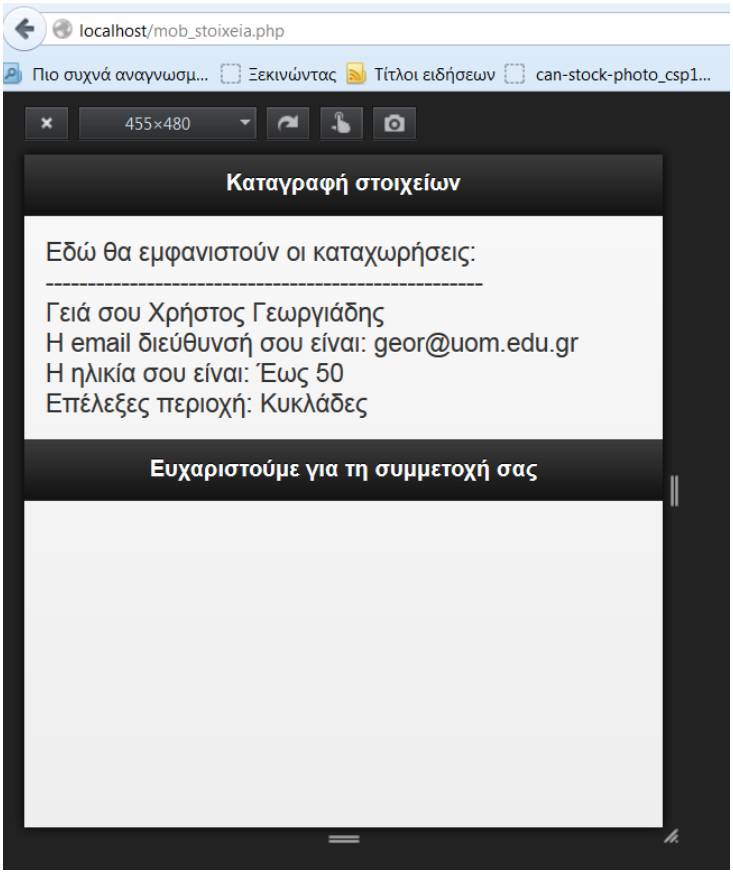
```
<!DOCTYPE html>
<html>
 <head>
   <meta charset="utf-8"/>
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <title>Φόρμα</title>
   <link rel="stylesheet" href="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.css" />
   <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
   <script src="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.js"></script>
 </head>

<body>

 <div data-role="page" id="homePage">
         <div data-role="header"><h1>Φόρμα Επικοινωνίας</h1></div>
         <div data-role="content">
         <h3>Συμπληρώστε παρακαλώ τα παρακάτω στοιχεία</h3>
    <form name="gbForm" method = "post" action = "mob_stoixeia.php">
<div>Ονοματεπώνυμο:</div>
     <div> <input type="text" name="name" size="50" required /></div>
```

```
            <br/>
<div >Ηλικία (διαλέξτε ένα):</div>
      <div data-role="fieldcontain">
          <fieldset data-role="controlgroup">
             <input type="radio" name="age" id="OS30" value="Έως 30"/>
             <label for="OS30">Από 18 έως 30</label>

             <input type="radio" name="age" id="OS50" value="Έως 50" />
             <label for="OS50">Από 31 έως 50</label>

             <input type="radio" name="age" id="OLDER" value="Πάνω από 50"/>
             <label for="OLDER">Από 51 και πάνω</label>

          </fieldset>
      </div>
      <br/>
```



Εικόνα 8.1 *Παράδειγμα φόρμας επικοινωνίας*

```
      <div>Περιοχή που σας ενδιαφέρει:</div>
      <div>
        <select name="region">
          <option value="Μακεδονία" selected="selected">Μακεδονία</option>
          <option value="Κυκλάδες">Κυκλάδες</option>
          <option value="Θράκη">Θράκη</option>
          <option value="Άλλη">'Αλλη</option>
        </select>
      </div>
      <br/>

      <div>e-mail επικοινωνίας:</div>
      <div ><input type="email" name="email" size="50" required /></div>
          <br/>
      <div style="clear: left;">
            <input type="submit" name = "submit" value="Υποβολή - Δείξε στοιχεία" />
            <input type="reset" value="Καθάρισμα" />
      </div>
    </form>
  </div>
  <div data-role="footer">
        <h4>Ευχαριστούμε για τη συμμετοχή σας</h4>
  </div>
 </div>
</body>
</html>
```



Εικόνα 8.2 *Εμφάνιση των καταχωρήσεων σε φόρμα*

Αρχείο *mob_stoixeia.php*

```
<!DOCTYPE html>
<html>
        <head>
          <meta name="viewport" content="width=device-width, initial-scale=1">
          <link rel="stylesheet" href="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.css" />
          <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
          <script src="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.js"></script>
          <meta charset="utf-8"/>
          <title>Apotelesmata</title>
          <style>
                div#problemArea {border:solid 2px red; margin:7px;
                        padding:5px;
                        background-color:#000000;
                        color: white;
                        float:left;
                        width:40%;}
           </style>
          </head>
         <body>
           <div data-role="page">
                <div data-role="header">
                   <h1>Καταγραφή στοιχείων</h1>
            </div>
            <div data-role="content">
            <div id="problemArea"> Εδώ θα εμφανιστούν οι καταχωρήσεις:
                        <br/>---------------------------------------------------- </div>
               Γειά σου <?php echo $_POST["name"]; ?><br/>
               Η email διεύθυνσή σου είναι: <?php echo $_POST["email"]; ?><br/>
               Η ηλικία σου είναι: <?php echo $_POST["age"]; ?><br/>
               Επέλεξες περιοχή: <?php echo $_POST["region"]; ?><br/>
             </div>
            <div data-role="footer">
              <h4>Ευχαριστούμε για τη συμμετοχή σας</h4>
            </div>
      </div>
   </body>
</html>
```

Sound 8.2.mp3	Ηχητικό απόσπασμα (audio)
Περιγραφή εφαρμογής Mobile Web	

3. Ανάπτυξη υβριδικών κινητών εφαρμογών

Ένα από τα πιο διαδεδομένα περιβάλλοντα ανάπτυξης υβριδικών κινητών εφαρμογών είναι το αποκαλούμενο Apache Cordova PhoneGap. Είναι ένα περιβάλλον προγραμματισμού που βασίζεται στο πολύ γνωστό περιβάλλον Eclipse (Eclipse JBoss).

3.1. Εισαγωγή

Σε αυτή την εργαστηριακή άσκηση θα αναφερθούμε στα βασικά στοιχεία του περιβάλλοντος PhoneGap, για την κατασκευή μιας υβριδικής εφαρμογής.

1. Δημιουργούμε το πρώτο μας project (File → New → Other → Mobile → Hybrid …). Βλέπουμε τα αρχεία που δημιουργούνται - τις βασικές αλλαγές τις κάνουμε αρχικά στο αρχείο index.html.

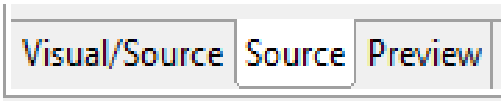
2. Ανοίγουμε το αρχείο config.xml - εκεί υπάρχει το όνομα του project μας και ρυθμίσεις

3. Πηγαίνοντας πάλι στο αρχείο index.html κάτω στα tabs βλέπουμε:

    - Visual / Source που έχει και τις δυο οπτικές τόσο του development όσο και της εκτέλεσης της εφαρμογής (πώς φαίνεται στο front-end).

    - Source: βλέπουμε καθαρά μόνο τον κώδικα (αυτή είναι η κατάσταση που θα χρησιμοποιούμε γιατί τα αποτελέσματα θα τα βλέπουμε μέσα από τη χρήση προσομοιωτών)

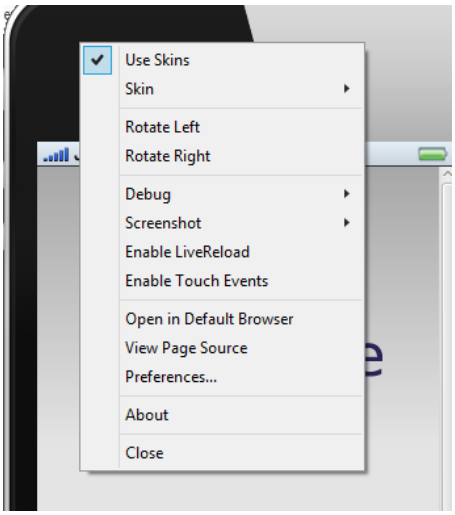
    - Preview: βλέπουμε μόνο την οθόνη εκτέλεσης της εφαρμογής



Εικόνα 8.3 *Καρτέλες προβολής περιεχομένου στο αρχείο index.html*

4. Ενδεικτικά, για να δούμε πώς φαίνεται και το αρχικό παράδειγμα της εφαρμογής μας πάμε και κάνουμε δεξί κλικ στο συνολικό μας project και μετά Run As και επιλέγουμε τον CordovaSim. Η κατασκευή αυτού του emulator βασίζεται ουσιαστικά στον Ripple emulator αλλά έχει προσαρμοστεί για να φαίνεται ως συσκευή.

5. Αυτή τη συσκευή που βλέπουμε, αν θέλουμε μπορούμε να την προσαρμόσουμε στις δικές μας προτιμήσεις. Έστω ότι κάνουμε δεξί κλικ επάνω στη συσκευή και βλέπουμε την παρακάτω λίστα:



Εικόνα 8.4 *Λίστα παραμετροποίησης εικονικής συσκευής*

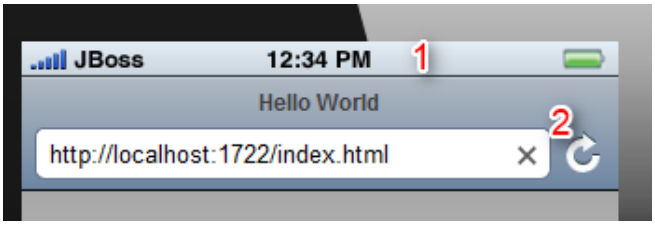
6. Αν αφαιρέσουμε το tick από το skin θα δούμε ότι ουσιαστικά αυτός ο προσομοιωτής τρέχει σε localhost. Για να επαναφέρουμε το Skin πάμε Device → Use Skin.

7. Μετά μπορούμε να χρησιμοποιήσουμε το skin για να επιλέξουμε μια άλλη συσκευή. Ενδεικτικά, πειραματιστείτε με κάποιες αλλαγές.

8. Επίσης μπορούμε να εφαρμόσουμε και την περιστροφή χρησιμοποιώντας το Rotate Left ή Rotate Right.

9. Αφήνουμε ανοικτό τον προσομοιωτή, τον τακτοποιούμε σε μια άκρη της οθόνης και πάμε στη σελίδα index.html. Ρυθμίζουμε το JBoss παράθυρο να μην τον καλύπτει, και πάμε για να κάνουμε μια μικρή αλλαγή στο h1: αντί για το υπάρχον περιεχόμενο <h1>Apache Cordova application powered by Eclipse …</h1>, βάζουμε ένα δικό μας κείμενο έστω "Η πρωτη μας εφαρμογη".

10. Αποθηκεύουμε το περιεχόμενο της σελίδας και πάμε στον προσομοιωτή που έχουμε ανοικτό και πατάμε ένα κλικ επάνω στην μπάρα που γράφει JBoss μια φορά και βλέπουμε ότι εμφανίζεται μια γραμμή με τη διεύθυνση και πατάμε του κουμπί της ανανέωσης για να δούμε την αλλαγή μας.



Εικόνα 8.5 *Διαδικασία προβολής αλλαγών στην εφαρμογή μας*

11. Ωστόσο μπορούμε να βλέπουμε τις αλλαγές μας και με μεγαλύτερη ευκολία: δηλαδή, αμέσως μόλις σώζουμε τις αλλαγές να ενημερώνεται αυτόματα ο προσομοιωτής χωρίς να χρειάζεται να ζητάμε τη διαδικασία της ανανέωσης. Αυτό μπορούμε να το καταφέρουμε αν χρησιμοποιήσουμε τον LiveReload Sever που τρέχει τοπικά. Για να τον δούμε πάμε στο tab Servers. Βλέπουμε εκεί ότι υπάρχει ο server. ΠΡΟΣΟΧΗ: Αν δε βλέπουμε το tab του server, πάμε (στο περιβάλλον του eclipse) από το επάνω οριζόντιο μενού Window → Show View → Other → Server. Κάνουμε expand και επιλέγουμε "Servers".

12. Βλέπουμε ότι είναι σε stopped mode - για να τον ενεργοποιήσουμε πάμε και κάνουμε δεξί κλικ πάνω στον server και πατάμε start.

13. Χρειάζεται ένα ακόμα βήμα για να αρχίσει ο LiveReload Server να αλληλοεπιδρά με τον simulator μας. Πάμε στον προσομοιωτή μας και κάνουμε δεξί κλικ και επιλέγουμε Enable LiveReload.

14. Για να δούμε ότι λειτουργεί με επιτυχία πάμε να κάνουμε μια αλλαγή, έστω πάλι στο <h1> και να γράψουμε πχ. "η πρωτη δοκιμή με τον LiveReload server". Αποθηκεύουμε την αλλαγή μας και βλέπουμε ότι η οθόνη του προσομοιωτή ενημερώθηκε αυτόματα.

15. Προσέξτε ότι στην πρώτη μας εφαρμογή υπάρχει μια ένδειξη που δείχνει "Device is Ready". Όταν κάνουμε μια αλλαγή και την αποθηκεύουμε, στο παράθυρο Console καταγράφεται το «!JavaScript LOG: Received Event: deviceready» ενώ επίσης η ένδειξη στον προσομοιωτή προς στιγμή αναγράφει "Connection to Device".

16. Αυτό οφείλεται σε έναν συνδυασμό κώδικα events στην JavaScript: στέλνεται ένα μήνυμα για να εξεταστεί ότι η συσκευή είναι ενεργή και αν αυτή είναι ενεργή επιστρέφει το αντίστοιχο μήνυμα. Πιο συγκεκριμένα στο index.html βλέπουμε ότι υπάρχει:

```
<div id="deviceready" class="blink">
    <p class="event listening">Connecting to Device</p>
    <p class="event received">Device is Ready</p>
```

```
</div>
```

Όταν αποθηκεύουμε το περιεχόμενό μας, συμβαίνει το event listening και στον προσομοιωτή μας καταγράφεται το Connecting to Device. Αφού επιστραφεί η τιμή true ότι η συσκευή μας είναι ενεργή, εκτελείται το event στην JavaScript “received” και έτσι εμείς βλέπουμε το μήνυμα “Device is Ready”.

17. Όλα αυτά τα events στην JavaScript που περιγράψαμε στο προηγούμενο βήμα, έχουν δημιουργηθεί στον φάκελο index.js που βρίσκεται μέσα στον φάκελο js. Στο αρχείο μας index.html συνδέεται με τη χρήση του κώδικα

```
<script type="text/javascript" src="js/index.js"></script>
```

Παρατηρήστε το κώδικα .js. Κάντε στη συνέχεια μια προσθήκη (εντολή console.log('Event listening: '); μέσα στην onDeviceReady), ώστε να έχουμε την παρακάτω μορφή:

```
…
onDeviceReady: function() {
        app.receivedEvent('deviceready');
                console.log('Event listening: ');
  }
  …
```

Τι άλλαξε στη συμπεριφορά της εφαρμογής;

18. Υπάρχει ο φάκελος res → icon που περιέχει την εικόνα έναρξης όταν ξεκίνησε η εικόνα μας. Εκεί αν φτιάχνουμε μια δικιά μας εφαρμογή μπορούμε να τοποθετούμε τις δικές μας εικόνες.

19. Παρατηρούμε ξανά το αρχείο index.html. Βλέπουμε ότι υπάρχει η μορφή που γνωρίζουμε για την κατασκευή σελίδων html. Υπάρχει το <title>, υπάρχει και η σύνδεση με ένα εξωτερικό css αρχείο index.css που βρίσκεται στον φάκελο css:

```
<link rel="stylesheet" type="text/css" href="css/index.css" />
```

20. Βλέπουμε ότι το index.css δίνει τη μορφή που βλέπουμε στη συσκευή μας.

21. Αν θέλουμε να έχουμε έλεγχο στις μορφοποιήσεις css, στο index.css σβήνουμε από το body όλο το περιεχόμενο εκτός από το ακόλουθο:

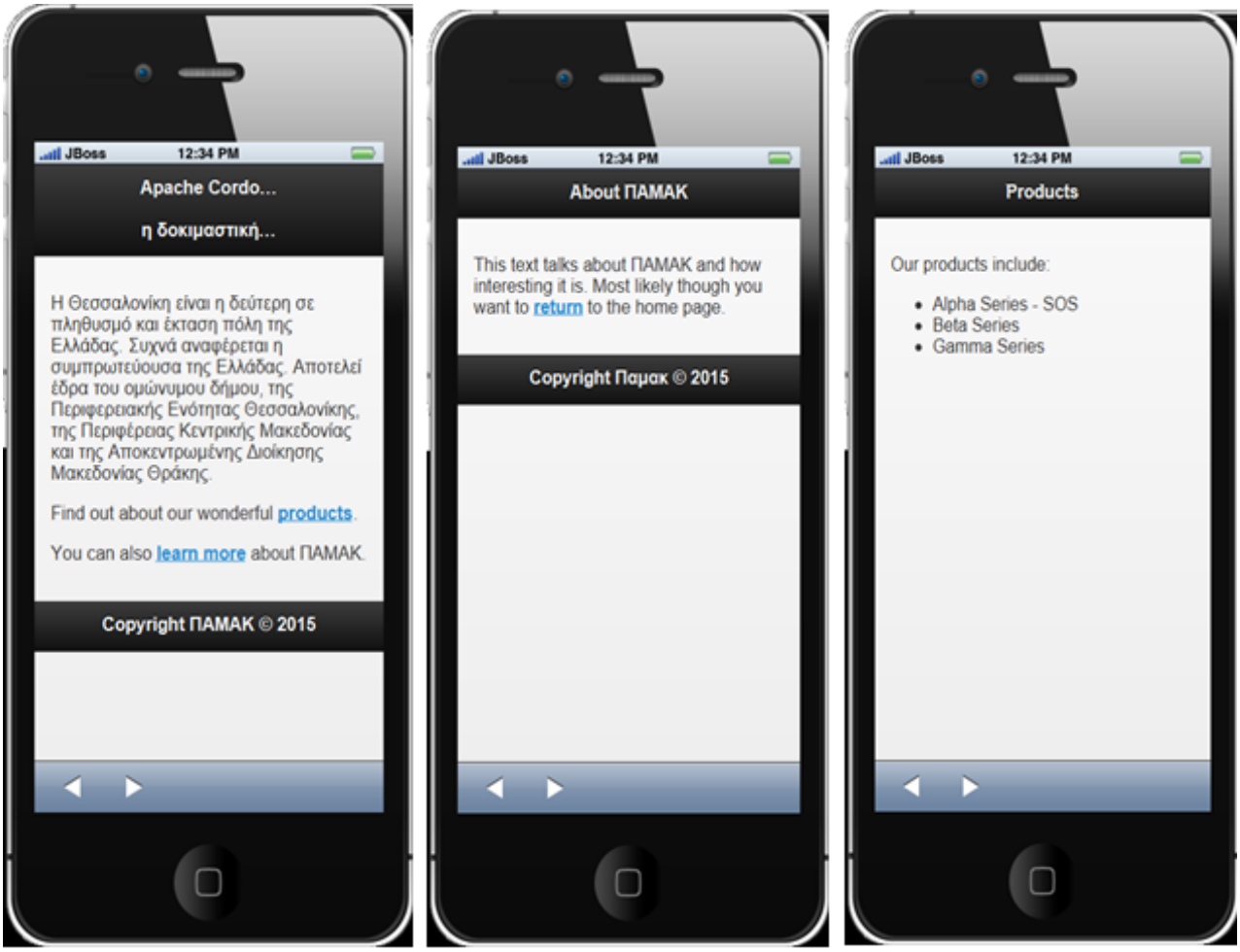
```
body {
-webkit-touch-callout: none;        /* prevent callout to copy image, etc when tap to hold */
-webkit-text-size-adjust: none;     /* prevent webkit from resizing text to fit */
-webkit-user-select: none;          /* prevent copy paste, to allow, change 'none' to 'text' */
  }
```

3.2. Εσωτερικοί και εξωτερικοί σύνδεσμοι

Σε αυτή την εργαστηριακή άσκηση θα δούμε κάποια βασικά στοιχεία στην κατασκευή υβριδικής εφαρμογής, όπως η κεφαλίδα, το υποσέλιδο και οι σύνδεσμοι (προς μια εξωτερική σελίδα ή προς ένα άλλο τμήμα της ίδιας σελίδας, προς μια ‘σελίδα’ μέσα στη σελίδα). Προσέξτε την αναμενόμενη πληθώρα των ‘κοινών στοιχείων’ ανάμεσα στον απαιτούμενο κώδικα κατά την ανάπτυξη υβριδικής εφαρμογής και τον απαιτούμενο κώδικα για την ανάπτυξη ιστοσελίδας κινητού Ιστού.

1. Στο index.html πηγαίνουμε στο και αλλάζουμε τον τίτλο της εφαρμογής μας, πχ. <h1>Apache Cordova

 η δοκιμαστική εφαρμογή μου</h1>



Εικόνα 8.6 *Χρήση εσωτερικών και εξωτερικών συνδέσμων*

2. Κάνουμε τις απαραίτητες προσθήκες/διαγραφές ώστε να φτάσουμε στο ακόλουθο περιεχόμενο:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet"  href="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.css" />
        <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
        <script src="http://code.jquery.com/mobile/1.3.2/jquery.mobile-
          1.3.2.min.js"></script>
     <title>Δοκιμές σε PhoneGap</title>
   </head>
   <body>
     <div class="app" data-role="page" id="homePage" >
       <div data-role="header">
        <h1>Apache Cordova <br/><br/> η δοκιμαστική εφαρμογή μου</h1>
       </div>
      <div data-role="content">
      <p>
       Η Θεσσαλονίκη είναι η δεύτερη σε πληθυσμό και έκταση πόλη της
            Ελλάδας. Συχνά αναφέρεται η συμπρωτεύουσα της Ελλάδας. Αποτελεί
```

```
            έδρα του ομώνυμου δήμου, της Περιφερειακής Ενότητας Θεσσαλονίκης,
            της Περιφέρειας Κεντρικής Μακεδονίας και της Αποκεντρωμένης
            Διοίκησης Μακεδονίας Θράκης.
        </p>
    <p>
        Find out about our wonderful <a href="products.html"
            data-prefetch="true" data-transition="pop"
            data-direction="reverse">products</a>.
        </p>
    <p>
            You can also <a href="#aboutPage">learn more</a> about ΠΑΜΑΚ.
        </p>

    </div>
    <div data-role="footer">
        <h4>Copyright ΠΑΜΑΚ &copy; 2015</h4>
    </div>
</div>

<div data-role="page" id="aboutPage">

    <div data-role="header"><h1>About ΠΑΜΑΚ</h1></div>

    <div data-role="content">
    <p>
    This text talks about ΠΑΜΑΚ and how interesting it is. Most likely
      though you want to <a href="#homePage">return</a> to the home page.
    </p>
    </div>

    <div data-role="footer">
      <h4>Copyright Παμακ &copy; 2015</h4>
    </div>
</div>
</body>

</html>
```

3. Παρατηρήστε την ενσωμάτωση των css κανόνων μορφοποίησης mobile jQuery και των βιβλιοθηκών jQuery και mobile jQuery με τις εντολές:

```
<link rel="stylesheet"
      href="http://code.jquery.com/mobile/1.3.2/jquery.mobile-1.3.2.min.css"/>
  <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.3.2/jquery.mobile- 1.3.2.min.js"></script>
```

4. Παρατηρήστε τη χρήση ετικετών <meta> για τα Ελληνικά και τη ρύθμιση για mobile συσκευές:

```
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1">
```

5. Παρατηρήστε τα τμήματα , και που ορίζουν κεφαλίδα, περιεχόμενο και υποσέλιδο σε mobile web pages.

6. Παρατηρήστε τα τμήματα και που ορίζουν δυο διαφορετικές σελίδες μέσα σε μία σελίδα html.

7. Παρατηρήστε πώς με απλές ετικέτες υπερσύνδεσης, <a href="#aboutPage"> και <a href="#homePage"> γίνεται η ανακατεύθυνση προς και από τέτοιες σελίδες.

8. Η σελίδα products.html έχει το παρακάτω περιεχόμενο:

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset="UTF-8">
    <title>Products</title>
  </head>
  <body>
     <div data-role="page" id="productsPage">

<div data-role="header"><h1>Products</h1></div>

<div data-role="content">
<p>
        Our products include:
</p>
<ul>
        <li>Alpha Series - SOS</li>
        <li>Beta Series</li>
        <li>Gamma Series</li>
</ul>
      </div>
   </div>
  </body>
</html>
```

4. Ανάπτυξη εγγενών κινητών εφαρμογών (Android programming)

4.1. Εισαγωγή

Σε αυτή την ενότητα θα ασχοληθούμε με την ανάπτυξη εγγενών εφαρμογών για το λειτουργικό σύστημα Android. Θα γίνει παρουσίαση της μεθοδολογίας ανάπτυξης εφαρμογών τόσο σε περιβάλλον Eclipse (σε συνδυασμό με το απαραίτητο Android plugin), όσο και στο νεότερο περιβάλλον Android Studio της Google.

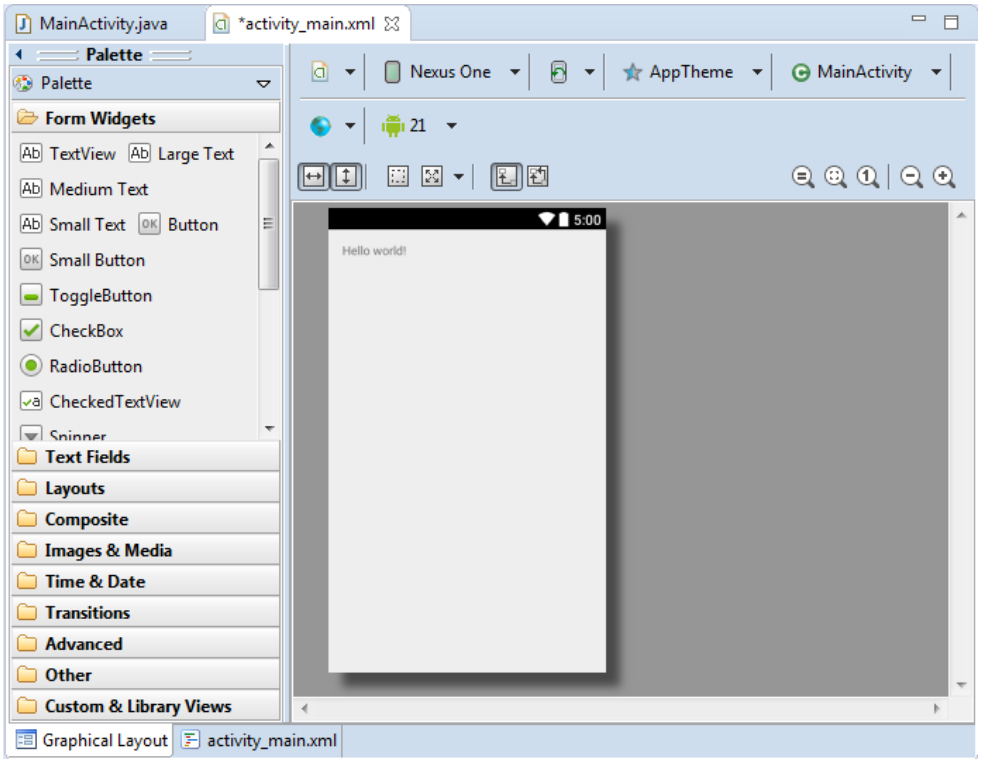
4.2. Χρήση layouts και buttons

Σε αυτή την εργαστηριακή άσκηση θα ασχοληθούμε με τα layouts, τη χρήση buttons και button events, μέσα από την ανάπτυξη μιας σχετικής εφαρμογής. Το προγραμματιστικό περιβάλλον που θα χρησιμοποιήσουμε θα είναι το Eclipse IDE, έκδοση Luna.

1. Δημιουργούμε ένα νέο project. Επιλέγουμε File->New->Other->Android->Android Application Project.
2. Δίνουμε σαν όνομα “Application With Buttons And Layouts” και πατάμε Next.
3. Πατάμε Next στις επόμενες επιλογές και Finish.

Gif 8.1.gif	Κινούμενη εικόνα (interactive)
Δημιουργία ενός Android Application Project στο περιβάλλον Eclipse	

4. Έχοντας ανοιχτό το project «Application With Buttons And Layouts» θα ανατρέξουμε στο αρχείο:
   res->layout->activity_main.xml
   Και θα το ανοίξουμε με διπλό κλικ. Έτσι θα ανοίξει στην οθόνη μας ο layout manager.



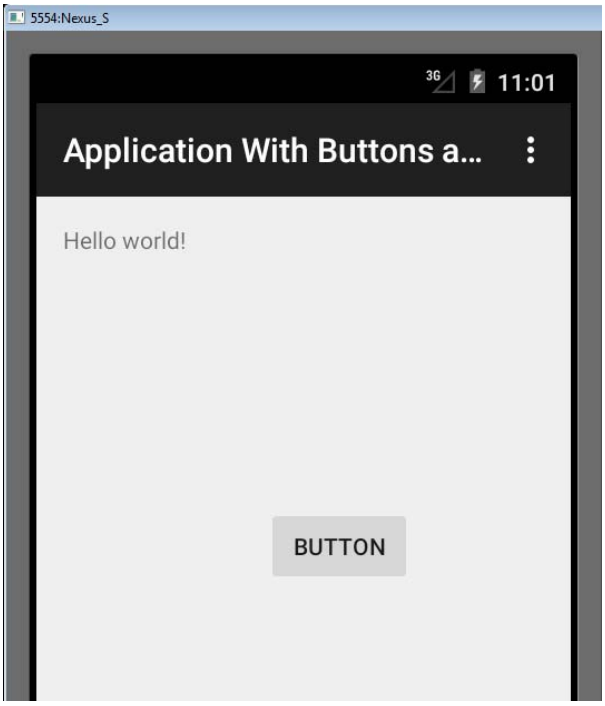
Εικόνα 8.7 *Προβολή του Layout Manager στο περιβάλλον Eclipse IDE*

Sound 8.3.mp3	Ηχητικό απόσπασμα (audio)
Περιγραφή Layout manager	

5. Εδώ φαίνεται πως έχουμε ως root layout το relative layout, στο οποίο κάθε αντικείμενο έχει σχετική θέση σε σχέση με τα υπόλοιπα.
6. Από το μενού στα αριστερά μπορούμε να «σύρουμε» (με drag and drop) νέα στοιχεία όπως για παράδειγμα πλήκτρα (buttons).
7. Θα δούμε πως προσθέτοντας ένα πλήκτρο αυτό αποκτά σχετική θέση με το στοιχείο text που ήδη υπάρχει. Κάνοντας δεξί κλικ στο πλήκτρο που μόλις προσθέσαμε και επιλέγοντας “Edit Text”, μπορούμε να αλλάξουμε το κείμενο που εμφανίζεται στο εσωτερικό του πλήκτρου.

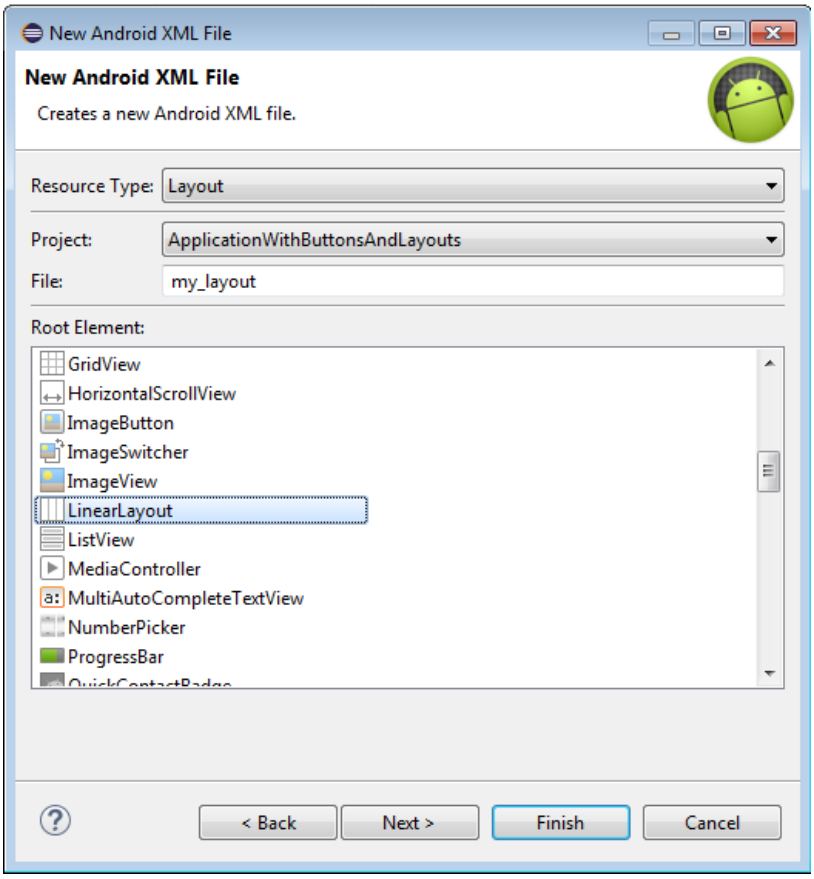
Gif 8.2.gif	Κινούμενη εικόνα (interactive)
Δημιουργία ενός button	

8. Κάνουμε save και εκτελούμε την εφαρμογή μας, για να δούμε πως αυτή εμφανίζεται στον emulator. Θα δούμε πως υπάρχει το πλήκτρο που μόλις προσθέσαμε, αλλά δεν έχει κάποια λειτουργικότητα ακόμη. Στη συνέχεια της άσκησης θα προσθέσουμε λειτουργικότητα μέσω των Button Events.



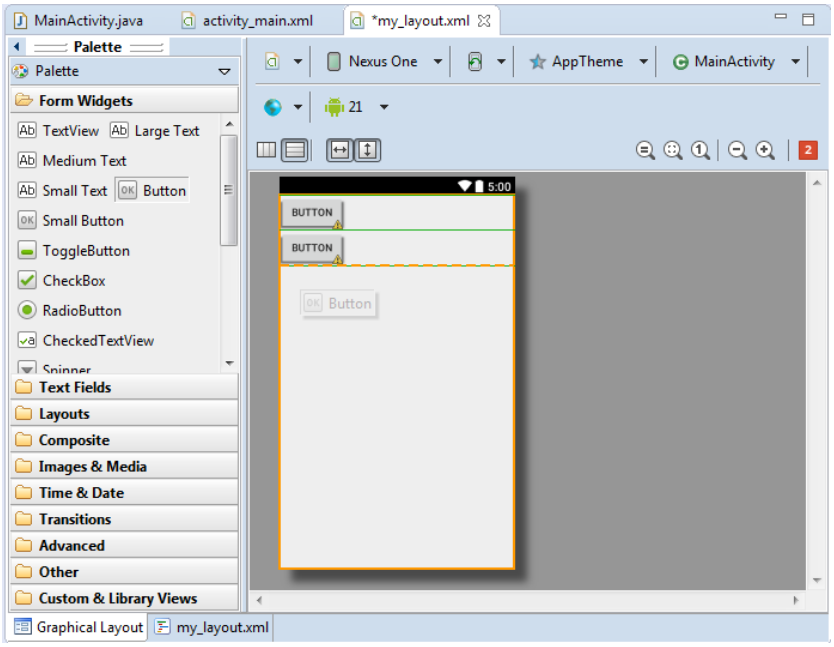
Εικόνα 8.8 *Ενδεικτική εμφάνιση της εφαρμογής μας*

9. Δημιουργούμε ένα νέο layout πηγαίνοντας στον φάκελο layouts και κάνοντας δεξί κλικ->new-> other->Android->Android XML file. Επιλέγουμε linear layout και δίνουμε ως όνομα mylayout.xml.



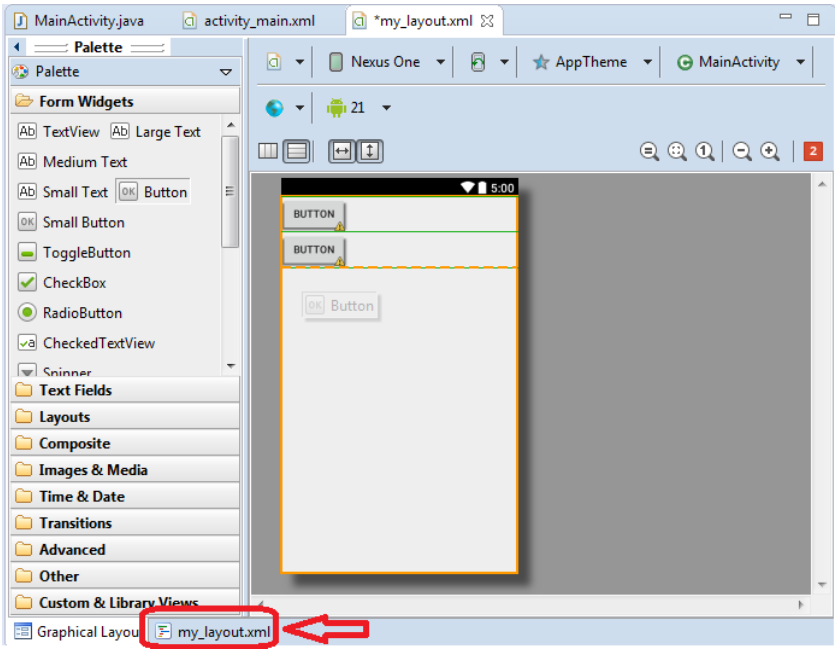
Εικόνα 8.9 *Εισαγωγή ενός νέου Layout, τύπου Linear*

10. Επιλέγοντας στο νέο layout να προσθέσουμε κάποια στοιχεία (πχ. Buttons) θα δούμε πως μας επιτρέπει να προσθέσουμε Button μόνο το ένα κάτω από το άλλο.



Εικόνα 8.10 *Επεξεργασία ενός Linear Layout*

11. Πέρα από το γραφικό περιβάλλον που μας προσφέρεται από την καρτέλα “Graphical Layout” μπορούμε να επεξεργαστούμε το xml αρχείο και μέσω κώδικα κάνοντας κλικ στην καρτέλα my_layout.xml.

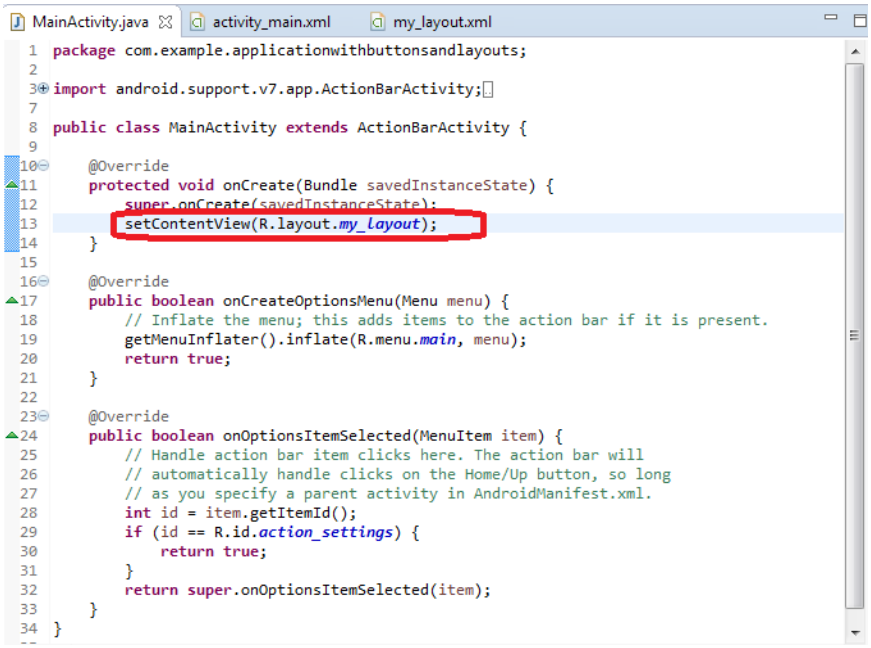


Εικόνα 8.11 *Καρτέλα προβολής XML κώδικα*

Στην καρτέλα αυτή μπορούμε με χρήση κώδικα να κάνουμε αλλαγές στην εμφάνιση της εφαρμογής. Για παράδειγμα για να αλλάξουμε το χρώμα του background προσθέτουμε τη σειρά:

android:background=”#00cc00”

12. Στη συνέχεια τροποποιούμε το αρχείο Main_Activity.java ώστε να δείχνει στο my_layout αρχείο. Τρέχουμε ξανά την εφαρμογή μας για να δούμε τις αλλαγές.



Εικόνα 8.12 *Τροποποίηση του αρχείου Main_Activity.java*

13. Εναλλακτικά μπορούμε να προσθέσουμε κάποια εικόνα. Θα πρέπει να έχουμε τροποποιήσει την εικόνα έτσι ώστε να λειτουργεί σε διάφορες αναλύσεις και σε διάφορα μεγέθη οθονών.

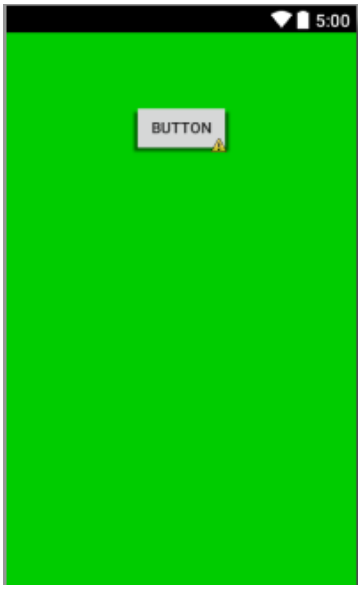
Για να γίνει αυτό τοποθετούμε την τροποποιημένη εικόνα σε καθένα από τους φακέλους drawable που βρίσκονται στον φάκελο res. Για να το κάνουμε αυτό μπορούμε να χρησιμοποιήσουμε το εργαλείο Android Asset Studio.

Link: http://romannurik.github.io/AndroidAssetStudio/icons-launcher.html

14. Στη συνέχεια θα προσθέσουμε λειτουργικότητα στο στοιχείο Button. Στο αρχείο activity_main, μέσα στο αντικείμενο Button, εισάγουμε την παρακάτω σειρά κώδικα:
android:onClick=”onClick”

15. Ενώ στο αρχείο java προσθέτουμε την παρακάτω μέθοδο:

```
public void onClick(View v){
switch(v.getId()){
case R.id.button1:
String message = “Καλημέρα”;
Toast.makeText(this,message,Toast.LENGTH_SHORT).show();
default:
break;
}
}
```



Εικόνα 8.13 *Εμφάνιση εφαρμογής Android*

16. Αφήνοντας το ποντίκι πάνω από την πρώτη σειρά κώδικα, μας δίνεται η δυνατότητα να εισάγουμε τις βιβλιοθήκες view που απαιτούνται. Σε περίπτωση που δε γίνει αυτόματα η εισαγωγή τους τις προσθέτουμε χειροκίνητα.

    import android.view.View;
    import android.view.Menu;
    import android.widget.Toast;

17. Εκτελούμε την εφαρμογή μας και βλέπουμε πως πατώντας το button εμφανίζεται το μήνυμα που ορίσαμε.

Κάντε κλικ στα εικονίδια του σχήματος για επεξήγηση

Dynamic 8.1.zip	Διαδραστική εικόνα (interactive)
Περιγραφή των στοιχείων της εφαρμογής	

Αρχείο xml:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#00cc00"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        tools:context="com.example.appicationwithlayoutsandbuttons.MainActivity" >

        <Button
         android:id="@+id/button1"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_alignParentTop="true"
```

```
        android:layout_centerHorizontal="true"
        android:layout_marginTop="48dp"
        android:onClick="onClick"
        android:text="Button"/>

</RelativeLayout>
```

Αρχείο java:

```
package com.example.appicationwithlayoutsandbuttons;
import android.R.string;
import android.support.v7.app.ActionBarActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Toast;

public class MainActivity extends ActionBarActivity {

@Override
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();
        if (id == R.id.action_settings) {
                return true;
        }
        return super.onOptionsItemSelected(item);
}
public void onClick(View v){
        switch(v.getId()){
        case R.id.button1:
        String message = "Καλημέρα";
        Toast.makeText(this,message,Toast.LENGTH_SHORT).show();
        default:
        break;
        }
        }
}
```

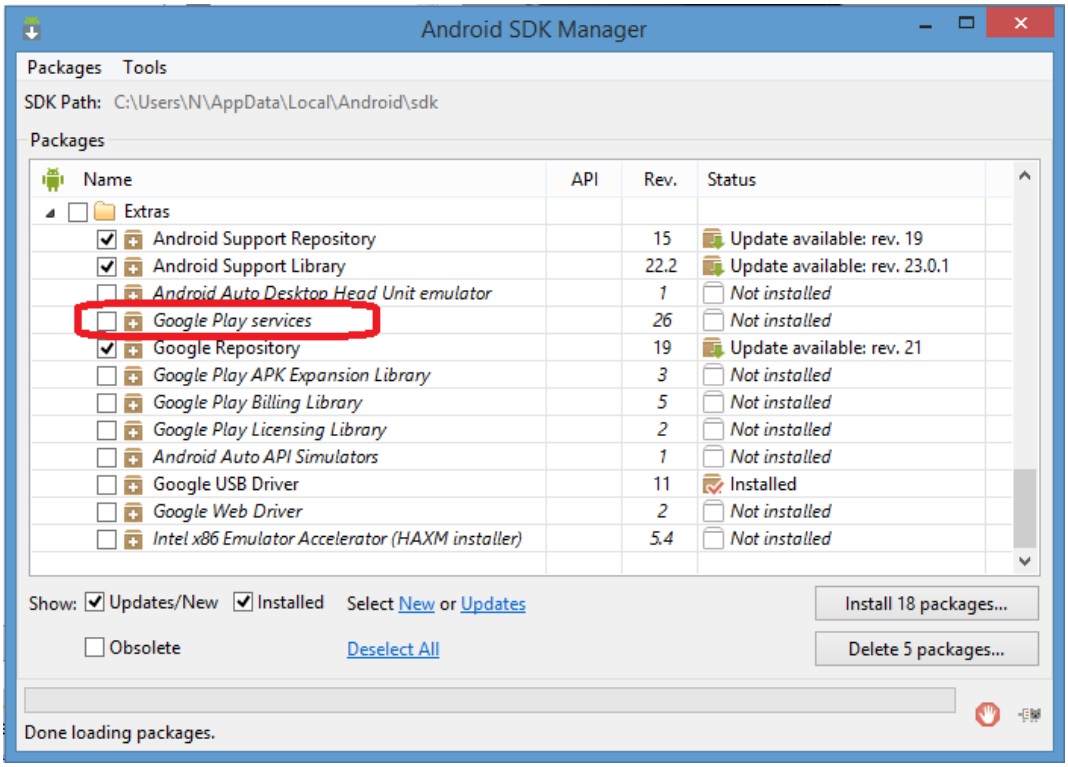
4.3. Άσκηση με αξιοποίηση του Google Maps API

Σε αυτή την εργαστηριακή άσκηση, θα ασχοληθούμε με την ενσωμάτωση ενός χάρτη σε μια εφαρμογή, αξιοποιώντας το Google Maps API. Για την ενσωμάτωση και τη χρήση της υπηρεσίας των χαρτών της Google, σε μια android εφαρμογή, είναι απαραίτητη η λήψη ενός κλειδιού (Google Maps API key). Σε περίπτωση χρήσης του Eclise IDE και του Android plugin, είναι απαραίτητο να ακολουθηθεί η διαδικασία που περιγράφεται στο παρακάτω link:

https://developers.google.com/maps/documentation/android-api/signup

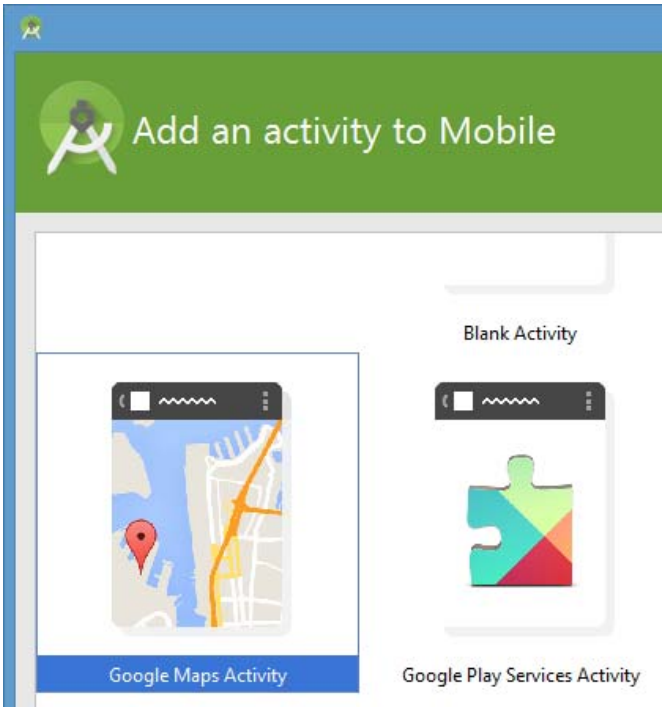
Παρακάτω θα δούμε μια πιο απλοποιημένη μέθοδο, με τη χρήση του προγραμματιστικού περιβάλλοντος Android Studio. Το Android Studio είναι μια πλατφόρμα που έχει αναπτύξει η Google, και σύντομα θα αποτελέσει τη μοναδική πλατφόρμα που θα λαμβάνει τις τελευταίες ενημερώσεις (πχ. ενημερώσεις ασφαλείας).

1. Αφού εκκινήσουμε το Android Studio, εγκαθιστούμε το πακέτο Google Play services μέσω του SDK manager.



Εικόνα 8.14 *Επιλογή των Google Play Services, μέσω του Android SDK manager*

2. Αφού έχουμε βεβαιωθεί πως έχουμε εγκατεστημένο το πακέτο Google Play services, δημιουργούμε ένα νέο project. Επιλέγουμε File -> New -> New Project.
3. Δίνουμε όνομα στην εφαρμογή μας και πατάμε το πλήκτρο Next.
4. Επιλέγουμε σαν τύπο συσκευών Phone and Tablet και πατάμε το πλήκτρο Next.
5. Επιλέγουμε ως activity ένα Google Maps Activity στο παράθυρο 'Add an activity to Mobile'. Έπειτα πατάμε το πλήκτρο Next.

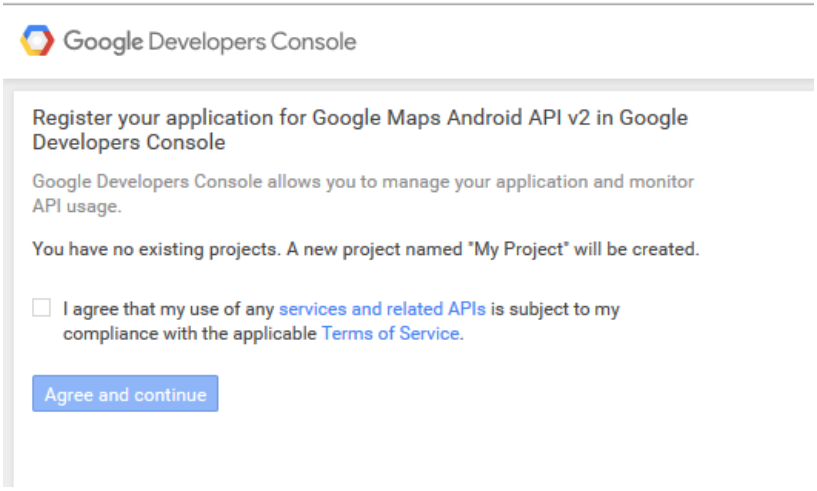


Εικόνα 8.15 *Επιλογή ενός Google Maps Activity κατά την αρχικοποίηση της εφαρμογής*

6. Αντιγράφουμε τον σύνδεσμο που έχει δημιουργήσει αυτόματα το Android Studio και που βρίσκεται στο αρχείο google_maps_api.xml.

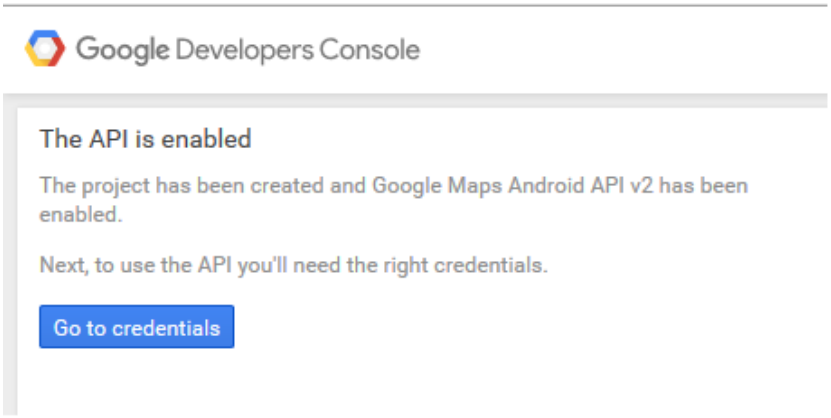
Εικόνα 8.16 *Ο σύνδεσμος που πρέπει να δοθεί στην Google για τη λήψη του κατάλληλου κλειδιού*

7. Επικολλούμε τον σύνδεσμο σε έναν φυλλομετρητή (web browser).



Εικόνα 8.17 *Διεπιφάνεια του Google Developers Console*

8. Ακολουθούμε τις οδηγίες και δημιουργούμε ένα Android API key για το project που εμφανίζεται.



Εικόνα 8.18 *Λήψη διαπιστευτηρίων (κλειδιού)*

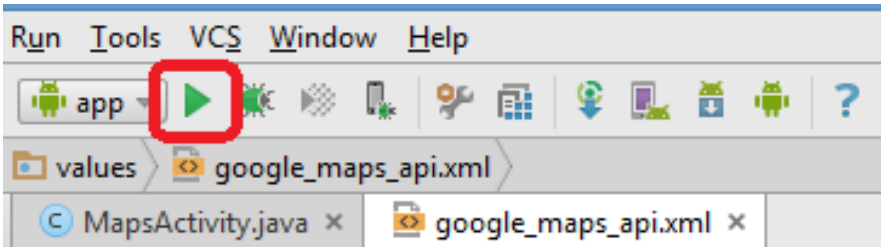
9. Αντιγράφουμε το κλειδί που δημιουργήθηκε στο αρχείο google_maps_api.xml στο στοιχείο <string> (αντικαθιστώντας το κείμενο «YOUR_KEY_HERE»).
10. Ανοίγουμε το αρχείο MapsActivity.java και εντός της μεθόδου MapsActivity προσθέτουμε την παρακάτω μέθοδο (αντικαθιστώντας τις setUpMapIfNeeded και setUpMap) :

```
@Override
public void onMapReady(GoogleMap map) {
    LatLng thessaloniki = new LatLng(40.625161, 22.960073);
    map.addMarker(new MarkerOptions().position(thessaloniki).title("You are in
Thessaloniki"));
    map.moveCamera(CameraUpdateFactory.newLatLng(thessaloniki));
}
```

11. Δημιουργούμε μια εικονική μηχανή που να περιέχει τα Google Play services

Video 8.1.mp4	Βίντεο (video)
Δημιουργία εικονικής συσκευής στο περιβάλλον Android Studio	

12. Πατούμε το πλήκτρο εκτέλεσης της εφαρμογής και επιλέγουμε να την εκτελέσουμε στην εικονική μηχανή που μόλις δημιουργήσαμε



Εικόνα 8.19 *Πλήκτρο εκτέλεσης της εφαρμογής*

13. Στην εικονική μηχανή που ανοίγει βλέπουμε πως η εφαρμογή μας φορτώνει έναν χάρτη και εστιάζει και τοποθετεί έναν marker στη Θεσσαλονίκη, με το μήνυμα "You are in Thessaloniki". Κάνοντας zoom in θα δούμε λεπτομέρειες για το σημείο που φαίνεται στον χάρτη.

Εικόνα 8.20 *Αποτέλεσμα εκτέλεσης της εφαρμογής*

Εικόνα 8.21 *Αποτέλεσμα «μεγέθυνσης» (zoom) στον χάρτη*

Τελική μορφή αρχείου MapsActivity:

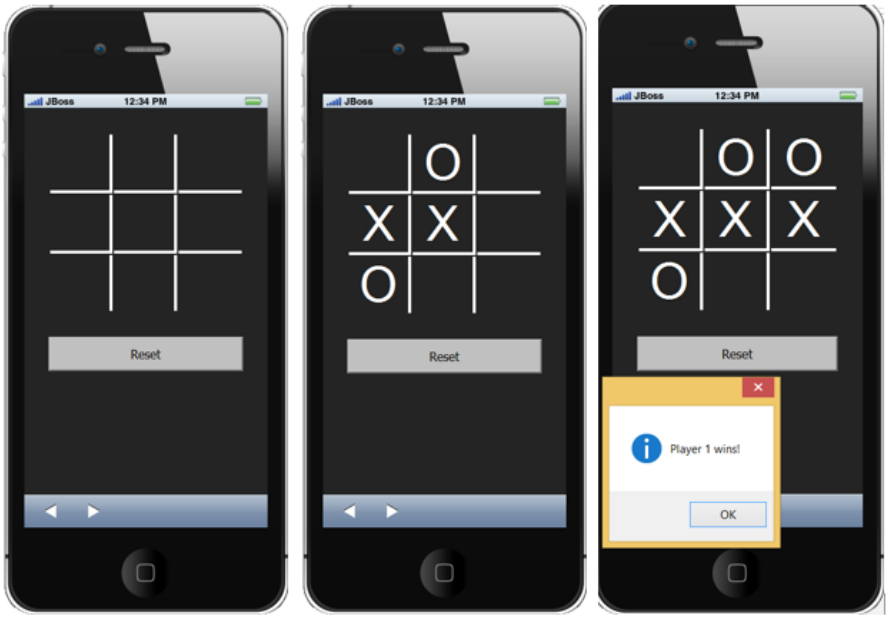
```
package com.example.uom.mymapapplication;

import android.os.Bundle;
import android.support.v4.app.FragmentActivity;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_maps);
      SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
            .findFragmentById(R.id.map);
      mapFragment.getMapAsync(this);
   }

   @Override
   public void onMapReady(GoogleMap map) {
   LatLng thessaloniki = new LatLng(40.625161, 22.960073);
   map.addMarker(new MarkerOptions().position(thessaloniki).title("You are in Thessaloniki"));
   map.moveCamera(CameraUpdateFactory.newLatLng(thessaloniki));
}

}
```

5. Ασκήσεις αυτοαξιολόγησης

5.1. Ανάπτυξη εφαρμογής για το παιχνίδι τρίλιζα

Εκφώνηση:

Να υλοποιήσετε στο περιβάλλον Phone Gap το γνωστό παιχνίδι τρίλιζα. Για να θεωρείται πλήρης η άσκηση, θα πρέπει όταν ένας παίχτης κερδίζει τον γύρο να εμφανίζεται το κατάλληλο μήνυμα. Ενδεικτικά screenshots για το επιθυμητό τρόπο εμφάνισης της εφαρμογής είναι τα παρακάτω:

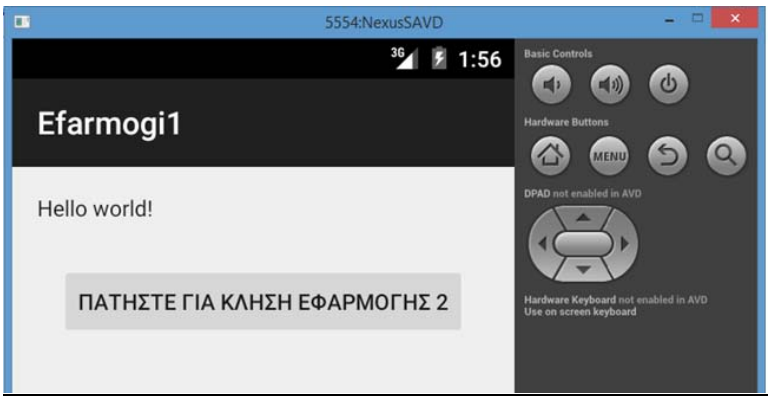


Εικόνα 8.22 *Ενδεικτικές οθόνες της εφαρμογής*

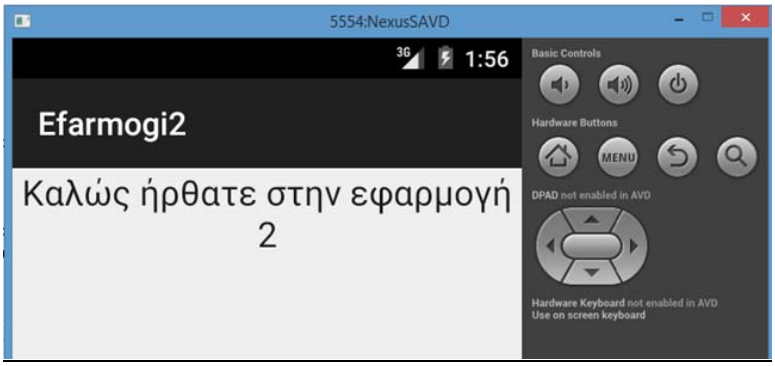
5.2. Άσκηση με αξιοποίηση Intents

Εκφώνηση:

Να αναπτυχθεί εφαρμογή που με χρήση Intents, θα είναι δυνατή η κλήση μιας εξωτερικής εφαρμογής. Συγκεκριμένα θα πρέπει να αναπτυχθούν δύο εφαρμογές. Η πρώτη θα περιλαμβάνει ένα στοιχείο τύπου button. Με την ενεργοποίηση αυτού του στοιχείου από τον χρήση θα καλείται ένα layout μιας δεύτερης εφαρμογής που θα περιλαμβάνει ένα μήνυμα καλωσορίσματος. Ενδεικτικές εικόνες των δύο εφαρμογών είναι οι παρακάτω:



Εικόνα 8.23 *Ενδεικτική οθόνη πρώτης εφαρμογής*



Εικόνα 8.24 *Ενδεικτική οθόνη δεύτερης εφαρμογής*

Σημείωση: Στον Ιστότοπο του συγγράμματος (http://ec-tech.uom.gr/WT-ECOM), θα βρείτε τον πηγαίο κώδικα για όλες τις υποδειγματικά λυμένες ασκήσεις, καθώς και για τις ασκήσεις αυτοαξιολόγησης.

6. Συμπεράσματα

Στο κεφάλαιο αυτό αναλύθηκαν και περιγράφηκαν εκτενώς οι κατηγορίες ανάπτυξης περιεχομένου για κινητές συσκευές, ενώ παράλληλα παρουσιάστηκε ένα σύνολο εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών). Σκοπός τον εργαστηριακών ασκήσεων ήταν η κατανόηση τεχνικών προγραμματισμού ανάπτυξης περιεχομένου για κινητές συσκευές, κυρίως από την πλευρά του πελάτη (client-side scripting). Οι ασκήσεις που αφορούν την ανάπτυξη υβριδικών εφαρμογών έχουν υποδειγματικά επιλυθεί στο περιβάλλον PhoneGap, ενώ οι ασκήσεις που αφορούν την ανάπτυξη εγγενών εφαρμογών έχουν επιλυθεί στα περιβάλλοντα Eclipse (σε συνδυασμό με το Android Plugin) αλλά και Android Studio, κάνοντας χρήση της γλώσσας προγραμματισμού Java. Μέσω των εργαστηριακών ασκήσεων παρουσιάζονται τεχνικές ανάπτυξης κινητών εφαρμογών ενσωματώνοντας στοιχεία διαδεδομένα σε εφαρμογές που υποστηρίζουν συναλλαγές Ηλεκτρονικού Εμπορίου (εισαγωγή στοιχείων, φόρμες, διάδραση με χάρτη κ.α.).

Βιβλιογραφία/Αναφορές

Annuzzi, J. Jr., Darcey, L., Conder, S. (2014). *Introduction to Android Application Development, 4th edition*, Addison-Wesley.

Cameron, D. (2013). *A Software Engineer Learns HTML5, JavaScript and jQuery*, CreateSpace.

Firtman, M. (2013). *Programming the Mobile Web, 2nd edition*, O' Reilly.

Firtman, M. (2012). *jQuery Mobile: Up and Running*, O' Reilly.

Frederick, G. R., & Lal, R. (2009). *Beginning Smartphone Web Development*, Apress.

Gasston, P. (2013). *The Modern Web: Multi-Device Web Development with HTML5, CSS3, and JavaScript*, No Starch Press.

Harwani, B. (2013). *PhoneGap Build: Developing Cross Platform Mobile Applications in the Cloud*, Auerbach Publications.

Jackson, W. (2013). *Learn Android App Development*, Apress,

Matthews, A., & Gliser, S. (2015). *Creating Mobile Apps with jQuery Mobile, 2nd edition*, Packt Publishing.

Γαβαλάς, Δ., Κασαπάκης, Β., & Χατζηδημήτρης, Θ. (2015). *Κινητές Τεχνολογίες*, Εκδόσεις Νέων Τεχνολογιών.

Χρήσιμοι δικτυακοί τόποι:

World Wide Web Consortium, W3C:

W3C Web Design (http://www.w3schools.com/website/web_design.asp)

W3C HTML5 Tutorial (http://www.w3schools.com/html/html5_intro.asp)

W3C CSS3 Tutorial (http://www.w3schools.com/css/css3_intro.asp)

W3C jQuery Mobile Tutorial (http://www.w3schools.com/jquerymobile/default.asp)

W3C AJAX Tutorial (http://www.w3schools.com/ajax/default.asp)

W3C Google Maps API Tutorial (http://www.w3schools.com/googleAPI/default.asp)

Official jQuery Mobile Resources (http://jquerymobile.com/resources)

Android tutorials:

https://developer.android.com/training/basics/firstapp/index.html

https://developer.android.com/training/basics/fragments/index.html

https://developer.android.com/training/building-graphics.html

https://developer.android.com/training/building-location.html

https://developer.android.com/training/best-ux.html

Quiz8.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Οι υβριδικές κινητές εφαρμογές, χρησιμοποιούν τις ίδιες τεχνολογίες ανάπτυξης περιεχομένου που χρησιμοποιούν και οι κινητές ιστοσελίδες

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Α) Σωστό

Κριτήριο αξιολόγησης 2

[*] Οι εγγενείς κινητές εφαρμογές:

Α) Γράφονται συνήθως σε HTML5 κώδικα σε συνδυασμό με τη χρήση CSS3 και Javascript

Β) Αναπτύσονται αποκλειστικά σε Java και έτσι εκτελούνται σε όλα τα γνωστά λειτουργικά συστήματα κινητών συσκευών.

Γ) Αναπτύσονται σε κατάλληλη γλώσσα προγραμματισμού ανάλογα με το λειτουργικό σύστημα στο οποίο στοχεύουν.

Απάντηση/Λύση

Γ) Αναπτύσονται σε κατάλληλη γλώσσα προγραμματισμού ανάλογα με το λειτουργικό σύστημα στο οποίο στοχεύουν.

Κριτήριο αξιολόγησης 3

[*] Ο τρόπος διανομής και η συνδεσιμότητα είναι παρόμοια για τις υβριδικές και τις εγγενείς κινητές συσκευές:

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Α) Σωστό

Κριτήριο αξιολόγησης 4

[*] Η εντολή «Toast.makeText(this,message,Toast.LENGTH_SHORT).show()», σε μια εφαρμογή android επιτρέπει:

Α) Την εμφάνιση ενός toast μηνύματος στην οθόνη

Β) Την είσοδο ενός sting από τον χρήστη και την αποθήκευση του στη μεταβλητή message

Γ) Τη μεταφορά ενός toast μηνύματος μέσω intent σε μια άλλη εφαρμογή

Απάντηση/Λύση
Α) Την εμφάνιση ενός toast μηνύματος στην οθόνη

Κριτήριο αξιολόγησης 5
[*] Για την ενσωμάτωση και τη χρήση της υπηρεσίας των χαρτών της Google, σε μια android εφαρμογή, είναι απαραίτητη η λήψη ενός κλειδιού (Google Maps API key).

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Α) Σωστό

Κριτήριο αξιολόγησης 6
[**] Το παρακάτω τμήμα κώδικα:
** <input type="reset" value="Λειτουργία" /> **

Α) Χρησιμοποιείται για τη στοίχιση κειμένου στα αριστερά της οθόνης

Β) Χρησιμοποιείται για την υποστήριξη καθαρίσματος μιας φόρμας από τις επιλογές που έχει δώσει ο χρήστης

Γ) Χρησιμοποιείται για την υποβολή των στοιχείων που έχει δώσει ο χρήστης και στη συνέχεια την επαναφορά σε προηγούμενη οθόνη

Απάντηση/Λύση
Β) Χρησιμοποιείται για την υποστήριξη καθαρίσματος μιας φόρμας από τις επιλογές που έχει δώσει ο χρήστης

Κριτήριο αξιολόγησης 7
[**] Η εντολή «map.addMarker(new MarkerOptions().position(X).title("Message"))» χρησιμοποιείται για
την εμφάνιση της τρέχουσας τοποθεσίας του χρήστη στον χάρτη, εμφανίζοντάς του παράλληλα ένα μήνυμα.

α) Σωστό

β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 8
[*] Η JavaScript είναι μια δυναμική γλώσσα προγραμματισμού η οποία:

Α) χρησιμοποιείται ευρέως από τους Web browsers, όσον αφορά μάλιστα την πλευρά του πελάτη θεωρείται η κύρια γλώσσα του Διαδικτύου

Β) είναι ικανή να ελέγξει όλα τα στοιχεία του Document Object Model μια ιστοσελίδας και να τα πυροδοτήσει προκαλώντας διάφορα συμβάντα

Γ) είναι ικανή να υποστηρίξει την έννοια της φορητότητας στις κινητές εφαρμογές για όλες τις πλατφόρμες και συσκευές

Δ) συνδυάζει όλα τα προαναφερόμενα χαρακτηριστικά

Ε) συνδυάζει τα προαναφερόμενα χαρακτηριστικά των επιλογών Α) και Γ)

Απάντηση/Λύση
Δ) συνδυάζει όλα τα προαναφερόμενα χαρακτηριστικά

Κριτήριο αξιολόγησης 9
[*] Η mobile jQuery είναι μια βιβλιοθήκη, ένα πλαίσιο (framework), η οποία:

Α) δε βασίζεται στη βιβλιοθήκη jQuery

Β) χρησιμοποιεί τις HTML5, CSS3 και JavaScript

Γ) διαχειρίζεται συμβάντα των κινητών συσκευών για τη δημιουργία κινητών εφαρμογών

Δ) συνδυάζει όλα τα προαναφερόμενα χαρακτηριστικά

Ε) συνδυάζει τα προαναφερόμενα χαρακτηριστικά των επιλογών Β) και Γ)

Απάντηση/Λύση
Ε) συνδυάζει τα προαναφερόμενα χαρακτηριστικά των επιλογών Β) και Γ)

Κριτήριο αξιολόγησης 10
[**] Ποια από τις ακόλουθες επιλογές δεν αποτελεί σημείο προβληματισμού στην ανάπτυξη κινητών ιστοσελίδων:

Α) Είναι διαφορετικά τα πλαίσια κινητών χρηστών σε σχέση με τους χρήστες σταθερών υπολογιστών

Β) Υπάρχουν στην κυκλοφορία πολλά κινητά, με διαφορετικούς περιηγητές, με διαφορετικές εκδοχές

Γ) Είναι διαφορετικές οι γλώσσες προγραμματισμού σε σχέση με αυτές που απαιτούνται για την ανάπτυξη ιστοσελίδων για σταθερούς υπολογιστές (δηλ. ιστοσελίδων του παραδοσιακού ενσύρματου Ιστού)

Δ) Τα ασύρματα/κινητά δίκτυα είναι πιο αργά, ενώ και το υλικό είναι πιο αργό και με λιγότερη διαθέσιμη μνήμη σε σχέση με τους αντίστοιχους διαθέσιμους πόρους των σταθερών υπολογιστών για πρόσβαση σε ιστοσελίδες

Απάντηση/Λύση
Γ) Είναι διαφορετικές οι γλώσσες προγραμματισμού σε σχέση με αυτές που απαιτούνται για την ανάπτυξη ιστοσελίδων για σταθερούς υπολογιστές (δηλ. ιστοσελίδων του παραδοσιακού ενσύρματου Ιστού)

Κεφάλαιο 9: Τεχνολογία Υπηρεσιών Ιστού και Ηλεκτρονικό Εμπόριο

Σύνοψη
Στο κεφάλαιο αυτό γίνεται αναφορά στην τεχνολογία των Υπηρεσιών Ιστού (ΥΙ), τις υπάρχουσες κατηγορίες ΥΙ, στις τεχνικές επιλογής και σύνθεσης αλλά και στους ειδικότερους τομείς της επιστήμης της πληροφορικής, στους οποίους έχουν εφαρμογή οι ΥΙ, όπως είναι για παράδειγμα το «Διαδίκτυο των Αντικειμένων». Στόχος του κεφαλαίου είναι να αναδειχθούν οι δυνατότητες των ΥΙ που επιτρέπουν την απρόσκοπτη επικοινωνία ανάμεσα σε συστήματα ηλεκτρονικού εμπορίου, αλλά και την «ολοκλήρωση» τους. Καθώς στη σύγχρονη εποχή το επιχειρείν βασίζεται σε μεγάλο βαθμό στις συναλλαγές μέσω Διαδικτύου (η-επιχειρείν), είναι απαραίτητη η αντιμετώπιση των προβλημάτων που εγείρονται από την ανάγκη διαλειτουργικότητας ανάμεσα σε συστήματα, τα οποία έχουν αναπτυχθεί σε διαφορετικές πλατφόρμες. Οι ΥΙ αποτελούν την τεχνολογία που επιτρέπει μέσω χαλαρής σύζευξης την επικοινωνία αυτή μεταξύ ετερογενών συστημάτων, και για τον λόγο αυτό έχουν υιοθετηθεί σε μεγάλο βαθμό από πληθώρα επιχειρήσεων και οργανισμών.

Προαπαιτούμενη γνώση
Τα κεφάλαια 1 και 2 του παρόντος συγγράμματος

1. Αναγκαιότητα ολοκλήρωσης & διαλειτουργικότητα σε συστήματα ηλεκτρονικού εμπορίου

Στη σημερινή εποχή, περισσότερο από ποτέ, η χρήση των τεχνολογιών του Διαδικτύου παίζει καθοριστικό ρόλο στο επιχειρείν. Αυτό συμβαίνει διότι οι τεχνολογίες αυτές συμβάλλουν στην αντιμετώπιση προβλημάτων που πιθανόν να προκύπτουν από τη γεωγραφική απόσταση με τους πελάτες και τους προμηθευτές μιας επιχείρησης, τη διαρκώς αυξανόμενη ανάγκη προβολής προϊόντων και υπηρεσιών με τον πλέον βέλτιστο και ακριβή τρόπο αλλά και από την ανάγκη για απροβλημάτιστη ανταλλαγή πληροφοριών με τους συνεργάτες της επιχείρησης. Η μετάβαση λοιπόν σε ψηφιακές πλατφόρμες διευρύνει τους ορίζοντες μιας επιχείρησης, καταργεί περιορισμούς και δίνει πρόσβαση σε δυνατότητες που θα ήταν αδύνατον να εκμεταλλευτεί μια εταιρία χωρίς τη χρήση του Διαδικτύου.

Πέρα όμως από τις πολλές δυνατότητες που προσφέρονται με τη χρήση του Διαδικτύου, η μεταφορά του επιχειρείν στο η-επιχειρείν εγείρει και ένα πλήθος από προκλήσεις. Μια ειδικότερη πρόκληση, αφορά την επικοινωνία με πιθανώς ετερογενή συστήματα προμηθευτών και συνεργατών, εφόσον εκεί συχνά παρουσιάζονται προβλήματα ασυμβατότητας. Αυτό συμβαίνει διότι είναι αναγκαία η ανταλλαγή πληροφοριών ανάμεσα σε πλατφόρμες οι οποίες έχουν αναπτυχθεί με διαφορετικές προδιαγραφές και με χρήση διαφορετικών συστημάτων ανάπτυξης. Όπως είναι φυσικό οι πλατφόρμες αυτές δεν μπορούν εκ κατασκευής να αναγνωρίσουν ως είσοδο αρχεία παραγόμενα από τις πλατφόρμες συνεργατών της επιχείρησης. Ειδικότερα σε περιβάλλοντα ηλεκτρονικού εμπορίου (ΗΕ), το πλήθος τέτοιων συστημάτων είναι πολύ ευρύ και ποικίλει από συστήματα ηλεκτρονικών πληρωμών και τραπεζικών συστημάτων, μεταφορικών εταιριών μέχρι και υπηρεσίες ανατροφοδότησης εφοδιαστικής αλυσίδας. Γίνεται λοιπόν εύκολα αντιληπτή η ανάγκη για απρόσκοπτη επικοινωνία ανάμεσα σε αυτά τα συστήματα ΗΕ, ή πιο συγκεκριμένα γίνεται φανερή η ανάγκη για ολοκλήρωση & διαλειτουργικότητα σε συστήματα ΗΕ.

Μια σημαντική προσέγγιση προς αυτή την κατεύθυνση είναι το Enterprise Service Bus (ESB), το οποίο ουσιαστικά είναι μια αρχιτεκτονική λογισμικού η οποία μπορεί να χρησιμοποιηθεί ως ενδιάμεσος (middleware), που θα ενισχύει και θα υποστηρίζει την αλληλεπίδραση σύνθετων αρχιτεκτονικών, απλοποιώντας τις απαιτήσεις των διαφορετικών interfaces ετερογενών συστημάτων. Κάποια από τα πλεονεκτήματα που παρουσιάζει η υιοθέτηση του ESB είναι:

- η αυξημένη ευελιξία και η δυνατότητα πραγματοποίησης αλλαγών στα πληροφοριακά συστήματα που επικοινωνούν μέσω του ESB, καθώς αυτά μπορούν να προσαρμόζονται ευκολότερα στις απαιτήσεις του επιχειρησιακού περιβάλλοντος, που αλλάζει με ραγδαίους ρυθμούς.

- η δυνατότητα αυτόματης εισαγωγής ή εξαγωγής συστημάτων ηλεκτρονικού εμπορίου, αλλά και επεκτάσεων αυτών στη συνολική αρχιτεκτονική με αυτοματοποιημένο τρόπο και χωρίς την ανάγκη παραμετροποιήσεων στον τρόπο επικοινωνίας.

Σημαντικά συστήματα ESB που χρησιμοποιούνται και σήμερα είναι τα SAP Process Integration, Oracle Enterprise Service Bus (BEA Logic), Mule ESB (Enterprise Edition) αλλά και το open-source λογισμικό Open ESB.

Πώς όμως επιτρέπουν τα συστήματα που βασίζονται στο ESB τη διευκόλυνση των συναλλαγών συστημάτων τα οποία έχουν αναπτυχθεί με χρήση διαφορετικών πλατφόρμων και κάνοντας χρήση διαφορετικών προτύπων; Τα συστήματα ESB επιτρέπουν κάτι τέτοιο καθώς βασίζονται στη χρήση της Αρχιτεκτονικής βασισμένη-σε-Υπηρεσίες (SOA). Η αρχιτεκτονική αυτή επιτρέπει την αντιμετώπιση της τεχνολογικής πρόκλησης της διασύνδεσης ετερογενών συστημάτων και αποτελεί μια αρχιτεκτονική με διαρκώς αυξανόμενη υιοθέτηση από τη βιομηχανία και τις επιχειρήσεις.

2. Αρχιτεκτονική βασισμένη-σε-Υπηρεσίες (SOA): εξέλιξη στοιχείων λογισμικού για κατανεμημένα συστήματα

Η αρχιτεκτονική βασισμένη-σε-Υπηρεσίες (SOA) είναι μια προσέγγιση σχεδιασμού αρχιτεκτονικής λογισμικού, η οποία έχει ως επίκεντρο τις υπηρεσίες (Papazoglou, et al., 2008). Στηρίζεται δηλαδή στην εξής προσέγγιση: τμήματα λογισμικού μπορούν να προσφέρουν τη λειτουργικότητα τους ως υπηρεσία σε άλλα τμήματα λογισμικού ή σε ολοκληρωμένες εφαρμογές. Πιο συγκεκριμένα, ένα σύστημα σχεδιασμένο με SOA δίνει τη δυνατότητα παροχής υπηρεσιών σε χρήστες ή σε άλλες υπηρεσίες στο Διαδίκτυο μέσα από δημοσιευμένες και εύκολα προσβάσιμες διεπαφές. Οι επιχειρήσεις μπορούν να ωφεληθούν από τη σχεδίαση συστημάτων με SOA καθώς αυτά μπορούν να αναπαραστήσουν μεμονωμένες επιχειρηματικές δραστηριότητες σαν υπηρεσίες δίνοντας τη δυνατότητα επαναχρησιμοποίησης αλλά και παραμετροποίησης, κάτι που καθιστά τις επιχειρήσεις ευέλικτες μπροστά σε αλλαγές του περιβάλλοντος στο οποίο δρουν. Η συγκεκριμένη αρχιτεκτονική έκανε την εμφάνιση της στα μέσα της δεκαετίας του 90, λόγω της ανάγκης για διαλειτουργικότητα ανάμεσα σε ετερογενή συστήματα, μια ανάγκη που διογκώθηκε ακόμη περισσότερο λόγω της αναδιάρθρωσης της επιχειρηματικής λογικής πολλών εταιριών και επιχειρήσεων, στην προσπάθεια τους να επιτύχουν την απαιτούμενη ευελιξία. Η αναδιάρθρωση αυτή σήμαινε την ανάθεση ορισμένων λειτουργιών της επιχείρησης σε τρίτους, για την ελαχιστοποίηση του κόστους. Κατ' επέκταση ήταν απαραίτητη η διαλειτουργικότητα και η συνεχής επικοινωνία με το πληροφοριακό σύστημα των εξωτερικών συνεργατών. Η διαλειτουργικότητα αυτή κατέστη δυνατή με τη χρήση της αρχιτεκτονικής SOA.

2.1. Χαλαρή σύζευξη

Οι Υπηρεσίες Ιστού χαρακτηρίζονται από τη λεγόμενη χαλαρή σύζευξη, δηλαδή τη δυνατότητα επικοινωνίας ανάμεσα σε εφαρμογές ή τμήματα λογισμικού, χωρίς να υπάρχει πρότερη γνώση των προδιαγραφών που χαρακτηρίζουν την κάθε εμπλεκόμενη εφαρμογή (Weerawarana, et al., 2008).

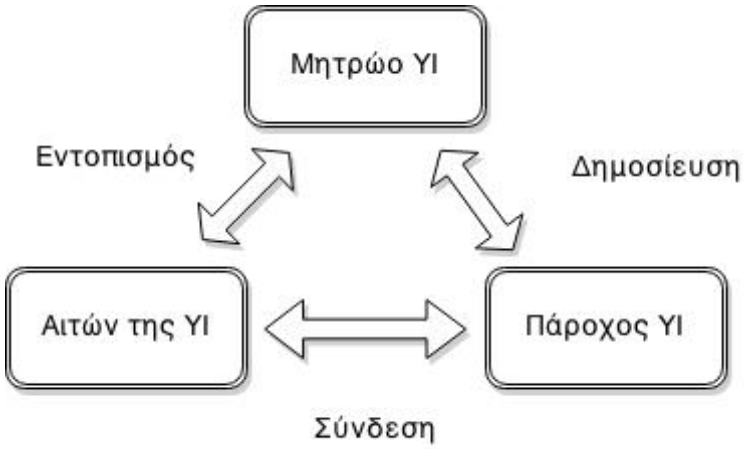
Σε συστήματα που έχουν αναπτυχθεί με βάση την αρχιτεκτονική SOA, η κυριότερη μέθοδος επικοινωνίας μεταξύ των εφαρμογών είναι με μηνύματα σε μορφή XML. Τα μηνύματα αυτά περιέχουν πληροφορίες σχετικά με κάποια λειτουργία προς εκτέλεση, όπως για παράδειγμα, ποιες εφαρμογές θα συνεργαστούν για αυτόν τον σκοπό και ποια θα είναι τα δεδομένα που θα διαμοιραστούν.

Η χαλαρή σύζευξη λοιπόν επιτρέπει τη διαλειτουργικότητα ανάμεσα σε συστήματα λογισμικού συνεργατών ανεξάρτητα από την πλατφόρμα στην οποία έχει αναπτυχθεί κάθε λογισμικό και τα πρωτόκολλα επικοινωνίας που χρησιμοποιεί. Με τον τρόπο αυτό, οι Υπηρεσίες Ιστού, προσφέρουν σημαντικές επιχειρηματικές ευκαιρίες, κάτι που καθιστά εύκολα κατανοητό τον λόγο για τον οποίο θεωρούνται ιδανικές για επιχειρηματικές συναλλαγές.

2.2. Δημοσίευση, εντοπισμός και σύνδεση σε αρχιτεκτονικές τύπου SOA

Για να λειτουργήσει μια αρχιτεκτονική SOA βασισμένη σε ΥΙ είναι απαραίτητη η ύπαρξη μεθόδων δημοσίευσης, εντοπισμού και σύνδεσης των ΥΙ. Χαρακτηριστικό είναι το σχήμα 9.1 που δείχνει τα τρία

βασικά αυτά συστατικά μιας SOA αρχιτεκτονικής, τον τρόπο σύνδεσης τους και τον τρόπο με τον οποίο αυτές επηρεάζουν τους κύριους συμμετέχοντες σε μια συναλλαγή βασισμένη σε μια SOA αρχιτεκτονική.



Σχήμα 9.1 *Το τρίγωνο της αρχιτεκτονικής SOA*

2.3. Δημοσίευση

Όταν ένας πάροχος μιας υπηρεσίας θέλει να την καταστήσει διαθέσιμη προς κατανάλωση θα πρέπει να τη δημοσιεύσει σε κάποιο μητρώο ΥΙ (repository). Το δημοφιλέστερο μητρώο ΥΙ είναι το Universal Description, Discovery and Integration (UDDI) μητρώο στο οποίο είναι δυνατή η αποθήκευση πληροφοριών σχετικά με την ΥΙ αλλά και τον ίδιο τον πάροχο. Το UDDI στηρίζεται στη γλώσσα XML και χρησιμοποιείται ως ένα μέσο για την αποθήκευση πληροφοριών σχετικά με ΥΙ ενώ ταυτόχρονα διευκολύνει και τον εντοπισμό τους. Για την περιγραφή των διεπαφών αλλά και στοιχείων σχετικά με τη λειτουργικότητα μιας ΥΙ γίνεται χρήση ενός αρχείου WSDL. Ένα αρχείο WSDL (Web Services Description Language), αποτελεί ουσιαστικά μια περιγραφή μιας ΥΙ σε XML μορφή. Περιγράφει την τοποθεσία της ΥΙ, τις διάφορες λειτουργίες και μεθόδους που αυτή περιέχει, τις αναμενόμενες εισόδους και εξόδους της καθώς και την αναμενόμενη συμπεριφορά της. Δίνει έτσι τη δυνατότητα σε όποιον επιθυμεί να κάνει χρήση αυτής της ΥΙ καθώς θα μπορεί να στείλει τα κατάλληλα μηνύματα που θα ενεργοποιήσουν τις αντίστοιχες μεθόδους. Περισσότερες πληροφορίες για τα αρχεία WSDL θα δοθούν σε επόμενη ενότητα.

Sound 9.1.mp3	Ηχητικό απόσπασμα (audio)
Η δημοσίευση Υπηρεσιών Ιστού	

2.4. Εντοπισμός

Οι ίδιοι μηχανισμοί είναι αυτοί που βοηθούν τον εντοπισμό τον υπηρεσιών από τους ενδιαφερόμενους clients. Όπως είδαμε προηγουμένως, για την περιγραφή και τη δημοσίευση των λειτουργιών μιας ΥΙ γίνεται χρήση της γλώσσας WSDL. Όταν λοιπόν ένας χρήστης κάνει αναζήτηση για κάποια υπηρεσία, η οποία να ικανοποιεί τα λειτουργικά κριτήρια που επιθυμεί, απευθύνεται συνήθως σε κάποιο ειδικό μητρώο (μέ χρήση πχ. του πρωτοκόλλου UDDI). Καθώς ένα τέτοιο μητρώο, μέσω της υλικοτεχνικής υποδομής του, προσφέρει τη δυνατότητα αποθήκευσης πληροφοριών για διαθέσιμες ΥΙ και για τους παρόχους τους, μέσω WSDL αρχείων, είναι δυνατή και ταυτόχρονα εύκολη η ανακάλυψη και η πρόσβαση σε αυτές.

Sound 9.2.mp3	Ηχητικό απόσπασμα (audio)
Ο εντοπισμός Υπηρεσιών Ιστού	

2.5. Σύνδεση

Αφού ένας client έχει εντοπίσει μέσω UDDI την YI που επιθυμεί να χρησιμοποιήσει, θα πρέπει να πραγματοποιήσει τη σύνδεση του με αυτή. Η σύνδεση πραγματοποιείται με βάση τις προδιαγραφές μηνυμάτων και τις πληροφορίες πρωτοκόλλων που αναφέρονται ρητά στο WSDL αρχείο της υπηρεσίας. Το αρχείο αυτό, όπως έχει ήδη αναφερθεί, περιγράφει την YI και τις διεπαφές (interfaces) της. Ένα παράδειγμα περιγραφής προδιαγραφών βασισμένο στην έκδοση SOAP 1.2 δίνεται παρακάτω:

```
<binding name="SimpeE-CommerceBind" type="tns:SimpleE-Commerce">
        <soap12:binding transport=http://schemas.xmlsoap.org/soap/http style="document" />
        <operation name="ViewProduct">
                <soap12:operation soapAction=http://example.org/ViewProduct soapActionRequired="true"
                style="rpc" />
                <input>
                        <soap12:body use="encoded" namespace=http://example.org/
                        encodingStyle="http://www.w3.org/2001/12/soap-encoding" />
                </input>
                <output>
                        <soap12:body use="encoded" namespace=http://example.org/
                        encodingStyle="http://www.w3.org/2001/12/soap-encoding" />
                </output>
        </operation>
</binding>
```

Η παραπάνω προδιαγραφή ορίζει τον τρόπο με τον οποίο θα πρέπει να γίνεται η σύνδεση και η ανταλλαγή μηνυμάτων μέσω SOAP για μια απλή υπηρεσία ηλεκτρονικού εμπορίου. Ιδιαίτερη σημασία έχει η παράμετρος "transport", η οποία ορίζει το πρωτόκολλο επιπέδου εφαρμογής που θα χρησιμοποιηθεί για την εκπομπή των μηνυμάτων. Όπως θα δούμε στην επόμενη ενότητα, τα πιο συχνά χρησιμοποιούμενα πρωτόκολλα είναι τα Hypertext Transfer Protocol (HTTP) και Simple Mail Transfer Protocol (SMTP). Στο συγκεκριμένο παράδειγμα, γίνεται χρήση του πρωτοκόλλου HTTP.

Sound 9.3.mp3	Ηχητικό απόσπασμα (audio)
Η σύνδεση Υπηρεσιών Ιστού	

3. SOAP-based υπηρεσίες Ιστού: αρχιτεκτονική πλατφόρμας υπηρεσιών Ιστού (web services)

Με τον όρο Υπηρεσίες Ιστού, αναφερόμαστε σε μια υλοποίηση της αρχιτεκτονικής SOA που αναφέρθηκε προηγουμένως. Συγκεκριμένα, αποτελεί μια μέθοδο διαλειτουργικής επικοινωνίας μεταξύ ηλεκτρονικών συσκευών και εφαρμογών μέσω ενός δικτύου, και ακόμα ειδικότερα μια επικοινωνία τύπου μηχανής-προς-μηχανή. Καθώς οι YI αποτελούν μια αρχιτεκτονική ανταλλαγής μηνυμάτων, η οποία βασίζεται στη διαλειτουργικότητα και η οποία δε στηρίζεται σε συγκεκριμένα πρωτόκολλα μεταφοράς για τη μεταφορά μηνυμάτων κατά τη χρήση YI μπορούν να αξιοποιηθούν πολλά γνωστά πρωτόκολλα. Όπως έχει ήδη αναφερθεί τα πιο συχνά χρησιμοποιούμενα πρωτόκολλα είναι τα Hypertext Transfer Protocol (HTTP και HTTPS) και Simple Mail Transfer Protocol (SMTP).

Μια πολύ βασική κατηγορία YI είναι οι βασισμένες σε SOAP μηνύματα YI ή αλλιώς WS-* YI. Η ονομασία προκύπτει λόγω του γεγονότος πως στηρίζονται στην ανταλλαγή μηνυμάτων μέσω του πρωτοκόλλου Simple Object Access Protocol (SOAP). Το SOAP σαν μέσο μετάδοσης πληροφορίας επιτρέπει τη διαλειτουργικότητα ανάμεσα σε εξυπηρετητές (servers) και πελάτες (clients), κατά βάση με χρήση μεθόδων RPC(remote procedure calls), μέσω της ανταλλαγής δομημένων μηνυμάτων XML. Περισσότερες πληροφορίες για το πρωτόκολλο SOAP θα δοθούν στην επόμενη υποενότητα.

Πρέπει να αναφερθεί πως οι Υπηρεσίες Ιστού τύπου SOAP θεωρούνται πολύ αξιόπιστες, αφού προσφέρουν τα μέσα για ασύγχρονη επεξεργασία και επιπρόσθετα παρέχουν τη δυνατότητα ανταλλαγής της

τρέχουσας κατάστασης των λειτουργιών ανάμεσα στον πελάτη και τον εξυπηρετητή που προσφέρει την Υπηρεσία (Pimenidis & Georgiadis, 2010).

Το παρακάτω σχήμα φανερώνει συνοπτικά τα επίπεδα της αρχιτεκτονικής SOA, και συγκεκριμένα τα πρωτόκολλα, τις γλώσσες και τις προδιαγραφές που σχετίζονται με τις ΥΙ τύπου WS-* (Weerawarana et al., 2008).

Σχήμα 9.2 *Τα επίπεδα της αρχιτεκτονικής SOA*

3.1. Περιγραφή πρωτοκόλλου SOAP

Το πρωτόκολλο SOAP σχεδιάστηκε το 1998 από τους Dave Winer, Don Box, Bob Atkinson και Mohsen Al-Ghosein για λογαριασμό της Microsoft. Τα αρχικά SOAP αντιστοιχούσαν στον όρο Simple Object Access Protocol, ο οποίος όμως όρος σταμάτησε να χρησιμοποιείται από την έκδοση 1.2.

Το πρωτόκολλο SOAP επιτρέπει την ανταλλαγή μηνυμάτων μεταξύ ΥΙ. Τα μηνύματα αυτά χαρακτηρίζονται από την ύπαρξη συγκεκριμένης δομής που βασίζεται στη γλώσσα XML. Με τη χρήση του πρωτοκόλλου SOAP, μειώνεται η πολυπλοκότητα (και κατ' επέκταση το απαιτούμενο κόστος) της επικοινωνίας των ετερογενών συστημάτων.

Κάθε SOAP μήνυμα αποτελείται από τρία συστατικά μέλη:

- Τον περιβάλλοντα φάκελο (envelope)
- Την κεφαλίδα (header)
- Το σώμα (body).

Ο περιβάλλοντας φάκελος ορίζει τη δομή του μηνύματος και τον τρόπο επεξεργασίας του. Η κεφαλίδα περιέχει πληροφορίες σχετικά με την εφαρμογή, όπως πληροφορίες σχετικές με συναλλαγές, πληρωμές, κτλ. Επιπρόσθετα, στην κεφαλίδα μπορούν να οριστούν στοιχεία που καθορίζουν τον τρόπο δρομολόγησης του μηνύματος, ενώ αποτελεί παράλληλα και το σημείο στο οποίο πρέπει να αναφέρεται η προσθήκη επεκτάσιμων λειτουργιών στο SOAP. Στο σώμα του μηνύματος αποθηκεύεται το περιεχόμενο του μηνύματος που πρέπει να ληφθεί και να υποστεί επεξεργασία από τον παραλήπτη. Αποτελεί ουσιαστικά το λεγόμενο ωφέλιμο φορτίο (payload) του μηνύματος, τη χρήσιμη δηλαδή πληροφορία.

Ένα παράδειγμα SOAP μηνύματος, όπου διακρίνονται χαρακτηριστικά τα κυρίως συστατικά μέρη του, είναι το παρακάτω:

```
<?xml version="1.0"?>
<soap:Envelope>
        xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
        soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
```

```
        <soap:body pb="http://www.example.uom.gr/members">
            <pb:GetPriceDetails>
            <pb:ProductID>2</pb:ProductID>
            </pb:GetPriceDetails>
        </soap:Body>
</soap:Envelope>
```

Μετά τη λήψη του παραπάνω SOAP μηνύματος, και την επεξεργασία των πληροφοριών που περιέχονται στο κυρίως σώμα (body), δηλαδή του ωφέλιμου φορτίου, η απάντηση ενός εξυπηρετητή, μπορεί να είναι της μορφής:

```
<?xml version="1.0"?>
<soap:Envelope>
    xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
    soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
        <soap:body pb="http://www.example.uom.gr/members">
            <pb:GetPriceDetailsResponse>
            <pb:Price>100</pb: Price >
            </pb:GetPriceDetailsResponse>
        </soap:Body>
</soap:Envelope>
```

3.2. Διευθυνσιοδότηση YI

Η Διευθυνσιοδότηση-YI (WS-Addressing) αποτελεί την περιγραφή ενός μηχανισμού που επιτρέπει την ανταλλαγή πληροφοριών προσδιορισμού, σχετικά με τα εμπλεκόμενα μέλη κατά τη διάρκεια ανταλλαγής μηνυμάτων από YI. Αυτό είναι εφικτό καθώς η περιγραφή ορίζει συγκεκριμένα XML στοιχεία τα οποία χρησιμοποιούνται για την αναφορά ακραίων σημείων (endpoints) δηλαδή σημείων στα οποία μπορεί να στοχεύσει ένα μήνυμα YI. Η περιγραφή αυτή είναι ανεξάρτητη από το μέσο μεταφοράς του μηνύματος, κάτι που προσφέρει ευελιξία στην ανταλλαγή τέτοιων πληροφοριών.

Ουσιαστικά, με τη χρήση της Διευθυνσιοδότησης-YI διαχωρίζεται η λογική της περιγραφής των εμπλεκόμενων μελών μιας επικοινωνίας από το μέσο επικοινωνίας, καθώς οι πληροφορίες διευθυνσιοδότησης όπως είναι ο αποστολέας, ο παραλήπτης κ.τ.λ. αποθηκεύονται στην κεφαλίδα του SOAP μηνύματος. Καθώς η κεφαλίδα θα περιέχει τα απαραίτητα μεταδεδομένα σχετικά με τη διευθυνσιοδότηση του μηνύματος, το μέσο μετάδοσης σε επίπεδο δικτύου είναι υπεύθυνο μόνο για την παράδοση του μηνύματος σε έναν dispatcher ικανό να επεξεργαστεί τα μεταδεδομένα αυτά.

3.3. WSDL

Η γλώσσα Περιγραφής Υπηρεσιών Ιστού (WSDL), είναι μια γλώσσα βασισμένη στην XML η οποία επιτρέπει την περιγραφή διεπαφών. Συγκεκριμένα περιγράφει τις YI ως σύνολα ακραίων σημείων, με τα οποία είναι δυνατή η αλληλεπίδραση. Χρησιμοποιείται για την περιγραφή των λειτουργικών χαρακτηριστικών που προσφέρονται από μια YI. Με τη χρήση μεταδεδομένων επιτρέπει την περιγραφή YI ανεξαρτήτως της πλατφόρμας ανάπτυξης της.

Όταν αναφερόμαστε σε ένα αρχείο WSDL εννοούμε μια περιγραφή μιας YI σε μια μορφή την οποία μπορεί να επεξεργαστεί από μια μηχανή (πχ. μια μηχανή σύνθεσης YI), και η οποία περιλαμβάνει πληροφορίες σχετικά με:

- Την τοποθεσία της YI (π.χ. πληροφορίες για τον πάροχο της αλλά και το μητρώο στο οποίο βρίσκεται)
- Τα λειτουργικά χαρακτηριστικά της
- Τις αναμενόμενες εισόδους που μπορεί να λάβει
- Την αναμενόμενη συμπεριφορά της για δεδομένες εισόδους

- Τις αναμενόμενες εξόδους που μπορεί να δώσει
- Τις μεθόδους που αυτή περιέχει

Για τη δημοσίευση ΥΙ στο Διαδίκτυο και της αντίστοιχης περιγραφής τους, συνήθως χρησιμοποιούνται τα αρχεία WSDL σε συνδυασμό με το πρωτόκολλο SOAP και τα λεγόμενα XML σχήματα. Καθώς τα WSDL αρχεία μπορούν να υποστούν επεξεργασία από μια μηχανή, διαβάζονται από μια εφαρμογή που επιθυμεί να λάβει πληροφορίες για τις επιτρεπόμενες λειτουργίες μιας ΥΙ. Το XML σχήμα είναι υπεύθυνο για την περιγραφή ειδικών τύπων δεδομένων που μπορεί να απαιτούνται από την ΥΙ, ενώ με τη χρήση του SOAP πρωτοκόλλου γίνεται τελικά η επικοινωνία με την ΥΙ, μέσω της κλήσης μιας εκ των λειτουργιών που περιγράφονται στο WSDL αρχείο.

Η τρέχουσα έκδοση της γλώσσας WSDL είναι η έκδοση 2.0. Σε αντίθεση με την έκδοση 1.1., επιτρέπει την περιγραφή συνδέσεων με όλες τις HTTP μεθόδους (GET, POST, PUT, UPDATE) και όχι μόνο με τις μεθόδους GET και POST. Με τον τρόπο αυτό, πέρα από την υποστήριξη των ΥΙ βασισμένων στο SOAP, επιτρέπει την καλύτερη υποστήριξη RESTful ΥΙ, που βασίζονται σε μια ολοένα και δημοφιλέστερη αρχιτεκτονική και στις οποίες θα αναφερθούμε στη συνέχεια αυτού του κεφαλαίου.

4. Συναλλαγές υπηρεσιών Ιστού

Για τα ζητήματα ποιότητας ΥΙ, όπως φαίνεται και στο σχήμα 9.2, ένας παράγοντας ιδιάζουσας σημασίας αποτελεί το ζήτημα των συναλλαγών ΥΙ. Όπως έχει αναφερθεί, οι ΥΙ έχουν αλλάξει τον τρόπο με τον οποίο ετερογενή συστήματα αλληλεπιδρούν παρέχοντας μηχανισμούς διαλειτουργικότητας. Παρά ταύτα, είναι απαραίτητη η ύπαρξη ενός μηχανισμού με τον οποίο θα εξασφαλίζεται η συνεκτικότητα και η αξιοπιστία των εφαρμογών που στηρίζονται στις ΥΙ. Ένας τέτοιος μηχανισμός είναι οι συναλλαγές ΥΙ, καθώς αυτές εξασφαλίζουν πως τα αποτελέσματα της χρήσης κατανεμημένων και βασισμένων-στις-ΥΙ εφαρμογών, θα είναι αυτά στα οποία είχαν εξ' αρχής συμφωνήσει οι συμμετέχοντες. Οι ιδιότητες που παρέχονται από τις συναλλαγές ΥΙ, οι οποίες συχνά αναφέρονται ως ACID είναι οι ακόλουθες:

- **Ατομικότητα (Atomicity)** – Σε περίπτωση επιτυχίας της συναλλαγής, όλες οι ενέργειες της εφαρμογής εκτελούνται, ενώ σε αντίθετη περίπτωση δεν εκτελείται καμία ενέργεια.
- **Συνέπεια (Consistency)** – Τα αποτελέσματα της εφαρμογής χαρακτηρίζονται από συνέπεια και η εφαρμογή επιτελεί ορθές μεταβάσεις καταστάσεων κατά την ολοκλήρωση της.
- **Απομόνωση (Isolation)** – Μέχρι την ολοκλήρωση της συναλλαγής και την παραγωγή της τελικής απάντησης μιας εφαρμογής, μεσολαβούν ενδιάμεσες καταστάσεις. Αυτές, εξασφαλίζεται πως, δεν είναι ορατές από τρίτους ή από άλλες συναλλαγές. Επιπρόσθετα, οι χρησιμοποιούμενοι πόροι μιας συναλλαγής δεν είναι διαθέσιμοι σε άλλες συναλλαγές μέχρι το πέρας της τρέχουσας συναλλαγής.
- **Διάρκεια (Durability)** – Αφότου έχει ολοκληρωθεί μια συναλλαγή, οι αλλαγές που έχει προκαλέσει διατηρούνται ακόμα και αν υπάρξει σφάλμα σε επόμενες ενέργειες.

Οι συναλλαγές με τις ιδιότητες ACID χαρακτηρίζονται ως ατομικές συναλλαγές.

Όπως θα γίνει φανερό σε επόμενο κεφάλαιο, θεμελιώδης σημασίας για την παροχή ΥΙ προστιθέμενης αξίας είναι η σύνθεση ΥΙ. Καθώς όμως πραγματοποιούνται συναλλαγές με χρήση συνθέσεων ΥΙ, εγείρονται ζητήματα σχετικά με το κατά πόσο θα πρέπει οι ιδιότητες ACID να εφαρμόζονται με την ίδια αυστηρότητα. Ο λόγος είναι πως τα συστατικά μέλη μιας σύνθεσης ΥΙ είναι χαλαρά συνδεδεμένες, κατανεμημένες ΥΙ των οποίων η αλληλεπίδραση δημιουργεί την ανάγκη για πιο ευέλικτες ιδιότητες συναλλαγών. Η χρήση των προδιαγραφών Συντονισμού ΥΙ, Ατομικής Συναλλαγής ΥΙ και Επιχειρηματικής Δραστηριότητας ΥΙ εξασφαλίζει την εφαρμογή ενός συνόλου πρωτοκόλλων που επιτρέπουν αυτή την ευελιξία στις συναλλαγές με χρήση ΥΙ.

4.1. Συντονισμός ΥΙ

Ο Συντονισμός ΥΙ αποτελεί μια προδιαγραφή, η οποία περιγράφει τις λειτουργίες οριοθέτησης μιας δραστηριότητας και συγκεκριμένα περιγράφει τα ακόλουθα στοιχεία:

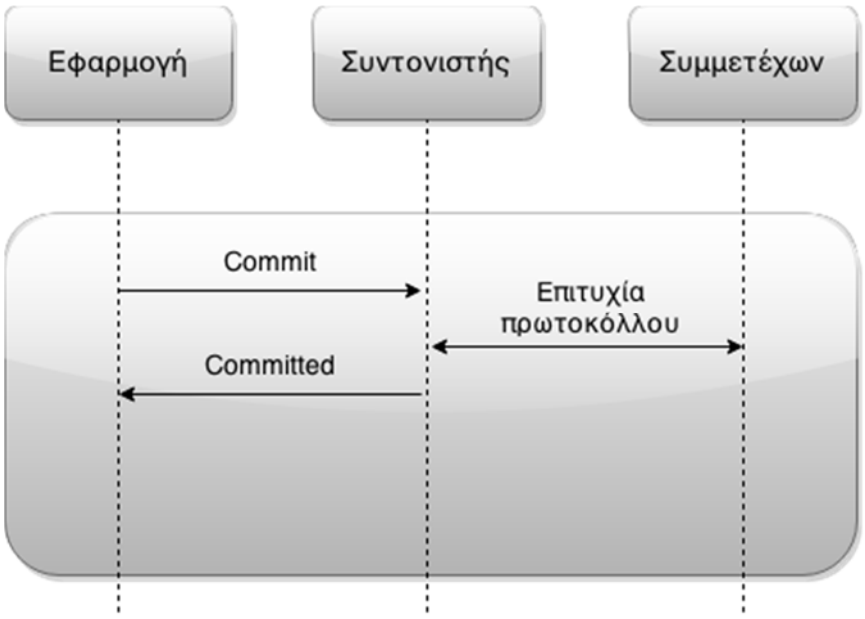
- Ενεργοποίηση – Αφορά τη δημιουργία μιας νέας δραστηριότητας και τη θέσπιση του θεματικού πλαισίου της.
- Θεματικό πλαίσιο – Μέσω του θεματικού πλαισίου καθορίζεται η λειτουργία της δραστηριότητας, λαμβάνοντας υπόψη τις λειτουργίες που έχουν καθοριστεί στην εμβέλεια της. Περιέχει πληροφορίες όπως ένα αναγνωριστικό, τη χρονική στιγμή λήξης της προθεσμίας για την ολοκλήρωση μιας δραστηριότητας, τον τύπο συντονισμού, την υπηρεσία εγγραφής αλλά και στοιχεία για πιθανές επεκτάσεις λειτουργιών.
- Εγγραφή – Με την εγγραφή, μια YI δηλώνει πως συμμετέχει στην απαραίτητη επεξεργασία της δραστηριότητας προκειμένου αυτή να ολοκληρωθεί.
- Πρωτόκολλο συντονισμού – Αφορά στον τρόπο επεξεργασίας των δραστηριοτήτων, έτσι ώστε αυτές να ολοκληρωθούν. Παραδείγματα πρωτοκόλλων συντονισμού είναι η Ατομική Συναλλαγή και η Επιχειρηματική Δραστηριότητα YI που θα περιγραφούν παρακάτω.

4.2. Ατομική συναλλαγή YI

Το πρωτόκολλο Ατομικής Συναλλαγής YI στηρίζεται στις ιδιότητες ACID και ορίζει πως η ολοκλήρωση μιας συναλλαγής θα γίνει ομοιόμορφα για όλους τους συμμετέχοντες. Κατά την εκτέλεση μιας συναλλαγής και σε περίπτωση που μια δραστηριότητα είναι επιτυχής, δίνεται ένα σήμα Commit από την εφαρμογή έτσι ώστε η συναλλαγή να ολοκληρωθεί. Σε αντίθετη περίπτωση δίνεται ένα σήμα Rollback και καμία αλλαγή δε λαμβάνει θέση. Συγκεκριμένα μετά την ολοκλήρωση της οποιασδήποτε δραστηριότητας σε επίπεδο εφαρμογής, δίνεται εντολή στον συντονιστή της συναλλαγής να εκτελέσει τη λειτουργία commit (δέσμευση) και με τον τρόπο αυτό να οριστεί ότι η συναλλαγή ολοκληρώνεται με επιτυχία (Weerawarana et al, 2008).

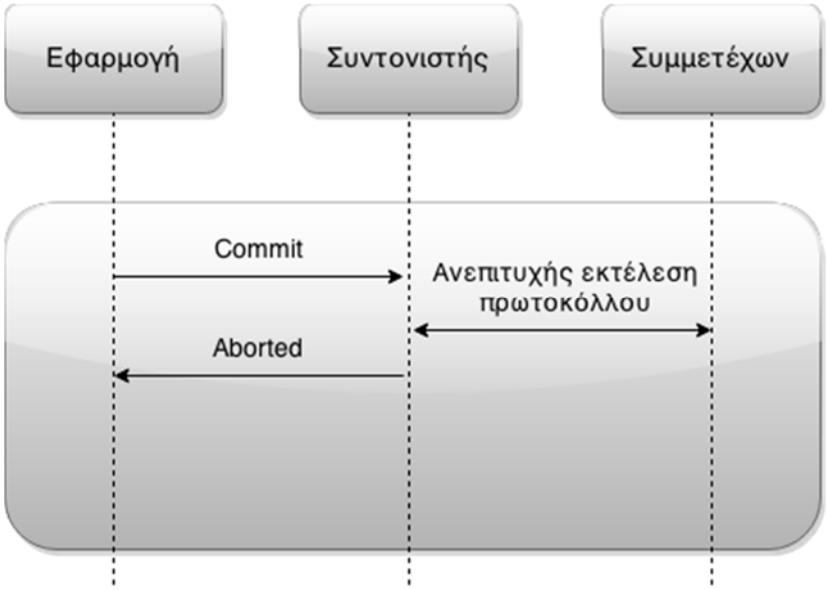
Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

Gif 9.1.gif	Κινούμενη εικόνα (interactive)
Σχήμα 9.3 Επιτυχής συναλλαγή	



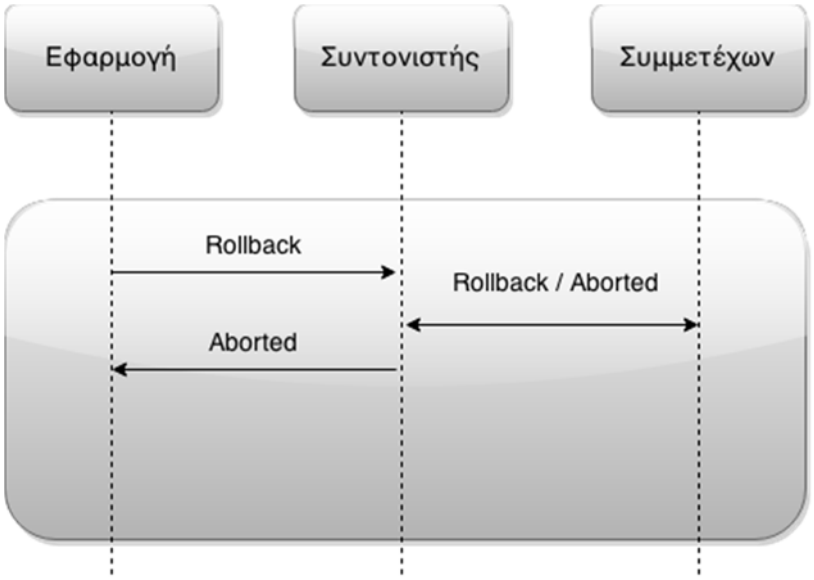
Σχήμα 9.3 *Επιτυχής συναλλαγή*

Σε αντίθετη περίπτωση υπάρχει ανεπιτυχής συναλλαγή όπως φαίνεται στο παρακάτω σχήμα:



Σχήμα 9.4 *Ανεπιτυχής συναλλαγή*

Διαφορετικά, σε περίπτωση σφάλματος σε επίπεδο εφαρμογής έχουμε ανεπιτυχή συναλλαγή λόγω αστοχίας εφαρμογής, όπως φαίνεται στο σχήμα 9.5

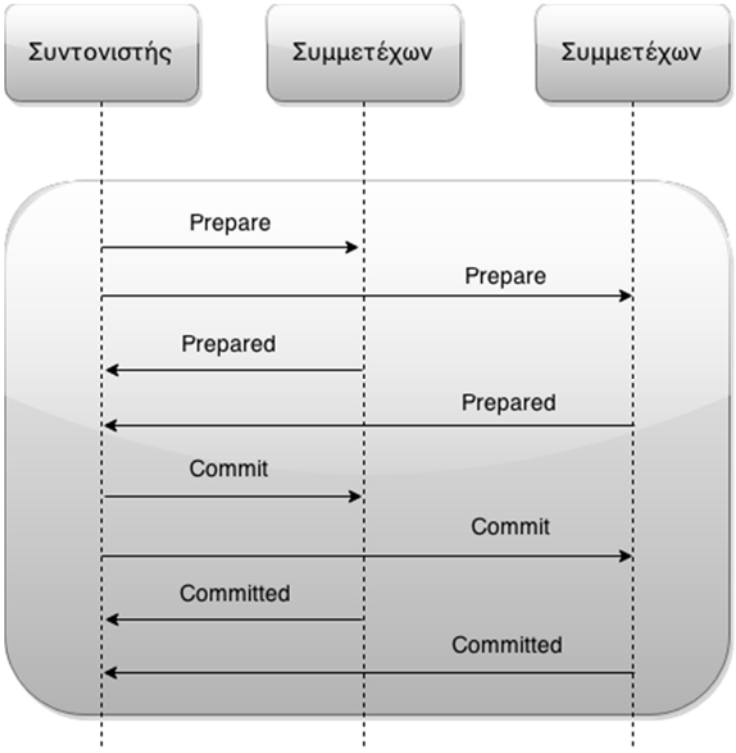


Σχήμα 9.5 *Ανεπιτυχής συναλλαγή λόγω αστοχίας εφαρμογής*

Για την ολοκλήρωση συναλλαγών ατομικού τύπου χρησιμοποιείται το Σταθερό Πρωτόκολλο Δέσμευσης Δύο Φάσεων (Durable Two-Phase Commit). Ένα σενάριο επιτυχούς έκβασης ατομικής συναλλαγής μέσω του συγκεκριμένου πρωτοκόλλου φαίνεται στο επόμενο σχήμα:

Κάντε κλικ για επανάληψη της κίνησης στην παρακάτω εικόνα:

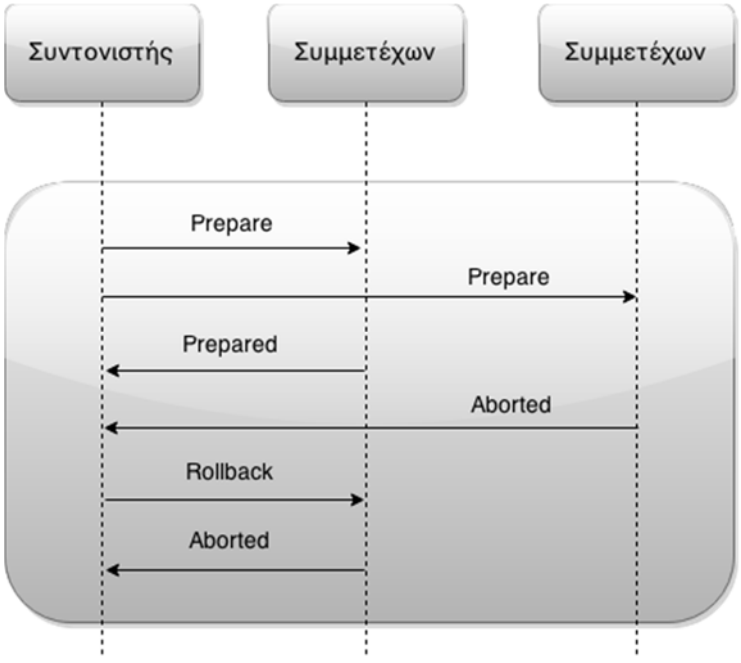
Gif 9.2.gif	Κινούμενη εικόνα (interactive)
Σχήμα 9.6 Επιτυχής δέσμευση, μέσω σταθερού πρωτοκόλλου δύο φάσεων	



Σχήμα 9.6 *Επιτυχής δέσμευση, μέσω σταθερού πρωτοκόλλου δύο φάσεων*

Αρχικά, ο συντονιστής στέλνει ένα σήμα Prepare σε όλους τους συμμετέχοντες. Σε περίπτωση που όλοι οι συμμετέχοντες στείλουν σήμα Prepared, ο συντονιστής στέλνει σε όλους το σήμα commit, σηματοδοτώντας την επιτυχή έκβαση του πρώτου σκέλους της συναλλαγής. Σε αυτό το σημείο, όλοι οι συμμετέχοντες στέλνουν σήμα Commited ως επιβεβαίωση.

Σε περίπτωση που τουλάχιστον ένας συμμετέχων σε μια συναλλαγή δώσει σήμα Aborted στον συντονιστή, αυτός ενημερώνει τους υπόλοιπους συμμετέχοντες για την ανεπιτυχή έκβαση της συναλλαγής στέλνοντας σήμα Rollback. Ως επιβεβαίωση λήψης αυτού του σήματος και οι υπόλοιποι συμμετέχοντες στέλνουν σήμα Aborted.



Σχήμα 9.7 *Ανεπιτυχής δέσμευση, μέσω σταθερού πρωτοκόλλου δύο φάσεων*

Τέλος, αξίζει να αναφερθεί πως υπάρχει ένα ακόμη πρωτόκολλο που περιγράφεται στην προδιαγραφή Ατομική Συναλλαγή YI, το Ασταθές Σταθερό Πρωτόκολλο Δέσμευσης Δύο Φάσεων. Το συγκεκριμένο πρωτόκολλο υιοθετεί τη χρήση προσωρινής μνήμης για την αποθήκευση πληροφοριών, με σκοπό τη βελτίωση της απόδοσης.

4.3. Επιχειρηματική δραστηριότητα YI

Σε εφαρμογές ηλεκτρονικού επιχειρείν που αφορούν σενάρια B2B, υπάρχει η απαίτηση για την ομαλή εκτέλεση των συναλλαγών και την παροχή εγγυήσεων για αυτή. Ταυτόχρονα, η αυξημένη πολυπλοκότητα των B2B συναλλαγών σε συνδυασμό με την ύπαρξη πολλαπλών διασυνδεδεμένων ετερογενών συστημάτων καθιστά αδύνατη τη χρήση ατομικών συναλλαγών. Για τον λόγο αυτό έχει αναπτυχθεί το πρότυπο της Επιχειρηματικής-Δραστηριότητας YI (Weerawarana et al., 2008). Μια εφαρμογή που υιοθετεί το συγκεκριμένο πρότυπο μπορεί να χωριστεί στις λεγόμενες «εμβέλειες», δηλαδή σε συλλογές λειτουργιών YI. Οι εμβέλειες αυτές, οι οποίες χαρακτηρίζονται από σχέσης γονέα-παιδιού, δίνουν στις επιχειρηματικές συναλλαγές επιπρόσθετες δυνατότητες, καθώς επιτρέπουν τη λήψη αποφάσεων σε επίπεδο επιχείρησης. Συγκεκριμένα, δύο σημαντικές δυνατότητες που προσφέρουν περιγράφονται παρακάτω:

Απομόνωση αστοχιών – Σε περίπτωση που ένα σύνολο λειτουργιών YI επιστρέψει μήνυμα ανεπιτυχούς έκβασης, είναι δυνατό η αστοχία αυτή να απομονωθεί και να μην επηρεάσει την εσωτερική εμβέλεια. Κάτι τέτοιο καθιστά εφικτή τη μη ακύρωση των ενεργειών που έχουν εκτελεστεί μέχρι τη δεδομένη εκείνη στιγμή.

Τμηματικότητα – Αφορά στον σωστό καταμερισμό των εργασιών σε εμβέλειες, με τέτοιο τρόπο ώστε δραστηριότητες που βρίσκονται έξω από τα πλαίσια της επιχείρησης να μπορούν να εκτελεστούν μέσω μηχανισμών ροής εργασιών. Παράλληλα, αφορά στον ορισμό των ένθετων εμβελειών, δηλαδή εμβελειών που αρχικοποιούνται μέσα στη ροή εξωτερικών εμβελειών, και στις οποίες σχηματίζονται σχέσεις γονέα-παιδιού.

Στο πρότυπο της Επιχειρηματικής-Δραστηριότητας YI, όταν μια εμβέλεια-παιδί ολοκληρώνεται, στέλνει μήνυμα ολοκλήρωσης στην εμβέλεια-γονέα. Σε αντίθεση με τις Ατομικές Συναλλαγές YI, υπάρχει η δυνατότητα αντιστάθμισης από τον γονέα, δηλαδή της αναστροφής της λειτουργίας που επιτέλεσε η εμβέλεια-παιδί.

Δύο σημαντικά πρωτόκολλα του προτύπου Επιχειρηματικής-Δραστηριότητας YI, είναι τα παρακάτω:

- Επιχειρηματική Συμφωνία με Ολοκλήρωση από τους Συμμετέχοντες (Business Agreement with Participant Completion)
- Επιχειρηματική Συμφωνία με Ολοκλήρωση από τον Συντονιστή (Business Agreement with Coordinator Completion)

4.3.1. Επιχειρηματική Συμφωνία με Ολοκλήρωση από τους Συμμετέχοντες

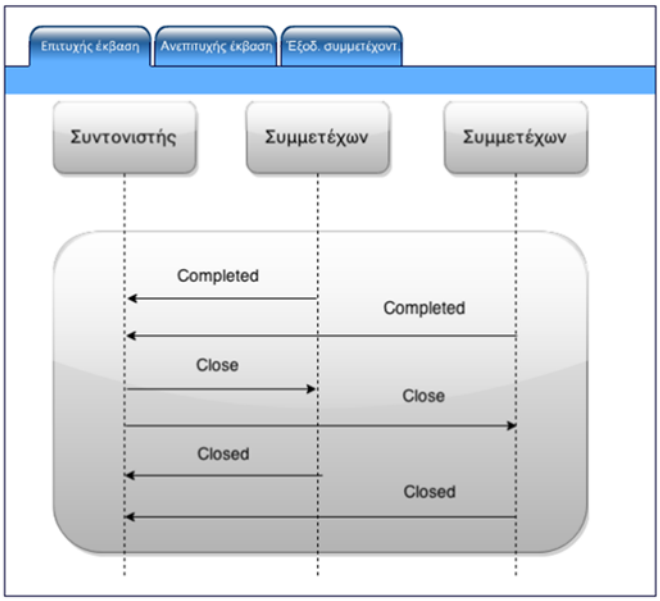
Σε αυτό το πρωτόκολλο δημιουργείται μια θυγατρική δραστηριότητα, η οποία πρέπει να έχει τη δυνατότητα να επιτελέσει την αντιστάθμιση των ενεργειών που εκτελέστηκαν. Συγκεκριμένα με την ολοκλήρωση της, στέλνει ένα μήνυμα Completed προς τη δραστηριότητα-γονέα και αναμένει τη λήψη μηνύματος που θα την πληροφορήσει για την έκβαση της επιχειρηματικής δραστηριότητας. Σε περίπτωση που λάβει μήνυμα Close αυτό θα σημαίνει την επιτυχή ολοκλήρωση της επιχειρηματικής δραστηριότητας, κάτι που σημαίνει πως δεν απαιτούνται πρόσθετες ενέργειες από μέρους της. Σε περίπτωση του λάβει μήνυμα Compensate η θυγατρική δραστηριότητα θα πρέπει να αντιστρέψει τα αποτελέσματα των ενεργειών που έχει επιτελέσει.

Στο παρακάτω σχήμα φαίνονται περιγραφικά διαφορετικά σενάρια εκτέλεσης επιχειρηματικών συμφωνιών με ολοκλήρωση από τους συμμετέχοντες. Συγκεκριμένα φαίνονται τα παρακάτω τρία σενάρια:

- Επιτυχής έκβαση μιας επιχειρηματικής συμφωνίας – ολοκλήρωση από τους συμμετέχοντες.
- Ανεπιτυχής έκβαση μιας επιχειρηματικής συμφωνίας – ολοκλήρωση από τους συμμετέχοντες.
- Έξοδος συμμετέχοντος από επιχειρηματική συμφωνία

Στο παρακάτω σχήμα επιλέξτε μια από τις τρεις καρτέλες

Flash 9.1.swf	Αρχείο flash (interactive)
Σχήμα 9.8 Σενάρια εκτέλεσης επιχειρηματικών συμφωνιών	

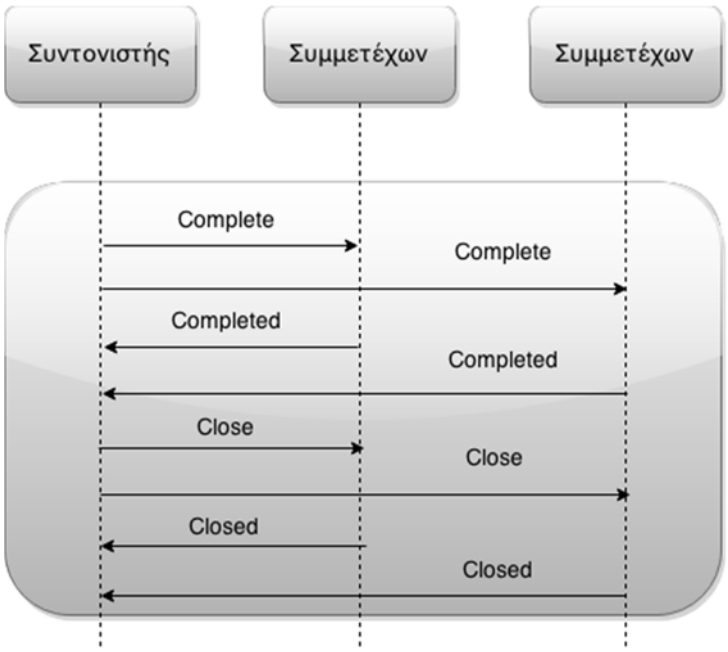


Σχήμα 9.8 *Σενάρια εκτέλεσης επιχειρηματικών συμφωνιών*

4.3.2. Επιχειρηματική Συμφωνία με Ολοκλήρωση από το Συντονιστή

Η κυρίαρχη διαφορά αυτού του πρωτοκόλλου σε σχέση με το πρωτόκολλο ολοκλήρωσης από τους συμμετέχοντες εντοπίζεται στο γεγονός πως μια θυγατρική διαδικασία δεν έχει τη δυνατότητα να τερματίσει τη συμμετοχή της στην επιχειρηματική δραστηριότητα αυτοβούλως. Αντιθέτως, η συγκεκριμένη διαδικασία αναμένει ένα μήνυμα Complete από τη διαδικασία γονέα, η οποία σημαίνει και την επιτυχή λήψη όλων των αιτήσεων για την εκτέλεση μιας συγκεκριμένης εργασίας.

Το σχήμα 9.9 παρουσιάζει ένα σενάριο εκτέλεσης επιχειρηματικών συμφωνιών με ολοκλήρωση από τον συντονιστή.



Σχήμα 9.9 *Επιτυχής έκβαση επιχειρηματικής συμφωνίας – ολοκλήρωση από τον συντονιστή.*

5. Ασφάλεια υπηρεσιών Ιστού

Μια ακόμη σημαντική παράμετρος σε ζητήματα ποιότητας ΥΙ αποτελεί η ασφάλεια ΥΙ. Στα πλαίσια της διασφάλισης των ΥΙ, έχει οριστεί ένα σύνολο από πρωτόκολλα και προδιαγραφές, που έχει ως στόχο τη διασφάλιση των συναλλαγών που πραγματοποιούνται με χρήση ΥΙ, και ειδικότερα την αντιμετώπιση κινδύνων που μπορούν να προκληθούν από εξωτερικούς κακοπροαίρετους χρήστες. Πιθανοί κίνδυνοι προκύπτουν από επιθέσεις ασφαλείας οι οποίοι εστιάζουν είτε στην ΥΙ και σε πιθανές αδυναμίες της είτε στο δίκτυο που χρησιμοποιείται για τη διεξαγωγή των συναλλαγών.

Παραδείγματα τέτοιων επιθέσεων αποτελούν οι υποκλοπές αλλά και οι τροποποιήσεις μηνυμάτων, ζητήματα σχετικά με την αυθεντικοποίηση χρηστών που συμμετέχουν σε μια συναλλαγή αλλά και οι μαζικές επιθέσεις κατά των δικτύων, οι γνωστές επιθέσεις κατανεμημένης άρνησης παροχής υπηρεσιών (Distributed Denial of Service-DDoS).

Το σύνολο των προδιαγραφών ασφαλείας φαίνονται στο παρακάτω σχήμα (Weerawarana et al., 2008):

Σχήμα 9.10 *Πρωτόκολλα και προδιαγραφές ασφαλείας των ΥΙ*

5.1. Ασφάλεια ΥΙ

Η ασφάλεια ΥΙ έχει εκδοθεί από την OASIS και η τωρινή της έκδοση είναι η 1.1.1. Περιέχει οδηγίες γνωστές ως SOAP message security, οι οποίες περιγράφουν τρεις κύριους μηχανισμούς:

- Την υπογραφή μηνυμάτων SOAP για την εξασφάλιση της ακεραιότητας. Με τον τρόπο αυτό εξασφαλίζεται πως τα μηνύματα δεν τροποποιούνται από μη-εξουσιοδοτημένους χρήστες.
- Την κρυπτογράφηση των SOAP μηνυμάτων για τη διασφάλιση της εμπιστευτικότητας. Με τον τρόπο αυτό εξασφαλίζεται πως μόνο ο τελικός παραλήπτης του μηνύματος θα είναι σε θέση να δει το περιεχόμενο του μηνύματος.
- Τη χρήση tokens ασφαλείας για την πιστοποίηση της ταυτότητας του αποστολέα. Είναι απαραίτητη προϋπόθεση για την ασφαλή επικοινωνία το να γνωρίζει ο παραλήπτης την προέλευση ενός μηνύματος SOAP. Παραδείγματα tokens ασφαλείας που υποστηρίζονται αποτελούν τα πιστοποιητικά X.509 και τα Kerberos tickets.

5.2. Εμπιστοσύνη ΥΙ

Η Εμπιστοσύνη ΥΙ παρέχει επεκτάσεις στο πρωτόκολλο της Ασφάλειας ΥΙ και αποτελεί ένα standard της OASIS. Καθώς η εμπιστοσύνη αποτελεί το θεμελιώδη λίθο για την εγκαθίδρυση της ασφάλειας είναι απαραίτητη η εξασφάλιση της σε κάθε συναλλαγή που κάνει χρήση ΥΙ.

Πριν από τη χρήση μιας ΥΙ, ο τελικός χρήστης ελέγχει τα πρωτόκολλα και τις δηλώσεις Πολιτικής Ασφάλειας ΥΙ του παρόχου της ΥΙ. Για τη χρήση της πρέπει να υπάρχει συμφωνία στα λεγόμενα δελτία ασφαλείας που χρησιμοποιεί η ΥΙ, τα οποία πρέπει να κατέχει και ο χρήστης. Σε περίπτωση που δεν τα κατέχει θα πρέπει να τα προμηθευτεί από έναν Security Token Server (διακομιστή δελτίων ασφαλείας) ο

οποίος να θεωρείται έμπιστος από τον πάροχο της YI. Η ύπαρξη αυτών των δελτίων εξασφαλίζει την εμπιστοσύνη ανάμεσα στον πάροχο της YI και τον τελικό χρήστη. Επιπρόσθετα, η Εμπιστοσύνη YI περιγράφει τη δομή των μηνυμάτων αίτησης δελτίων ασφαλείας αλλά και τον τρόπο ανταλλαγής κλειδιών ανάμεσα στους εμπλεκόμενους μιας συναλλαγής.

Σε σχέση με τα προαναφερθέντα δελτία ασφάλειας η Εμπιστοσύνη YI βασίζεται σε τρεις λειτουργίες:

- Έκδοση (Issuance) – Αφορά στην έκδοση ενός νέου δελτίου ασφαλείας
- Ανανέωση (Renewal) – Αφορά στην ανανέωση της ισχύς ενός υπάρχοντος δελτίου ασφαλείας
- Επικύρωση (Validation) – Αφορά στην επικύρωση ενός δελτίου ασφαλείας ως προς τη συμμόρφωση του με τις πολιτικές του παρόχου της YI.

6. REST υπηρεσίες Ιστού

Η αρχιτεκτονική Representational State Transfer (REST), είναι μια service-oriented αρχιτεκτονική για κατανεμημένα συστήματα. Η αρχιτεκτονική REST ορίζει συγκεκριμένες αρχές για τον σχεδιασμό YI, με βάση τους πόρους (resources) και τις αναπαραστάσεις τους (representations), και έχει σαν στόχο την απροβλημάτιστη διαλειτουργικότητα ετερογενών συστημάτων χαλαρής ζεύξης (Fielding, 2000).

Με τον όρο resource μπορεί να χαρακτηριστεί οποιαδήποτε πληροφορία ή έννοια. Έτσι, σε συστήματα ηλεκτρονικού εμπορίου, ως resource μπορούμε να χαρακτηρίσουμε μια παραγγελία, ένα σύνολο παραγγελιών (πχ. το ιστορικό ενός πελάτη) , έναν συγκεκριμένο χρήστη, ένα προϊόν κ.α.

Ωστόσο, ενώ resource μπορεί να είναι οποιαδήποτε πληροφορία ή έννοια, η αναπαράσταση αυτής (representation) είναι το έγγραφο ή το αρχείο συγκεκριμένης μορφοποίησης που περιγράφει την τρέχουσα κατάσταση του resource (Pautasso, 2014). Για κάθε resource είναι δυνατό να υπάρχουν πολλές διαφορετικές αναπαραστάσεις, όπου οι πιο συχνά χρησιμοποιούμενες είναι οι απλές HTML σελίδες και οι σελίδες που ακολουθούν τις μορφοποιήσεις XML και JSON. Αυτό επιτρέπει σε κάθε πελάτη να μπορεί κατά την αποστολή ενός αιτήματος για την τρέχουσα κατάσταση ενός πόρου, να ζητήσει και συγκεκριμένη μορφοποίηση για την επιστρεφόμενη αναπαράσταση. Βεβαίως υπάρχει περίπτωση ο εξυπηρετητής να μην μπορεί να επιστρέψει την αναπαράσταση στην αιτούμενη μορφοποίηση, ενημερώνοντας τον πελάτη με κατάλληλο μήνυμα του HTTP πρωτοκόλλου.

Όπως ήδη αναφέρθηκε, η αιτούμενη μορφοποίηση της αναπαράστασης βρίσκεται μέσα στο απεσταλμένο μήνυμα. Αυτό συμβαίνει διότι η αρχιτεκτονική REST επιβάλει τα μηνύματα να είναι αυτό-περιγραφόμενα, δηλαδή να περιέχουν όλη την πληροφορία που χρειάζονται ώστε να επεξεργαστούν, για παράδειγμα δίχως να γνωρίζει ο εξυπηρετητής κάτι για την κατάσταση του πελάτη (Pautasso et al., 2014). Έτσι λοιπόν όλα τα μηνύματα θα πρέπει να περιέχουν πληροφορία για τη μορφοποίηση των αναπαραστάσεων, πληροφορίες για την προσωρινή αποθήκευση κ.α.

Στην REST αρχιτεκτονική είναι δυνατή η πρόσβαση στις λειτουργίες των YI μέσω των Universal Resource Identifier (URI). Μέσω της κλήσης μιας YI και κάνοντας χρήση του HTTP πρωτοκόλλου και των HTTP μεθόδων (GET/PUT/POST/DELETE) είναι δυνατή η πρόσβαση και ο χειρισμός των πόρων. Σε μια αρχιτεκτονική τύπου REST δεν υπάρχει ανάγκη υιοθέτησης του μητρώου UDDI καθώς το μόνο προαπαιτούμενο προκειμένου να είναι δυνατή η πρόσβαση στην υπηρεσία, είναι η γνώση του URI της. Για να μπορεί μια YI να θεωρηθεί RESTful, θα πρέπει αυτή να ικανοποιεί τους περιορισμούς της REST αρχιτεκτονικής. Οι περιορισμοί αυτοί είναι:

1. Client–server: Κάθε σύστημα που βασίζεται στην αρχιτεκτονική REST θα πρέπει να χαρακτηρίζεται από διαχωρισμό των ευθυνών (separation of concerns). Πιο συγκεκριμένα, ζητήματα όπως η αποθήκευση των δεδομένων θα πρέπει να αποτελούν έγνοια μόνο του εξυπηρετητή, ενώ ο εξυπηρετητής δε θα πρέπει να κρατάει πληροφορία σχετική με την κατάσταση στην οποία βρίσκεται ο κάθε πελάτης.
2. Uniform interface: Η διεπιφάνεια (interface) κάθε συστατικού μέλους μιας REST αρχιτεκτονικής θα πρέπει να ακολουθεί συγκεκριμένους γενικούς κανόνες ομοιογένειας, κάτι που διευκολύνει την πρόσβαση στις υπηρεσίες τους, ενώ παράλληλα συμβάλει στη διευκόλυνση της κλιμακωσιμότητας (scalability).

3. Stateless: Κάθε αίτημα ενός client περιέχει όλες τις απαραίτητες πληροφορίες για την επεξεργασία του μηνύματος καθώς και για την κατάσταση που βρίσκεται τη συγκεκριμένη στιγμή ο client. Δεν είναι λοιπόν απαραίτητο ο server να κρατάει πληροφορίες σχετικά με την κατάσταση του κάθε client, κάτι που έχει μεγάλο αντίκτυπο στην κλιμακωσιμότητα και την αξιοπιστία του συστήματος.
4. Cacheable: Οι εξυπηρετητές επιτρέπουν κάποιες απαντήσεις τους να αποθηκευθούν να προσωρινή μνήμη στην πλευρά του πελάτη. Αυτό συμβαίνει ιδιαίτερα σε πληροφορίες που δεν αλλάζουν με ταχείς ρυθμούς. Αντίθετα, είναι συχνό φαινόμενο το να μην επιτρέπουν την προσωρινή αποθήκευση απαντήσεων, ιδιαίτερα όταν αφορούν πληροφορίες οι οποίες έχουν μικρή διάρκεια ζωής. Με τον τρόπο αυτό, οι clients μπορούν σε ορισμένες περιπτώσεις να αποφύγουν την άσκοπη επικοινωνία με τον server, κάτι που μπορεί να βελτιώσει σε μεγάλο βαθμό την απόδοση ενός συστήματος ιδιαίτερα σε συστήματα μεγάλης κλίμακας, ενώ ταυτόχρονα προστατεύονται από τη χρήση μη επικαιροποιημένων πληροφοριών, κάτι που θα μπορούσε να οδηγήσει σε αλλοίωση δεδομένων.
5. Layered system: Στις αρχιτεκτονικές τύπου REST οι clients μπορούν να συνδεθούν είτε απευθείας στον server που παρέχει μια υπηρεσία, είτε σε ενδιάμεσους κόμβους-εξυπηρετητές δίχως να το γνωρίζουν. Αυτό συμβαίνει διότι η αρχιτεκτονική REST χρησιμοποιεί ιεραρχικά επίπεδα για την κατανομή των κόμβων που συμμετέχουν σε ένα σύστημα, όπου κάθε κόμβος μπορεί να επικοινωνεί μόνο με τους κόμβους στους οποίους βρίσκεται πλησιέστερα.
6. Code on demand (προαιρετικό): Ο κώδικας κατά απαίτηση (Code on demand) είναι ο μοναδικός περιορισμός που είναι προαιρετικός για την REST αρχιτεκτονική. Πέρα από τις ζητούμενες πληροφορίες και τις αναπαραστάσεις πόρων, οι server σε κάποιες περιπτώσεις μπορούν να προσφέρουν και εκτελέσιμο κώδικα στον client, με τις πιο συνήθης υλοποιήσεις να αφορούν κώδικα σε JavaScript, αλλά και Java applets.

7. Επιλογή και σύνθεση υπηρεσιών Ιστού

Όπως γίνεται φανερό, τα τελευταία χρόνια οι ΥΙ έχουν φέρει επανάσταση στον τρόπο με τον οποίο ετερογενή συστήματα επικοινωνούν και αλληλεπιδρούν διαδικτυακά. Μια από τις μεγαλύτερες ευκαιρίες που παρέχει η χρήση της τεχνολογίας των ΥΙ είναι η σύνθεση ΥΙ για τη δημιουργία ΥΙ προστιθέμενης αξίας με προσανατολισμό σε συγκεκριμένο πεδίο εφαρμογής. Ειδικότερα στον τομέα του ηλεκτρονικού εμπορίου, παραδείγματα σύνθεσης ΥΙ θα μπορούσαν να αποτελούν συστήματα που συνδυάζουν υπηρεσίες ηλεκτρονικού καλαθιού, μεταφορικών εταιριών, υπηρεσιών διαφήμισης και πληρωμής. Ωστόσο, η προαναφερθείσα επιτυχία των ΥΙ έχει οδηγήσει στην ύπαρξη πληθώρας ΥΙ, γεγονός που καθιστά δύσκολη ,τόσο για τους χρήστες όσο και για τις επιχειρήσεις, την επιλογή των ιδανικών ΥΙ, είτε αυτές πρόκειται να χρησιμοποιηθούν μεμονωμένα είτε σαν μέλος μιας ευρύτερης σύνθεσης.

Πολλές τεχνολογίες και μέθοδοι έχουν προταθεί στη βιβλιογραφία για την αντιμετώπιση ζητημάτων επιλογής και σύνθεσης (Sheng et al., 2014). Οι σημασιολογίες, μέσω της χρήσης οντολογιών, αποτελούν μια πολύ δημοφιλή μέθοδο επιλογής (Hatzi et al., 2012). Άλλες προσεγγίσεις αφορούν στη χρήση αλγορίθμων όπως ο skyline αλγόριθμος για το φιλτράρισμα ΥΙ (Alrifai et al., 2010) αλλά και μέθοδοι που προέρχονται από τον τομέα της επιχειρησιακής έρευνας και επιτρέπουν την επιλογή της βέλτιστης εναλλακτικής με βάση προκαθορισμένα και συχνά αντικρουόμενα κριτήρια. Αυτές οι τεχνικές ανήκουν στην κατηγορία των πολυκριτηριακών μεθόδων απόφασης και θα αναπτυχθούν περισσότερο στο επόμενο κεφάλαιο.

Επιπρόσθετα χαρακτηριστικά ποιότητας (Quality of Service – QoS), μιας υπηρεσίας μπορούν να ληφθούν υπόψη κατά τη διαδικασία βελτιστοποίησης μιας σύνθεσης ΥΙ.

7.1. Σύνθεση ΥΙ τύπου WS*

Υπάρχουν διάφορες προσεγγίσεις για τη σύνθεση ΥΙ, και για τον λόγο αυτό η επιλογή της κατάλληλης σχετίζεται με τις επιχειρηματικές ανάγκες των εμπλεκομένων μερών. Πριν εισάγουμε τις σύγχρονες προσεγγίσεις όμως, πρέπει να αναλυθούν οι όροι ενορχήστρωση και χορογραφία.

Η Χορογραφία (Choreography) σχετίζεται με την ανταλλαγή δημοσίων μηνυμάτων, τις διαπραγματεύσεις και τις συμφωνίες μεταξύ των επιχειρηματικών διαδικασιών και τέλος των κανόνων που εφαρμόζονται. Κατά κύριο λόγο έχει προσανατολισμό προς τους clients των ΥΙ, είτε αυτοί είναι οι τελικοί

χρήστες, είτε πρόκειται απλά για ΥΙ που «καταναλώνουν» άλλες ΥΙ. Καθώς οι συναλλαγές ανάμεσα σε ΥΙ θα πρέπει να καθορίζονται με τρόπο ξεκάθαρο πριν την υλοποίηση τους, όλοι οι συμμετέχοντες στη σύνθεση των υπηρεσιών έχουν ίσα δικαιώματα και μπορούν, ανά πάσα στιγμή, να έχουν πρόσβαση σε πληροφορίες σχετικά με το πώς κάθε υπηρεσία μπορεί να συνεργαστεί με μια άλλη.

Από την άλλη πλευρά, η Ενορχήστρωση, καθορίζεται κυρίως από XML-based γλώσσες ορισμού, όπως η BPEL, και ουσιαστικά περιγράφει το πώς οι υπηρεσίες μπορούν να αλληλεπιδρούν εστιάζοντας όμως σε συγκεκριμένες ΥΙ και δε διαθέτει τον δημόσιο χαρακτήρα της Χορογραφίας.

7.2. Χειροκίνητες συνθέσεις (manual compositions)

Η πιο κοινή μέθοδος σύνθεσης είναι η λεγόμενη χειροκίνητη σύνθεση ΥΙ, κατά την οποία ο χρήστης εισάγει μία λίστα παραμέτρων (η οποία περιγράφει τις λειτουργίες που επιθυμεί από την ΥΙ ή ακόμη και τα μη-λειτουργικά χαρακτηριστικά που επιθυμεί όπως είναι τα QoS χαρακτηριστικά), και λαμβάνει μια λίστα από δυνητικά κατάλληλες ΥΙ για τη σύνθεση. Στη συνέχεια, ο χρήστης καλείται να συγχωνεύσει αυτές τις ΥΙ δίνοντας ταυτόχρονα σαφείς οδηγίες σχετικά με τον τρόπο σύνδεσης και αλληλεπίδρασης. Παράλληλα, ο χρήστης θα πρέπει να μεριμνήσει για τον ορισμό της σειράς επίκλησης των ΥΙ αλλά και για τον καθορισμό των εξόδων συγκεκριμένων ΥΙ που θα μπορούν να χρησιμοποιηθούν ως είσοδοι για άλλες ΥΙ. Παρότι αποτελεί μια αρκετά αξιόπιστη μέθοδο σύνθεσης, η οποία εξασφαλίζει πως το τελικό αποτέλεσμα της σύνθεσης θα είναι πολύ κοντά στο επιθυμητό, παρουσιάζει το σημαντικό μειονέκτημα της απαίτησης της ανθρώπινης παρέμβαση σε κάθε βήμα. Είναι όμως μια μέθοδος που έχει ευρεία απήχηση σε συγκεκριμένους τομείς, όπου η αλλαγή μιας σύνθεσης μπορεί να είναι επιζήμια.

7.3. Μερικώς αυτοματοποιημένες συνθέσεις (partially automated composition)

Στην περίπτωση των μερικώς αυτοματοποιημένων συνθέσεων, ο τελικός χρήστης μπορεί να επιλέξει από μια σειρά ΥΙ που έχουν ανακαλυφθεί, με βάση κάποια υποκειμενικά κριτήρια, χωρίς κατ' ανάγκη να γνωρίζει κάθε πτυχή σχετικά με τα λειτουργικότητα και μη-λειτουργικά χαρακτηριστικά της επιλεγμένης ΥΙ. Έτσι, ο χρήστης έχει μικρότερη εμπλοκή και χρειάζεται να επέμβει περισσότερο μόνο σε περιπτώσεις κατά τις οποίες υπάρχει ελλιπής κατάλογος επιστρεφόμενων αποτελεσμάτων ή υπάρχουν ασυμβατότητες ανάμεσα σε ΥΙ, κάτι που θα καθιστούσε αδύνατη τη σύνθεση μιας υπηρεσίας προστιθέμενης αξίας. Στην ημι-αυτοματοποιημένη σύνθεση βασικό ρόλο επιτελούν τα μεταδεδομένα και οι σημασιολογικές περιγραφές των ΥΙ, γεγονός που διευκολύνει την αυτοματοποιημένη ανακάλυψη τους.

7.4. Αυτοματοποιημένη σύνθεση (automated composition)

Η προσέγγιση της αυτοματοποιημένης σύνθεσης (Wang, et al., 2014) χαρακτηρίζεται από την εισαγωγή μιας μηχανής σύνθεσης. Μια τέτοια μηχανή συνθέτει ΥΙ με αυτοματοποιημένο τρόπο και χωρίς τη μεσολάβηση του ανθρώπινου παράγοντα. Αυτό καθίσταται εφικτό με τη χρήση προηγμένων σημασιολογικών κανόνων, αλλά και αλγορίθμων. Στην περίπτωση αυτή, ο τελικός χρήστης περιγράφει μόνο τις επιχειρηματικές του ανάγκες και τους επιδιωκόμενους στόχους, και δε συμμετέχει στη διαδικασία ανακάλυψης, επιλογής και σύνθεσης. Σε ακόμα πιο προχωρημένες περιπτώσεις αυτοματοποιημένων συνθέσεων, η μηχανή σύνθεσης μπορεί να λάβει υπόψη στοιχεία του εξωτερικού περιβάλλοντος και προκλήσεις της επιχείρησης και αναλόγως να προσαρμόσει τη διαδικασία σύνθεσης. Ένα τέτοιο παράδειγμα σε περιβάλλοντα ηλεκτρονικού εμπορίου θα ήταν η λήψη στοιχείων σχετικά με την τιμολογιακή πολιτική ανταγωνιστών και η σύνθεση υπηρεσιών με τέτοιο τρόπο έτσι ώστε να εξασφαλίσει η επιχείρηση ανταγωνιστικό πλεονέκτημα.

7.5. Σύνθεση βασισμένη στη μοντελοποίηση (model-based composition)

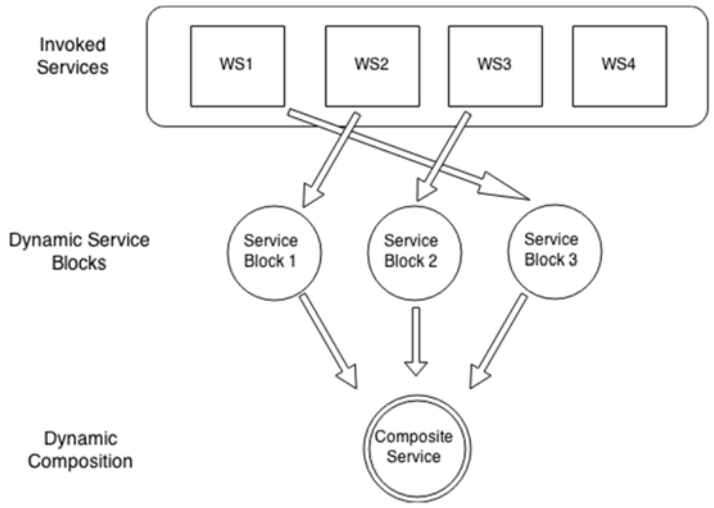
Μια διαφορετική προσέγγιση για τη σύνθεση ΥΙ είναι η προσέγγιση με βάση τη μοντελοποίηση. Αφορά στη δημιουργία σε πρώτο στάδιο ενός θεωρητικού μοντέλου που θα περιγράφει την ενορχήστρωση των ΥΙ όπως αυτή προκύπτει από τα στάδια της ανακάλυψης και της επιλογής. Το μοντέλο αυτό μπορεί να δημιουργηθεί σε μια καθιερωμένη γλώσσα μοντελοποίησης όπως για παράδειγμα στη UML, στη συνέχεια όμως μπορεί να

γίνει μετασχηματισμός του μοντέλου αυτού σε μια μορφή περισσότερο domain specific με απώτερο στόχο τον έλεγχο ιδιοτήτων αλλά και την αυτόματη παραγωγή κώδικα.

7.6. Δυναμικές συνθέσεις (dynamic composition)

Ενώ τα παραπάνω περιγράφουν στατικές συνθέσεις ΥΙ, είναι δυνατόν να έχουμε και δυναμικές συνθέσεις που εκτελούνται κατά τη διάρκεια της εκτέλεσης του συστήματος, κάτι που είναι εφικτό χάρη στη λεγόμενη «δυναμική σύνδεση υπηρεσιών». Δεδομένου ότι πολλές αλλαγές μπορούν να συμβούν στο περιβάλλον μιας επιχείρησης κατά τη διάρκεια της εκτέλεσης ενός συστήματος και καθώς νέες ΥΙ συνεχώς προστίθενται στο σύνολο των διαθέσιμων ΥΙ, η δυναμική σύνδεση υπηρεσιών παρέχει τα μέσα για τη δημιουργία ευέλικτων και συνεχώς αναπροσαρμοζόμενων ΥΙ. Με τον τρόπο αυτό είναι δυνατή η συνεχής βελτίωση της απόδοσης της σύνθεσης. Η μεθοδολογία αυτή βασίζεται στην αντικατάσταση υπηρεσιών από νέες κατά τη διάρκεια εκτέλεσης του συστήματος, καθώς αυτές ίσως να εκπληρώνουν καλύτερα τις διαρκώς μεταβαλλόμενες ανάγκες των επιχειρήσεων. Η μεθοδολογία ουσιαστικά στηρίζεται στη δημιουργία δυναμικών συνδέσμων, δημιουργώντας ουσιαστικά τις θέσεις στις οποίες θα τοποθετηθούν οι ΥΙ κατά τη διάρκεια της εκτέλεσης, όπως φαίνεται και στο σχήμα 9.11.

Dynamic 9.1.zip	Διαδραστική εικόνα (interactive)
Σχήμα 9.11 Σύνθεση ΥΙ με χρήση δυναμικών συνδέσμων	



Σχήμα 9.11 *Σύνθεση ΥΙ με χρήση δυναμικών συνδέσμων*

Οι δυναμικές συνθέσεις είναι δυσκολότερες στον χειρισμό καθώς στηρίζονται στη χρήση πολύπλοκων αλγορίθμων για τη δυναμική σύνδεση, οι οποίοι απαιτούν περισσότερους πόρους (όπως μεγαλύτερη επεξεργαστική ισχύ και μνήμη) από το σύστημα για να εκτελεστούν.

7.7. Σύνθεση βασισμένη σε χαρακτηριστικά ποιότητας υπηρεσιών (QoS-based composition)

Με τον όρο χαρακτηριστικά ποιότητας υπηρεσιών (QoS) περιγράφουμε μη-λειτουργικά χαρακτηριστικά και ιδιότητες μιας ΥΙ. Δεδομένου ότι ολοένα και περισσότερες υπηρεσίες είναι διαθέσιμες στο Διαδίκτυο, πολλές φορές ο τελικός χρήστης, ή η μηχανή σύνθεσης που αναλαμβάνει να διεκπεραιώσει τη σύνθεση, ενδεχομένως να πρέπει να επιλέξει ανάμεσα από πολλές ΥΙ που παρέχουν την ίδια λειτουργία αλλά παρ' όλα αυτά παρουσιάζουν διαφορετικές παραμέτρους QoS. Αυτές οι παράμετροι διαφοροποιούν σε μεγάλο βαθμό τις υπηρεσίες και είναι ζωτικής σημασίας για τη δημιουργία μιας σύνθετης ΥΙ που να ανταποκρίνεται στα κριτήρια που θεσπίζει ο χρήστης.

Αναλυτικά οι κυριότερες παράμετροι QoS είναι:

1. Διαθεσιμότητα: αναφέρεται στο χρονικό διάστημα κατά το οποίο η ΥΙ είναι διαθέσιμη για τις μηχανές σύνθεσης και για τον τελικό χρήστη.
2. Χρόνος απόκρισης: αντανακλά τον χρόνο που μεσολαβεί μεταξύ της κλήσης της ΥΙ και της στιγμής κατά την οποία αυτή ολοκληρώνει τη λειτουργία της. Αν ο χρόνος αυτός είναι πολύ υψηλός, η ποιότητα και η απόδοση της τελικής σύνθεσης επηρεάζεται σημαντικά.
3. Αξιοπιστία: Με τον όρο αυτό αναφερόμαστε στην ικανότητα της ΥΙ να ολοκληρώνει τη λειτουργία της, οποτεδήποτε καλείται και πάντα εντός της προκαθορισμένης χρονικής προθεσμίας.
4. Επεκτασιμότητα: Μια ΥΙ θα πρέπει να είναι σε θέση να ολοκληρώσει τη λειτουργία της στα προκαθορισμένα χρονικά πλαίσια, ακόμα και όταν ένα μεγάλο πλήθος χρηστών καλούν και επιχειρούν να «καταναλώσουν» την υπηρεσία την ίδια στιγμή.
5. Απαιτήσεις/Κόστος: Αντανακλά το χρηματικό αντίτιμο, δηλαδή το κόστος που θα έχει η κλήση και η «κατανάλωση» της ΥΙ στον τελικό χρήστη. Ωστόσο, σε κάποιες περιπτώσεις αυτό το χαρακτηριστικό ποιότητας μπορεί να χρησιμοποιηθεί για να περιγράψει το απαιτούμενο κόστος σε επεξεργαστική ισχύ, μνήμη αλλά και άλλους πόρους του συστήματος.
6. Σχόλια: Πολλά μητρώα ΥΙ ενσωματώνουν μηχανισμούς άμεσης ανατροφοδότησης από τους τελικούς χρήστες. Έτσι ανάλογα με τη βαθμολόγηση προηγούμενων χρηστών, ο τελικός χρήστης ή η μηχανή σύνθεσης μπορεί αποφασίσει το κατά πόσο επιθυμεί να επιλέξει μια ΥΙ για να τη συμπεριλάβει σε μια σύνθεση. Με την ανάπτυξη και την υιοθέτηση των τεχνολογιών που εισήγαγε το Web 2.0, όπως είναι οι τεχνολογίες κοινωνικής δικτύωσης, παρατηρείται μια αύξηση στη χρήση έμμεσων τεχνικών ανατροφοδότησης κατά τις οποίες γίνεται άντληση στοιχείων από κοινωνικά δίκτυα τα οποία μετά από επεξεργασία μπορούν να δείξουν τάσεις και προθέσεις χρηστών προς συγκεκριμένες εφαρμογές. Τέτοιες τεχνολογίες έχουν αρχίσει να συμπεριλαμβάνονται και στα πιο σύγχρονα μητρώα ΥΙ, γεγονός που βοηθά ακόμα περισσότερο τον τελικό χρήστη στη διαδικασία επιλογής ΥΙ.
7. Ασφάλεια: Αυτή η παράμετρος περιγράφει τις τεχνικές, τις μεθόδους αλλά και τα πρωτόκολλα ασφαλείας που χρησιμοποιούνται από την ΥΙ και τον πάροχό της. Παραδείγματα τεχνικών περιλαμβάνουν τους αλγόριθμους κρυπτογράφησης και ιδιωτικότητας που απαιτούνται, ενώ στον τομέα των πρωτοκόλλων περιλαμβάνονται τα πρωτόκολλα WS-Security για την περίπτωση των Soap-based ΥΙ ενώ για την περίπτωση των REST ΥΙ είναι απαραίτητη η χρήση SSL από την υποδομή του παρόχου της ΥΙ.

Σε μια σύνθεση υπηρεσιών θα πρέπει να εξασφαλίζεται πως παράμετροι ποιότητας μιας ΥΙ δε θα έρχονται σε αντίθεση με τις παραμέτρους ποιότητας των άλλων ΥΙ κατά την αλληλεπίδραση τους. Έτσι για παράδειγμα, η χρήση διαφορετικών πρωτοκόλλων και μηχανισμών ασφαλείας μπορεί να προκαλέσει ασυμβατότητα και αδυναμία επεξεργασίας της συνολικής σύνθεσης. Για τον λόγο αυτό είναι απαραίτητη η χρήση ευέλικτων φίλτρων από την εκάστοτε μηχανή αναζήτησης, τα οποία θα μπορούν να αναπροσαρμόζονται σε τακτά χρονικά διαστήματα, ανάλογα με το πλήθος των αποτελεσμάτων, και τις περιγραφές των επιστρεφόμενων ΥΙ.

7.8. Business-driven automated composition

Μία από τις μεγαλύτερες θεωρητικές προκλήσεις στις αρχιτεκτονικές τύπου SOA, όπως αυτή φαίνεται από τη βιβλιογραφία, είναι ο διαχωρισμός του «επιχειρησιακού επιπέδου» από το «επίπεδο συστήματος». Ως επίπεδο συστήματος ορίζονται τα χαρακτηριστικά των υπηρεσιών που δε σχετίζονται με τις επιχειρηματικές λειτουργίες της υπηρεσίας, όπως για παράδειγμα οι παράμετροι QoS που αναφέρθηκαν παραπάνω.

Ένας τέτοιος διαχωρισμός αυτών των δύο επιπέδων, θα μπορούσε να καταστήσει εφικτή μια πιο ευέλικτη προσέγγιση συνθέσεων ΥΙ που θα στηρίζονται στο ημι-αυτοματοποιημένο σύστημα συνθέσεων, που παρά ταύτα θα είχαν και τη δυνατότητα αναπροσαρμογής κατά την εκτέλεση. Η σύνθεση των υπηρεσιών στο επιχειρηματικό επίπεδο θα απαιτούσε σημαντικό βαθμό συμμετοχής από τον χρήστη, καθώς μέσα από τη χρήση εργαλείων επιλογής, σύνθεσης και παρακολούθησης θα μπορούσε να εξασφαλίσει πως πληρούνται οι επιχειρηματικές του ανάγκες, δηλαδή τα λειτουργικά χαρακτηριστικά των ΥΙ ανταποκρίνονται στις απαιτήσεις του. Στη συνέχεια, η μηχανή σύνθεσης θα μπορούσε με αυτοματοποιημένο τρόπο, και δίχως την

περαιτέρω συμμετοχή του χρήστη, να πραγματοποιήσει τη σύνθεση στο επίπεδο συστήματος εξασφαλίζοντας πως δε θα υπάρξει ασυμβατότητα ανάμεσα στα μη λειτουργικά χαρακτηριστικά των YI.

Σε ένα τέτοιο σύστημα, η μηχανή σύνθεσης θα ήταν σε θέση να αντικαθιστά υπηρεσίες όταν αυτές παύουν να πληρούν τις ανάγκες των επιχειρήσεων σε επίπεδο λειτουργικών χαρακτηριστικών, ενώ ταυτόχρονα θα αναπροσάρμοζε το επίπεδο συστήματος.

Σε κάθε περίπτωση η ανάπτυξη ενός τέτοιου μοντέλου σύνθεσης έχει πολλές προκλήσεις, αποτελεί όμως ένα ενδιαφέρον ερευνητικό πεδίο, αφού συνδυάζει τα σημαντικότερα πλεονεκτήματα των άλλων μοντέλων σύνθεσης.

8. BPEL και OWL-S

Πολλές γλώσσες και πρωτόκολλα έχουν προταθεί για τη διευκόλυνση της ενορχήστρωσης συνθέσεων YI. Ωστόσο, οι δημοφιλέστερες επιλογές είναι η BPEL και η OWL-S.

8.1. BPEL

Η Business Process Execution Language (BPEL) είναι μια γλώσσα ροής υψηλού επιπέδου, βασισμένη στην XML, που επιτρέπει την περιγραφή επιχειρηματικών διαδικασιών και παρέχει τα μέσα για την ενορχήστρωση και τη σύνθεσή τους. Η BPEL τυποποιήθηκε από τις εταιρίες IBM και Microsoft τον Απρίλιο του 2003. Ένα αρχείο γραμμένο σε BPEL είναι σε θέση να περιγράψει το πώς οι YI αλληλεπιδρούν μεταξύ τους, αλλά και με τις μηχανές σύνθεσης. Παράλληλα, μπορεί να ορίσει τη σειρά με την οποία θα γίνεται η κλήση και η «κατανάλωση» των υπηρεσιών. Έτσι μπορεί να ειπωθεί πως η γλώσσα BPEL εστιάζει στον τρόπο συντονισμού επιχειρηματικών διαδικασιών, μέσω διαγραμμάτων ροής, και όχι σε λεπτομέρειες σχετικές με τη λειτουργικότητα της κάθε YI που συμμετέχει σε μια ενορχήστρωση YI.

Το κυριότερο δομικό στοιχείο μιας διαδικασίας περιγραφόμενης σε γλώσσα BPEL είναι η δραστηριότητα (activity). Οι δραστηριότητες χωρίζονται σε δύο κύριες κατηγορίες:

- Βασικές (Basic) – Οι βασικές δραστηριότητες χρησιμοποιούνται για τον χειρισμό των δεδομένων κατά την εκτέλεση μιας ενορχήστρωσης αλλά και για τον χειρισμό των αλληλεπιδράσεων μεταξύ των YI. Μερικές βασικές δραστηριότητες είναι Compensate, Assign, Wait, Throw, ReThrow, και Exit
- Δομημένες (Structured) – Οι δομημένες δραστηριότητες συνήθως περιέχουν άλλες δραστηριότητες και χρησιμοποιούνται για την περιγραφή του συντονισμού των επιχειρηματικών διαδικασιών. Ενδεικτικές δομημένες δραστηριότητες είναι οι If, While, Pick, Flow, Sequence και Scope.

Video 9.1.mp4	Βίντεο (video)
Οι δραστηριότητες στην BPEL	

Κάθε διαδικασία BPEL αλληλεπιδρά με ένα πλήθος συνεργατών (partners). Με τον όρο αυτό εννοείται κάθε μέλος με το οποίο πραγματοποιούνται συναλλαγές και ανταλλαγές μηνυμάτων. Η αλληλεπίδραση με τους συνεργάτες πραγματοποιείται μέσω των λεγόμενων συνδέσμων συνεργατών (partner-Links). Στα partner links γίνεται ο καθορισμός των portTypes που προσφέρονται από μια YI αλλά και αυτών που απαιτούνται να υπάρχουν από τους συνεργάτες με τους οποίους αλληλεπιδρά.

Όπως φαίνεται στο σχήμα 9.2., η BPEL βρίσκεται στο υψηλότερο επίπεδο της στοίβας προδιαγραφών WS. Σε κάθε αρχείο BPEL γίνεται χρήση των περιγραφών WSDL για τον καθορισμό των αλληλεπιδράσεων των YI που συμμετέχουν σε μια σύνθεση.

8.2. OWL-S

Η γλώσσα OWL-S είναι μια γλώσσα σημασιολογικής σήμανσης που στοχεύει στην περιγραφή υπηρεσιών βάση σημασιολογικών χαρακτηριστικών (Farrag et al., 2013; Martin et al., 2005). Παρέχει τον τρόπο στους παρόχους να περιγράψουν σημασιολογικά τις YI που παρέχουν με τη χρήση οντολογιών. Με τον τρόπο αυτό,

η ανακάλυψη και η σύνθεση διευκολύνεται καθώς οι μηχανές σύνθεσης μπορούν να ερμηνεύσουν αυτόματα τις λειτουργίες, τις ιδιότητες, τις προϋποθέσεις και τις δυνατότητες μιας ΥΙ υπό αυστηρούς περιορισμούς.

Πιο συγκεκριμένα, η μηχανή σύνθεσης, μέσα από την επεξεργασία των οντολογιών, μπορεί να έχει πρόσβαση σε πληροφορίες σχετικές με το τι κάνει η υπηρεσία, ποιες είναι οι παράμετροι εισόδου που απαιτούνται, ποια είναι τα αποτελέσματα που θα πρέπει να αναμένονται καθώς και το τι είδους μηνύματα και δεδομένα ανταλλάσσονται κατά την εκτέλεση κάθε ΥΙ. Το σημαντικό είναι πως αυτές οι πληροφορίες μπορούν να γίνουν κατανοητές και να αξιοποιηθούν από τη μηχανή σύνθεσης χωρίς να υπάρχει η ανάγκη για ανθρώπινη παρέμβαση. Πολλές επεκτάσεις έχουν προταθεί για την OWL-S. Μια από τις πιο αξιοσημείωτες είναι η OWL-Q, η οποία βασίζεται σε αρχές σχεδιασμού αλλά και σε απαιτήσεις που προκύπτουν από χαρακτηριστικά ποιότητας ΥΙ.

9. Υπηρεσίες Ιστού και το Διαδίκτυο των Αντικειμένων

Τα τελευταία χρόνια, η έννοια του Web of Things (WoT), ως μια συλλογή από καινοτόμες ιδέες και τεχνολογίες, έχει προσελκύσει την προσοχή των ερευνητικών και επιχειρηματικών κοινοτήτων, αφού φαίνεται πως θα αποτελέσει για τις επιχειρήσεις ένα νέο τρόπο για να παρέχουν υπηρεσίες προστιθέμενης αξίας για την εκπλήρωση των αναγκών των πελατών τους (Vesyropoulos & Georgiadis, 2013).

Sound 9.4.mp3	Ηχητικό απόσπασμα (audio)
Περιγραφή του «Διαδικτύου των Αντικειμένων»	

Με τη δημιουργία εικονικών αναπαραστάσεων για αντικείμενα του πραγματικού κόσμου, δίνεται η δυνατότητα πρόσβασης στη λειτουργικότητα τους μέσω υπηρεσιών τύπου REST. Με τον τρόπο αυτό υπηρεσίες που θα προέρχονται από τα «έξυπνα» πλέον αυτά αντικείμενα, οι αποκαλούμενες «φυσικές Υπηρεσίες Ιστού» (π.χ. που προέρχονται από έξυπνα ψυγεία ή συστήματα θέρμανσης) προστίθενται στην ήδη υπάρχουσα αφθονία ΥΙ που υπάρχουν στο Διαδίκτυο. Όπως αυτό είναι φυσικό, κάτι τέτοιο μπορεί να περιπλέξει την επιλογή και τη σύνθεση ΥΙ για τους τελικούς χρήστες.

Ως εκ τούτου, προκύπτει η ανάγκη για νέες τεχνικές επιλογής ΥΙ που θα αφορούν τόσο τις φυσικές ΥΙ (physical WS), όσο και τις παραδοσιακές ΥΙ (virtual WS), και που θα είναι σε θέση να παράγουν συνθέσεις ή συγκλίσεις ΥΙ, με βάση τις εξατομικευμένες ανάγκες σε χαρακτηριστικά ποιότητας. Τέτοιες προσεγγίσεις έχουν ήδη παρουσιαστεί στην ερευνητική βιβλιογραφία.

Το WoT, ως επέκταση του «Διαδικτύου των Αντικειμένων» (IoT) (Da Xu et al., 2014) έχει κερδίσει πρόσφατα μεγάλη προσοχή, καθώς υπόσχεται μια πληθώρα από οφέλη που επέφερε το IoT, σε συνδυασμό με την ευκολία χρήσης των γνωστών προτύπων του Παγκοσμίου Ιστού. Τα πρότυπα αυτά χρησιμοποιούνται προκειμένου να καταστεί δυνατή η επικοινωνία και η διαλειτουργικότητα μεταξύ των ΥΙ και των αντικειμένων του πραγματικού κόσμου (Guinard et al., 2011).

Για να επιτευχθεί μια σύνδεση με ένα τέτοιο «έξυπνο» αντικείμενο, το μόνο που χρειάζεται είναι μια απλή διεύθυνση URL που θα παρέχει πρόσβαση στην εικονική αναπαράσταση του αντικειμένου. Αυτή η αναπαράσταση μπορεί να έχει τη μορφή μιας ιστοσελίδας που παρέχει πρόσβαση σε υπηρεσίες που αντιστοιχούν σε λειτουργίες του αντικειμένου. Εκμεταλλευόμενοι τις καθιερωμένες αρχιτεκτονικές και τα πρωτόκολλα του Διαδικτύου, είναι δυνατή η δημιουργία ενός δικτύου εικονικών αναπαραστάσεων αντικειμένων, το οποίο θα στηρίζεται στην ενσωμάτωση αισθητήρων αλλά και των λεγόμενων «έξυπνων πυλών» που υποβοηθούν την επικοινωνία αντικειμένων που βασίζονται σε ετερογενή πρωτόκολλα επικοινωνίας.

Η έννοια του WoT προτείνει τη χρήση του γνωστού πρωτοκόλλου HTTP ως πρωτόκολλο εφαρμογής (application protocol), που καθιστά ιδανικές ΥΙ για χρήση, της υπηρεσίες τύπου REST. Οι υπηρεσίες αυτές μπορούν να παρέχουν πρόσβαση στη λειτουργικότητα της εικονικής αναπαράστασης ενός "έξυπνου" φυσικού αντικειμένου χρησιμοποιώντας URIs και ένα σύνολο μεθόδων HTTP (GET, POST, PUT, DELETE). Έτσι, είναι δυνατή η ανταλλαγή δεδομένων μεταξύ των "έξυπνων αντικειμένων», συνήθως σε JSON, Atom ή XML. Αυτό επιτρέπει τον χειρισμό αυτών των εικονικών αναπαραστάσεων από Web client που είναι συμβατοί με το πρωτόκολλο HTTP, όπως για παράδειγμα οι φυλλομετρητές που χρησιμοποιούνται για την περιήγηση στο Web.

10. Σύνθεση και σύγκλιση υπηρεσιών REST στα πλαίσια του Διαδικτύου των Αντικειμένων

Σε αντίθεση με τις υπηρεσίες τύπου SOAP, οι υπηρεσίες τύπου REST συντίθεται συνήθως με τη μορφή των Web 2.0 mashups. Με τον όρο Web mashup, αναφερόμαστε σε μια Web εφαρμογή ή ιστοσελίδα που συνήθως χρησιμοποιεί application programming interfaces (APIs) για να ενώσει πληροφορίες από πολλαπλές πηγές για τη δημιουργία υπηρεσιών προστιθέμενης αξίας. Ενώ υπάρχουν και εναλλακτικές μέθοδοι για τη σύνθεση υπηρεσιών REST, η χρήση των mashups που οδηγεί στην αποκαλούμενη "σύγκληση υπηρεσιών" παραμένει η πιο δημοφιλής μέθοδος.

Καθώς ολοένα και περισσότερες συσκευές με ενσωματωμένους αισθητήρες θα είναι σε θέση να παρέχουν, μέσω YI, πρόσβαση στη λειτουργικότητα τους και ένα μεγάλο πλήθος πραγματικών αντικειμένων θα έχει τη δυνατότητα της επικοινωνίας και διαλειτουργικότητας μέσω TCP / IP δικτύων, η ανάγκη για τη δημιουργία υπηρεσιών προστιθέμενης αξίας ως αποτέλεσμα σύνθεσης «φυσικών υπηρεσιών», αλλά και virtual YI, αυξάνεται εκθετικά.

10.1. Physical-virtual mashups

Οι συγκλίσεις τύπου Physical-Virtual Mashups αφορούν τη σύνθεση YI που παρέχονται όχι μόνο από τις παραδοσιακές Web-based υπηρεσίες (εικονικές YI), αλλά κι από τις υπηρεσίες που παρέχονται από φορητές συσκευές αλλά και από τα φυσικά αντικείμενα του πραγματικού κόσμου. Οι YI «φυσικού τύπου» επιτρέπουν σε επιχειρήσεις, τελικούς χρήστες αλλά και σε άλλες έξυπνες συσκευές να αλληλεπιδρούν με συσκευές που παρέχουν τέτοιες YI, με την αποστολή αιτήσεων HTTP.

Ένα παράδειγμα σύνθεσης «φυσικών» και «εικονικών» υπηρεσιών με τη μορφή σύγκλισης αφορά την προβολή πληροφοριών σχετικά με την κατανάλωση ενέργειας από διάφορες συσκευές μιας επιχείρησης σε συνδυασμό με τη χρήση ενός εξωτερικού (ή μελλοντικά εσωτερικού) χάρτη. Έτσι οι «φυσικές» YI, δηλαδή οι πληροφορίες που προκύπτουν από τους αισθητήρες μέτρησης κατανάλωσης ενέργειας που θα βρίσκονται ενσωματωμένοι στις συσκευές αυτές θα απεικονίζονται σε έναν εικονικό χάρτη (μέσω μιας υπηρεσίας χάρτη όπως π.χ. μέσω της χρήσης του API των χαρτών της Google) με τέτοιο τρόπο έτσι ώστε να φανερώνεται η γεωγραφική τους τοποθεσία. Με τον τρόπο αυτό είναι ευκολότερη η συντήρηση των συσκευών και ο γρήγορος εντοπισμός προβλημάτων.

10.2. Physical-physical mashups

Οι συγκλίσεις τύπου Physical-Physical Mashups, προκύπτουν από τη σύνθεση YI που δίνουν πρόσβαση στις λειτουργίες συσκευών με ενσωματωμένους αισθητήρες. Τέτοιες συγκλίσεις προβάλουν ή δίνουν πρόσβαση σε στοιχεία που παρέχονται από αντικείμενα του πραγματικού κόσμου. Καθώς τέτοιες συσκευές μπορούν να αλληλεπιδρούν με τη χρήση του HTTP πρωτοκόλλου, είναι δυνατή η λεγόμενη μηχανή-προς-μηχανή αλληλεπίδραση (machine-to-machine interaction), στην οποία δεν είναι υποχρεωτική η ανθρώπινη παρέμβαση. Πολλές φορές λοιπόν τα Physical-Physical Mashups, αποτελούν και ένα μέσο παρακολούθησης της διαλειτουργικοτητας μεταξύ των έξυπνων συσκευών, πέρα από τη χρήση τους σαν μέσα σύνθεσης των «φυσικών» YI των συσκευών αυτών.

Ένα παράδειγμα τέτοιας εφαρμογής θα περιελάμβανε τη συνεργασία μεταξύ των αισθητήρων της γραμμής παραγωγής μιας επιχείρησης. Σε μια τέτοια σύγκλιση YI, η αύξηση της θερμοκρασίας σε ένα συγκεκριμένο μηχάνημα παραγωγής θα μπορούσε να οδηγήσει σε μια ανταλλαγή μηνυμάτων μεταξύ του μηχανήματος και των συστημάτων διαχείρισης εξαερισμού και κλιματισμού. Σε κάθε περίπτωση μέσω της σύγκλισης θα ήταν δυνατή η παρακολούθηση της διαδικασίας ανταλλαγής μηνυμάτων από τον αρμόδιο υπάλληλο ο οποίος θα είχε πρόσβαση στη σύγκλιση αυτή.

Sound 9.5.mp3	Ηχητικό απόσπασμα (audio)
Περιγραφή των Physical-Physical Mashups	

10.3. Business intelligence mashups

Οι συγκλίσεις τύπου Business Intelligence αφορούν κυρίως επιχειρήσεις. Αποτελούν Web εφαρμογές, οι οποίες ενσωματώνουν αποτελεσματικά ένα πλήθος από τοπικές εφαρμογές της επιχείρησης οι οποίες ικανοποιούν τις διάφορες επιχειρηματικές ανάγκες της. Επιπρόσθετα, οι συγκλίσεις αυτού του τύπου μπορεί να περιλαμβάνουν και εξωτερικές ΥΙ, με σκοπό τη δημιουργία υπηρεσιών προστιθέμενης αξίας. Λόγω της ανάγκης ενσωμάτωσης επιχειρηματικής λογικής, στα Business Intelligence Mashups συχνά γίνεται χρήση σημασιολογικών κανόνων, ενώ είναι απαραίτητη και η ενορχήστρωση μέσω BPEL. Είναι συχνά απαραίτητη η χρήση ΥΙ τύπου SOAP, και θα πρέπει να υπάρχει δυνατότητα αναπροσαρμογής της σύγκλισης, με την τροποποίηση ΥΙ κατά την εκτέλεση, έτσι ώστε να μπορεί αυτή να ανταποκριθεί στις ταχύτατες αλλαγές του επιχειρηματικού περιβάλλοντος.

11. Συμπεράσματα

Καθώς το επιχειρείν βασίζεται σε ολοένα και πιο μεγάλο βαθμό στις συναλλαγές μέσω Διαδικτύου (η-επιχειρείν), είναι απαραίτητη η αντιμετώπιση των προβλημάτων που εγείρονται από την ανάγκη διαλειτουργικότητας ανάμεσα σε συστήματα, τα οποία έχουν αναπτυχθεί σε διαφορετικές πλατφόρμες. Οι Υπηρεσίες Ιστού αποτελούν την τεχνολογία που επιτρέπει μέσω χαλαρής ζεύξης την επικοινωνία αυτή μεταξύ ετερογενών συστημάτων, και για τον λόγο αυτό έχουν υιοθετηθεί σε μεγάλο βαθμό από πληθώρα επιχειρήσεων και οργανισμών.

Ένα μεγάλο πλεονέκτημα των ΥΙ είναι πως επιτρέπουν την ολοκλήρωση μεμονωμένων ΥΙ, από διαφορετικούς παρόχους, έτσι ώστε τελικά να παρέχουν τη βέλτιστη εμπειρία στους χρήστες, μέσω υπηρεσιών προστιθέμενης αξίας. Επιπλέον, διάφορες ποιοτικές παράμετροι (όπως ασφάλεια, απόδοση κ.ά.), αποτελούν βασικά κριτήρια στην αξιολόγηση των ΥΙ και της δυνατότητας που έχουν στο να συμμετέχουν σε ευρύτερες συνθέσεις.

Τόσο σαν μεμονωμένες υπηρεσίες όσο και σαν μέλη μιας ευρύτερης σύνθεσης, οι ΥΙ μπορούν να προσφέρουν σημαντικά οφέλη στον τρόπο με τον οποίο οι επιχειρήσεις επικοινωνούν με τους καταναλωτές και τους συνεργάτες τους, αλλά και να βελτιώσουν τον σχεδιασμό και την προβολή των επιχειρηματικών λειτουργιών τους.

Στο κεφάλαιο αυτό έγινε αναφορά στις υπάρχουσες κατηγορίες ΥΙ, στις τεχνικές επιλογής και σύνθεσης αλλά και στους ειδικότερους τομείς της επιστήμης της πληροφορικής, στους οποίους έχουν εφαρμογή οι ΥΙ, όπως είναι το Διαδίκτυο των Αντικειμένων.

Βιβλιογραφία/Αναφορές

Alrifai, M., Skoutas, D., & Risse, T. (2010). Selecting skyline services for QoS-based web service composition. In Proceedings of the 19th international conference on World Wide Web (pp. 11-20). ACM.

Da Xu, L., He, W., & Li, S. (2014). Internet of Things in industries: A survey. *Industrial Informatics*, IEEE Transactions on, 10(4), 2233-2243.

Martin, D., Paolucci, M., McIlraith, S., Burstein, M., McDermott, D., McGuinness, D., ... & Sycara, K. (2005). Bringing semantics to web services: The OWL-S approach. In *Semantic Web Services and Web Process Composition,* Springer Berlin Heidelberg, pp. 26-42.

Farrag, T. A., Saleh, A. I., & Ali, H. A. (2013). Toward SWSs discovery: mapping from wsdl to owl-s based on ontology search and standardization engine. *Knowledge and Data Engineering*, IEEE Transactions on, 25(5), 1135-1147.

Fielding, R. (2000). *Architectural Styles and The Design of Network-based Software Architectures*. PhD Thesis, University of California, Irvine

Guinard, D., Trifa, V., Mattern, F., & Wilde, E. (2011). From the internet of things to the web of things: Resource-oriented architecture and best practices. In *Architecting the Internet of Things,* pp. 97-129. Springer Berlin Heidelberg.

Hatzi, O., Vrakas, D., Nikolaidou, M., Bassiliades, N., Anagnostopoulos, D., & Vlahavas, L. (2012). An integrated approach to automated semantic web service composition through planning. *Services Computing*, IEEE Transactions on, 5(3), 319-332.

Papazoglou, M. P., Traverso, P., Dustdar, S., & Leymann, F. (2008). Service-oriented computing: a research roadmap. *International Journal of Cooperative Information Systems*, 17(02), 223-255

Pautasso, C. (2014). RESTful web services: principles, patterns, emerging technologies. In *Web Services Foundations,* Springer New York, pp. 31-51.

Pautasso, C., Wilde, E., & Alarcon, R. (2014). *REST: Advanced Research Topics and Practical Applications*. Springer.

Pimenidis, E., & Georgiadis, C. K. (2010). Web services for rural areas—Security challenges in development and use. *Computers and electronics in agriculture*, 70(2), 348-354.

Sheng, Q. Z., Qiao, X., Vasilakos, A. V., Szabo, C., Bourne, S., & Xu, X. (2014). Web services composition: A decade's overview. *Information Sciences*, 280, 218-238.

Vesyropoulos, N., & Georgiadis, C. K. (2013). Web of things: understanding the growing opportunities for business transactions. In Proceedings of the 6th Balkan Conference in Informatics, ACM, pp. 267-274.

Wang, P., Ding, Z., Jiang, C., & Zhou, M. (2014). Automated web service composition supporting conditional branch structures. *Enterprise Information Systems*, 8(1), 121-146.

Weerawarana, S., Curbera, F., Leymann, F., Storey, T., & Ferguson, D. F. (2008). *Αρχιτεκτονική Πλατφόρμας Υπηρεσιών Ιστού*, Κλειδάριθμος, Επιστ. επιμ. Ελλ. εκδ. Χ. Κ. Γεωργιάδης, αρχική έκδοση: *Web services platform architecture*. Prentice Hall PTR 2005.

Quiz9.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Η χαλαρή σύζευξη επιτρέπει τη διαλειτουργικότητα ανάμεσα σε ετερογενή συστήματα λογισμικού.

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Α) Σωστό

Κριτήριο αξιολόγησης 2

[*] Ένα μήνυμα SOAP αποτελείται μόνο από τον περιβάλλοντα φάκελο (envelope) και την κεφαλίδα (header)

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 3

[*] Η γλώσσα Περιγραφής Υπηρεσιών Ιστού (WSDL) χρησιμοποιείται για:

Α) Την ενορχήστρωση ΥΙ

Β) Την περιγραφή διεπαφών ΥΙ

Γ) Τη χορογραφία ΥΙ

Δ) Τον καθορισμό επιχειρηματικών κανόνων

Απάντηση/Λύση

Β) Την περιγραφή διεπαφών ΥΙ

Κριτήριο αξιολόγησης 4

[*] Με τον όρο χαρακτηριστικά ποιότητας υπηρεσιών (QoS) περιγράφουμε:

Α) Μη-λειτουργικά χαρακτηριστικά και ιδιότητες μιας ΥΙ

Β) Λειτουργικά χαρακτηριστικά και ιδιότητες μιας ΥΙ

Απάντηση/Λύση
Α) Μη-λειτουργικά χαρακτηριστικά και ιδιότητες μιας ΥΙ

Κριτήριο αξιολόγησης 5
[*] Η γλώσσα BPEL είναι μια γλώσσα σημασιολογικής σήμανσης που στοχεύει στην περιγραφή υπηρεσιών βάσει σημασιολογικών χαρακτηριστικών

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 6
[*] Οι τεχνολογίες στις οποίες στηρίζεται το Διαδίκτυο των Αντικειμένων (Internet of Things) καθιστούν ιδανικές για χρήση τις ΥΙ τύπου WS-*
Α) Σωστό

Β) Λάθος

Απάντηση/Λύση
Β) Λάθος

Κριτήριο αξιολόγησης 7
[**] Ποιος από τους παρακάτω δεν είναι περιορισμός που επιβάλλει η REST αρχιτεκτονική;

Α) Client-server

Β) Uniform interface

Γ) Security

Δ) Stateless

Απάντηση/Λύση
Γ) Security

Κριτήριο αξιολόγησης 8
[**] Στα πλαίσια των ιδιοτήτων ACID που παρέχονται από τις συναλλαγές ΥΙ, η ατομικότητα αναφέρεται στο εξής:

Α) Τα αποτελέσματα της εφαρμογής χαρακτηρίζονται από συνέπεια και η εφαρμογή επιτελεί ορθές μεταβάσεις καταστάσεων κατά την ολοκλήρωση της.

Β) Οι ενδιάμεσες καταστάσεις είναι διαθέσιμες και μετά την ολοκλήρωση της συναλλαγής.

Γ) Σε περίπτωση επιτυχίας της συναλλαγής, όλες οι ενέργειες της εφαρμογής εκτελούνται, ενώ σε αντίθετη περίπτωση δεν εκτελείται καμία ενέργεια.

Δ) Αφότου έχει ολοκληρωθεί μια συναλλαγή, οι αλλαγές που έχει προκαλέσει διατηρούνται ακόμα και αν υπάρξει σφάλμα σε επόμενες ενέργειες.

Απάντηση/Λύση
Γ) Σε περίπτωση επιτυχίας της συναλλαγής, όλες οι ενέργειες της εφαρμογής εκτελούνται, ενώ σε αντίθετη περίπτωση δεν εκτελείται καμία ενέργεια.

Κριτήριο αξιολόγησης 9
[**] Ποιο από τα ακόλουθα δεν αποτελεί πρωτόκολλο του προτύπου Επιχειρηματικής-Δραστηριότητας ΥΙ;

Α) Επιχειρηματική Συμφωνία με Ολοκλήρωση από το Μητρώο ΥΙ

Β) Επιχειρηματική Συμφωνία με Ολοκλήρωση από τους Συμμετέχοντες

Γ) Επιχειρηματική Συμφωνία με Ολοκλήρωση από τον Συντονιστή

Απάντηση/Λύση
Α) Επιχειρηματική Συμφωνία με Ολοκλήρωση από το Μητρώο ΥΙ

Κριτήριο αξιολόγησης 10
[**] Ποια από τα παρακάτω δεν αποτελεί επίπεδο της αρχιτεκτονικής SOA:

Α) Ανταλλαγή Μηνυμάτων

Β) Περιγραφή ΥΙ

Γ) Ασφάλεια ΥΙ

Δ) Ποιότητα ΥΙ

Ε) Συστατικά Στοιχεία

Απάντηση/Λύση
Γ) Ασφάλεια ΥΙ

Κεφάλαιο 10: Επιλογή και Σύνθεση Υπηρεσιών Ιστού για Επιχειρηματικές Διαδικασίες: Εργαστηριακές Ασκήσεις

Σύνοψη
Στο κεφάλαιο αυτό παρουσιάζεται ένα σύνολο εργαστηριακών ασκήσεων (εκφωνήσεις και οι υποδειγματικές λύσεις αυτών), με σκοπό την κατανόηση τεχνικών ανάπτυξης, επιλογής και σύνθεσης Υπηρεσιών Ιστού (ΥΙ). Επιπρόσθετα, παρουσιάζεται αναλυτικά μια προσέγγιση επιλογής ΥΙ, η οποία αξιοποιεί μια δημοφιλή μέθοδο πολυκριτήριας ανάλυσης αποφάσεων, την AHP. Οι επιμέρους τεχνολογίες που χρησιμοποιούνται είναι: Το περιβάλλον ανάπτυξης Eclipse IDE for JavaEE developers (έκδοση Luna), το framework ανάπτυξης REST εφαρμογών RESTlet και το πακέτο OpenESB 2.3.1, το οποίο συμπεριλαμβάνει το περιβάλλον ανάπτυξης NetBeans (έκδοση 7) μαζί με τον Glassfish Server και υποστήριξη για σύνθεση ΥΙ μέσω BPEL. Στα παραδείγματα ανάπτυξης ΥΙ χρησιμοποιείται ως βασική γλώσσα προγραμματισμού η Java, ενώ στο πακέτο OpenESB γίνεται χρήση της BPEL και μελέτη αρχείων WDSL και XML schema, τα οποία προϋποθέτουν μια βασική κατανόηση της XML.

Προαπαιτούμενη γνώση
Το κεφάλαιο 9 του παρόντος συγγράμματος, και επιπλέον θα είναι χρήσιμη κάποια προηγούμενη εμπειρία προγραμματισμού σε Java.

1. Εισαγωγή

Αξιοποιώντας τα ενορατικά/οπτικά (visual) εργαλεία και τη γενικότερη διάδραση και ευχρηστία που παρέχουν τα περιβάλλοντα Eclipse αλλά και NetBeans (μέσω του πακέτου OpenESB), ακολουθούμε και σε αυτό το κεφάλαιο τη μέθοδο εκπαίδευσης από παράδειγμα (example-based learning) για να παρουσιάσουμε σημαντικές έννοιες και τεχνικές σχετικές με την ανάπτυξη, την επιλογή και τη σύνθεση Υπηρεσιών Ιστού (ΥΙ).

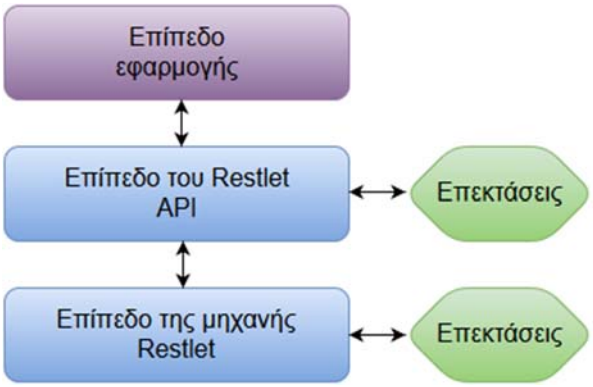
Συγκεκριμένα, θα παρουσιαστεί μια μεθοδολογία ανάπτυξης εφαρμογών REST, μέσω του RESTlet framework, μια μέθοδος επιλογής ΥΙ που βασίζεται στην AHP καθώς και η μέθοδος σύνθεσης ΥΙ μέσω του γραφικού περιβάλλοντος που παρέχει το OpenESB και με τη χρήση της γλώσσας BPEL. Η γνώση αυτών των εργαλείων και τεχνικών είναι πολύ σημαντική καθώς, όπως έγινε φανερό και στο προηγούμενο κεφάλαιο, η αξιοποίηση και η χρήση ΥΙ αλλά και συνθέσεων αυτών, καθιστά δυνατή την υποστήριξη τόσο απλών όσο και πιο σύνθετων συναλλαγών Ηλεκτρονικού Εμπορίου.

2. Ανάπτυξη Υπηρεσιών Ιστού REST

2.1. Εισαγωγή στο RESTlet framework

Στην πρώτη ενότητα αυτού του κεφαλαίου θα ασχοληθούμε με τη χρήση ενός framework για την ανάπτυξη RESTful εφαρμογών Ιστού. Το Restlet Framework (http://restlet.com) είναι ένα ανοιχτού κώδικα framework, για την ανάπτυξη API (application programming interface) σε Java (Louvel et al., 2012). Με τη χρήση του συγκεκριμένου framework, είναι δυνατή η ανάπτυξη εφαρμογών τόσο από την πλευρά του πελάτη, όσο και από την πλευρά του εξυπηρετητή. Παράλληλα, το συγκεκριμένο framework, επιτρέπει την ανάπτυξη εφαρμογών που θα μπορούν να αξιοποιήσουν μια πληθώρα πρωτοκόλλων μετάδοσης, όπως είναι τα HTTP και SMTP αλλά και προτύπων σύνδεσης βάσεων δεδομένων (με κυριότερο το JDBC client).

Το framework αποτελείται από δύο συστατικά μέρη, το Restlet API και το Restlet Engine. Το πρώτο τμήμα προσφέρει την υποστήριξη των εντολών χειρισμού REST κλήσεων ενώ παράλληλα διαχειρίζεται τις κλήσεις για τις client-side και τις server-side εφαρμογές. Το Restlet Engine είναι υπεύθυνο για την εκτέλεση των REST μεθόδων και υποστηρίζει το Restlet API. Συνολικά απαρτίζουν το framework και ενσωματώνονται σε ένα JAR, το “org.restlet.jar”.



Σχήμα 10.1 *Αρχιτεκτονική του RESTlet framework*

2.2. Ανάπτυξη ΥΙ διαχείρισης λογαριασμών χρηστών

Εκφώνηση:

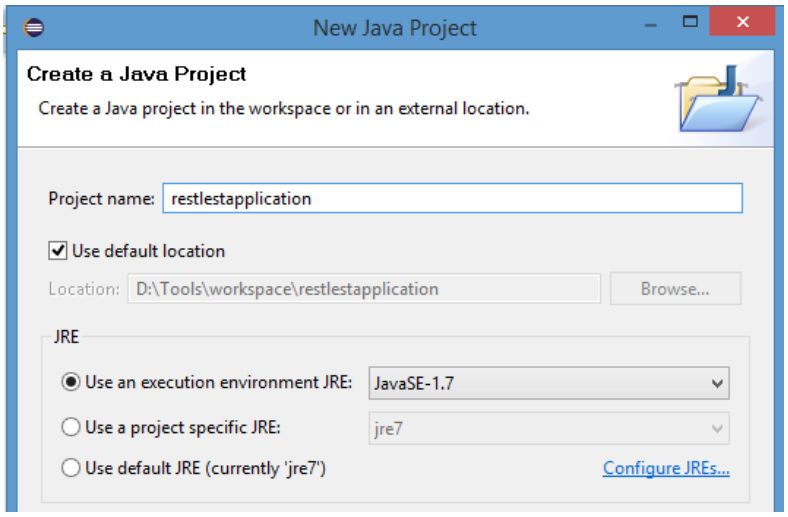
Αναπτύξτε ΥΙ που να επιτρέπει τον χειρισμό αναπαραστάσεων πόρων. Συγκεκριμένα, θα επιτρέπει τη λήψη, την τροποποίηση και τη διαγραφή στοιχείων μιας λίστας, μέσω HTTP requests.

Για την ανάπτυξη της ΥΙ προτείνεται η χρήση του περιβάλλοντος Eclipse IDE (έκδοση Luna) και του RESTlet framework («http://restlet.com/downloads/current/»).

Σημείωση: Καθώς από έναν απλό φυλλομετρητή είναι δύσκολο να γίνουν HTTP κλήσεις πέραν της GET, είναι απαραίτητη η χρήση κάποιου browser plugin, έτσι ώστε να είστε σε θέση να επιτελείτε όλα τα επιθυμητά HTTP requests (GET, POST, PUT, DELETE). Το προτεινόμενο plugin για τον Mozilla Firefox είναι το HttpRequester, το οποίο είναι διαθέσιμο στη διεύθυνση https://addons.mozilla.org/el/firefox/addon/httprequester/?src=api και το οποίο εγκαθίσταται σαν toolbar button στον Firefox. Πατώντας το εικονίδιο που δημιουργείται μετά την εγκατάσταση του, είναι δυνατή η κλήση HTTP requests, αλλά και η διατήρηση ιστορικού αυτών.

Υποδειγματική λύση:

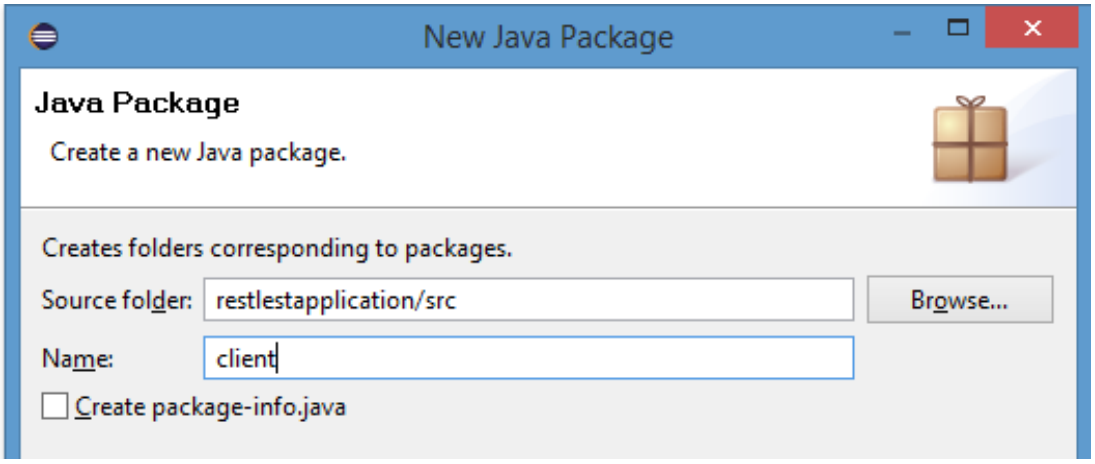
- Δημιουργία Java Project. Επιλογή File –> New –> Java Project.



Εικόνα 10.1 *Δημιουργία νέου Java project*

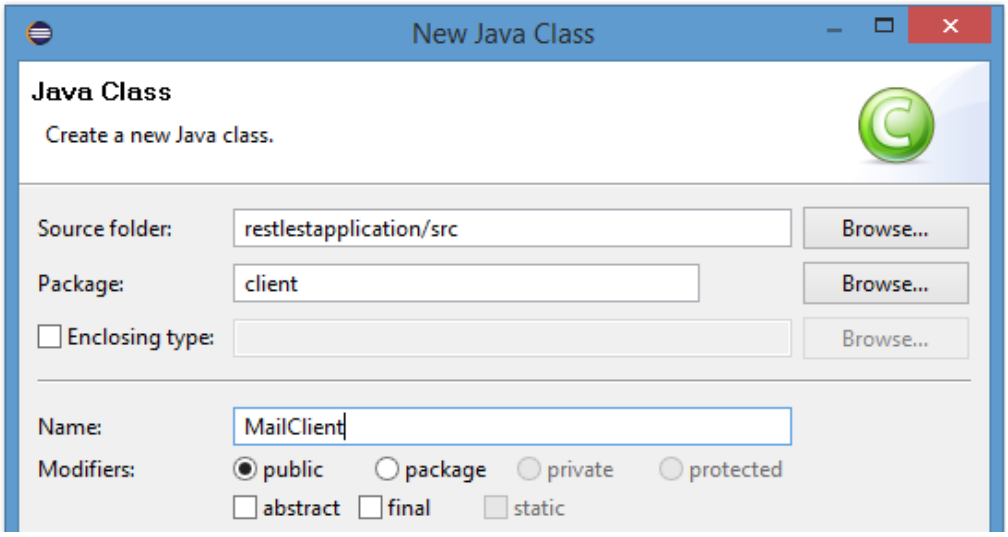
- Ονομασία του project: restlestapplication. Ως execution environment, επιλέγουμε το JavaSE-1.7.

- Κάνουμε δεξί κλικ στο project μας, επιλέγουμε New->Package και προχωρούμε σε δημιουργία package με την ονομασία client.



Εικόνα 10.2 *Δημιουργία νέου package*

- Επαναλαμβάνουμε τη διαδικασία για 2 ακόμα packages, τα common και server.
- Κάνουμε δεξί κλικ στο package client και επιλέγουμε New-> class. Προχωράμε έτσι στη δημιουργία της κλάσης MailClient.



Εικόνα 10.3 *Δημιουργία νέας Java class*

MailClient.java

```
package client;
import org.restlet.Client;
import org.restlet.Context;
import org.restlet.data.Protocol;
import common.AccountResource;
import common.AccountsResource;
import common.RootResource;
import org.restlet.resource.ClientResource;
public class MailClient {
        public static void main(String[] args) throws Exception {
                Client client = new Client(new Context(), Protocol.HTTP);
                ClientResource service = new ClientResource("http://localhost:8111");
```

```
        service.setNext(client);
        RootResource mailRoot = service.getChild("/", RootResource.class);
        System.out.println(mailRoot.represent());
        System.out.println("\n1) Λίστα Ενεργών Χρηστών\n");
        AccountsResource mailAccounts = service.getChild("/accounts/",
        AccountsResource.class);
        String list = mailAccounts.represent();
        System.out.println(list == null ? "<Άδεια λίστα>\n" : list);
        System.out.println("2) Πρόσθεση νέων χρηστών\n");
        mailAccounts.add("Χρήστος Γεωργιάδης");
        mailAccounts.add("Νικόλαος Παπαδόπουλος");
        mailAccounts.add("Αναστασία Παπαδοπούλου");
        mailAccounts.add("Μαρία Γεωργίου");
        mailAccounts.add("Κωνσταντίνος Κωνσταντίνου");
        mailAccounts.add("Γιώργος Δημητρίου");
        mailAccounts.add("Παναγιώτης Παναγιώτου");
        mailAccounts.add("Αλίκη Παναγιώτου");
        System.out.println("Οκτώ νέοι χρήστες προστέθηκαν!");
        System.out.println("\n3) Ενημερωμένη λίστα χρηστών\n");
        System.out.println(mailAccounts.represent());
        System.out.println("4) Εμφάνιση 5ου χρήστη\n");
        AccountResource mailAccount = service.getChild("/accounts/4",
        AccountResource.class);
        System.out.println(mailAccount.represent());
        System.out.println("\n5) Αλλαγή στοιχείων χρήστη\n");
        mailAccount.store("Κωνσταντίνα Κωνσταντίνου");
        System.out.println(mailAccount.represent());
        System.out.println("\n6) Διαγραφή του 8ου χρήστη και ανανέωση προβολής της λίστας\n");
        mailAccount = service.getChild("/accounts/7", AccountResource.class);
        mailAccount.remove();
        System.out.println(mailAccounts.represent());
        }
}
```

Sound 10.1.mp3	Ηχητικό απόσπασμα (audio)
Επεξήγηση της MailClient κλάσης	

Η κλάση MailClient.java, η οποία μπορεί να εκτελεστεί αυτόνομα ως Java application, επιτελεί αυτόματα ενέργειες δημιουργίας – αναβάθμισης – διαγραφής λογαριασμών.

- Δημιουργία με την ίδια μέθοδο των κλάσεων AccountResource, AccountsResource και RootResource στο package common.

AccountResource.java

```
package common;
import org.restlet.resource.Delete;
import org.restlet.resource.Get;
import org.restlet.resource.Put;
public interface AccountResource {
        @Get("txt")
        public String represent();
        @Put ("txt")
        public void store(String account);
        @Delete
        public void remove();
```

```
	}
```

AccountsResource.java

```
package common;
import org.restlet.resource.Get;
import org.restlet.resource.Post;
public interface AccountsResource {
	@Get("txt")
	public String represent();
	@Post("txt")
	public String add(String account );
	}
```

RootResource.java

```
package common;
import org.restlet.resource.Get;
public interface RootResource {
	@Get("txt")
	public String represent();
	}
```

- Δημιουργία των κλάσεων AccountServerResource, AccountsServerResource, RootServerResource στο package server.
- Δημιουργία της κλάσης MailServerApplication στο package server. Η συγκεκριμένη κλάση είναι η εκτελέσιμη κλάση της εφαρμογής.

AccountServerResource.java

```
package server;
import common.AccountResource;
import org.restlet.resource.ResourceException;
import org.restlet.resource.ServerResource;
public class AccountServerResource extends ServerResource implements
AccountResource {
	private int accountId;
	@Override
	protected void doInit() throws ResourceException {
		this.accountId = Integer.parseInt(getAttribute("accountId"));
		}
	public String represent() {
		return AccountsServerResource.getAccounts().get(this.accountId);
		}
	public void store(String account) {
		AccountsServerResource.getAccounts().set(this.accountId, account);
		}
	public void remove() {
		AccountsServerResource.getAccounts().remove(this.accountId);
		}
	}
```

AccountsServerResource.java

```
package server;
import java.util.List;
import java.util.concurrent.CopyOnWriteArrayList;
import common.AccountsResource;
import org.restlet.resource.ServerResource;
```

```
public class AccountsServerResource extends ServerResource implements
AccountsResource {
        private static final List<String> accounts = new CopyOnWriteArrayList<String>();
        public static List<String> getAccounts() {
        return accounts;
        }
public String represent() {
        StringBuilder result = new StringBuilder();
        for (String account : getAccounts()) {
                result.append((account == null) ? "" : account).append('\n');
                }
        return result.toString();
        }
public String add(String account) {
        getAccounts().add(account);
        return Integer.toString(getAccounts().indexOf(account));
        }
}
```

RootServerResource.java

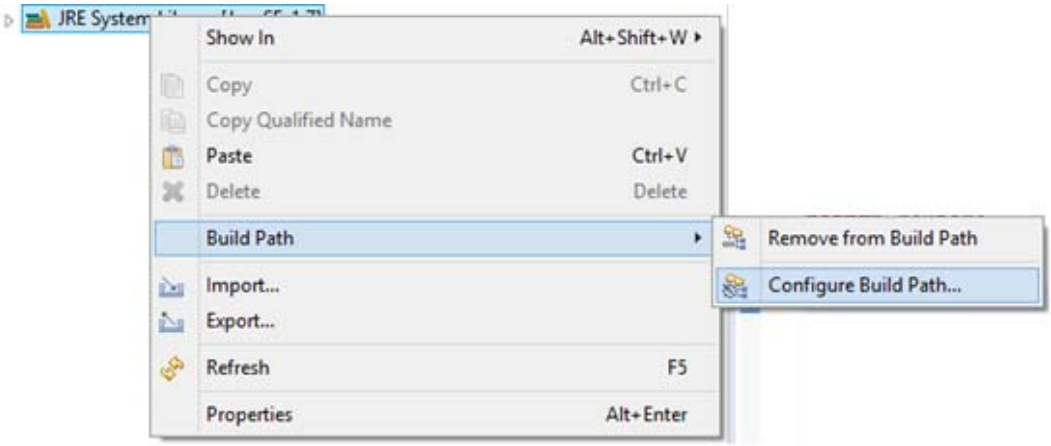
```
package server;
import org.restlet.resource.Get;
import org.restlet.resource.Options;
import org.restlet.resource.ServerResource;
public class RootServerResource extends ServerResource {
        @Get ("txt")
        public String represent(){
                return " ";
                }
        @Options ("txt")
        public String describe(){
                throw new RuntimeException("Not yet implemented");
                }
        }
```

MailServerApplication.java

```
package server;
import org.restlet.Application;
import org.restlet.Restlet;
import org.restlet.Server;
import org.restlet.data.Protocol;
import org.restlet.routing.Router;
public class MailServerApplication extends Application {
        public static void main(String[] args) throws Exception {
                Server mailServer = new Server(Protocol.HTTP, 8111);
                mailServer.setNext(new MailServerApplication());
                mailServer.start();
                }
        @Override
        public Restlet createInboundRoot() {
                Router router = new Router(getContext());
                router.attach("http://localhost:8111/",
                RootServerResource.class);
                router.attach("http://localhost:8111/accounts/",
                AccountsServerResource.class);
```

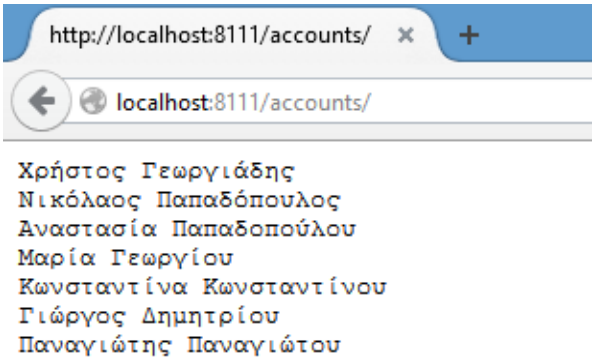
```
            router.attach("http://localhost:8111/accounts/{accountId}",
            AccountServerResource.class);
        return router;
        }
}
```

- Όπως βλέπουμε το IDE μας ενημερώνει για ένα πλήθος σφαλμάτων, που οφείλονται στη χρήση πακέτων που ορίζει το restlet framework. Για να λυθούν λοιπόν θα πρέπει να εισάγουμε το συγκεκριμένο framework στο project.
- Προσθήκη της βιβλιοθήκης org.restlet.jar ως external jar file: Αφού κατεβάσουμε το restlet framework από τον σύνδεσμο «http://restlet.com/downloads/current/», και αποσυμπιέσουμε το αρχείο zip, κάνουμε δεξί κλικ στο JRE System Library->Build Path->Configure Build Path->Add external JARs, και επιλέγουμε το org.restlet.jar από τον υπολογιστή μας.



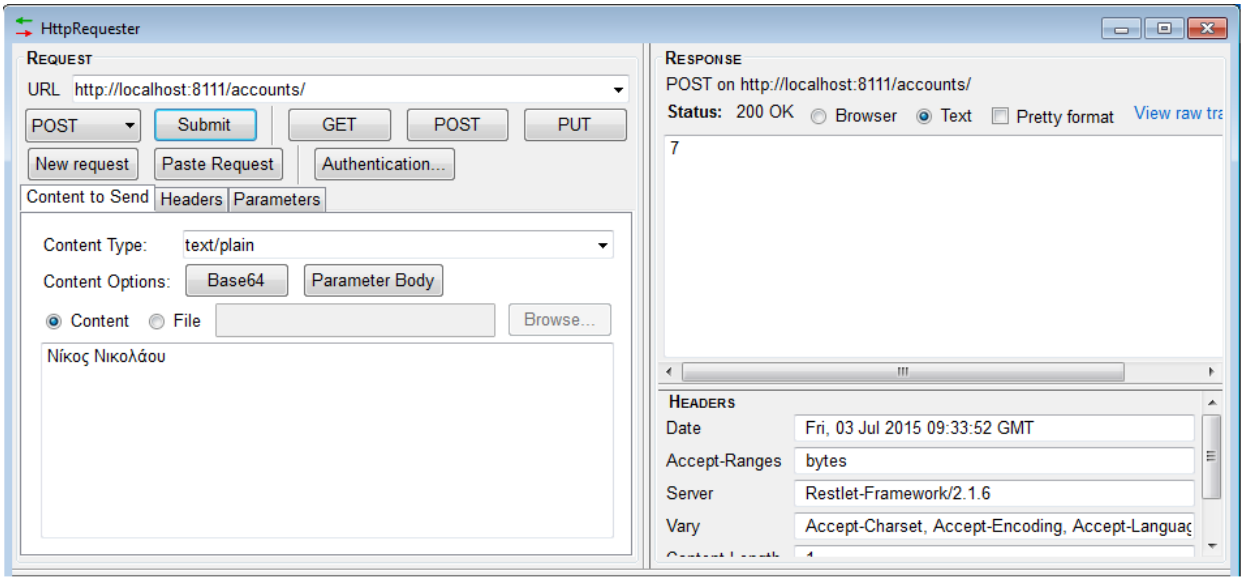
Εικόνα 10.4 *Εισαγωγή του org.restlet.jar ως external jar*

- Εκτέλεση της εφαρμογής στο Eclipse σαν Java Application. Δεξί κλικ στην κύρια κλάση της εφαρμογής (MailServerApplication.java) και επιλέγουμε Run As-> Java Application. Ξεκινάει ο server στο port 8111. Είναι πλέον σε θέση να δεχτεί HTTP requests (GET, POST, PUT, DELETE)
- Εκτέλεση με τον ίδιο τρόπο της MailClient.java. Αυτόματα εκτελούνται οι λειτουργίες που ορίσαμε στην κλάση. Συγκεκριμένα:
    - Παρουσίαση της λίστας χρηστών (αρχικά κενή) με χρήση ενός GET request (AccountsResource.java) στο http://localhost:8111/accounts/
    - Προσθήκη 8 νέων χρηστών που έχουν οριστεί στην κλάση με χρήση POST request (AccountsResource.java) στο http://localhost:8111/accounts/
    - Εμφάνιση της ενημερωμένης λίστας χρηστών με τους 8 χρήστες με χρήση GET (AccountsResource.java) στο http://localhost:8111/accounts/
    - Εμφάνιση του 5ου χρήστη http://localhost:8111/accounts/4 (Καθώς η αρίθμηση ξεκινά από το 0).
    - Επικαιροποίηση του 5ου χρήση με τη χρήση PUT request (AccountResource.java)
    - Διαγραφή του 8ου χρήστη στη σειρά (ξεκινώντας από το 0) με DELETE (AccountResource.java)
- Προβολή λίστας χρηστών με GET
    - Εισαγωγή του URL http://localhost:8111/accounts/ στον Mozilla Firefox. Εμφάνιση της λίστας χρηστών
    - Εναλλακτικά, επιλογή μέσω του HttpRequester plugin, από το drop-down menu της επιλογής GET και αποστολή του HTTP request πατώντας το πλήκτρο Submit. Με τον τρόπο αυτό βλέπουμε την απάντηση στο request μας.
- Πρόσθεση χρήστη με κλήση POST



Εικόνα 10.5 *Αποτελέσματα προβολής χρηστών μέσω GET (μετά τη διαγραφή του 8ου χρήστη)*

- Στην διεύθυνση http://localhost:8111/accounts/ εισαγωγή στο παράθυρο Content ονόματος χρήστη και επιλογή από το drop-down menu (κάτω από τη γραμμή διεύθυνσης) το POST. Σαν επιλογή στο Content Type επιλέγουμε text/plain και πατάμε το πλήκτρο Submit για την αποστολή του HTTP request.
- Βλέπουμε στο Content το όνομα λογαριασμού που επιλέξαμε. Στο response τμήμα γίνεται φανερό πως ο νέος χρήστης θα έχει ως νούμερο το 7, καθώς είναι ο 8ος χρήστης και η αρίθμηση ξεκινάει από το 0.



Εικόνα 10.6 *Εκτέλεση POST request μέσω HttpRequester plugin*

- Για επιβεβαίωση αν ζητήσουμε ένα GET θα λάβουμε τη συνολική λίστα των 8 λογαριασμών.

- Διαγραφή λογαριασμού χρήστη με το DELETE
  - Επιλέγουμε να διαγράψουμε τον 3ο λογαριασμό οπότε πληκτρολογούμε στο URL τη διεύθυνση http://localhost:8111/accounts/2.
  - Επιλογή του DELETE και πατάμε το πλήκτρο Submit
  - Πλέον στη λίστα χρηστών δε θα υπάρχει ο τρίτος χρήστης.
- Τροποποίηση λογαριασμού χρήστη με το PUT
  - Επιλέγουμε να τροποποιήσουμε τον 1ο λογαριασμό, οπότε πληκτρολογούμε στο URL τη διεύθυνση http://localhost:8111/accounts/0.
  - Πληκτρολογούμε στο πεδίο Content το νέο όνομα του χρήστη και επιλέγουμε PUT και πατάμε το πλήκτρο Submit.

Video 10.1.mp4	Βίντεο (video)
Η λειτουργία του HttpRequester plugin, μέσω παραδειγμάτων χρήσης	

3. Υπολογιστική Νέφους (Cloud Computing) και σχετικά Ζητήματα Ασφάλειας για Εφαρμογές Ηλεκτρονικού Εμπορίου

Τα τελευταία χρόνια η «υπολογιστική νέφους» έχει γίνει κοινά αποδεκτή και υιοθετείται από ολοένα και περισσότερους οργανισμούς, καθώς παρέχει μια εναλλακτική μέθοδο για επιχειρήσεις (αλλά και απλούς χρήστες) αποθήκευσης δεδομένων, επικοινωνίας και λειτουργικότητας. Η τεχνολογία cloud προσφέρει τη δυνατότητα ελαχιστοποίησης των εξόδων μιας επιχείρησης, αφού με το λεγόμενο Software as a Service (SaaS), προσφέρει ένα εναλλακτικό μοντέλο παροχής λογισμικού, αφού τόσο η εφαρμογή όσο και τα δεδομένα προς επεξεργασία μπορούν να αποθηκευθούν και να εκτελεστούν στους servers του cloud παρόχου.

Σε αυτή την ενότητα θα παρουσιαστούν σύντομα ζητήματα ασφαλείας που σχετίζονται με τη χρήση υπηρεσιών «υπολογιστικής νέφους», ενώ παράλληλα θα γίνει επίδειξη του τρόπου μεταφοράς μιας Υπηρεσίας Ιστού σε έναν cloud πάροχο.

3.1. Ζητήματα ασφαλείας στην υπολογιστική νέφους

Η εμφάνιση της τεχνολογίας του «υπολογιστικού νέφους», έχει δημιουργήσει την ανάγκη λήψης επιπλέον μέτρων ασφαλείας, για τη διασφάλιση ευαίσθητων δεδομένων, καθώς νέες προκλήσεις κάνουν την εμφάνιση τους (Subashini & Kavitha, 2011). Ακόμη και προκλήσεις ασφαλείας που είναι πάγιες σε υπολογιστικά συστήματα, όπως είναι η αυθεντικοποίηση, η αδειοδότηση και η διαθεσιμότητα, μεγεθύνονται ραγδαία σε συστήματα «υπολογιστικού νέφους». Για παράδειγμα, στα ζητήματα αυθεντικοποίησης και αδειοδότησης, νέες ανησυχίες εγείρονται από τις πολιτικές και τα πρωτόκολλα που χρησιμοποιεί ο εκάστοτε πάροχος. Αντίστοιχα στο ζήτημα της διαθεσιμότητας, καθώς τα υλικά τμήματα των υποδομών πολλών επιχειρήσεων ανήκουν πλέον στην ευθύνη τρίτων, μια πιθανή μηχανική βλάβη ή αστοχία υλικού μπορεί να έχει αντίκτυπο σε περισσότερους τελικούς χρήστες σε σχέση με την περίπτωση που δε χρησιμοποιείται η «υπολογιστική νέφους» (Vesyropoulos et al., 2014).

Πιο αναλυτικά, μερικά από τα κυριότερα ζητήματα ασφαλείας που προκύπτουν είναι τα παρακάτω:

Ζητήματα εμπιστευτικότητας (Confidentiality):
Η εμπιστευτικότητα δεδομένων αφορά την εξουσιοδότηση (authorization) ενός χρήστη, προτού αυτός να μπορεί να έχει πρόσβαση σε ευαίσθητα δεδομένα. Εάν υπάρξει πρόσβαση ενός μη-εξουσιοδοτημένου χρήστη σε αυτά τα δεδομένα, υπάρχει κίνδυνος τόσο από την έκθεση αυτών, όσο και από πιθανές τροποποιήσεις τους, που τελικά μπορούν να προκαλέσουν ανεπανόρθωτη ζημιά στον κάτοχο τους. Καθώς η αποθήκευση δεδομένων πραγματοποιείται από την πλευρά του παρόχου, μεταφέρεται σε αυτόν ένα μεγάλο τμήμα της ευθύνης για την εμπιστευτικότητα τους.

Πέρα όμως από την εμπιστευτικότητα δεδομένων, σε περιβάλλοντα «υπολογιστικής νέφους» είναι ιδιαίτερα σημαντική και η εμπιστευτικότητα λογισμικού. Καθώς για την πρόσβαση σε ευαίσθητα δεδομένα γίνεται χρήση λογισμικού του παρόχου, ο τελικός χρήστης πολλές φορές δεν μπορεί να είναι ενημερωμένος για τα πρωτόκολλα και τις λειτουργίες αυτών των προγραμμάτων και δεν είναι σε θέση να επιλέξει το λογισμικό που προτιμά. Για τον λόγο αυτό είναι σημαντικό οι εφαρμογές αυτός να παρέχουν εγγυήσεις σχετικά με τα πρωτόκολλα ασφαλείας που προσφέρουν, ενώ παράλληλα είναι θεμιτό να γίνεται καταγραφή των αλλαγών που πραγματοποιούνται σε κάθε έκδοση των προγραμμάτων αυτόν σε αρχεία καταγραφής (log files).

Επιπρόσθετα, κατά τη χρήση της τεχνολογίας «υπολογιστικής νέφους» τα ζητήματα εμπιστευτικότητας αποκτούν ιδιάζουσα βαρύτητα, καθώς η υιοθέτηση αυτού του κατανεμημένου μοντέλου παροχής υπηρεσιών συνεπάγεται την παρουσία περισσοτέρων εμπλεκομένων ενδιαφερομένων (χρηστών, προγραμματιστών, υπαλλήλων των παρόχων κ.α.) αλλά και προγραμμάτων και συσκευών, κάτι που οδηγεί και στην ύπαρξη περισσοτέρων πιθανών κινδύνων ασφαλείας.

Τέλος, σχετικά με το ζήτημα της εμπιστευτικότητας στην «υπολογιστική νέφους», εγείρονται επιπρόσθετες προκλήσεις που οφείλονται στην πολυμίσθωση (multitenancy) και στην παραμένουσα

μαγνήτιση των σκληρών δίσκων του παρόχου, κάτι που μπορεί να οδηγήσει σε ανάκτηση διαγραμμένων δεδομένων (Data remanence):

- Η πολυμίσθωση αναφέρεται στον διαμοιρασμό πόρων ανάμεσα σε πολλούς χρήστες. Τέτοιοι πόροι μπορεί να περιλαμβάνουν εφαρμογές, ευαίσθητα δεδομένα, αποθηκευτικούς χώρους και μνήμη. Καθώς οι διάφοροι χρήστες μπορεί να χειρίζονται διαφορετικά «στιγμιότυπα» των ίδιων πόρων, κάνουν και χρήση των ίδιων συσκευών υλικού. Έτσι είναι, υπό προϋποθέσεις, εφικτή η μη-εξουσιοδοτημένη πρόσβαση ενός χρήστη στις πληροφορίες ενός άλλου χρήστη, όταν γίνεται ταυτόχρονα χρήση του ίδιου υλικού.
- Η παραμένουσα μαγνήτιση αναφέρεται στην ύπαρξη «ίχνους» διαγραμμένων δεδομένων από έναν σκληρό δίσκο, τα οποία μέσω ειδικού λογισμικού μπορούν να ανακτηθούν. Καθώς λοιπόν ένας χρήστης αποκτά πρόσβαση σε ένα τμήμα ενός σκληρού δίσκου ενός παρόχου, είναι δυνατό μέσω ενός τέτοιου ίχνους να αποκτήσει πρόσβαση σε πληροφορίες που φαινομενικά είχαν διαγραφεί και που ανήκαν σε κάποιον προηγούμενο χρήστη.

Ζητήματα Ακεραιότητας (Integrity)
Με τον όρο αυτό αναφερόμαστε στη διασφάλιση ότι τα μη εξουσιοδοτημένα άτομα δεν έχουν δυνατότητες τροποποίησης σε δεδομένα αλλά και σε ρυθμίσεις του υλικού στο οποίο αυτά αποθηκεύονται. Και ενώ μια επιχείρηση είναι σε θέση να ελέγχει την πρόσβαση στους τοπικούς υπολογιστές και στα δεδομένα της από το προσωπικό της (μέσω της καταγραφής log files), αυτό είναι σαφώς δυσκολότερο σε σενάρια χρήσης «υπολογιστικής νέφους».

Οι πιθανές παραβιάσεις δεδομένων και υλικού, όταν αυτά αποθηκεύονται απομακρυσμένα μπορούν να είναι τόσο από εσωτερικές όσο και από εξωτερικές πηγές. Συγκεκριμένα μπορεί να οφείλονται σε κακοπροαίρετη συμπεριφορά των εργαζομένων ενός παρόχου αλλά και σε ενέργειες χρηστών και προγραμμάτων που έχουν ως στόχο την παραποίηση ευαίσθητων πληροφοριών.

Ζητήματα Διαθεσιμότητας (Availability)
Με τον όρο διαθεσιμότητα αναφερόμαστε στη δυνατότητα πρόσβασης σε έναν συγκεκριμένο πόρο τη στιγμή που αυτός ζητείται από τον χρήστη (on-demand). Ένα υψηλό επίπεδο διαθεσιμότητας από έναν πάροχο, θα σήμαινε πως ο πάροχος είναι θέση να προσφέρει κρίσιμες λειτουργίες και πρόσβαση σε πόρους, ακόμη κι αν έχει προηγηθεί μια βλάβη υλικού ή μια εξωτερική επίθεση. Κι ενώ σε παραδοσιακά τοπικά συστήματα υπολογιστών, η διαθεσιμότητα αφορά κυρίως τη συχνότητα εμφάνισης βλαβών υλικού/λογισμικού, σε περιβάλλοντα «υπολογιστικής νέφους» το ζήτημα αυτό μετατοπίζεται στη συχνότητα εμφάνισης προβλημάτων στη δομή του δικτύου, στα πρωτόκολλα που χρησιμοποιούνται και στο εύρος ζώνης που διατίθεται από τον πάροχο. Καθώς ένα πολύ μεγάλο πλήθος πληροφοριών ανταλλάσσεται καθημερινά, ανάμεσα σε έναν σημαντικά μεγάλο αριθμό χρηστών, ο πάροχος θα πρέπει να λάβει τα κατάλληλα μέτρα έτσι ώστε να μπορεί να ανταπεξέλθει στη μεγάλη αυτή ζήτηση για εύρος ζώνης, ενώ παράλληλα θα πρέπει να είναι προετοιμασμένος για εξωτερικές επιθέσεις από κακόβουλους χρήστες που θα προσπαθήσουν να προκαλέσουν βλάβη στο δίκτυο (πχ. μέσω επιθέσεων DDOS).

Ζητήματα εμπιστοσύνης (Trust)
Ο όρος εμπιστοσύνη σε μια συναλλαγή αναφέρεται στην προσδοκία από πλευράς του κάθε εμπλεκόμενου πως τα υπόλοιπα εμπλεκόμενα μέλη της συναλλαγής θα έχουν την αναμενόμενη συμπεριφορά κατά τη διάρκεια της. Από την πλευρά των χρηστών, η εμπιστοσύνη σε σενάρια χρήσης της τεχνολογίας «υπολογιστικής νέφους» αφορά την υπόθεση πως ο πάροχος θα είναι σε θέση να παραδώσει τις αναμενόμενες υπηρεσίες και τα προϊόντα. Παράλληλα όμως αφορά και την υπόθεση πως ο πάροχος θα πάρει όλα εκείνα τα μέτρα και τις προφυλάξεις που απαιτούνται για τη διασφάλιση των δεδομένων του χρήστη. Ανάμεσα λοιπόν στα εμπλεκόμενα μέλη της συναλλαγής θα πρέπει να δημιουργείται η αίσθηση πως οι συναλλαγές θα διενεργούνται με μεγάλο επίπεδο ασφάλειας.

Παρόλα αυτά κατά τη χρήση της «υπολογιστικής νέφους» η ευθύνη για τους μηχανισμούς ασφαλείας μεταφέρεται σε μεγάλο βαθμό στην πλευρά του παρόχου. Έτσι ο τελικός χρήστης έχει περιορισμένη πρόσβαση σε πληροφορίες σχετικά με τους μηχανισμούς αυτούς και με τον τρόπο που αυτοί χρησιμοποιούνται από τον πάροχο, κάτι που μπορεί να μειώσει το αίσθημα ασφαλείας που νιώθει ο χρήστης με αποτέλεσμα να μειωθεί και το επίπεδο εμπιστοσύνης του προς τον πάροχο.

3.2. Εφαρμογή Restlet στο Google App Engine

Εκφώνηση:

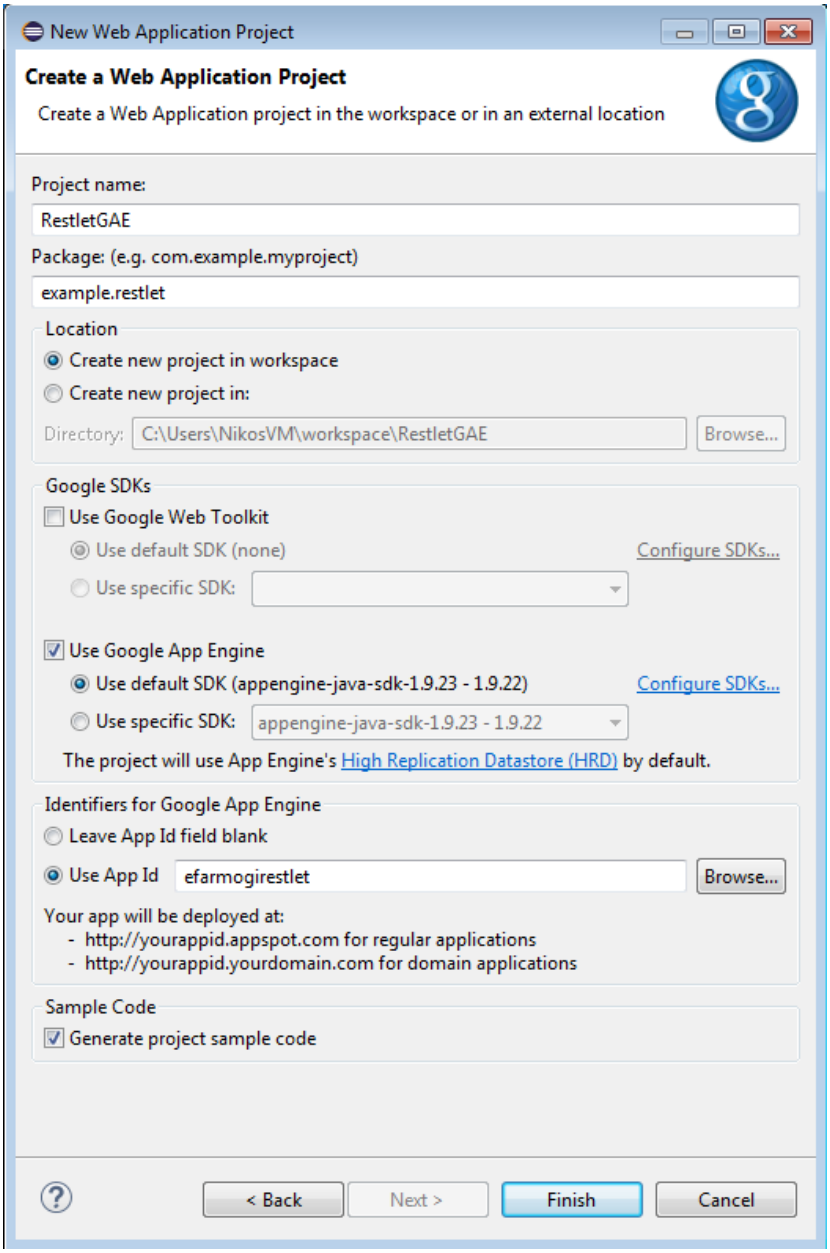
Αναπτύξτε μια εφαρμογή προβολής πληροφοριών χρηστών, με χρήση του RESTlet framework και «ανεβάστε» τη σε πάροχο υπηρεσιών υπολογιστικής νέφους. Προτείνεται η χρήση του Google App Engine.

Υποδειγματική λύση:

- Πρέπει να κατεβάσουμε το Google App Engine SDK (έκδοση για Java), από τη διεύθυνση https://cloud.google.com/appengine/downloads?csw=1. Αποσυμπιέζουμε το αρχείο που κατέβηκε.
- Πρέπει επίσης να κατεβάσουμε το απαραίτητο plugin της Google μέσω του περιβάλλοντος Eclipse. Πατάμε Help > Install New Software, δίνουμε ως διεύθυνση την «https://dl.google.com/eclipse/plugin/4.4» και από τη λίστα που μας επιστρέφεται επιλέγουμε το Google plugin for eclipse.
- Δημιουργία ενός νέου Web Application Project, με την ονομασία RestletGAE, με ονομασία package example.restlet, και App Id, το efarmogirestlet. Επιλέγουμε File->New->Project->Google->Web Application Project.
- Ρυθμίζουμε το Google App Engine πατώντας την επιλογή Configure SDK, και επιλέγοντας τον φάκελο που προέκυψε από την αποσυμπίεση του αρχείου. Δίνουμε τις επιλογές που φαίνονται στην εικόνα 10.7 και πατάμε Finish.
- Διαγράφουμε την sample code κλάση RestletGAEServlet.java που δημιουργήθηκε αυτόματα.
- Δημιουργούμε τις κλάσεις της εφαρμογής μας.
    - Αρχικά δημιουργούμε την **RESTLet.java** που δημιουργεί ένα Restlet component και συνδέει έναν HTTP server connector σε αυτό. Ο HTTP server μας είναι ο 8888.

RESTLet.java

```
package userRest;
import org.restlet.Component;
import org.restlet.data.Protocol;
public class RESTLet {
public static void main(String[] args) throws Exception {
// Create a new Restlet component and add a HTTP server connector to it
Component component = new Component();
component.getServers().add(Protocol.HTTP, 8888);
// Attach the application to the component and start it
component.getDefaultHost().attach(new RouterApplication());
component.start();
}
}
```



Εικόνα 10.7 *Δημιουργία Web application project*

- Στη συνέχεια δημιουργούμε την **RouterApplication.java** η οποία θα λαμβάνει τις εισερχόμενες REST κλήσεις.

RouterApplication.java

```
package userRest;
import org.restlet.Application;
import org.restlet.Restlet;
import org.restlet.routing.Router;
public class RouterApplication extends Application{
// Creates a root Restlet that will receive all incoming calls.
@Override
public synchronized Restlet createInboundRoot() {
// Create a router Restlet that routes each call to a new respective instance of resource.
Router router = new Router(getContext());
// Define three routes
```

```
router.attach("/user/{uid}", UserResource.class);
router.attach("/user/{uid}/complete", UserCompleteResource.class);
router.attach("/user/{uid}/pending", UserPendingResource.class);
return router;
}
}
```

- Τέλος ορίζουμε τις επόμενες 3 κλάσεις που περιγράφουν την ιδιότητα ενός χρήστη, το πλήθος τον αγορών που έχει κάνει και το πλήθος των παραγγελιών που εκκρεμούν.

UserResource.java

```
package userRest;
import org.restlet.resource.Get;
import org.restlet.resource.ServerResource;
public class UserResource extends ServerResource {
@Get
public String toString() {
String uid = (String) getRequestAttributes().get("uid");
return "Ο χρήστης \"" + uid + "\" είναι: εγγεγραμμένος χρήστης";
}
}
```

UserCompleteResource.java

```
package userRest;
import org.restlet.resource.Get;
import org.restlet.resource.ServerResource;
public class UserCompleteResource extends ServerResource {
@Get
public String toString() {
String uid = (String) getRequestAttributes().get("uid");
return "Οι ολοκληρωμένες παραγγελίες του χρήστη \"" + uid + "\" είναι: 12";
}
}
```

UserPendingResource.java

```
package userRest;
import org.restlet.resource.Get;
import org.restlet.resource.ServerResource;
public class UserPendingResource extends ServerResource {
@Get
public String toString() {
String uid = (String) getRequestAttributes().get("uid");
return "Οι εκκρεμείς παραγγελίες του χρήστη \"" + uid + "\" είναι: 2";
}
}
```

- Προσθέτουμε το **org.restlet.jar** και το **org.restlet.ext.servlet.jar** στα External JARs όπως κάναμε και στην τοπική εκτέλεση.

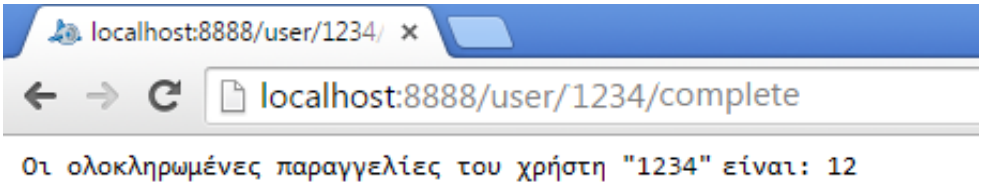
Sound 10.2.mp3	Ηχητικό απόσπασμα (audio)
Επεξήγηση των ιδιαιτεροτήτων του GAE, και της συμβατότητας με το RESTlet framework	

- Κάνοντας αντιγραφή-επικόλληση των αρχείων **org.restlet.jar** και **org.restlet.ext.servlet.jar** από τον υπολογιστή μας τα προσθέτουμε και στο /RestletGAE/war/WEB-INF/lib

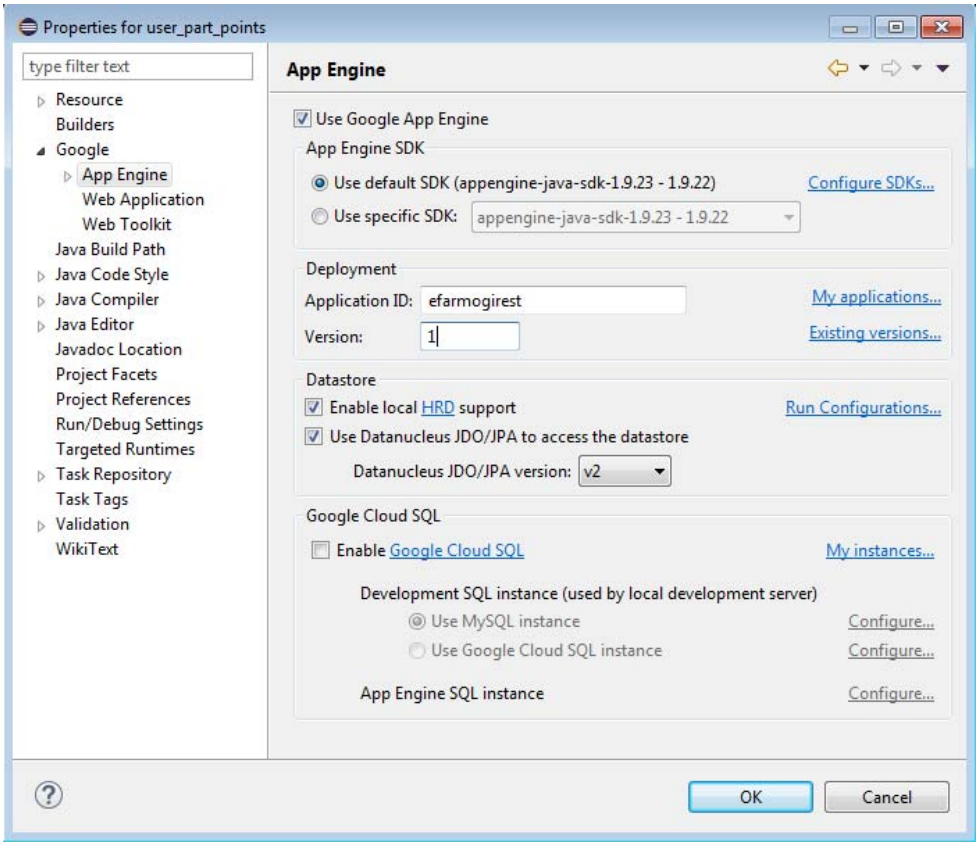
- Έλεγχος του αρχείου appengine-web.xml (βρίσκεται στο RestletGAE/war/WEB-INF/appengine-web.xml). Θέλουμε να βεβαιωθούμε πως αναφέρεται στο σωστό όνομα εφαρμογής με τη σωστή έκδοση
- Τροποποίηση του αρχείου web.xml (βρίσκεται στο /RestletGAE/war/WEB-INF/web.xml). Θέλουμε να βεβαιωθούμε πως αναφέρεται στην κύρια εκτελέσιμη κλάση της εφαρμογής, την **«RouterApplication»** και έτσι να έχει τη μορφή:

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE web-app PUBLIC
"-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app xmlns="http://java.sun.com/xml/ns/javaee" version="2.5">
	<display-name>restlet servlet</display-name>
	<servlet>
		<servlet-name>RestletServlet</servlet-name>
		<servlet-class>org.restlet.ext.servlet.ServerServlet</servlet-class>
		<init-param>
			<param-name>org.restlet.application</param-name>
			<param-value>tutorial.restlet.RouterApplication </param-value>
		</init-param>
	</servlet>
	<!-- Catch all requests -->
	<servlet-mapping>
		<servlet-name>RestletServlet</servlet-name>
		<url-pattern>/*</url-pattern>
	</servlet-mapping>
</web-app>
```

- Τροποποιούμε το αρχείο index.html που βρίσκεται στο /RestletGAE/war/index.html δίνοντάς του όποια μορφή θέλουμε να εμφανίζεται σαν αρχική σελίδα της εφαρμογής μας.
- Εκτέλεση της εφαρμογής μας τοπικά
  - Εκτελούμε την εφαρμογή μας τοπικά με δεξί κλικ – Run As – Web Application.
  - Από το μήνυμα που λαμβάνουμε στην Console του Eclipse φαίνεται να τρέχει κανονικά ο server στο http://localhost:8888/
  - Δοκιμάζουμε τα παρακάτω URI:
    - http://localhost:8888/user/1
    - http://localhost:8888/user/2/pending
    - http://localhost:8888/user/1234/complete
- Ανέβασμα και εκτέλεση της εφαρμογής στο Google App Engine.
- (Προαιρετικά) Κάνουμε δεξί κλικ – Google – App Engine Settings, βλέπουμε τις ρυθμίσεις της εφαρμογής για ανέβασμα στο Google App Engine, και ελέγχουμε αν θέλουμε να κάνουμε αλλαγές στο όνομα της εφαρμογής.



Εικόνα 10.8 *Τοπική εκτέλεση ενός GET request*



Εικόνα 10.9 *Οθόνη προαιρετικής παραμετροποίησης στοιχείων εφαρμογής πριν το deploy*

- Κάνουμε δεξί κλικ στο project μας – Google – Deploy to App Engine – Deploy
- Είναι απαραίτητη η χρήση ενός λογαριασμού Google. Στην κονσόλα του Eclipse βλέπουμε ότι το deployment έγινε επιτυχώς.
- Εκτελούμε τα παραπάνω URI αντικαθιστώντας το «http://localhost:8888» με το «efarmogirest.appspot.com» και βλέπουμε πως η εφαρμογή λειτουργεί κανονικά στο Google App Engine, δίνοντας τα ίδια αποτελέσματα με την τοπική εκτέλεση.

4. Δημιουργία ενορχήστρωσης BPEL για τη σύνθεση YI

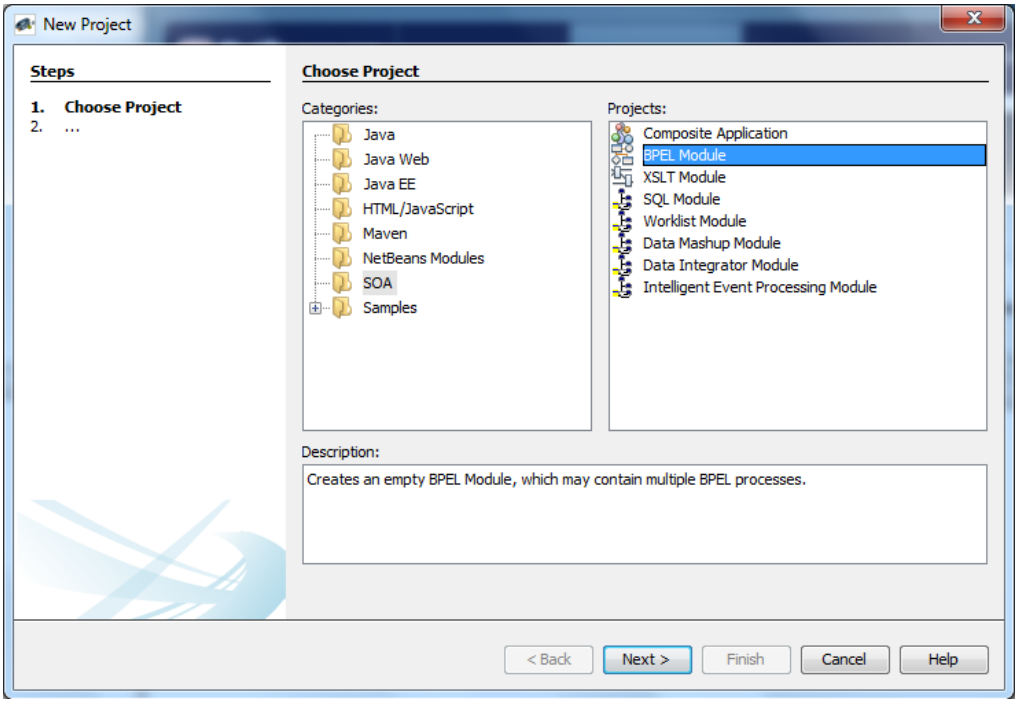
Εκφώνηση:

Δημιουργείστε μια ενορχήστρωση BPEL, για τον έλεγχο της τιμής μιας string μεταβλητής που θα δίνεται από ένα partner link προς ένα BPEL module.

Για την υλοποίηση της ενορχήστρωσης προτείνεται η χρήση του πακέτου OpenESB 2.3.1, το οποίο συμπεριλαμβάνει το περιβάλλον ανάπτυξης NetBeans (έκδοση 7) μαζί με τον Glassfish Server και παρέχει υποστήριξη για σύνθεση YI μέσω BPEL.

Υποδειγματική λύση:

- Δημιουργία BPEL module
  - Από το περιβάλλον NetBeans IDE, επιλέγουμε File → New Project.
  - Στο μενού Categories, επιλέγουμε Service Oriented Architecture.
  - Στο μενού Projects, επιλέγουμε BPEL Module και μετά Next.



Εικόνα 10.10 *Δημιουργία BPEL module*

- Στα Name και Location, δίνουμε το όνομα και την τοποθεσία στην οποία θέλουμε να αποθηκευθεί το project μας (σε αυτό το παράδειγμα ονομάζουμε το αρχείο CompareStrings).
- Τώρα στα projects εμφανίζεται το BPEL module όπως φαίνεται στην εικόνα 10.12.

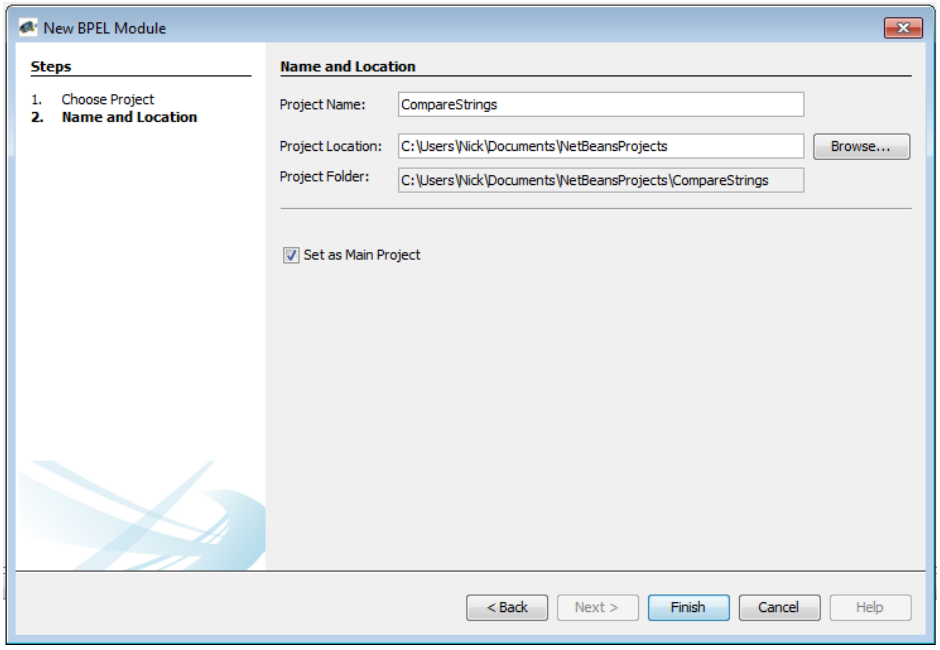
Δημιουργία του XML Schema

Γενικά, όταν δημιουργούμε ένα BPEL module project ακολουθούμε τα παρακάτω βήματα:

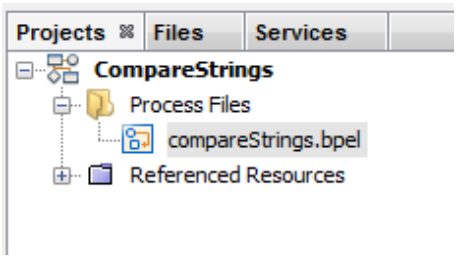
1. Δημιουργούμε ένα νέο BPEL Project
2. Δημιουργούμε το XML Schema ή το XSD αρχείο
3. Δημιουργούμε το WSDL αρχείο
4. Δημιουργούμε την BPEL διαδικασία

Το αρχείο XSD (XML Schema) βοηθά να ορίσουμε τη δομή μηνυμάτων του project. Σύνθετες δομές μηνυμάτων ορίζονται στο XSD αρχείο και στη συνέχεια εισάγονται στο αρχείο WSDL.

Σε αυτή την ενότητα προσθέτουμε ένα νέο αρχείο XML schema στο BPEL Module και προσθέτουμε components στο schema.



Εικόνα 10.11 *Ρύθμιση παραμέτρων του BPEL module*



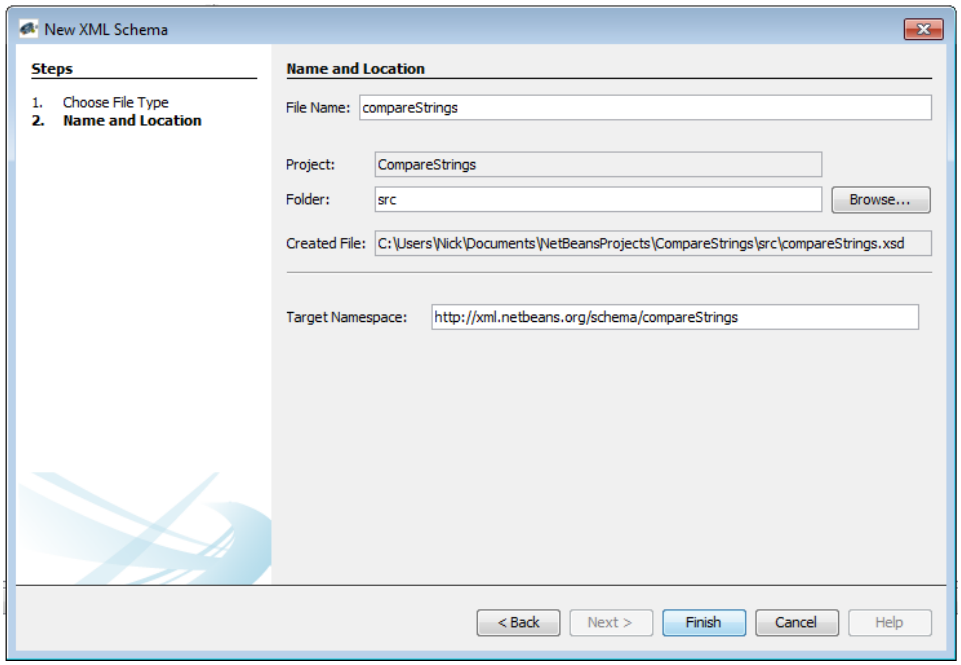
Εικόνα 10.12 *Εμφάνιση του BPEL module στη λίστα αρχείων του project*

Για να δημιουργήσουμε το `CompareStrings.xsd`:

- Στο παράθυρο Projects, κάνουμε expand το `CompareStrings` κόμβο, και στη συνέχεια κάνουμε δεξί κλικ στον κόμβο `Process Files` και επιλέγουμε New -> Other.
- Στο παράθυρο που ανοίγει:
  - Στη σελίδα Choose File Type, στη λίστα Categories, επιλέγουμε `XML` , και στη λίστα File Types, επιλέγουμε `XML Schema` και πατάμε Next.
  - Σαν όνομα στο πεδίο File Name, βάζουμε το `CompareStrings`.
  - Πατάμε Finish.
- Στο παράθυρο Projects, ο κόμβος Process Files πλέον περιέχει έναν υποκόμβο που ονομάζεται CompareStrings.xsd. Ο Source Editor περιέχει μια καρτέλα για το XML schema.
- Στο Schema view, κάντε κλικ στο κουμπί Design για να ανοίξει το Design view του XML schema editor.

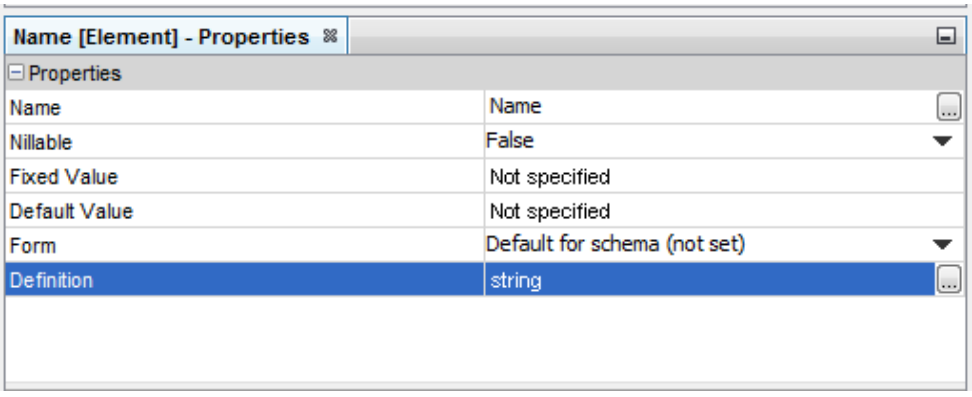
Προσθήκη ενός αντικειμένου τύπου string στο XML schema (τύπου global):

- Στο τμήμα XML Components της παλέτας επιλέξτε το εικονίδιο Element και σύρετε την επιλογή σας στο design area, ακριβώς κάτω από τον κόμβο Elements. Το IDE αυτόματα προσθέτει ένα στοιχείο με την ονομασία `newElement` κάτω από τον κόμβο Elements στο schema design.



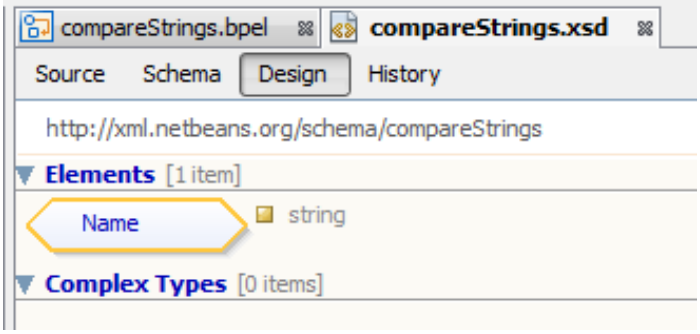
Εικόνα 10.13 Δημιουργία του XML schema

- Στην περιοχή του schema design του Design view, επιλέξτε τον κόμβο `newElement`.
- Στο παράθυρο Properties window, επιλέξτε το πεδίο ονόματος και δώστε σαν όνομα `Name`.
- Στο ίδιο παράθυρο, στο πεδίο `Definition` κάντε κλικ στο κουμπί της έλλειψης.
- Από τη λίστα επιλέξτε `Built-in Types-> string` και πατήστε OK.



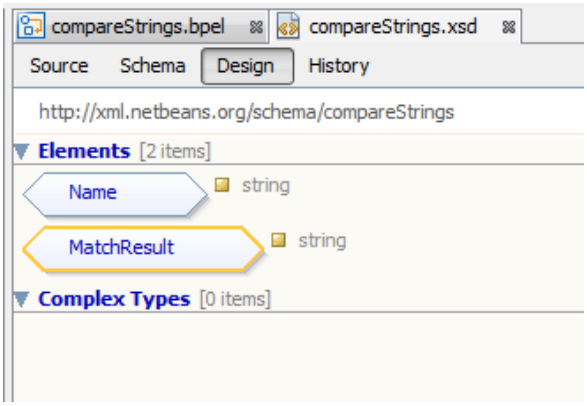
Εικόνα 10.14 *Προσθήκη νέου αντικειμένου τύπου string*

- Στο design view, to IDE πλέον δείχνει το επιλεγμένο Name και δίπλα τον τύπο string.



Εικόνα 10.15 *Εμφάνιση του νέου αντικειμένου τύπου string*

- Επαναλαμβάνουμε την παραπάνω διαδικασία για τη δημιουργία ενός ακόμα κόμβου που θα δέχεται το αποτέλεσμα της σύγκρισης, τον MatchResult, επίσης τύπου string.



Εικόνα 10.16 *Εμφάνιση του αντικειμένου MatchResult*

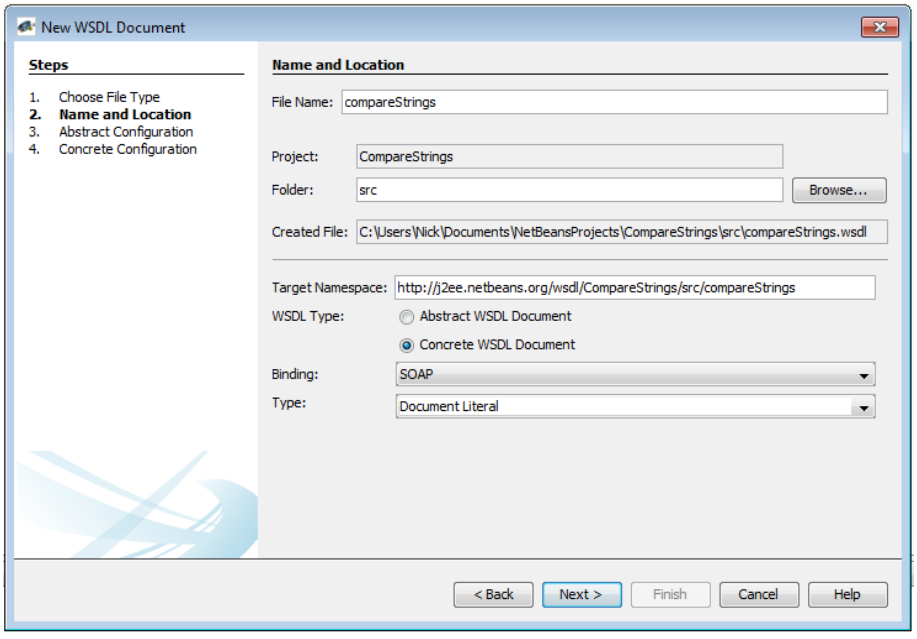
- Για να αποθηκεύσουμε την εργασία μας, από το παράθυρο Projects, επιλέγουμε το compareStrings project κόμβο και από την μπάρα στο πάνω μέρος της οθόνης επιλέγουμε File > Save All.

Δημιουργία του WSDL Document

- Στο παράθυρο Projects, κάνουμε expand τον κόμβο `compareStrings` project, και στη συνέχεια με δεξί κλικ στον κόμβο `Process Files` και επιλέγουμε New -> WSDL Document.

- Στη συνέχεια:

1. Σαν όνομα δίνουμε `compareStrings`.
2. Επιλέγουμε ως WSDL Type την επιλογή Concrete WSDL Document.
3. Στην επιλογή Binding επιλέγουμε SOAP.
4. Στην επιλογή Type δίνουμε την τιμή Document Literal.
5. Πατάμε Next.



Εικόνα 10.17 *Δημιουργία νέου WSDL αρχείου*

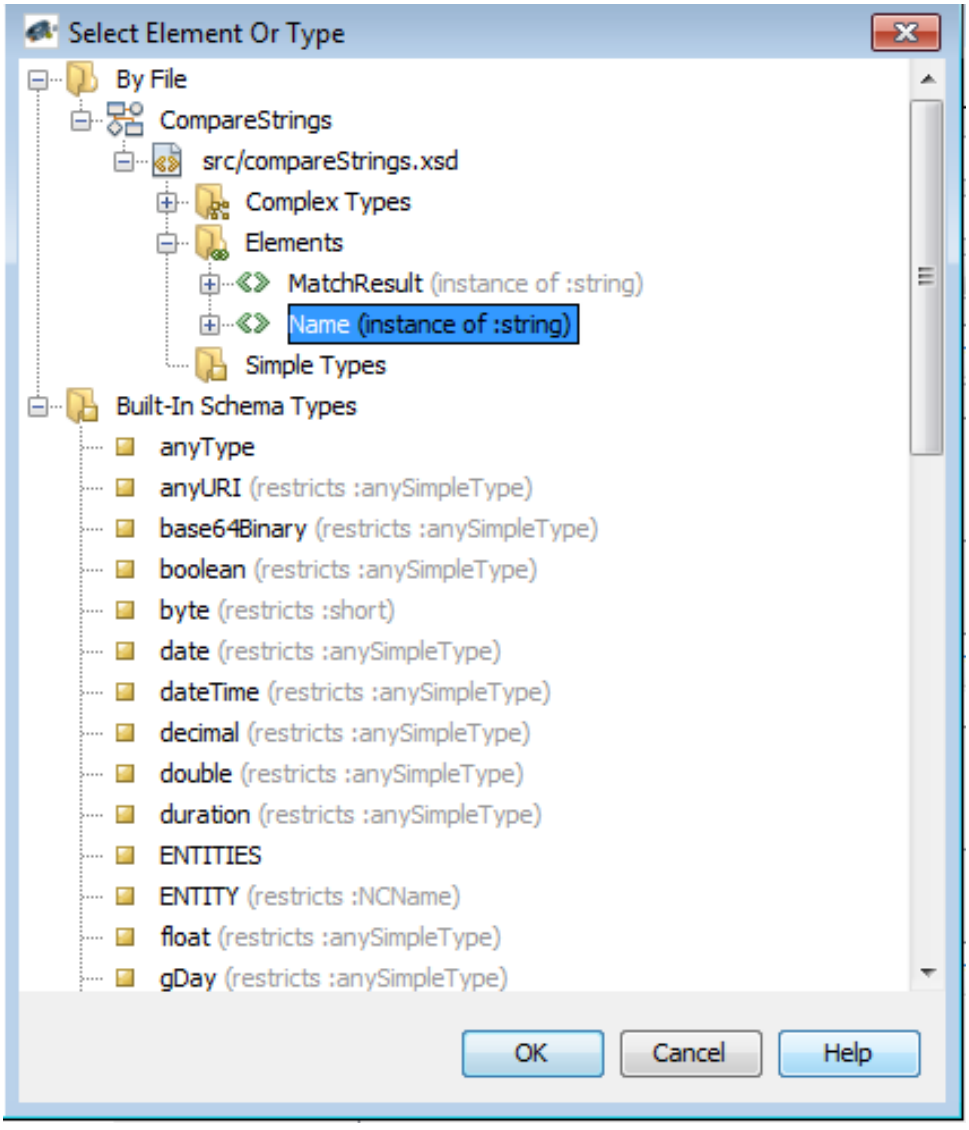
Ανοίγει η σελίδα Abstract Configuration.

- Σε αυτή τη σελίδα:

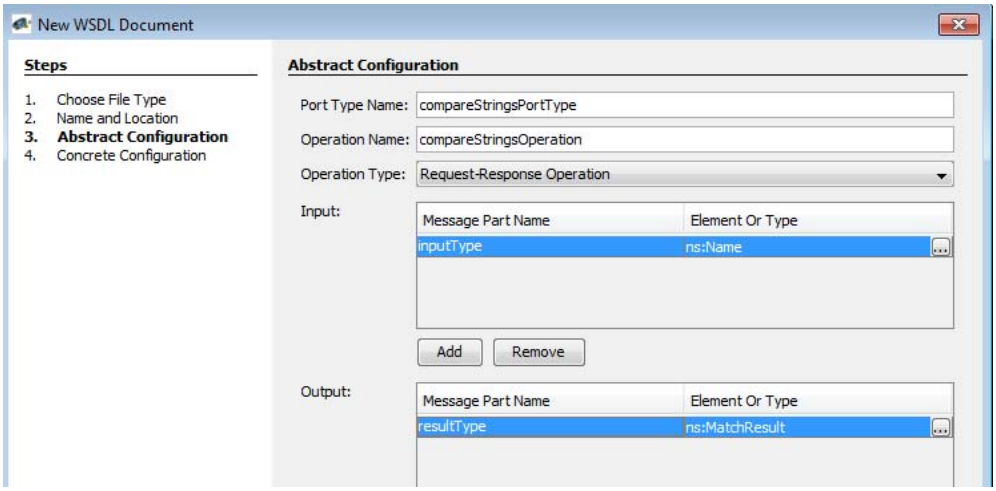
1. Στη στήλη Message Part Name, κάνουμε διπλό κλικ πάνω στο προεπιλεγμένο όνομα (`part1`) για να γίνει το πεδίο editable.
2. Δίνουμε σαν όνομα `inputType` και πατάμε Enter.
3. Στη στήλη Element Or Type, κάνουμε κλικ στο κουμπί της έλλειψης.
4. Επιλέγουμε `By File` > Elements > Name

- Στο πεδίο Output:

1. Στο τμήμα ονομασίας του Message Part Name, κάνουμε διπλό κλικ στην default τιμή (`part1`).
2. Δίνουμε σαν όνομα `resultType` και πατάμε Enter.
3. Στη στήλη Element Or Type, κάνουμε κλικ στο κουμπί της έλλειψης.
4. Επιλέγουμε `By File` > Elements > MatchResult



Εικόνα 10.18 *Επιλογή τύπου μεταβλητής μέσω του γραφικού περιβάλλοντος*



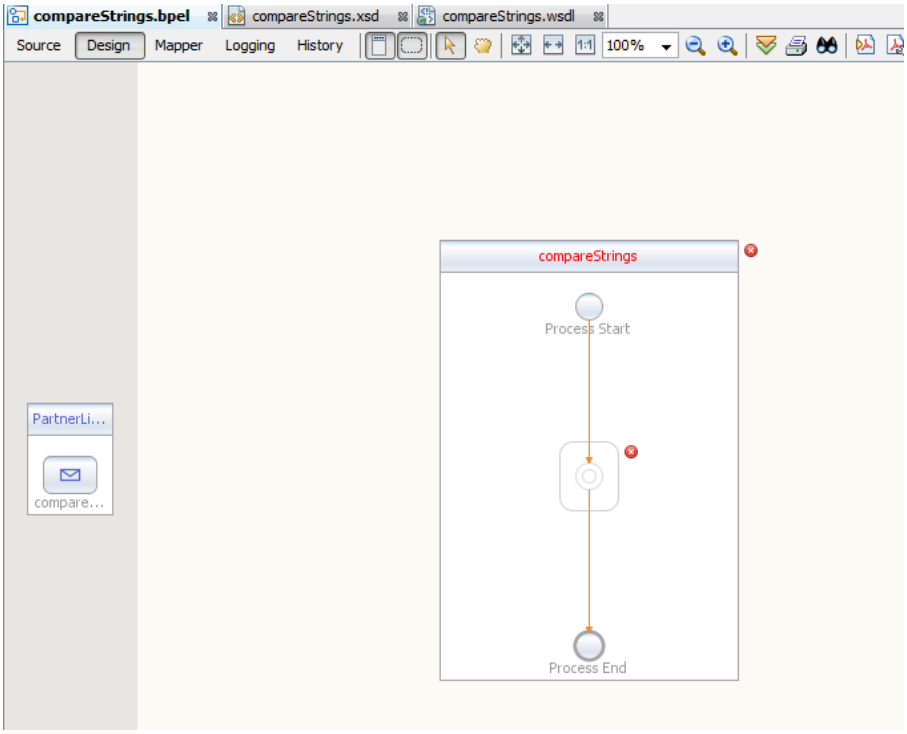
Εικόνα 10.19 *Η οθόνη Abstract Configuration*

- Πατάμε Next. Ανοίγει η σελίδα Concrete Configuration.
- Πατάμε Finish.

Στο Projects window, ο κόμβος `Process Files`, πλέον περιέχει έναν υποκόμβο με την ονομασία `compareStrings.wsdl`. Ο Source Editor περιέχει μια καρτέλα για το WSDL αρχείο, την `compareStrings.wsdl`

Προσθήκη ενός Partner Link στην BPEL διαδικασία

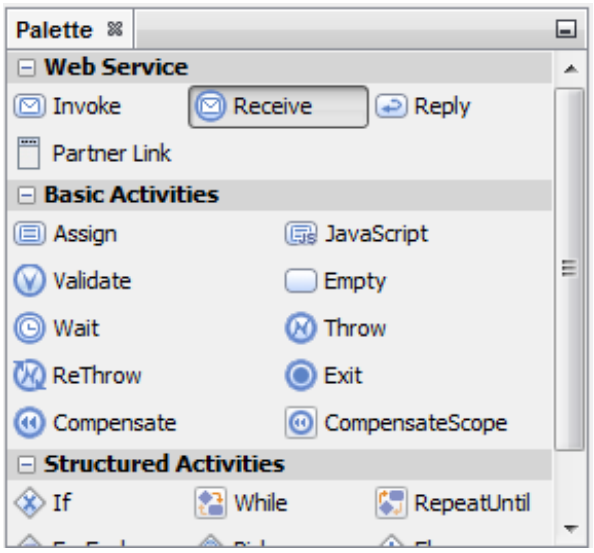
- Στο παράθυρο Projects, κάνουμε κλικ στον φάκελο Process Files για να φανούν τα περιεχόμενα του και επιλέγουμε το wsdl αρχείο (compareStrings.wsdl).
- Σύρουμε (drag n' drop) το αρχείο wsdl από το παράθυρο Projects, στην αριστερή πλευρά του καμβά του Design view. Ο BPEL Editor προσθέτει με αυτόν τον τρόπο ένα partner link στον καμβά.



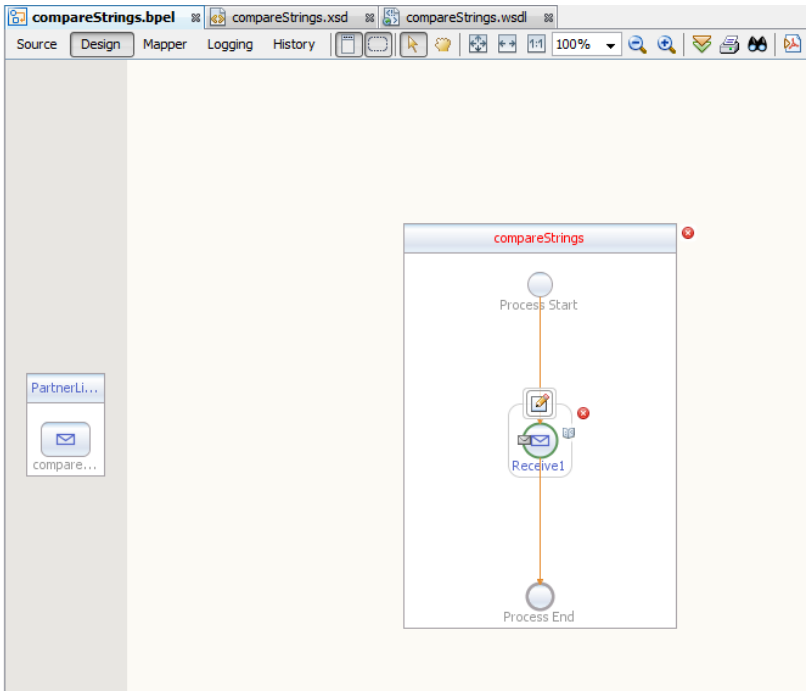
Εικόνα 10.20 *Αρχική οθόνη ενορχήστρωσης BPEL, με ένα Partner Link*

Προσθέστε μια δραστηριότητα Receive στο BPEL Process

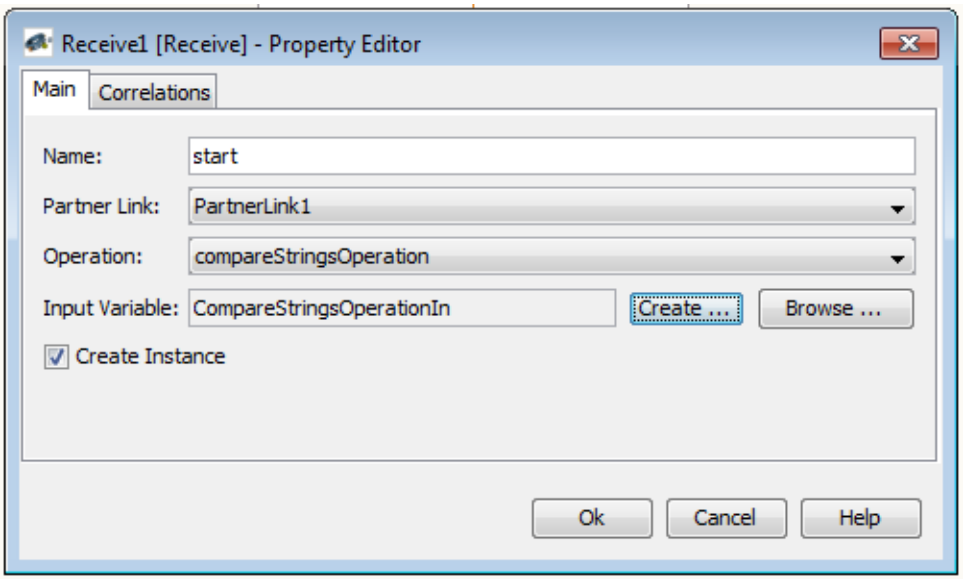
- Από το μενού Web Service του Palette window, επιλέγουμε τη δραστηριότητα Receive.
- Σύρουμε τη δραστηριότητα αυτή στο κουτί (process box) στον καμβά του Design view,ανάμεσα στο Process Start και το Process End.
- Κάνουμε κλικ στο εικονίδιο Edit του Receive1. Ανοίγει ο Property Editor.
- Στο tab Main, αλλάζουμε το όνομα σε start.
- Από το drop-down μενού του Partner Link, επιλέγουμε PartnerLink1. Το IDE προσθέτει στο πεδίο Operation το compareStringsOperation.
- Επιλέγουμε το κουμπί Create δίπλα στο Input Variable, έτσι ώστε να εμφανιστεί το μενού New Input Variable.Κάνουμε κλικ στο OK για να την αποδοχή των default τιμών.
- Κάνουμε κλικ στο OK ώστε να κλείσει ο Property Editor. Φαίνεται πλέον στο Design view η νέα σύνδεση ανάμεσα στο PartnerLink1 και τη δραστηριότητα Start.



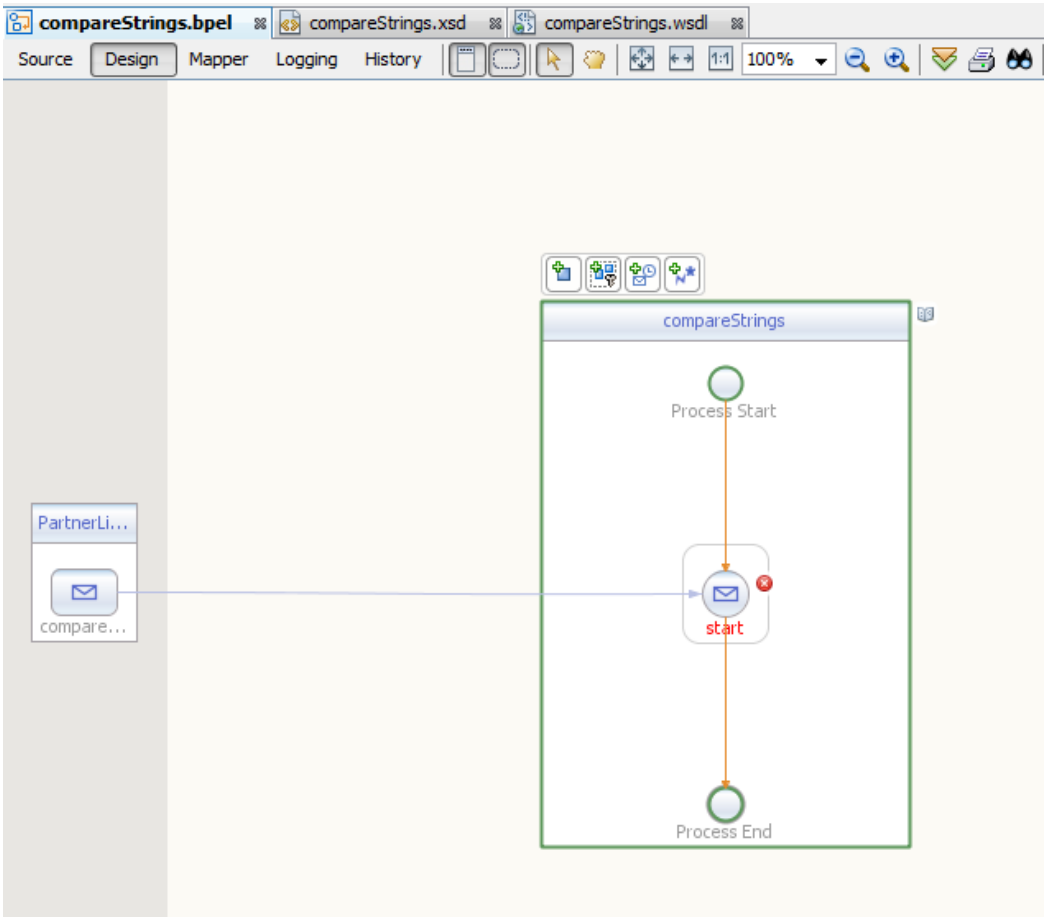
Εικόνα 10.21 *Το palette window, με επιλογές Web Service και Activities*



Εικόνα 10.22 *Εισαγωγή ενός receive activity*



Εικόνα 10.23 *Ρύθμιση του receive activity και ορισμός της μεταβλητής εισόδου*



Εικόνα 10.24 *Η παραμετροποιημένη receive activity με την ονομασία start*

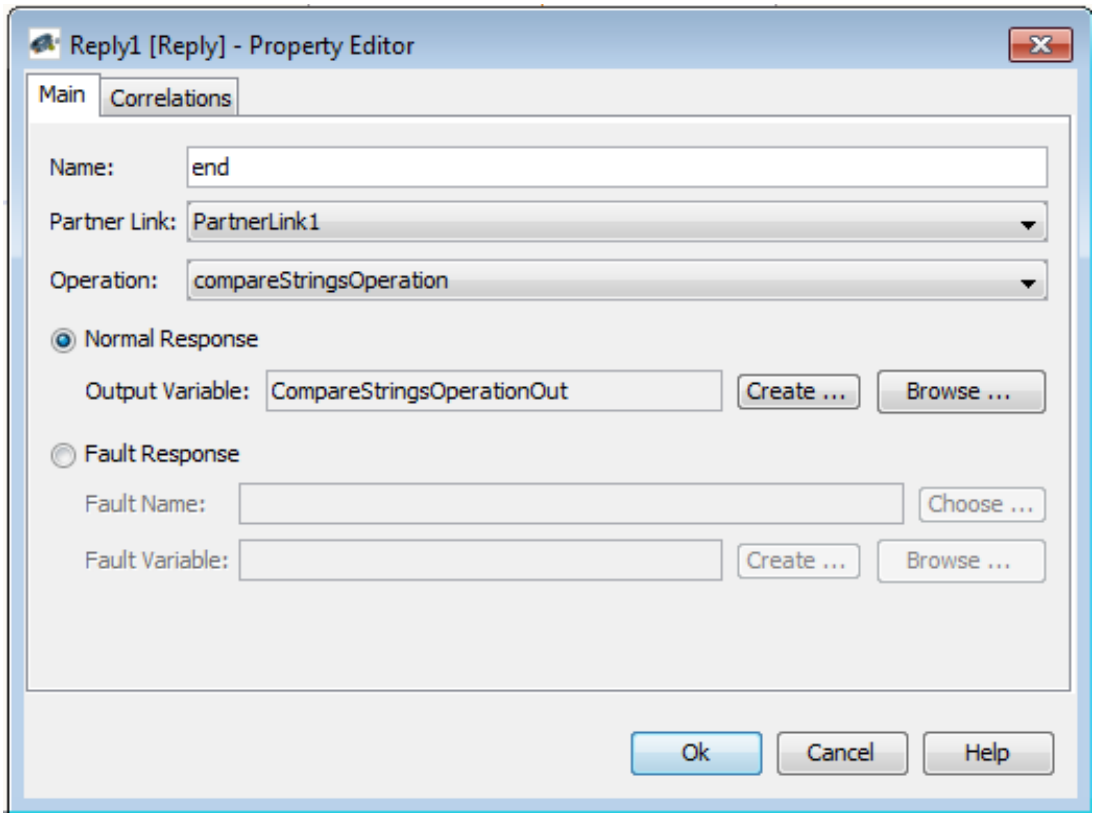
Προσθήκη μιας δραστηριότητας Reply στο BPEL Process

- Επιλέγουμε τη δραστηριότητα Reply στο τμήμα Web Service της παλέτας. Σύρουμε τη δραστηριότητα Reply ανάμεσα στο start και το Process End στον καμβά. Μια δραστηριότητα Reply1 εμφανίζεται στο design view.

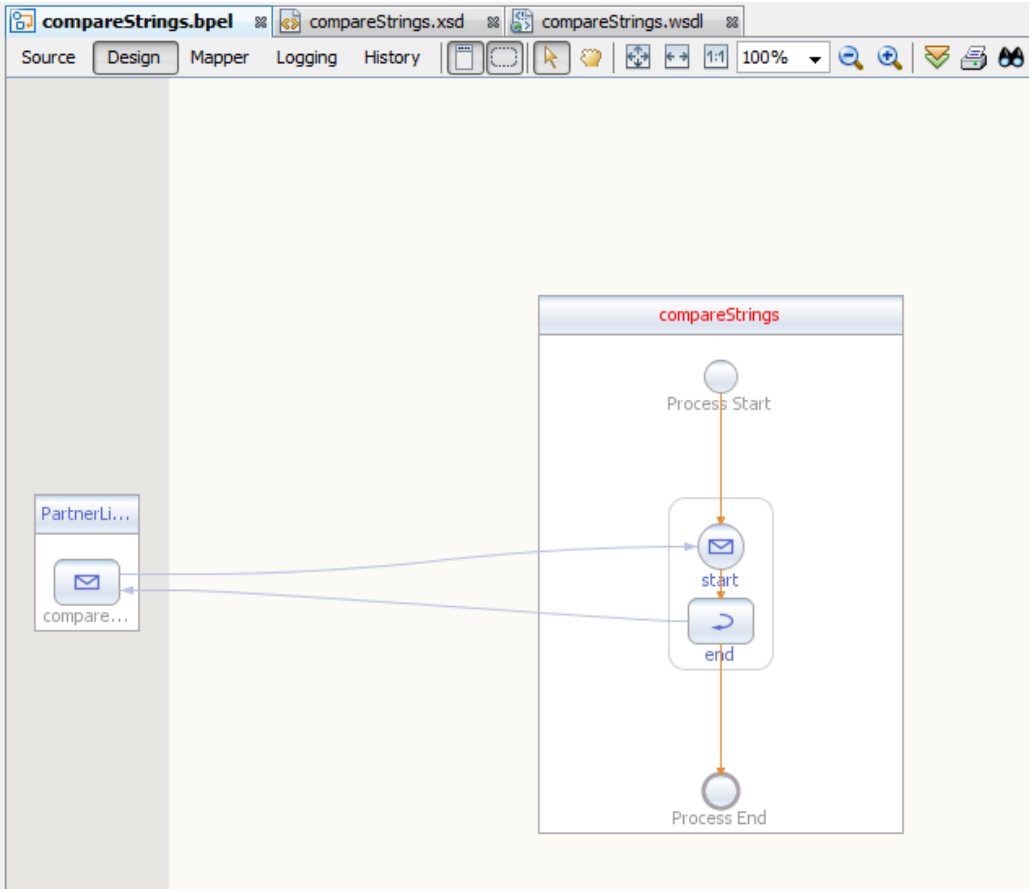
- Κάνουμε κλικ στο κουμπί edit του Reply1. Θα ανοίξει Ο Reply1 Property Editor.
- Στην καρτέλα Main, αλλάζουμε το όνομα σε end.
- Από την drop-down λίστα του Partner Link, επιλέγουμε PartnerLink1. Το IDE προσθέτει στο Operation field το compareStringsOperation.
- Για να δημιουργήσουμε μια νέα μεταβλητή τύπου output, επιλέγουμε Normal Response, και κάνουμε κλικ στο πλήκτρο Create δίπλα στο Input Variable πεδίο. Το μενού New Input Variable ανοίγει. Κάνουμε κλικ στο OK για αποδοχή των default τιμών.
- Κάνουμε κλικ στο OK για να κλείσουμε τον Reply1 Property Editor. Στο Design view πλέον φαίνεται η σύνδεση ανάμεσα στη δραστηριότητα end και το PartnerLink1.

Προσθήκη μιας δραστηριότητας If στο BPEL Process

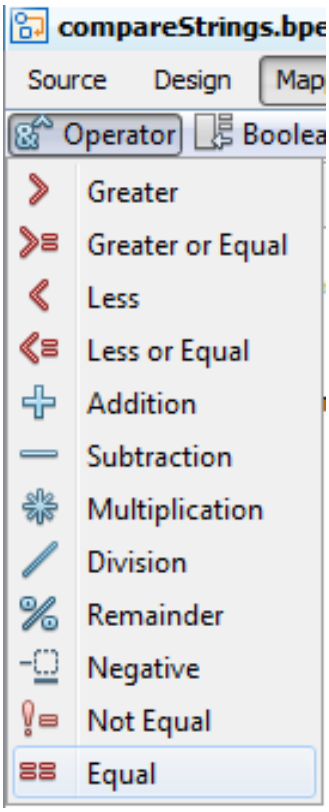
- Επιλέγουμε τη δραστηριότητα If στο Structured Activities τμήμα της παλέτας (Palette). Σύρουμε τη δραστηριότητα If ανάμεσα στο start και το end στον καμβά. Μια δραστηριότητα If1 εμφανίζεται στο design view.
- Κάντας διπλό κλικ στη δραστηριότητα If1, εμφανίζεται ο BPEL Mapper.
- Από τη γραμμή ενεργειών επιλέγουμε operator->Equal
- Στη συνέχεια από τη γραμμή ενεργειών επιλέγουμε String->String Literal
- Στο τμήμα String Literal που εμφανίζεται δίνουμε το string με το οποίο θα συγκρίνουμε την είσοδο της YI. Για τις ανάγκες αυτού του παραδείγματος δίνουμε “Georgiadis”.
- Στη συνέχεια κάνουμε τη σύνδεση των μεταβλητών, όπως φαίνεται στην εικόνα 10.29, με τη χρήση drag n’ drop.



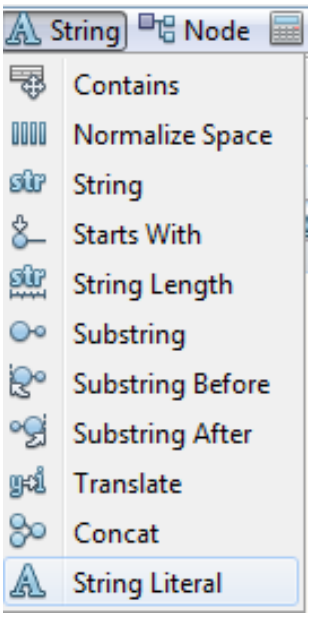
Εικόνα 10.25 *Παραμετροποίηση της reply activity*



Εικόνα 10.26 *Αλληλεπίδραση με το partner link και ανταλλαγή μηνυμάτων*



Εικόνα 10.27 *Λίστα διαθέσιμων operators*



Εικόνα 10.28 *Λίστα διαθέσιμων επιλογών που σχετίζονται με τον τύπο string*

Προσθήκη μιας δραστηριότητα Assign στο BPEL Process

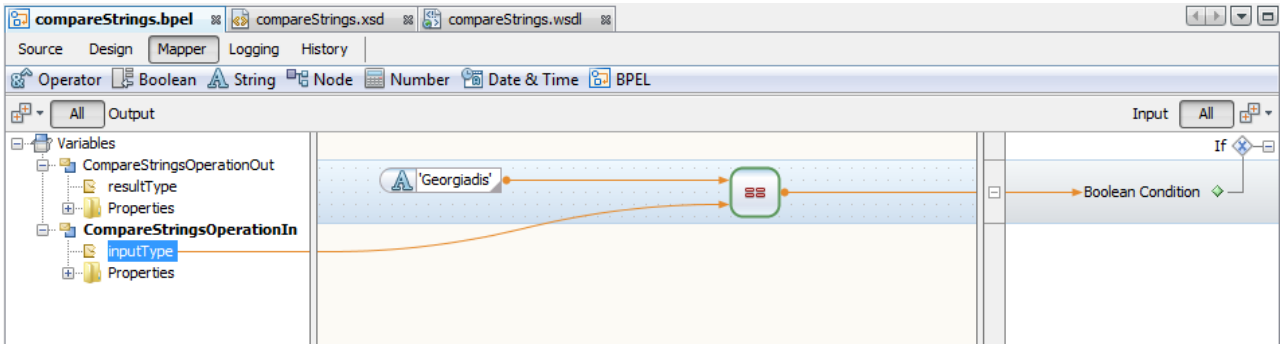
- Επιλέγουμε τη δραστηριότητα Assign στο Basic Activities τμήμα της παλέτας (Palette). Σύρουμε τη δραστηριότητα Assign στο εσωτερικό της If1, δηλαδή στην περίπτωση που ισχύει η συνθήκη. Μια δραστηριότητα Assign1 εμφανίζεται στο design view.
- Επιλέγουμε τη δραστηριότητα Assign1 και κάνουμε κλικ στο πλήκτρο Mapper στο editors toolbar. Εμφανίζεται ο BPEL Mapper.
- Από τη γραμμή ενεργειών επιλέγουμε String->String Literal
- Δίνουμε σαν τιμή "Strings Match" και τη συσχετίζουμε με την τιμή CompareStringsOperationOutresultType

Έτσι αντιγράφεται η τιμή του string στην έξοδο.

Στη συνέχεια θα προσθέσουμε ένα νέο assign activity για την περίπτωση που το string εισόδου δεν είναι το αναμενόμενο.

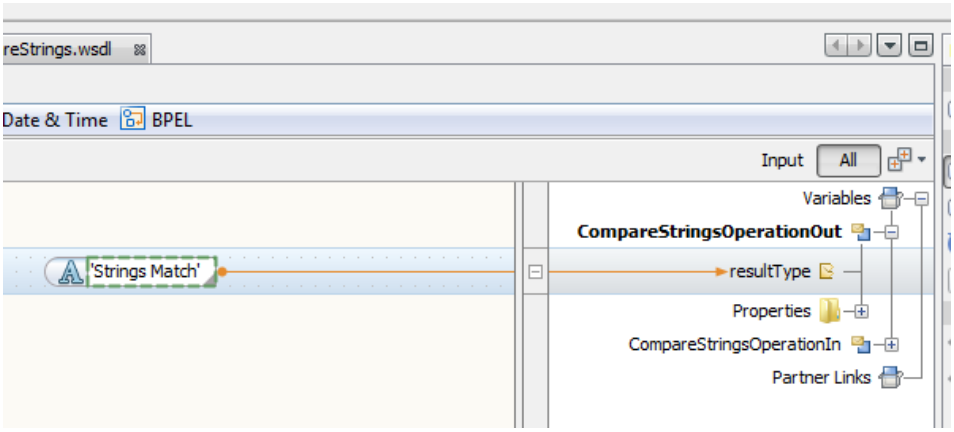
- Επιλέγουμε τη δραστηριότητα Assign στο Basic Activities τμήμα της παλέτας (Palette). Σύρουμε τη δραστηριότητα Assign στο δεξί τμήμα της If1, δηλαδή στην περίπτωση που δεν ισχύει η συνθήκη. Μια δραστηριότητα Assign2 εμφανίζεται στο design view.

Gif 10.1.gif	Κινούμενη εικόνα (interactive)
Σχήμα 10.29 Σύγκριση του string εισόδου με ένα προεπιλεγμένο string, μέσω του εργαλείου mapper	

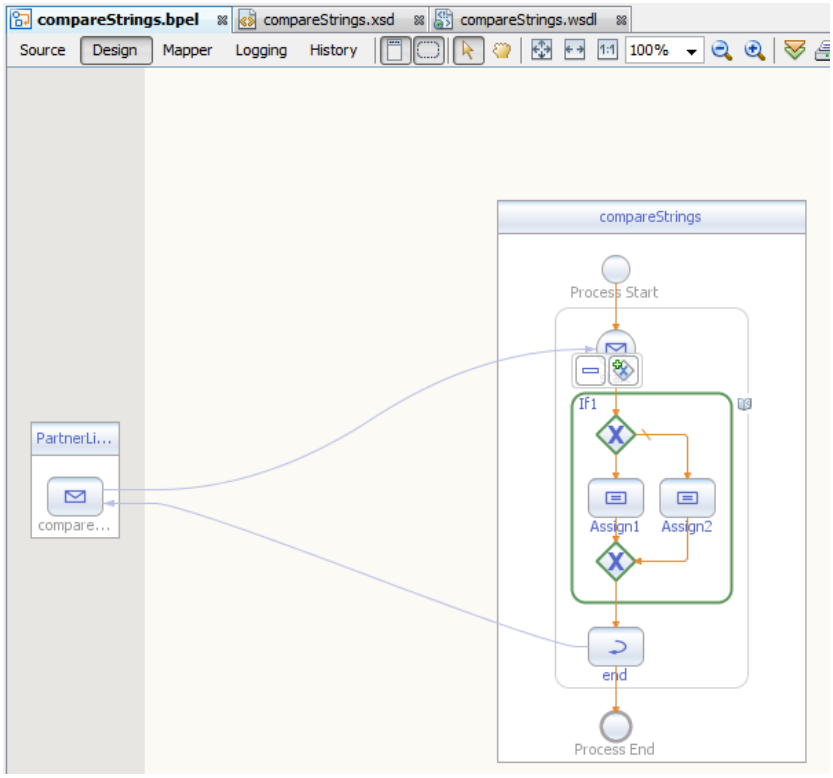


Εικόνα 10.29 *Σύγκριση του string εισόδου με ένα προεπιλεγμένο string, μέσω του εργαλείου mapper*

Gif 10.2.gif	Κινούμενη εικόνα (interactive)
Σχήμα 10.30 Εισαγωγή ενός string στη μεταβλητή εξόδου, μέσω του εργαλείου mapper	

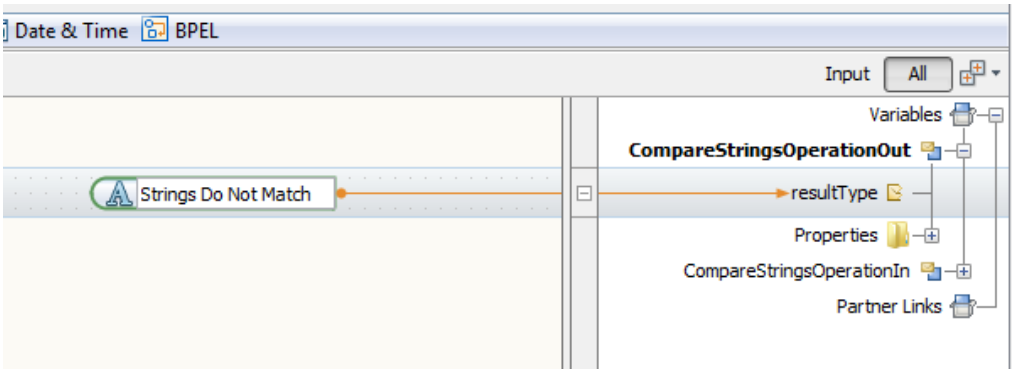


Εικόνα 10.30 *Εισαγωγή ενός string στη μεταβλητή εξόδου, μέσω του εργαλείου mapper*



Εικόνα 10.31 *Η ενορχήστρωση BPEL, μετά την εισαγωγή μιας If structured activity*

- Επιλέγουμε τη δραστηριότητα Assign1 και κάνουμε κλικ στο πλήκτρο Mapper στο editors toolbar. Εμφανίζεται ο BPEL Mapper.
- Από τη γραμμή ενεργειών επιλέγουμε String->String Literal
- Δίνουμε σαν τιμή "Strings Do Not Match" και τη συσχετίζουμε με την τιμή CompareStringsOperationOutresultType



Εικόνα 10.32 *Εισαγωγή ενός string στη μεταβλητή εξόδου, μέσω του εργαλείου mapper*

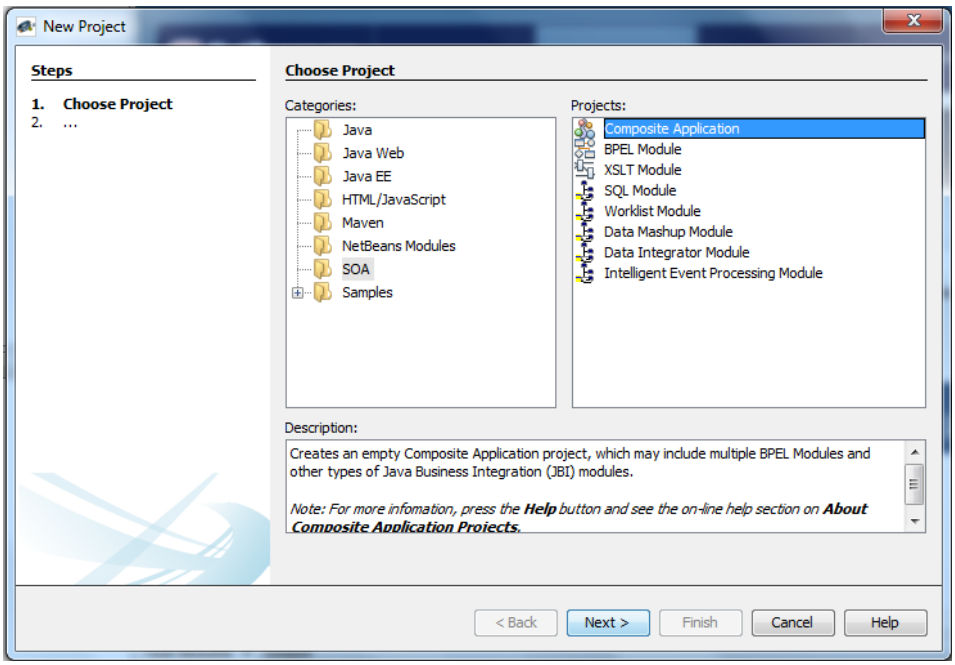
Κάνουμε κλικ στο εικονίδιο Save All στο κεντρικό μενού του IDE.

Δημιουργία ενός Composite Application Project

Ένα BPEL Module project δεν μπορεί να γίνει άμεσα deploy. Πρέπει πρώτα να προσθέσουμε το BPEL Module project, ως ένα JBI module, σε ένα σύνθετο project (Composite Application project). Στη συνέχεια μπορούμε να κάνουμε deploy το σύνθετο project. Κάνοντας Deploy το project κάνουμε την υπηρεσία διαθέσιμη και επιτρέπουμε στα συστατικά της μέρη να τρέξουν.

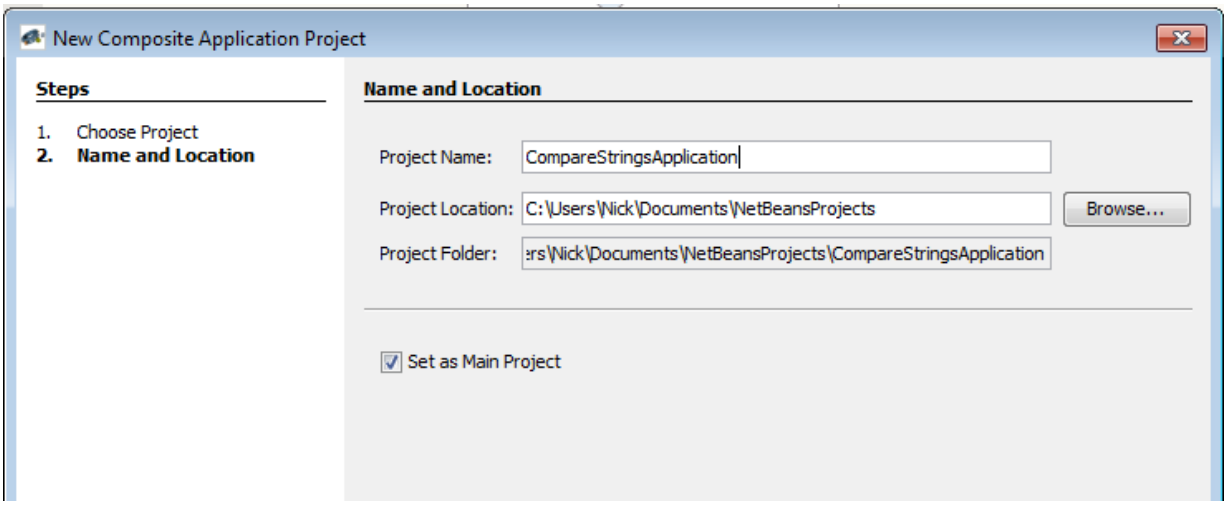
Δημιουργία ενός νέου σύνθετου project (Composite Application Project)

- Επιλέγουμε File → New Project (ή εναλλακτικά πατάμε Ctrl-Shift-N).
- Στη λίστα Categories επιλέγουμε Service Oriented Architecture, στη λίστα Projects επιλέγουμε Composite Application, και πατάμε Next.



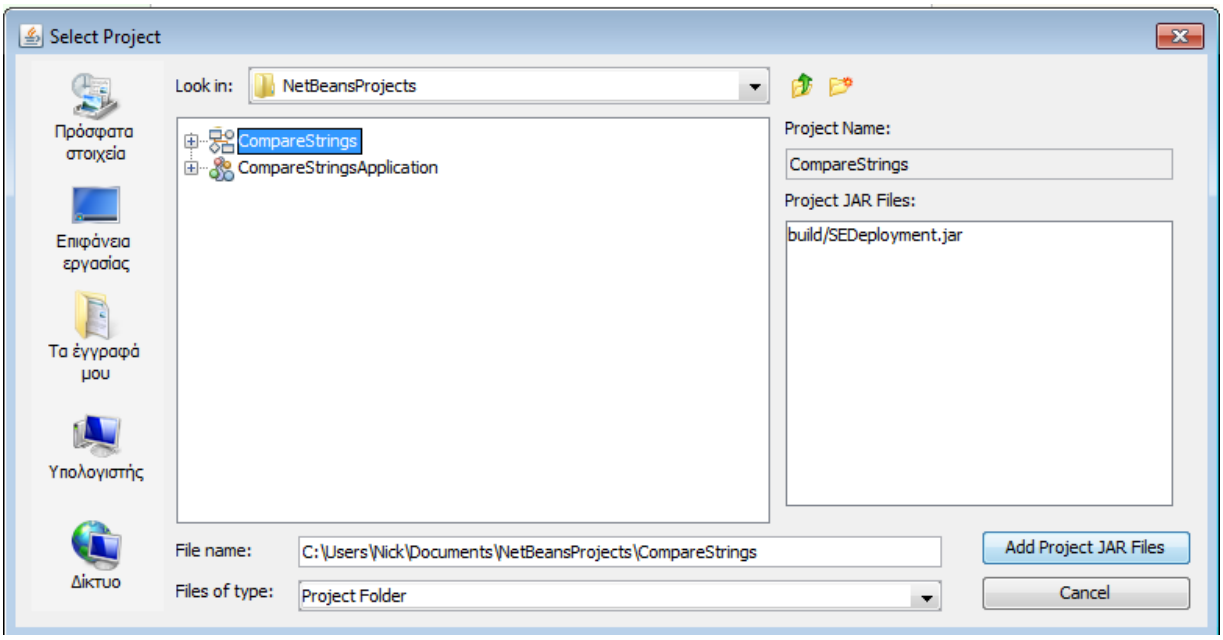
Εικόνα 10.33 *Δημιουργία Composite Application*

- Στη σελίδα Name and Location, αλλάζουμε το όνομα του project σε CompareStringsApplication.



Εικόνα 10.34 *Ορισμός των τιμών Name και Project Location*

- Αφήνουμε επιλεγμένο το Set as Main Project checkbox και πατάμε Finish.
- Για να προσθέσουμε το BPEL Module σαν JBI module στο σύνθετο project, κάνουμε δεξί κλικ στο Composite Application και επιλέγουμε Add JBI Module. Έτσι ανοίγει το παράθυρο Select Project.



Εικόνα 10.35 *Εισαγωγή του BPEL module, στο composite application ως JBI module*

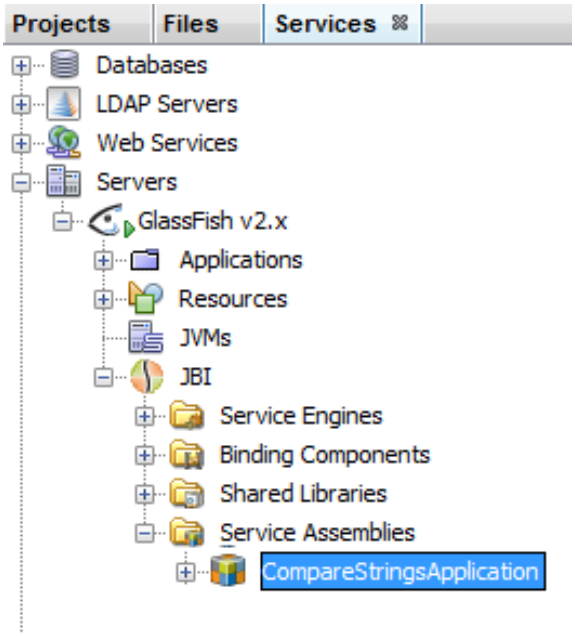
- Επιλέγουμε το CompareStrings project που δημιουργήσαμε νωρίτερα και κάνουμε κλικ στο Add Project JAR Files. Έτσι κλείνει το παράθυρο Select Project και το αρχείο compareStrings.jar προστίθεται στον κόμβο JBI Modules του CompareStringsApplication Composite Application.

Πως γίνεται το Build και Deploy του σύνθετου project (Composite Application Project)

Κάνοντας Build ένα project γίνεται αυτόματα μεταγλώττιση (compile) του αρχείου BPEL και γίνεται προσθήκη όλων των αρχείων (BPEL,WSDL,XSD) σε ένα JAR archive.
Κάνοντας Deploy ένα project γίνονται όλα τα παραπάνω και επιπρόσθετα στέλνονται τα αρχεία για εκτέλεση στον Application Server.

Build και Deploy

- Κάνουμε δεξί κλικ στον κόμβο Composite Application project και επιλέγουμε Build. Όταν ολοκληρωθεί αυτή η διαδικασία θα εμφανιστεί στο παράθυρο Output το μήνυμα "Build Successful."
- Κάνουμε δεξί κλικ στον ίδιο κόμβο και πατάμε Deploy.
- Επιλέγουμε το tab Services και ανοίγουμε (expand) την επιλογή Servers → GlassFish V2 → JBI → Service Assemblies για να δούμε σχηματικά το αποτέλεσμα του deploy.



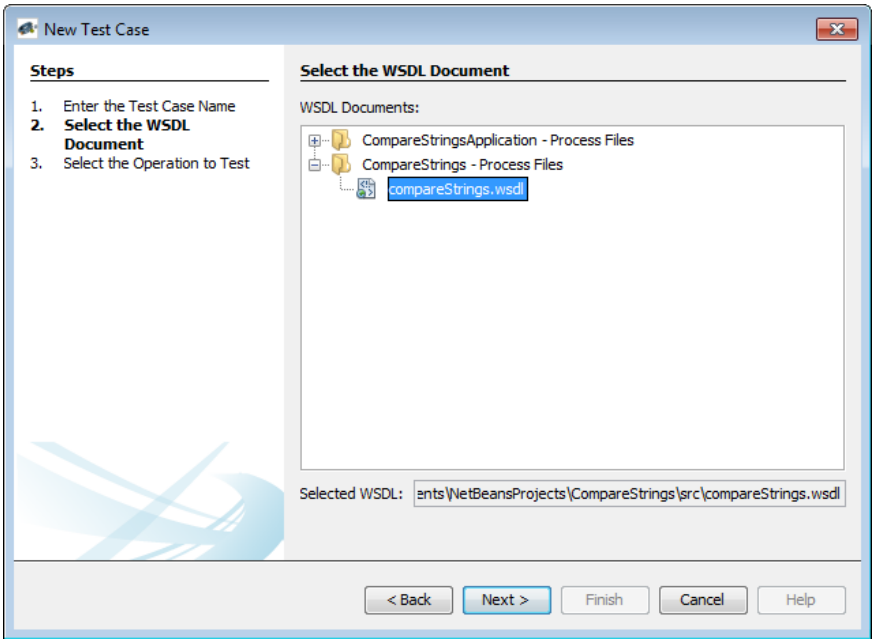
Εικόνα 10.36 *Η εμφάνιση της εφαρμογής στην καρτέλα services, ως αποτέλεσμα του deploy αυτής*

Δοκιμάζοντας το Composite Application

Μπορούμε να δοκιμάσουμε το Composite Application project, προσθέτοντας σενάρια (test cases).

Δοκιμή του CompareStringsApplication Composite Application Project

- Στο παράθυρο Projects, ανοίγουμε (expand) τον κόμβο CompareStringsApplication, κάνουμε δεξί κλικ στον κόμβο Test, και επιλέγουμε New Test Case.
- Αφήνουμε το προκαθορισμένο όνομα (TestCase1), και πατάμε Next.
- Από τη σελίδα Select the WSDL Document, κάνουμε expand τον κόμβο CompareStrings-Process Files, επιλέγουμε CompareStrings.wsdl, και πατάμε Next.
- Από τη σελίδα Select the Operation to Test, επιλέγουμε CompareStringsOperation και πατάμε Finish.



Εικόνα 10.37 *Επιλογή WSDL αρχείου για τη διενέργεια test*

Ένας κόμβος TestCase1 προστίθεται κάτω από τον κόμβο Test, που περιέχει δύο υποκόμβους, τον Input και Output.

Εμφανίζεται ο Source Editor που περιέχει το αρχείο Input.xml

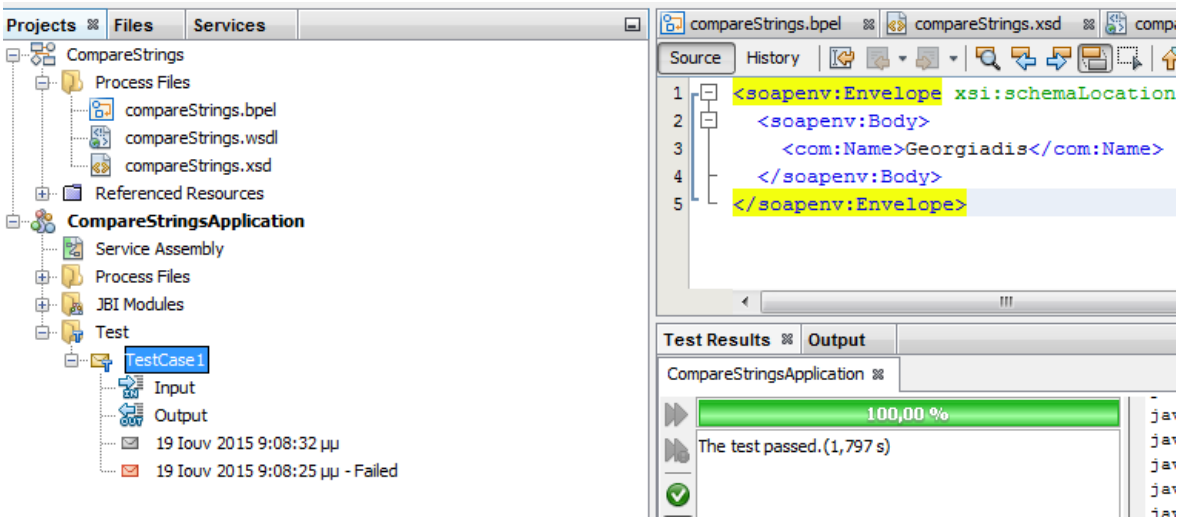
- Στην καρτέλα Input.xml του Source Editor, κάνουμε τα ακόλουθα:

1. Εντοπίζουμε τη γραμμή:
   <com:Name>?string?</com:Name>
2. Αντικαθιστούμε το ?string? με τη λέξη Georgiadis, έτσι ώστε να πάρει την ακόλουθη μορφή:
   <com:Name>Georgiadis </com:Name>

- Από τη menu bar, κάνουμε κλικ στην επιλογή Save All.
- Στο παράθυρο Projects, κάνουμε διπλό κλικ στον κόμβο Output κάτω από το Test → TestCase1.
  Ανοίγει το Output.xml στον Source Editor. Αρχικά το αρχείο είναι άδειο, μέχρι το πρώτο τεστ να το γεμίσει με δεδομένα.
- Στο παράθυρο Projects, κάνουμε δεξί κλικ στον κόμβο TestCase1 και επιλέγουμε Run. Θα εμφανιστεί ένα παράθυρο που θα ρωτάει αν επιθυμούμε να κάνουμε Overwrite, πατάμε Yes. Το πρώτο τεστ γεμίζει στοιχεία το αρχείο Output.xml.

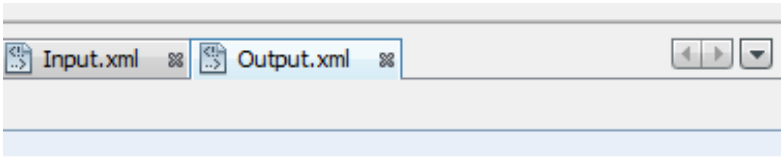
Sound 10.3.mp3	Ηχητικό απόσπασμα (audio)
Επεξήγηση των TestCases στο περιβάλλον NetBeans	

- Κάθε φορά το τεστ συγκρίνει τα αποτελέσματα με το περιεχόμενο του output file. Επειδή την πρώτη φορά το αρχείο αυτό είναι άδειο το πρώτο τεστ θα βγάλει αποτυχία. Τρέξτε κι άλλα τεστ το οποία θα πρέπει να βγάλουν σαν μήνυμα την επιτυχία.



Εικόνα 10.38 *Αποτελέσματα του TestCase1*

- Ανοίγοντας το αρχείο Output.xml θα δούμε πως αφού η είσοδος ήταν η αναμενόμενη, ως έξοδος περάστηκε το string “Strings Match”.

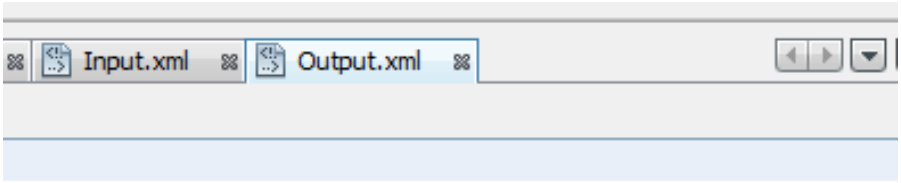



```
rc/compareStrings">Strings Match</MatchResult>
```

Εικόνα 10.39 *Το επιστρεφόμενο μήνυμα «Strings Match»*

- Δημιουργούμε ένα δεύτερο Test (το ονομάζουμε TestCase2) ακολουθώντας την ίδια διαδικασία.
- Αυτή τη φορά στην καρτέλα Input.xml του Source Editor, εντοπίζουμε τη γραμμή:
    - <com:Name>?string?</com:Name>
    - και αντικαθιστούμε το ?string? με τη λέξη Papadopoulos, έτσι ώστε να πάρει την ακόλουθη μορφή:
    - <com:Name>Papadopoulos </com:Name>

- Από την menu bar, κάνουμε κλικ στην επιλογή Save All.
- Στο παράθυρο Projects, κάνουμε δεξί κλικ στον κόμβο TestCase2 και επιλέγουμε Run, δύο φορές ώστε να γίνει σύγκριση των αποτελεσμάτων της δεύτερης φοράς με την πρώτη.
- Θα δούμε πάλι πως το τεστ θα δώσει επιτυχία. Αυτή τη φορά όμως ανοίγοντας το αρχείο Output.xml θα δούμε πως θα έχει περαστεί ως απάντηση το string “Strings Do Not Match”.



Εικόνα 10.40 *Το επιστρεφόμενο μήνυμα «Strings Do Not Match»*

5. Επιλογή YI με χρήση τεχνικών πολυκριτήριας ανάλυσης αποφάσεων

Για την επιλογή και σύνθεση YI, σε περιπτώσεις όπου υφίσταται πληθώρα υπηρεσιών με παρόμοια λειτουργικότητα (οι οποίες όμως διαφέρουν στα ποιοτικά χαρακτηριστικά τους), έχουν προταθεί πολλές μέθοδοι στη βιβλιογραφία. Στην πηγή (Vesyropoulos & Georgiadis, 2015) έχει παρουσιαστεί μια μέθοδος επιλογής με βάση τη μεθοδολογία Analytical Hierarchy Process (AHP), η οποία ανήκει στην κατηγορία των πολυκριτηριακών μεθόδων.

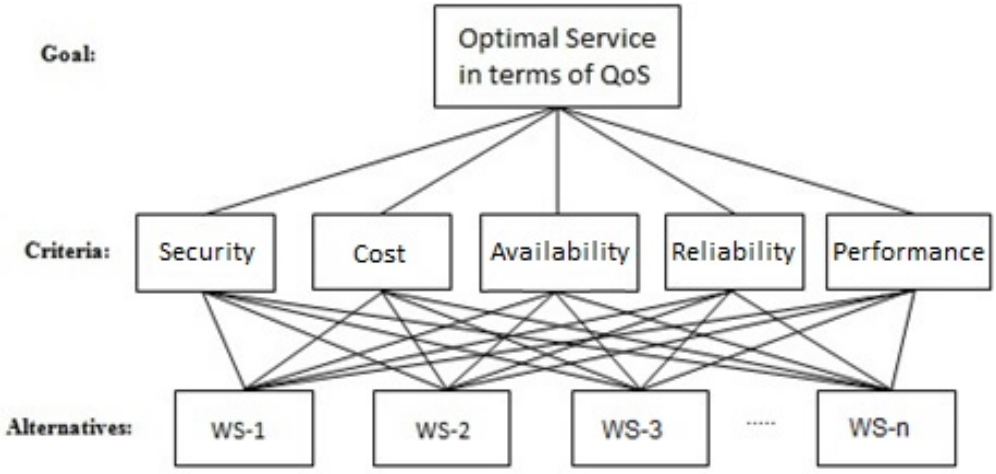
Η AHP έχει προταθεί το 1980 (Saaty, 1980), και έχει εφαρμοσθεί σε μια πληθώρα πεδίων όπως είναι η επιχειρησιακή έρευνα, η διαχείριση πόρων ενέργειας, η τραπεζική και άλλες. Μέσω ενός ιεραρχικού μοντέλου, παρέχει βοήθεια κατά την επιλογή μεταξύ πλήθους εναλλακτικών, με βάση κάποια προκαθορισμένα κριτήρια. Η μέθοδος μπορεί να χειριστεί τόσο ποιοτικές όσο και ποσοτικές τιμές. Για την εφαρμογή της μεθόδου ακολουθούνται κάποια συγκεκριμένα βήματα:

1. Σχεδιασμός της ιεραρχίας του προβλήματος όπως φαίνεται στο σχήμα 10.2.
2. Θέσπιση της προτεραιότητας των κριτηρίων με την εισαγωγή συγκρίσεων ανά ζεύγη, κάνοντας χρήση της κλίμακας από το 1 ως το 9. Με τον τρόπο αυτό φαίνονται τα βάρη που ορίζει ο χρήστης για κάθε κριτήριο, δηλαδή το πόσο σημαντικό το θεωρεί σε σχέση με τα υπόλοιπα.
3. Βαθμολόγηση των εναλλακτικών ως προς το κάθε κριτήριο, με χρήση συγκρίσεων ανά ζεύγη
4. Υπολογισμός της συνολικής βαθμολογίας για κάθε εναλλακτική και ταξινόμηση τους, ξεκινώντας από την πλησιέστερη προς το επιθυμητό αποτέλεσμα, και με φθίνουσα σειρά.

Έτσι λοιπόν γίνεται φανερό πως η AHP λαμβάνει ως εισόδους τιμές για διαφορετικά κριτήρια και έχει τη δυνατότητα υπολογισμού μιας συνολικής τιμής που αντιπροσωπεύει τη συνολική βαθμολογία κάθε εναλλακτικής επιλογής.

Όπως είναι φυσικό, η μεθοδολογία μπορεί να χρησιμοποιηθεί και για την επιλογή της κατάλληλης YI, μέσω ποιοτικών κριτηρίων, η οποία θα αποτελέσει τμήμα μιας σύνθεσης YI. Το μεγαλύτερο πλεονέκτημα που προκύπτει από την υιοθέτηση της μεθοδολογίας αυτής από μια μηχανή σύνθεσης, είναι η παροχή προσωποποιημένων συνθέσεων σε μεγαλύτερο βαθμό από άλλες μεθόδους καθώς μέσω των διαφόρων κριτηρίων η σύνθεση ανταποκρίνεται πλήρως στις ανάγκες του τελικού χρήστη ή της επιχείρησης. Είναι παράλληλα δυνατή η επιτάχυνση της διαδικασίας σύνθεσης, ειδικότερα σε συστήματα που θα ενσωματώνουν παράλληλα και αλγόριθμους τεχνητής νοημοσύνης, όπου επιτρέπεται η αυτόματη συμπλήρωση κάποιων συγκρίσεων από τη μηχανή σύνθεσης, καθώς αυτή θα «μαθαίνει» τις προτιμήσεις κάθε χρήστη.

Παρακάτω θα παρουσιαστεί η εφαρμογή της μεθοδολογίας για την επιλογή YI, με βάση τα κριτήρια αλλά και τα βάρη που εισάγει ο τελικός χρήστης. Τελικό στόχο αποτελεί η βέλτιστη σύγκλιση YI, με βάση τα χαρακτηριστικά ποιότητας. Επιπρόσθετα στη συγκεκριμένη εφαρμογή λαμβάνονται υπόψιν και εξωτερικές αξιολογήσεις YI που προκύπτουν από προηγούμενους χρήστες των YI, οι οποίοι βαθμολογούν την εμπειρία τους από τη χρήση των YI.



Σχήμα 10.2 *Η ιεραρχική δομή στην AHP, σε ένα παράδειγμα σύνθεσης βάσει ποιοτικών χαρακτηριστικών*

5.1. Εξατομικευμένη επιλογή και σύνθεση με βάση κριτήρια ποιότητας παρεχόμενων υπηρεσιών

Η συγκεκριμένη μεθοδολογία έχει βασιστεί σε μεγάλο βαθμό στις ΥΙ τύπου REST, τόσο λόγω της μεγάλης απήχησης που λαμβάνουν τα τελευταία χρόνια (Pautasso, 2014), όσο και λόγω του γεγονότος πως για την πρόσβαση σε πληροφορίες που προκύπτουν από «έξυπνες» συσκευές, μέσω κλήσεων HTTP, είναι επιβεβλημένη η χρήση τους (Want et al, 2015; Gubbi et al, 2013; Guinard et al, 2011). Καθώς όμως οι ΥΙ τύπου SOAP προσφέρουν οφέλη σε επίπεδο χειρισμού επιχειρηματικών κανόνων, και καθώς είναι δυνατή η ενσωμάτωση τους σε Web 2.0 mashups (βλ. http://www.ibm.com/developerworks/xml/library/x-mashups.html), η μεθοδολογία δεν αποκλείει τη χρήση και ΥΙ αυτού του τύπου.

Παράλληλα, σύμφωνα με τις αρχές του κοινωνικού ιστού (social web), και καθώς τα σχόλια και γενικότερα η ανάδραση χρηστών σχετικά με προϊόντα και υπηρεσίες είναι πλέον σήμερα περισσότερο προσβάσιμα από ποτέ, ο χρήστης και η μηχανή σύνθεσης θα πρέπει να μπορούν να λάβουν υπόψη αξιολογήσεις προηγούμενων χρηστών.

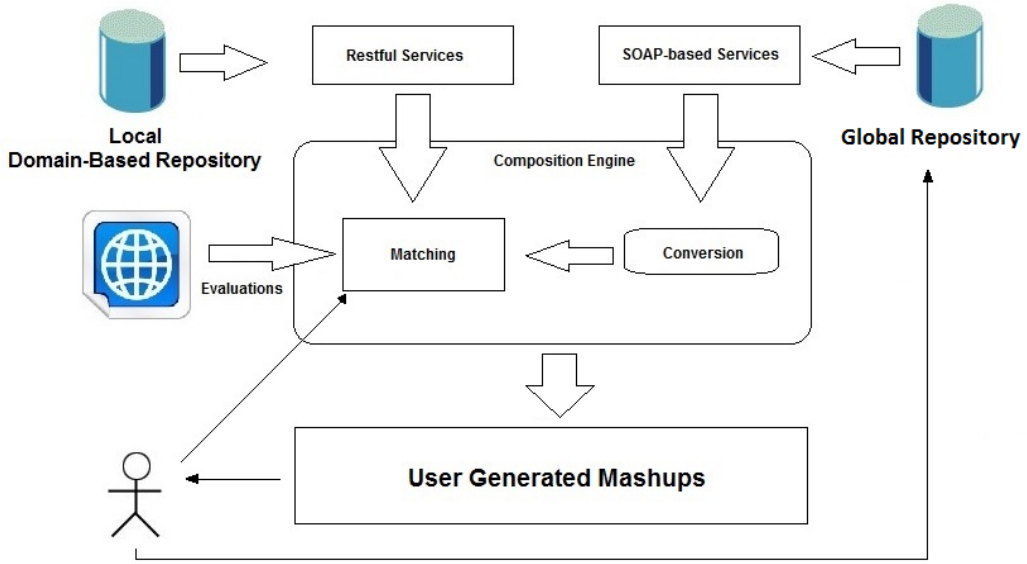
Μοντέλο συστήματος

Το προτεινόμενο μοντέλο αποτελείται από τις παρακάτω μεταβλητές:

1. UsQoS – Μια λίστα με τα αρχικά κριτήρια QoS που έχει ζητήσει ο χρήστης.
2. ProvQoS – Μια λίστα τιμών για συγκεκριμένα κριτήρια QoS, όπως π.χ. ο βαθμός ιδιωτικότητας, που παρέχεται από τους παρόχους των ΥΙ. Αυτές είναι προκαθορισμένες, σταθερές τιμές και αποτελούν ένα είδος αυτο-αξιολόγησης από την πλευρά των παρόχων.
3. EvQoS – Μια λίστα από τιμές για συγκεκριμένα κριτήρια QoS, που προσφέρονται από αξιολογήσεις προηγούμενων χρηστών.
4. UserWeights – Μια λίστα από βάρη, με τα οποία ο χρήστης καθορίζει τον βαθμό σημαντικότητας που δίνει σε διάφορα κριτήρια QoS και που βοηθούν στη λήψη εξατομικευμένων απαντήσεων από τη μηχανή σύνθεσης.

Όπως θα αναλυθεί παρακάτω η επιλογή των βαρών των κριτηρίων αποτελεί μια διαδικασία 2 βημάτων. Αρχικά, ο χρήστης επιλέγει ένα σύνολο λειτουργικών χαρακτηριστικών που επιθυμεί αλλά και ένα σύνολο χαρακτηριστικών ποιότητας, για τα οποία καθορίζει και τον βαθμό σημαντικότητας (βάρη).

Ως αποτέλεσμα η μηχανή σύνθεσης επιστρέφει ένα σύνολο από ΥΙ, οι οποίες ικανοποιούν τα απαιτούμενα λειτουργικά χαρακτηριστικά, ενώ μπορεί παράλληλα να ικανοποιούν μερικά ή όλα τα απαιτούμενα χαρακτηριστικά ποιότητας. Επιπρόσθετα, ενδέχεται να προσφέρουν και άλλα χαρακτηριστικά ποιότητας για τα οποία δεν έχει ορίσει βάρη ο χρήστης αλλά ενδεχομένως να αποτελούν χαρακτηριστικά που να τον ενδιαφέρουν. Σε αυτό το βήμα ο χρήστης έχει τη δυνατότητα να δώσει βάρη σε αυτά τα νέα κριτήρια και να κάνει αναπροσαρμογή των προηγούμενων βαρών.



Σχήμα 10.3 *Το μοντέλο σύνθεσης εξατομικευμένων υπηρεσιών*

Εφαρμογή

Η μεθοδολογία αποτελείται από συγκεκριμένες φάσεις. Η πρώτη φάση αφορά την εισαγωγή των λειτουργικών χαρακτηριστικών. Οι επόμενες δύο φάσεις αφορούν τον ορισμό των βαρών, ενώ οι τελευταίες φάσεις αφορούν την εφαρμογή της AHP μεθόδου.

Εδώ θα πρέπει να αναφερθεί πως ενώ η AHP δέχεται μόνο αριθμητικές τιμές για τις αξιολογήσεις των κριτηρίων, είναι δυνατή και η χρήση μη-αριθμητικών τιμών οι οποίες μπορούν να μετατραπούν σε αριθμητικές, μέσω κανονικοποίησης.

Το χαρακτηριστικό αυτό επιτρέπει μια πιο φιλική προς τον χρήστη προσέγγιση καθώς οι πιθανές επιλογές μπορούν να αποτελούν μέρος μιας κλίμακας Likert.

Φάση πρώτη:

Ο τελικός χρήστης ορίζει τα λειτουργικά χαρακτηριστικά που επιθυμεί να υφίστανται από την τελική σύνθεση. Η μηχανή σύνθεσης επιστρέφει ένα πλήθος από YI που ικανοποιούν τις λειτουργικές αυτές ανάγκες του χρήστη.

Φάση δεύτερη:

Ο χρήστης δίνει ως είσοδο ένα πλήθος από χαρακτηριστικά ποιότητας υπηρεσιών (UsQoS), μέσα από μια προκαθορισμένη λίστα χαρακτηριστικών QoSi (όπου i=1, 2…n) και τα οποία είναι καταχωρημένα στο μητρώο. Ο χρήστης επιπρόσθετα δίνει τις κατά-ζεύγη συγκρίσεις που προσδιορίζουν τα βάρη (UserWeights) των κριτηρίων. Έτσι για παράδειγμα για UsQoSi, όπου i = 3 ο χρήστης μπορεί να δώσει τις παρακάτω επιλογές:

{(UsQoS1, UsQoS2, PwC12), (UsQoS2, UsQoS3, PwC23), (UsQoS1, UsQoS3, PwC13)}, όπου τα πεδία UsQoS(a) και UsQoS(b) αντιστοιχούν στις ονομασίες παραμέτρων ποιότητας, ενώ το πεδίο PwC(ab) αντιστοιχεί σε μια τιμή που προσδιορίζει την προτίμηση του χρήστη στην παράμετρο ποιότητας UsQoSa σε σχέση με την παράμετρο UsQoSb.

Με βάση τις τιμές των βαρών αλλά και των αρχών της AHP μεθοδολογίας, ένας αρχικός πίνακας Eigenvector υπολογίζεται, ο οποίος φανερώνει ποια είναι τα κριτήρια τα οποία είναι πιο σημαντικά για τον χρήστη.

Επιπρόσθετα, ο χρήστης επιλέγει το πόσο αυστηρή επιθυμεί να είναι η μηχανή σύνθεσης σε ότι αφορά τις YI που έχει βαθμολογίες μόνο σε έναν αριθμό από τα ζητούμενα QoS κριτήρια και όχι σε όλα. Η επιστρεφόμενη απάντηση, με βάση και το βαθμό αυστηρότητας, μπορεί να είναι της μορφής:

ProvQoSWS1 = {(QoS1, value1), (QoS3, value3), (QoS4, value4)}

…..

ProvQoSWSm = {(QoS1, value1), (QoS4, value4), (QoS6, value6)}

Οι τιμές αυτές αποτελούν μια λίστα YI (πλήθους m) που έχουν τιμές ορισμένες από τους παρόχους (ProvQoS) για κάποια ή για όλα τα απαιτούμενα QoS χαρακτηριστικά. Τυπικά επιλέγουμε να απορρίψουμε YI που δεν έχουν βαθμολογήσεις σε χαρακτηριστικά με τη μεγαλύτερη βαρύτητα όπως αυτή προκύπτει από τον Eigenvector.

Φάση τρίτη:
Η επιστρεφόμενη λίστα YI μπορεί να διαφοροποιηθεί ως ακολούθως: κάποια κριτήρια μπορούν να εμφανίζονται στις επιστρεφόμενες YI, τα οποία να μην είχαν ληφθεί υπόψη από τον τελικό χρήστη στο προηγούμενο στάδιο. Έτσι, σε αυτή τη φάση ο χρήστης μπορεί να ενισχύσει τις προηγούμενες επιλογές του με κάποια επιπλέον QoS κριτήρια.

Είναι σημαντικό να τονιστεί πως τόσο τα ProvQoS κριτήρια όσο και τα UsQoS κριτήρια είναι υποσύνολα του συνολικού QoS, ενώ αντιπροσωπεύουν υποκειμενικές βαθμολογήσεις κριτηρίων από διαφορετικές σκοπιές, κι έτσι η τομή τους ισοδυναμεί με το μηδέν.

ProvQoS⊆ QoS
UsQoS⊆ QoS
UsQoS ∩ProvQoS = ∅

Φάση τέταρτη:
Σε κάθε σενάριο που ενσωματώνει τις τεχνολογίες του Web 2.0, είναι καθοριστικής σημασίας η αξιοποίηση των αξιολογήσεων των χρηστών. Προηγούμενοι χρήστες των YI που συμμετέχουν σε μια σύνθεση, μπορούν να δώσουν αξιολογήσεις σχετικές με το επίπεδο της ικανοποίησης τους. Αυτές οι αξιολογήσεις, όπως είναι φυσικό, είναι υποκειμενικές και διαφέρουν σημαντικά από τις ProvQoS και UsQoS τιμές. Έτσι, για παράδειγμα, το επίπεδο της ασφάλειας έτσι όπως το αντιλαμβάνεται κάποιος τελικός χρήστης, είναι ένα EvQoS χαρακτηριστικό, που διαφέρει από το ProvQoS χαρακτηριστικό της ασφάλειας. Σε αυτή τη φάση ένα πλήθος EvQoS προστίθενται.

Φάση πέμπτη:
Σε αυτή τη φάση η AHP μεθοδολογία εφαρμόζεται. Τα βήματα είναι τα ακόλουθα:

Βήμα 1
Σαν αποτέλεσμα της προσθήκης των ProvQoS και EvQoS κριτηρίων, ο δισδιάστατος πίνακας έχει συμπληρωθεί μερικώς με αξιολογήσεις. Στη συνέχεια θα πρέπει να συμπληρωθεί με τις επιπρόσθετες κατά-ζεύγη συγκρίσεις. Σε ένα παράδειγμα που περιλαμβάνει 3 UsQoS, 2 ProvQoS και 2 EvQoS κριτήρια, ο πίνακας συγκρίσεων διαμορφώνεται ως εξής:

$$\begin{bmatrix} 1 & PwC_{12} & PwC_{13} & PwC_{14} & PwC_{15} & PwC_{16} & PwC_{17} \\ PwC_{21} & 1 & PwC_{23} & PwC_{24} & PwC_{25} & PwC_{26} & PwC_{27} \\ PwC_{31} & PwC_{32} & 1 & PwC_{34} & PwC_{35} & PwC_{36} & PwC_{37} \\ PwC_{41} & PwC_{42} & PwC_{43} & 1 & PwC_{45} & PwC_{46} & PwC_{47} \\ PwC_{51} & PwC_{52} & PwC_{53} & PwC_{54} & 1 & PwC_{56} & PwC_{57} \\ PwC_{61} & PwC_{62} & PwC_{63} & PwC_{64} & PwC_{65} & 1 & PwC_{67} \\ PwC_{71} & PwC_{72} & PwC_{73} & PwC_{74} & PwC_{75} & PwC_{76} & 1 \end{bmatrix}$$

Πίνακας 10.1 *Πίνακας συγκρίσεων κατά-ζεύγη*

Βήμα 2
Στο επόμενο βήμα γίνεται ο υπολογισμός του eigenvector έτσι ώστε να ληφθούν οι προτεραιότητες (τα βάρη) των κριτηρίων. Σύμφωνα με τη μεθοδολογία της AHP, μετά τη μετατροπή των κλασμάτων σε δεκαδικούς αριθμούς, ο δισδιάστατος πίνακας υψώνεται στο τετράγωνο. Ο παρακάτω πίνακας Xij (πίνακας 10.2) συμβολίζει το αποτέλεσμα:

$$\begin{bmatrix} X_{11} & X_{12} & X_{13} & X_{14} & X_{15} & X_{16} & X_{17} \\ X_{21} & X_{22} & X_{23} & X_{24} & X_{25} & X_{26} & X_{27} \\ X_{31} & X_{32} & X_{33} & X_{34} & X_{35} & X_{36} & X_{37} \\ X_{41} & X_{42} & X_{43} & X_{44} & X_{45} & X_{46} & X_{47} \\ X_{51} & X_{52} & X_{53} & X_{54} & X_{55} & X_{56} & X_{57} \\ X_{61} & X_{62} & X_{63} & X_{64} & X_{65} & X_{66} & X_{67} \\ X_{71} & X_{72} & X_{73} & X_{74} & X_{75} & X_{76} & X_{77} \end{bmatrix}$$

Πίνακας 10.2 *Το αποτέλεσμα ύψωσης στο τετράγωνο, του δυαδικού πίνακα συγκρίσεων*

Στη συνέχεια γίνεται υπολογισμός του αθροισμάτων των σειρών του πίνακα Xij με βάση τον τύπο (1)

$$\text{Sumi} = \sum_{j=1}^{7} X_{ij} \quad (1)$$

για i=1,2,…n, όπου στο συγκεκριμένο παράδειγμα το n είναι ίσο με 7, καθώς αυτός είναι ο αριθμός των κριτηρίων.

Ο πίνακας eigenvector υπολογίζεται με βάση τον τύπο (2):

$$\text{EVi} = \frac{\text{SUM}_i}{\sum_1^7 \text{SUM}_i} \quad (2)$$

Η διαδικασία αυτή επαναλαμβάνεται υψώνοντας στο τετράγωνο τον πίνακα Xij και υπολογίζοντας το νέο πίνακα eigenvector, μέχρις ότου να μην υπάρχουν σημαντικές διαφορές ανάμεσα στους πίνακες eigenvectors.

Βήμα 3

Στο τρίτο βήμα σχηματίζεται ένας δυαδικός πίνακας κατά-ζεύγη συγκρίσεων, για κάθε ένα από τα κριτήρια, όπου κάθε πίνακας περιέχει συγκρίσεις των εναλλακτικών YI, ως προς το συγκεκριμένο κριτήριο.

Σε περίπτωση λοιπόν που η μηχανή σύνθεσης επιστρέψει τέσσερις YI που ικανοποιούν τα λειτουργικά κριτήρια και καθώς έχουμε επτά κριτήρια ποιότητας που λαμβάνουμε υπόψη στο συγκεκριμένο παράδειγμα, συνολικά θα δημιουργηθούν επτά δισδιάστατοι πίνακες AltEVk (για k=1,2,3,4) μεγέθους 4x4. Οι πίνακες αυτοί θα είναι της μορφής:

$$\begin{array}{c} \\ WS_1 \\ WS_2 \\ WS_3 \\ WS_4 \end{array} \begin{array}{c} \begin{array}{cccc} WS_1 & WS_2 & WS_3 & WS_4 \end{array} \\ \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \\ C_{41} & C_{42} & C_{43} & C_{44} \end{bmatrix} \end{array}$$

Πίνακας 10.3 *Ενδεικτικός δισδιάστατος πίνακας κατά-ζεύγη συγκρίσεων για ένα κριτήριο*

Στον παραπάνω πίνακα οι θέσεις C_{11}, C_{22}, C_{33} και C_{44} είναι ίσες με τη μονάδα καθώς δεν μπορεί να γίνει σύγκριση μιας YI με τον εαυτό της. Το βήμα ολοκληρώνεται με τον υπολογισμό του eigenvector (με χρήση του τύπου (1)) για κάθε έναν πίνακα AltEVk, κάτι που τελικά επιτρέπει τον υπολογισμό των βαρών για κάθε εναλλακτική YI για κάθε κριτήριο.

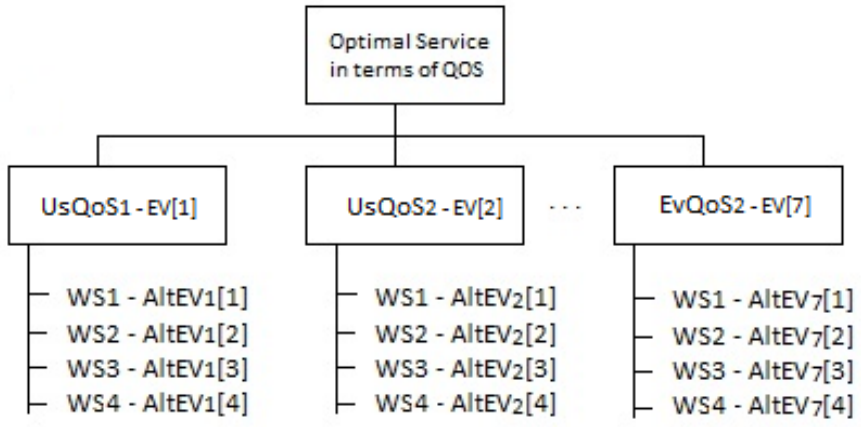
Βήμα 4

Το σχήμα 10.4 παρουσιάζει την ιεραρχική δομή του προβλήματος μετά την τοποθέτηση των τιμών του κάθε eigenvector (eigenvalues)

Το τελευταίο βήμα αυτής της διαδικασίας είναι ο υπολογισμός του πίνακα κατάταξης των εναλλακτικών AR[i], με χρήση του τύπου (3):

$$\text{AR[i]} = \sum_{j=1}^{7} \text{AltEv}_j[1] * \text{EV[j]} \quad (3)$$

Η YI με τη μεγαλύτερη βαθμολογία στον πίνακα AR[i] είναι η βέλτιστη YI σε σχέση με τα μη-λειτουργικά κριτήρια ποιότητας.



Σχήμα 10.4 *Εφαρμογή των Eigenvalues στο ιεραρχικό δέντρο*

5.2. Περίπτωση χρήσης μεθοδολογίας επιλογής YI στα πλαίσια χρήσης εντός ενός εμπορικού καταστήματος

Μελέτη περίπτωσης (Case study)
Η ενσωμάτωση των ιδεών του Web of Things (WoT) έχει ιδιαίτερο ενδιαφέρον σε σενάρια που περιλαμβάνουν συναλλαγές ηλεκτρονικού εμπορίου, μέσω φορητών συσκευών, σε συνδυασμό με τη φυσική παρουσία του χρήστη στο χώρο των συναλλαγών. Στο παρακάτω παράδειγμα θεωρούμε πως σε ένα εμπορικό κέντρο προσφέρονται πληροφορίες και συστάσεις μαζί με ένα πλήθος αποκλειστικών υπηρεσιών σε επισκέπτες που κάνουν χρήση της εφαρμογής (application) του εμπορικού καταστήματος για έξυπνες συσκευές. Οι επισκέπτες έχουν τη δυνατότητα χρήσης τόσο «φυσικών» όσο και «εικονικών» YI. «Φυσικές» YI μπορούν να είναι υπηρεσίες που προέρχονται από τη λειτουργικότητα φυσικών αντικειμένων, για παράδειγμα κάποιοι ψυγειοκαταψύκτες σε ένα κατάστημα προσφέρουν πληροφορίες για την κατάσταση των προϊόντων που περιέχουν, πληροφορίες οι οποίες συνδυάζονται εύκολα με πληροφορίες «εικονικών» YI που δείχνουν π.χ. μια έκπτωση στα προϊόντα αυτά. Σε ένα τέτοιο παράδειγμα μπορεί να γίνει χρήση τόσο YI τύπου SOAP όσο και YI τύπου REST.

Εκφώνηση:

Αξιοποιώντας τη μεθοδολογία που περιεγράφηκε στην προηγούμενη ενότητα, να περιγραφεί ένα σενάριο επιλογής YI, με βάση συγκεκριμένα κριτήρια ποιότητας, στα πλαίσια χρήσης ενός εμπορικού καταστήματος.

Υποδειγματική λύση:

- Φάση Πρώτη:
  - Στην πρώτη φάση υποθέτουμε πως ο χρήστης εισάγει μια λίστα από λειτουργικές απαιτήσεις, βασισμένες στις ανάγκες του αλλά και σε συστάσεις που δέχεται από τα καταστήματα. Έτσι λαμβάνει μια λίστα από πέντε YI (WSi για i=1, 2, ..,5) που ικανοποιούν τα λειτουργικά κριτήρια.
- Φάση Δεύτερη:
  - Ο χρήστης καλείται να επιλέξει τα χαρακτηριστικά ποιότητας που τον ενδιαφέρουν και να δώσει τις κατά-ζεύγη συγκρίσεις του, που καθορίζουν τα βάρη των κριτηρίων. Η μορφή εισόδου είναι του τύπου {(UsQoSi, UsQoSj, PwCij), …}
  - Σε αυτό το σενάριο ο χρήστης επιλέγει τα κριτήρια ‘ασφάλεια’ (Security - UsQoS1), ‘διαθεσιμότητα’ (Availability - UsQoS2) και ‘αξιοπιστία’ (Reliability - UsQoS3) ως εξής:{(Security, Availability - 6), (Security, Reliability - 8), (Reliability, Availability – 2)}

- Μια λίστα από YI που έχουν βαθμολογήσεις (από τους παρόχους) για τις τιμές αυτές επιστέφονται στον χρήστη.

  ProvQoSWS1 = {(Security, 8), (Availability, 6), (Reliability, 7), (Capacity, 6)}
  ProvQoSWS2 = {(Security, 7), (Reliability, 6), (Performance, 7), (Robustness, 4)}
  ProvQoSWS3 = {(Security, 7), (Availability, 7)}
  ProvQoSWS4 = {(Security, 9), (Availability, 8), (Reliability, 7)}
  ProvQoSWS5 = {(Availability, 8), (Performance, 8)}

- Η YI WS5 θα παραλειφθεί καθώς από αυτήν απουσιάζουν βαθμολογήσεις για τα πιο σημαντικά κριτήρια με βάση τα βάρη που δόθηκαν πριν.

- Φάση Τρίτη και φάση Τέταρτη:
  - Σε περίπτωση μη επιλογής επιπρόσθετων κριτηρίων ποιότητας από τον χρήστη, γίνεται η ενσωμάτωση των εξωτερικών βαθμολογήσεων από άλλους χρήστες.
  - Σε αυτό το παράδειγμα ο χρήστης λαμβάνει τιμές για την εκλαμβανόμενη από τους χρήστες ασφάλεια (EvSecurity) και την εκλαμβανόμενη από τους χρήστες αξιοπιστία (EvReliability).
- Φάση Πέμπτη:
  - Βήμα 1: Αφού προστεθούν τα ProvQoS και EvQoS κριτήρια, ο δισδιάστατος πίνακας των κατά-ζεύγη συγκρίσεων πρέπει να συμπληρωθεί από τον χρήστη. Σε αυτό το παράδειγμα ο πίνακας έχει ως εξής:

	$UsQoS_1$	$UsQoS_2$	$UsQoS_3$	$PrvQoS_1$	$PrvQoS_2$	$EvQoS_1$	$EvQoS_2$
$UsQoS_1$	1	6	8	7	9	3	8
$UsQoS_2$	1/6	1	1/2	3	7	1/2	7
$UsQoS_3$	1/8	2	1	5	7	2	5
$PrvQoS_1$	1/7	1/3	1/5	1	3	1/2	2
$PrvQoS_2$	1/9	1/7	1/7	1/3	1	1/2	1/2
$EvQoS_1$	1/3	2	1/2	2	2	1	3
$EvQoS_2$	1/8	1/7	1/5	1/2	2	1/3	1

Πίνακας 10.4 *Συνολικός πίνακας κατά-ζεύγη συγκρίσεων*

- Βήμα 2: Με τη μετατροπή των κλασμάτων σε δεκαδικούς αριθμούς και με την ύψωση του πίνακα στο τετράγωνο σχηματίζεται ο επόμενος πίνακας.

7.0006	38.7631	24.7861	85.0006	159	35.6672	125.5
2.644	7.001	5.5839	16.5007	43.0003	9.8339	28.8336
3.3674	12.1318	7.0003	25.7088	58.125	13.042	44.5
1.1164	3.6387	2.7886	7.0007	16.0199	4.1622	11.477
0.5407	2.4208	1.6627	3.8377	7.0007	2.0240	5.7704
1.9457	8.3817	5.953	17.0006	36.5006	7.0004	30.1672
0.7036	2.5551	2.0240	4.6373	9.6921	2.7633	7.0005

Πίνακας 10.5 *Ο πίνακας μετά την ύψωση στο τετράγωνο*

- Κάνοντας χρήση των τύπων (1) και (2) γίνεται υπολογισμός του Eigenvector, όπου:
- Ev = [0.4962, 0.1183, 0.1709, 0.0482, 0.0243, 0.1115, 0.0306]
- Βήμα 3: Σε αυτό το βήμα ο χρήστης καλείται να συμπληρώσει τιμές σε επτά πίνακες συγκρίσεων, έναν για κάθε ένα από τα επιλεχθέντα κριτήρια

UsQoS1

WS_1	1	3	6	8
WS_2	1/3	1	3	6
WS_3	1/6	1/3	1	4
WS_4	1/8	1/6	1/4	1

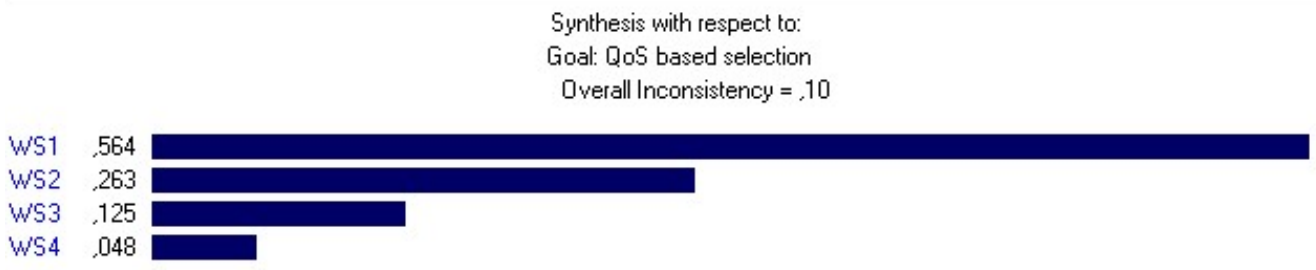
UsQoS2

WS_1	1	4	3	5
WS_2	1/4	1	3	7
WS_3	1/3	1/3	1	4
WS_4	1/5	1/7	1/4	1

Πίνακας 10.6 *Ενδεικτικοί πίνακες κατά-ζεύγη συγκρίσεων για 2 κριτήρια*

- Ακολουθώντας τη διαδικασία που είδαμε παραπάνω γίνεται υπολογισμός των επτά eigenvectors
- Βήμα 4: Με βάση τις πληροφορίες που συλλέχθηκαν στα προηγούμενα βήματα, γίνεται ο υπολογισμός της τελικής κατάταξης των εναλλακτικών YI με βάση τον (3): AR = [0.564, 0.263, 0.125, 0.048]
- Το σχήμα 10.5 δείχνει την τελική βαθμολόγηση και κατάταξη των τεσσάρων εναλλακτικών YI καθώς και την τιμή overall inconsistency όπως αυτή υπολογίζεται από το εργαλείο Expert Choice, και φανερώνει το κατά πόσο έχει γίνει με σωστό τρόπο η βαθμολόγηση των κριτηρίων και των εναλλακτικών, δηλαδή την ύπαρξη συνοχής. Η τιμή αυτή θα πρέπει να είναι ίση ή και μικρότερη του 0.10.

Σημείωση: Στον Ιστότοπο του συγγράμματος (http://ec-tech.uom.gr/WT-ECOM), θα βρείτε το πηγαίο κώδικα για όλες τις υποδειγματικά λυμένες ασκήσεις



Σχήμα 10.5 *Ταξινόμηση YI όπως αυτή υπολογίζεται μέσω του εργαλείου Expert Choice*

6. Συμπεράσματα

Στο κεφάλαιο αυτό παρουσιάστηκε ένα σύνολο εργαστηριακών ασκήσεων, με σκοπό την κατανόηση τεχνικών ανάπτυξης, επιλογής και σύνθεσης YI, ενώ ιδιαίτερη έμφαση δόθηκε στις YI τύπου REST. Οι ασκήσεις αυτές έχουν υποδειγματικά επιλυθεί στα περιβάλλοντα Eclipse IDE (με χρήση του RESTlet framework) και NetBeans 7 (όπως αυτό περιλαμβάνεται στο πακέτο OpenESB 2.3.1) και χρησιμοποιούν ως βασικές γλώσσες προγραμματισμού την Java και την BPEL. Επίσης, παρουσιάστηκε αναλυτικά μια μεθοδολογία επιλογής YI η οποία κάνει χρήση τεχνικών πολυκριτήριας ανάλυσης αποφάσεων. Στόχος του κεφαλαίου ήταν η ανάπτυξη δεξιοτήτων εκ μέρους των αναγνωστών, πάνω σε τεχνολογίες που έχουν ευρεία αποδοχή για την ανάπτυξη εφαρμογών και τεχνικών που υποστηρίζουν συναλλαγές Ηλεκτρονικού Εμπορίου.

Βιβλιογραφία/Αναφορές

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645-1660.

Guinard, D., Trifa, V., Mattern, F., & Wilde, E. (2011). From the internet of things to the web of things: Resource-oriented architecture and best practices. In *Architecting the Internet of Things* (pp. 97-129). Springer Berlin Heidelberg.

Louvel, J., Templier T., & Boileau, T. (2012). *Restlet in Action: Developing RESTful Web APIs in Java*. Manning Publications Co., 2012.

Pautasso, C. (2014). RESTful web services: principles, patterns, emerging technologies. *Web Services Foundations*. Springer New York, 2014. 31-51.

Saaty, T. L. (1980). *The analytic hierarchy process: planning, priority setting, resources allocation*. New York: McGraw.

Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of network and computer applications* 34.1: 1-11.

Vesyropoulos, N., Georgiadis C.K., & Pimenidis E. (2014). Ensuring Cloud Security: Current Concerns and Research Challenges. *E-Democracy, Security, Privacy and Trust in a Digital World*. Springer International Publishing, 3-10.

Vesyropoulos, N., Georgiadis, C. K. (2015). Customized QoS-based Mashups for the Web of Things: An Application of AHP. *Computer Science and Information Systems*, Vol. 12, No. 1, 115–133.

Want, R., Schilit, B. N., & Jenson, S. (2015). Enabling the Internet of Things. *Computer*, (1), 28-35.

Quiz10.htm	Διαδραστικό τεστ αξιολόγησης (Interactive)
Διαδραστικό τεστ αξιολόγησης	

Κριτήρια αξιολόγησης

Σημείωση: Η διαβάθμιση δυσκολίας των κριτηρίων αξιολόγησης δίνεται με το πλήθος των αναγραφόμενων αστερίσκων.

Κριτήριο αξιολόγησης 1

[*] Το Restlet Framework επιτρέπει την ανάπτυξη εφαρμογών με τη μορφή APIs που θα εκτελούνται:

Α) αποκλειστικά από την πλευρά του εξυπηρετητή

Β) αποκλειστικά από την πλευρά του πελάτη

Γ) και από την πλευρά του εξυπηρετητή και από την πλευρά του πελάτη

Απάντηση/Λύση

Γ) και από την πλευρά του εξυπηρετητή και από την πλευρά του πελάτη

Κριτήριο αξιολόγησης 2

[*] Με τη χρήση του Mozilla Plugin με την ονομασία HttpRequester είμαστε σε θέση να εκτελούμε Http Requests που θα ήταν δύσκολο να εκτελεστούν από έναν απλό φυλλομετρητή, όπως είναι το GET request.

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 3

[*] Η εμπιστευτικότητα σε περιβάλλοντα «υπολογιστικής νέφους» σχετίζεται άμεσα με:

Α) Την πολυμίσθωση των διαθέσιμων πόρων

Β) Την παραμένουσα μαγνήτιση των δεδομένων

Γ) Την εμπιστευτικότητα λογισμικού

Δ) Όλα τα παραπάνω

Ε) Κανένα από τα παραπάνω

Απάντηση/Λύση

Δ) Όλα τα παραπάνω

Κριτήριο αξιολόγησης 4

[*] Σε περιβάλλοντα «υπολογιστικής νέφους», η έννοια της διαθεσιμότητας σχετίζεται κυρίως με τη συχνότητα εμφάνισης βλαβών σε υλικό του παρόχου, όπως π.χ. ενός σκληρού δίσκου.

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 5

[*] Ποιο από τα παρακάτω στοιχεία δεν είναι φανερό σε μια ενορχήστρωση BPEL

Α) Η ανταλλαγή μηνυμάτων και η επικοινωνία μεταξύ των partner links

Β) Οι βασικές και δομημένες δραστηριότητες της BPEL

Γ) Τα πρωτόκολλα ασφαλείας που χρησιμοποιούνται από τον εξυπηρετητή

Δ) Οι κανόνες με τους οποίους γίνεται ανάθεση τιμών στις μεταβλητές των partner links

Απάντηση/Λύση

Γ) Τα πρωτόκολλα ασφαλείας που χρησιμοποιούνται από τον εξυπηρετητή

Κριτήριο αξιολόγησης 6

[*] Η μέθοδος Analytical Hierarchy Process (AHP), είναι μια γνωστή μεθοδολογία επιλογής Υπηρεσιών Ιστού (ΥΙ).

Α) Σωστό

Β) Λάθος

Απάντηση/Λύση

Β) Λάθος

Κριτήριο αξιολόγησης 7

[**] Το εργαλείο «mapper» επιτρέπει:

Α) Την εισαγωγή ενός string σε μια μεταβλητή εξόδου

Β) Τη σύγκριση της τιμής ενός string εισόδου με αυτή ενός προεπιλεγμένου string

Γ) Τη σύγκριση της τιμής μιας αριθμητικής μεταβλητής με μια προεπιλεγμένη τιμή

Δ) Τη σύγκριση των τιμών δύο μεταβλητών εισόδου

Ε) Τίποτα από τα παραπάνω

ΣΤ) Όλα τα παραπάνω

Απάντηση/Λύση
ΣΤ) Όλα τα παραπάνω

Κριτήριο αξιολόγησης 8
[**] Η κλήση της μεθοδου getRequestAttributes().get("uid") χρησιμοποιείται για την

Α) Ανάγνωση ενός στοιχείου από τη βάση δεδομένων του server

Β) Διαγραφή ενός στοιχείου από τη βάση δεδομένων του server

Γ) Ανανέωση ενός στοιχείου στη βάση δεδομένων του server

Δ) Εισαγωγή ενός στοιχείου στη βάση δεδομένων του server

Απάντηση/Λύση
Α) Ανάγνωση ενός στοιχείου από τη βάση δεδομένων του server

Κριτήριο αξιολόγησης 9
[**] Ποιο από τα ακόλουθα δεν απαιτείται για τη δημιουργία μιας σύνθεσης YI (μέσω BPEL module project);

Α) Δημιουργία WSDL αρχείου/αρχείων ή αναφορά σε υπάρχοντα αρχεία WSDL

Β) Δημιουργία κατάλληλων μεθόδων κλήσης του Restlet Framework

Γ) Δημιουργία BPEL διαδικασίας

Δ) Δημιουργία ενός νέου BPEL module project

Ε) Δημιουργία αρχείου/αρχείων XML Schema (ή XSD αρχείου/αρχείων) ή αναφορά σε υπάρχοντα αρχεία αυτού του τύπου

Απάντηση/Λύση
Β) Δημιουργία κατάλληλων μεθόδων κλήσης του Restlet Framework

Κριτήριο αξιολόγησης 10
[**] Με βάση της προδιαγραφές της γλώσσας BPEL, ποια από τις παρακάτω προτάσεις δεν είναι σωστή;

Α) Στις βασικές δραστηριότητες συμπεριλαμβάνονται οι εντολές: Assign, Wait και Throw

Β) Στις δομημένες δραστηριότητες συμπεριλαμβάνονται οι εντολές: Partner Link, If και While

Γ) Στις εντολές χειρισμού YI συμπεριλαμβάνονται οι εντολές: Receive, Reply και Invoke

Απάντηση/Λύση
Β) Στις δομημένες δραστηριότητες συμπεριλαμβάνονται οι εντολές: Partner Link, If και While