



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Σχεδιασμός και Ανάπτυξη Διαδικτυακής Εφαρμογής Αγοράς Προσφορών Φαγητού και Ποτού Design and Development of a Web Application for Special Offers of Food and Drink
Όνοματεπώνυμο Φοιτητή	Μπλετσογιάννη Χριστίνα
Πατρώνυμο	Φώπιος
Αριθμός Μητρώου	ΜΠΣΠ/ 15058
Υπεύθυνος Καθηγητής	Χρήστος Δουληγέρης
Επιβλέπων Καθηγητής	Χρήστος Δουληγέρης

Ημερομηνία Παράδοσης

Μάιος 2019

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Χρήστος Δουληγέρης
Καθηγητής
Πανεπιστημίου Πειραιά

(υπογραφή)

Σαράντης Μητρόπουλος
Αναπληρωτής Καθηγητής
Ιονίου Πανεπιστημίου

(υπογραφή)

Παναγιώτης Κοτζανικολάου
Επίκουρος Καθηγητής
Πανεπιστημίου Πειραιά

Περιεχόμενα

Ευχαριστίες	6
Περίληψη	7
Abstract	7
1. Εισαγωγή	8
1.1 Χώρος του προβλήματος	8
1.2 Στόχοι της εργασίας	9
1.3 Δομή της εργασίας και μεθοδολογία που ακολουθήθηκε	9
1.4 Παρόμοια Συστήματα	9
1.5 Σύγκριση Συστημάτων	14
2. Τεχνολογίες διαδικτυακών εφαρμογών	14
2.1 Αρχιτεκτονικά μοντέλα στη σχεδίαση διαδικτυακών εφαρμογών και διαδικτυακών υπηρεσιών	16
2.2 Διαδικτυακές εφαρμογές εμπλουτισμένου περιεχομένου και HTML5	21
3. Κινητές συσκευές και πλατφόρμες	24
3.1 Τεχνικά χαρακτηριστικά και λειτουργίες των έξυπνων κινητών τηλεφώνων	25
3.2 Εφαρμογές για κινητές πλατφόρμες	26
4. Android έναντι HTML5	27
4.1 Διαδικτυακές εφαρμογές	28
4.2 Εγγενείς εφαρμογές	29
4.3 Υβριδικές εφαρμογές κινητού	29
4.4 Σύγκριση	30
5. Location Based Services	31
5.1 Το επιστημονικό πεδίο των υπηρεσιών εντοπισμού	31
6. Μεθοδολογία υλοποίησης	33
6.1 Λογικός σχεδιασμός	33
6.1.1 Λειτουργία Business-To-Business	34
6.1.2 Λειτουργία Business-To-Customer	34
6.2 Αρχιτεκτονική	35
6.2.1 Model	35
6.2.2 View	35
6.2.3 Controller	35
6.3 Φυσικός σχεδιασμός	36
6.4 Αρχιτεκτονική λογισμικού	36
6.4.1 Διακομιστής διαδικτύου	37
6.4.2 PHP	37
6.4.3 JSON	38
6.4.4 Επικοινωνία με GPS	39
7. Υλοποίηση Συστήματος	39
7.1 Βάση δεδομένων	41
7.2 Συναρτήσεις διεπαφής με βάση δεδομένων	41
7.3 Επίπεδο επιχειρησιακής λογικής (Controller)	42
7.4 Επίπεδο παρουσίασης (View)	43
7.4.1 Ενεργοποίηση και χρήση Geolocation API	44

Μεταπτυχιακή Διατριβή	Μπλετσογιάννη Χριστίνα
7.5 Μοντέλο οντοτήτων – συσχετίσεων	45
7.6 Δομή της εφαρμογής	45
8. Συμπεράσματα – Μελλοντικές επεκτάσεις	51
8.1 Συμπεράσματα	51
8.2 Μελλοντικές Επεκτάσεις	52
Βιβλιογραφία	53

Πίνακας Εικόνων - Διαγραμμάτων

Εικόνα 1. Στιγμιότυπο της Regroup.gr.....	10
Εικόνα 2. Στιγμιότυπο του Deals36.....	11
Εικόνα 3. Στιγμιότυπο του happydeals.gr.....	11
Εικόνα 4. Στιγμιότυπο του dealsafari.....	12
Εικόνα 5. Στιγμιότυπο του OLADeals.....	13
Εικόνα 6. Στιγμιότυπο του GoodDeals.....	14
Εικόνα 7. Ένα παράδειγμα MVC.....	36
Εικόνα 8. Η αρχιτεκτονική του λογισμικού.....	37
Εικόνα 9. Η θέση της PHP μεταξύ των γλωσσών ανάπτυξης εφαρμογών διακομιστή.....	38
Εικόνα 10. Η μορφή ενός object σε JSON.....	39
Εικόνα 11. Η μορφή ενός array σε JSON.....	39
Εικόνα 12. Η αρχιτεκτονική τριών επιπέδων της πλατφόρμας προσφορών.....	40
Εικόνα 13. Επεξεργασία tags της HTML μέσω κανόνων CSS.....	43
Εικόνα 14. Στιγμιότυπο από την αρχική εικόνα της εφαρμογής.....	46
Εικόνα 15. Στιγμιότυπο από την φόρμα σύνδεσης στην εφαρμογή.....	47
Εικόνα 16. Στιγμιότυπο από την φόρμα εισαγωγής στοιχείων νέου χρήστη.....	47
Εικόνα 17. Στιγμιότυπο από την αρχική σελίδα του επισκέπτη.....	48
Εικόνα 18. Στιγμιότυπο από την προσφορά που έχει επιλέξει ο χρήστης.....	48
Εικόνα 19. Στιγμιότυπο από το καλάθι του χρήστη.....	48
Εικόνα 20. Στιγμιότυπο από την αρχική σελίδα του διαχειριστή.....	49
Εικόνα 21. Στιγμιότυπο από τους εγγεγραμμένους χρήστες όπως φαίνονται από το διαχειριστικό του administrator.....	49
Εικόνα 22. Στιγμιότυπο από την επεξεργασία των ήδη διαθέσιμων προϊόντων από το διαχειριστικό του administrator.....	50
Εικόνα 23. Στιγμιότυπο από την εισαγωγή νέου προϊόντος από το διαχειριστικό του administrator.....	50
Διάγραμμα 1. Use-Case.....	33
Διάγραμμα 2. Το σχεσιακό μοντέλο του συστήματος και οι συνδέσεις των πινάκων.....	45
Διάγραμμα 3. Αρχιτεκτονική επικοινωνίας της ενότητας διαχείρισης πληροφοριών με το σύστημα διαχείρισης βάσεων δεδομένων MySQL.....	51

Ευχαριστίες

Η ολοκλήρωση της συγκεκριμένης εργασίας δεν θα ήταν εφικτή χωρίς τη συμβολή κάποιων προσώπων. Έτσι, από αυτή τη θέση, θα ήθελα να ευχαριστήσω θερμά τον κύριο Δουληγέρη, Καθηγητή του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιά, για την ανάθεση αυτής της διπλωματικής, αλλά κυρίως για την υποστήριξή του σε όλη τη διάρκεια εκπόνησης αυτής της εργασίας.

Επιπλέον, θα ήθελα να ευχαριστήσω τον Δρ. Μητρόπουλο, Αναπληρωτή Καθηγητή του Ιονίου Πανεπιστημίου, για την αμέριστη συμπαράστασή του, την άρτια επιστημονική καθοδήγησή του και τις εξαιρετικά βοηθητικές συμβουλές του για την βελτίωση της εργασίας μου.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου και την αδερφή μου για την ηθική υποστήριξη που μου πάντα μου δίνουν.

Περίληψη

Η παρούσα μεταπτυχιακή διατριβή έχει ως κύριο στόχο την ανάπτυξη μιας διαδικτυακής εφαρμογής προσφορών που ονομάζεται «Snackmart» Συγκεκριμένα, παρουσιάζονται στον χρήστη της συγκεκριμένης εφαρμογής προσφορές που αφορούν φαγητό, καφέ ή ποτό.

Έγινε μια προσπάθεια δημιουργίας μιας εφαρμογής φιλικής προς τον χρήστη που θέλει να «ανακαλύψει» προσφορές καταστημάτων φαγητού ή ποτού/καφέ στην περιοχή που τον ενδιαφέρει.

Πιο κάτω αναλύονται οι τεχνολογίες και οι αρχιτεκτονικές που έχουν χρησιμοποιηθεί για την ανάπτυξη της εφαρμογής.

Τέλος, περιγράφονται λεπτομερώς και αναλύονται όλα τα στάδια της δημιουργίας της εφαρμογής, από τη μελέτη των λειτουργικών απαιτήσεων και την αρχιτεκτονική του συστήματος και την τεχνική υλοποίηση, μέχρι την παρουσίαση των συμπερασμάτων και των μελλοντικών προοπτικών της.

Abstract

The purpose of this master thesis is to develop a web application for special offers of food and drink. Through this application the visitors will be able to navigate in an application named "Snackmart", which provides them offers about food and drink in the area of their interest.

There has been an effort to create an application user friendly, which gives the opportunity to the users to find the offer that interests them.

In addition, in subsequent chapter, there is an analysis of the technologies that have been used in order to create this web application.

Finally, there is also a detailed analysis of all creation phases that have been followed, starting from the study of functional requirements and the architecture of the system and the technical implementation, until the presentation of its conclusions and future perspectives.

1. Εισαγωγή

Εδώ και αρκετά χρόνια, οι διαδικτυακές εφαρμογές αποτελούν πρωταρχικά εργαλεία στην ανάπτυξη της βιομηχανίας της πληροφορικής. Οι Εφαρμογές Ιστού (Web Applications), έχουν εξελιχθεί από απλές παρουσιάσεις ιστότοπων, σε πολύπλοκες αρχιτεκτονικές που προσφέρουν επιχειρηματική λειτουργικότητα σε μια εταιρεία και στους συνεργάτες της. Αποτελούν αναπόσπαστα εργαλεία για κάθε οργανισμό σήμερα. Μέσω αυτών, κάθε εργαζόμενος μπορεί να ενημερώνεται και να έρχεται σε επαφή με τους στόχους και τα σχέδια της επιχείρησης. Μέσω των Web Applications γίνεται η εκτέλεση των καθημερινών διαδικασιών εντός του οργανισμού, η προώθηση των πωλήσεων, καθώς και η συνεργασία της εκάστοτε επιχείρησης με εξωτερικούς φορείς και κατ'επέκταση με τους τελικούς χρήστες.

Σε ό,τι αφορά τις εφαρμογές προσφορών, υπάρχει μια μεγάλη αύξηση τα τελευταία χρόνια. Πλέον, τέτοιου τύπου εφαρμογές καλύπτουν ένα ευρύ φάσμα αναγκών του καταναλωτή. Από εφαρμογές προσφορών ένδυσης και υπόδησης, μέχρι web applications ταξιδιωτικών προσφορών, αλλά και προσφορών σε χώρους εστίασης με φαγητό ή ποτό. Οι περισσότερες από αυτές τις εφαρμογές που σχετίζονται με προσφορές φαγητού ή ποτού, συνήθως περιλαμβάνουν και πολλές άλλες κατηγορίες προσφορών ταυτόχρονα, κάτι που πιθανόν κουράζει τον χρήστη που μπαίνει στην εφαρμογή, έχοντας κάτι συγκεκριμένο στο μυαλό του και τελικά καταλήγει να περιηγηθεί στις εκατοντάδες άλλες προσφορές που τον κατακλύζουν.

Στην παρούσα μεταπτυχιακή διατριβή θα παρουσιαστεί η ανάπτυξη της εφαρμογής προσφορών φαγητού και ποτού/καφέ "SnackMart". Θα παρουσιαστούν οι τεχνολογίες, οι αρχιτεκτονικές καθώς και το business logic με το οποίο δημιουργήθηκε.

Στο παρόν κεφάλαιο θα αναφερθούμε στο σκοπό εκπόνησης της διπλωματικής εργασίας, στο χώρο προβλήματος και σε παρόμοιες εφαρμογές που υπάρχουν στο διαδίκτυο.

1.1 Χώρος του προβλήματος

Τις τελευταίες δεκαετίες, με την έξαρση του διαδικτύου και τη μεγάλη ανάπτυξη των διαδικτυακών εφαρμογών, έχουν δημιουργηθεί web applications που καλύπτουν κάθε πιθανή ανάγκη των χρηστών. Ιδιαίτερη άνθιση έχουν παρουσιάσει οι διαδικτυακές πλατφόρμες πωλήσεων και κατ'επέκταση προσφορών για κάθε είδος. Πλέον ο χρήστης είναι εξοικειωμένος με τη χρήση ηλεκτρονικού υπολογιστή και συνεχώς όλο και περισσότερο, του διαδικτύου. Τα ηλεκτρονικά καταστήματα δίνουν την ευκολία στο χρήστη να δει αναλυτικά τα χαρακτηριστικά ενός μεγάλου πλήθους προϊόντων που θα ήθελε να αγοράσει, χωρίς να χρειαστεί να επισκεφτεί το φυσικό κατάστημα. Λόγω της μεγάλης έξαρσης των διαδικτυακών καταστημάτων, ήταν φυσικό επακόλουθο η υλοποίηση διάφορων applications, που εντοπίζουν όλα τα φυσικά καταστήματα από τα οποία θα μπορούσε ο χρήστης- πελάτης, να αγοράσει αυτό που επιθυμεί στη χαμηλότερη τιμή.

Για παράδειγμα η Skrutz¹, λειτουργεί ακριβώς με αυτόν τον τρόπο. Ο χρήστης αναζητά το προϊόν που τον ενδιαφέρει μέσω αυτής της πλατφόρμας, η οποία το εντοπίζει στα αποθέματα των καταστημάτων που το πωλούν και εμφανίζει στο χρήστη μια λίστα με αυτά και τις τιμές στις οποίες πωλείται σε κάθε κατάσταση το συγκεκριμένο προϊόν. Έτσι, ο χρήστης μπορεί να αποφασίσει από πού θέλει να το αγοράσει, λαμβάνοντας υπόψη του εκτός από την χαμηλότερη τιμή και την τοποθεσία του καταστήματος, καθώς και την απόστασή του από αυτό.

Πέραν όμως από τέτοιου είδους πλατφόρμες όπως αυτή που αναφέραμε πιο πάνω, η οποία έχει κυρίως ενημερωτικό χαρακτήρα, υπάρχουν web applications μέσω των οποίων πωλείται και το προϊόν το οποίο προβάλλεται. Σε τέτοιου είδους εφαρμογές, η εταιρεία που δημιουργεί την πλατφόρμα, είτε κατέχει και κάποιο φυσικό χώρο, οπότε λειτουργεί σαν ηλεκτρονικό κατάστημα και κάνει προσφορές στα δικά της προϊόντα, είτε συνεργάζεται με διάφορες εταιρείες που θέλουν να προωθήσουν τα προϊόντα τους μέσω της συγκεκριμένης πλατφόρμας, κάνοντας κατά καιρούς διάφορες προσφορές για να προσελκύσουν καταναλωτές. Σε αυτήν την κατηγορία ανήκει και συγκεκριμένη εφαρμογή που έχει αναπτυχθεί για αυτή τη διπλωματική εργασία.

¹ www.skrutz.gr

1.2 Στόχοι της εργασίας

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί, η μελέτη και η ανάπτυξη μιας διαδικτυακής εφαρμογής προσφορών για προϊόντα φαγητού ή ποτού/καφέ. Σε αυτή την πλατφόρμα ο χρήστης θα έχει τη δυνατότητα να εντοπίσει διάφορες προσφορές για φαγητό ή ποτό στην περιοχή που τον ενδιαφέρει. Η ανάπτυξη της έγινε σε γλώσσα προγραμματισμού PHP χρησιμοποιώντας τις πλέον σύγχρονες αρχιτεκτονικές ανάπτυξης κώδικα. Το κύριο χαρακτηριστικό της συγκεκριμένης εφαρμογής, αποτελεί η ευχρηστία της. Στόχος της εργασίας είναι να παρουσιάσει μια νέα ιδέα σχετικά με τις πλατφόρμες προσφορών.

1.3 Δομή της εργασίας και μεθοδολογία που ακολουθήθηκε

Στο παρόν σημείο της μεταπτυχιακής διατριβής, θα αναφερθούμε σε παρόμοια συστήματα με την εφαρμογή που αναπτύχθηκε και θα γίνει μια σύγκριση μεταξύ τους, αναφέροντας τα θετικότερα στοιχεία τους, αλλά και κάποιες πιθανές βελτιώσεις όπως μπορεί να παρατηρηθεί από την πλευρά του χρήστη.

Στη συνέχεια, θα γίνει μια λεπτομερής αναφορά στις τεχνολογίες διαδικτυακών εφαρμογών, στα διάφορα αρχιτεκτονικά μοντέλα που χρησιμοποιούνται καθώς και για τη σχεδίασή τους, αλλά και για τις διαδικτυακές εφαρμογές εμπλουτισμένου περιεχομένου και στην HTML5. Ακόμη, θα γίνει αναφορά σε διάφορα τεχνικά χαρακτηριστικά και λειτουργίες των έξυπνων τηλεφώνων αλλά και σε εφαρμογές κινητών συσκευών.

Επιπλέον, θα γίνει αναλυτική αναφορά για τη μεθοδολογία υλοποίησης της εφαρμογής, με αναφορά στην επιχειρηματική λογική, την αρχιτεκτονική που ακολουθήθηκε, τη γλώσσα που χρησιμοποιήθηκε για την υλοποίησή της και γενικότερα για το φυσικό σχεδιασμό της.

Σε επόμενο κεφάλαιο, παρουσιάζεται σε μεγαλύτερα ανάλυση η δομή της εφαρμογής μέσα από κατάλληλα screenshots, σε κάθε στάδιο λειτουργίας της.

Στο τελευταίο κεφάλαιο, παρουσιάζονται τα συμπεράσματα που προέκυψαν από την υλοποίηση της διπλωματικής και την αξιολόγηση της. Επίσης γίνεται αναφορά σε μελλοντικές βελτιώσεις βάσει νεότερων τεχνολογιών, καθώς και των αναγκών των χρηστών.

1.4 Παρόμοια Συστήματα

Σε αυτό το κεφάλαιο θα γίνει αναφορά στις λειτουργίες συστημάτων που υπάρχουν αυτή τη στιγμή στο διαδίκτυο στην Ελλάδα, παρόμοιων με την παρούσα εφαρμογή που έχει αναπτυχθεί.

Regroup.gr

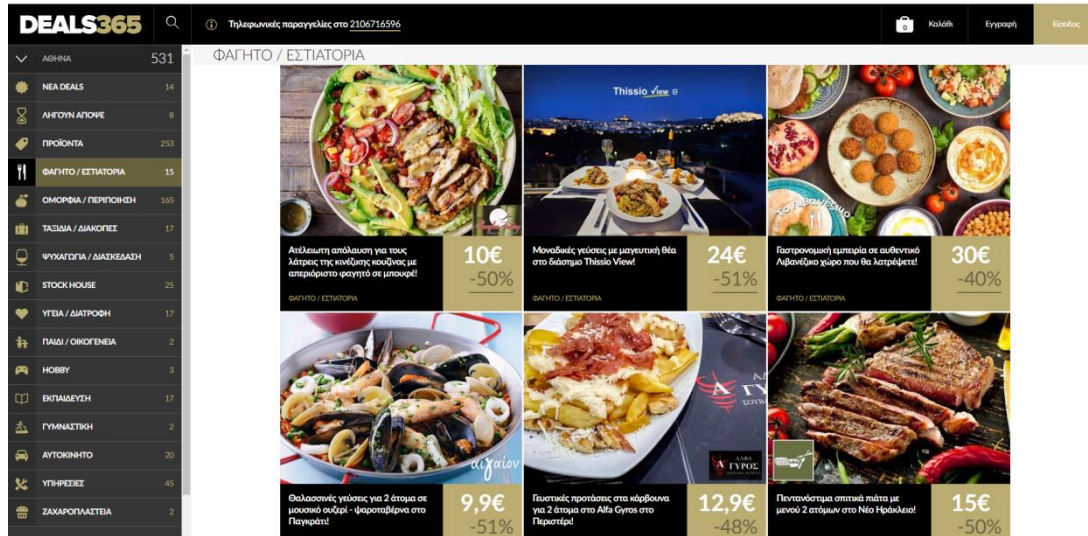
Η Regroup.gr είναι ένα από τα μεγαλύτερα site διαδικτυακών προσφορών. Μπαίνοντας στην αρχική σελίδα της πλατφόρμας, δίνεται η δυνατότητα στον χρήστη να επιλέξει την περιοχή όπου τον ενδιαφέρουν οι προσφορές που υπάρχουν. Η συγκεκριμένη πλατφόρμα, δεν είναι προσανατολισμένη σε μια συγκεκριμένη κατηγορία προσφορών, αλλά παρέχει προσφορές διαφόρων κατηγοριών όπως για παραστάσεις, ταξίδια, κατηγορία ομορφιά/fitness κτλ. Επιπλέον, αν προτείνει ο χρήστης τη regroup σε φίλο του κερδίζει 5 ευρώ όταν η αγορά κάθε φίλου του συμπληρώσει τα 20 ευρώ. Ακόμη, ο χρήστης που επιθυμεί να αγοράσει κάποια προσφορά, πρέπει να εγγραφεί στην πλατφόρμα και αργότερα συνδεθεί, να αγοράσει την προσφορά που τον ενδιαφέρει, επιλέγοντας ως τρόπο πληρωμής, πληρωμή με κάρτα, αντικαταβολή, PayPal ή Viva wallet κτλ. Στην περίπτωση που επιλέξει προσφορά που αφορά φαγητό, του αποστέλλεται μέσω email το κουπόνι της προσφοράς.

The screenshot shows the Regroup.gr website interface. At the top, there is a search bar with 'Αναζήτηση' and 'Αθήνα' selected. The navigation menu includes 'Αρχική', 'Τοπικά Deals', 'Προϊόντα', and 'Travel by Ekdromi.gr'. On the left, a sidebar lists various categories such as 'Προϊόντα (1429)', 'Τοπικά Deals (706)', 'Auto / Μото (32)', 'Παραστάσεις / Διασκέδαση (104)', 'Υγεία (64)', 'Fitness / Ομορφιά (266)', 'Υπηρεσίες (96)', 'Φαγητό & Ποτό (50)', 'Παιδί / Οικογένεια (52)', 'Ταξίδια & Εκδρομές (37)', and 'Travel by Ekdromi.gr'. The main content area features a large banner for 'CITY BAR RESTAURANT' with a deal for 'Γεύμα 2 Ατόμων Μεσογειακής Κουζίνας με Ελεύθερη Επιλογή' for 29€ for 3 people. Below the banner, there is a breadcrumb trail 'Αρχική > Τοπικά Deals > Φαγητό & Ποτό' and a section titled 'Φαγητό & Ποτό' with three featured items: 'Πλήρες Μενού Μοντέρνας Μεσογειακής Κουζίνας', 'Βάφλα με 2 Μπάλες Παγωτό-Τοπάζι & Βάφλας Κρέμας', and 'Μενού 2 Ατόμων Ελληνικής Κουζίνας με 2 Πιάτα'.

Εικόνα 1. Στιγμιότυπο της Regroup.gr (<https://www.regroup.gr/>)

Deals365

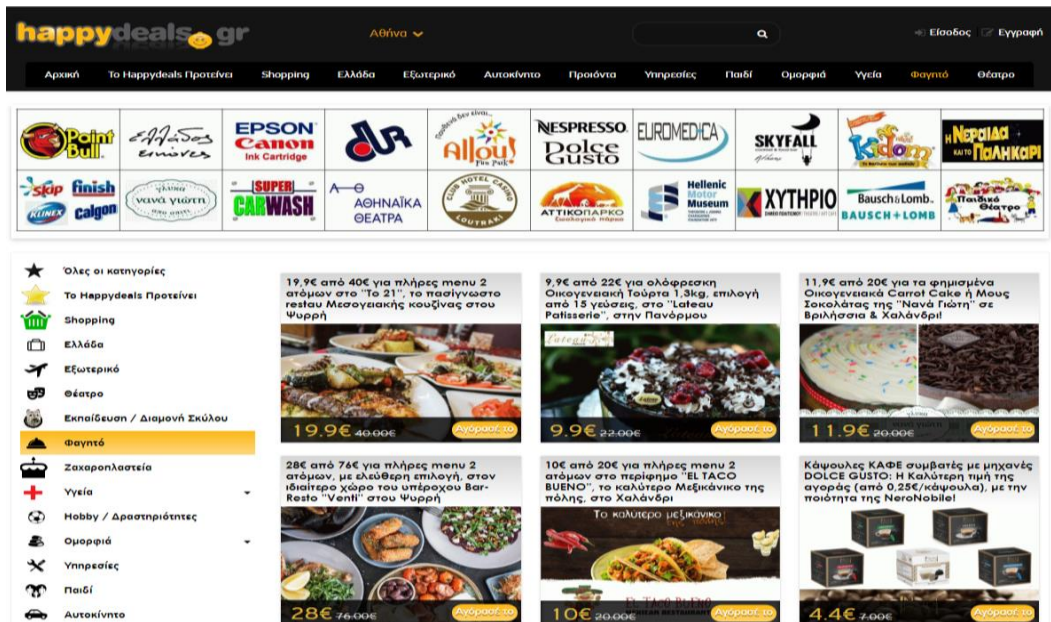
Το Deals365 είναι επίσης ένα από τα γνωστότερα site προσφορών. Όπως και η Regroup.gr, διαθέτει πολλές κατηγορίες προσφορών και δίνεται η δυνατότητα στο χρήστη να κάνει αναζήτηση με βάση την περιοχή που επιθυμεί. Ο χρήστης πρέπει να εγγραφεί ώστε να μπορέσει να χρησιμοποιήσει τις λειτουργίες της πλατφόρμας και εφόσον επιλέξει την προσφορά που τον ενδιαφέρει, μπορεί να επιλέξει ως τρόπο πληρωμής είτε μέσω χρεωστικής κάρτας, μέσω αντικαταβολής ή PayPal. Στη συνέχεια του αποστέλλεται ένα κουπόνι και αργότερα είτε μπορεί να το εκτυπώσει και να το επιδείξει στο εστιατόριο που έχει επιλέξει για παράδειγμα, είτε κατά την πληρωμή, είτε να αναφέρει στον υπάλληλο που θα τον εξυπηρετήσει το ονοματεπώνυμο του και τον κωδικό που αναγράφεται πάνω στο κουπόνι του. Η ανάπτυξη του site έχει γίνει σε γλώσσα PHP.



Εικόνα 2. Στιγμιότυπο του Deals365 (<https://www.deals365.gr/>)

happydeals.gr

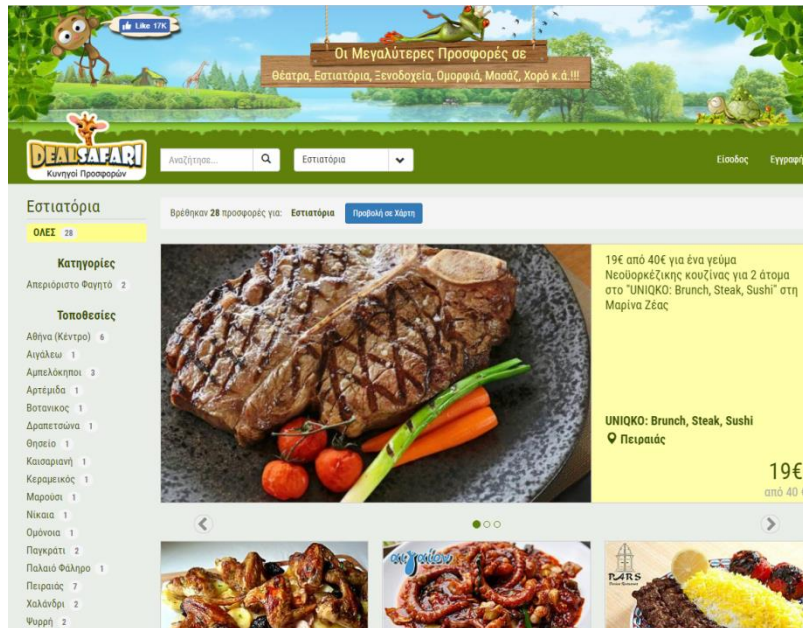
Το happydeals.gr, είναι μια ακόμη διαδικτυακή πλατφόρμα προσφορών, δημιουργημένη με τη γλώσσα PHP. Η πλατφόρμα αυτή, έχει διάφορες κατηγορίες, όπως διασκέδαση, φαγητό, ταξίδια κτλ. και δίνει τη δυνατότητα στον χρήστη να επιλέξει την περιοχή που επιθυμεί για να δει τις προσφορές που υπάρχουν σε αυτή. Επιπλέον, κάνει διάφορες προτάσεις προσφορών και ακόμη δίνει τη δυνατότητα στο χρήστη, εφόσον έχει προτείνει τη σελίδα σε φίλους του, να κερδίζει 1 ευρώ σε κάθε πρώτη αγορά που κάνουν οι φίλοι του, στους οποίους έχει προτείνει τη συγκεκριμένη πλατφόρμα προσφορών. Ο χρήστης επιλέγει τον τρόπο πληρωμής που επιθυμεί, είτε μέσω χρεωστικής/πιστωτικής κάρτας, αντικαταβολής, κατάθεσης στην τράπεζα, PayPal κτλ. Υπάρχει ένα πεδίο που ονομάζεται «Κουπόνια» και εκεί μπορεί να δει το σύνολο των αγορασμένων κουπονιών του. Για να λάβει την προσφορά του, χρειάζεται να εκτυπώσει και να προσκομίσει το κουπόνι, στην επιχείρηση από την οποία έχει αγοράσει την προσφορά που τον ενδιέφερε.



Εικόνα 3. Στιγμιότυπο του happydeals.gr (<https://www.happydeals.gr/>)

DEALSAFARI

Το συγκεκριμένο site δίνει τη δυνατότητα αναζήτησης προσφορών από το χρήστη, περιλαμβάνει διάφορες κατηγορίες, όπως και οι προηγούμενες ιστοσελίδες που παρουσιάστηκαν, και τις οποίες μπορεί επιπλέον να επιλέξει ο χρήστης από ένα dropdown που υπάρχει στην αρχική σελίδα. Ο τρόπος πληρωμής γίνεται είτε με κάρτα, είτε με αντικαταβολή, είτε με PayPal. Σε αυτή την πλατφόρμα, το κουπόνι που έχει αγοράσει ο χρήστης, εφόσον πληρώσει με κάρτα, του αποστέλλεται στο email του. Αν επιλέξει να πληρώσει με αντικαταβολή, του αποστέλλεται το κουπόνι μέσω courier στη διεύθυνσή του και καταβάλλει το αντίτιμο στο courier με οικονομική επιβάρυνση 3,5 ευρώ.



Εικόνα 4. Στιγμιότυπο του dealsafari (<https://www.dealsafari.gr>)

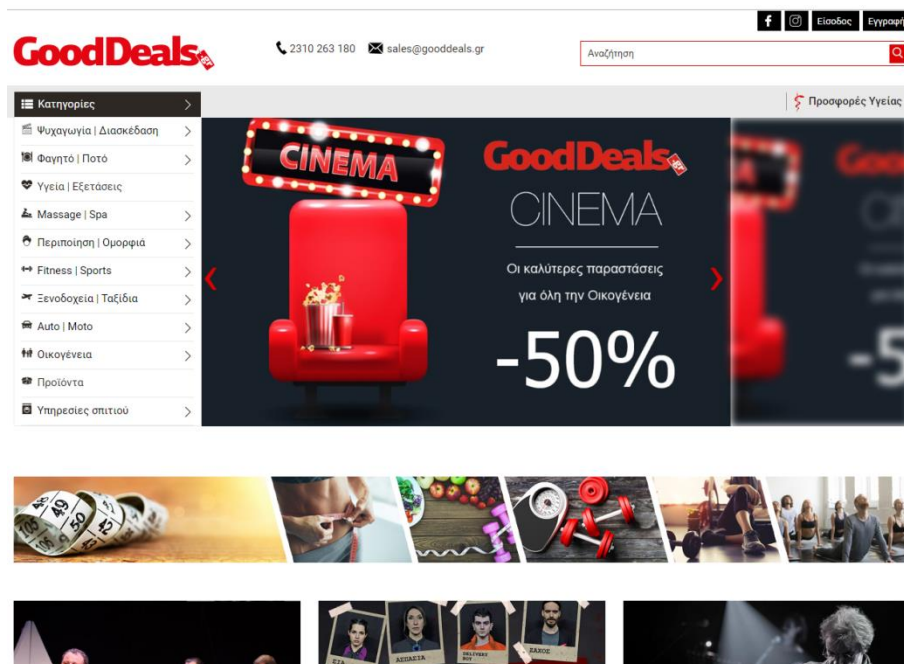
OLADeals

Στην πλατφόρμα προσφορών OLADeals, δίνεται η δυνατότητα στο χρήστη να επιλέξει την προσφορά που επιθυμεί αφού έχει εγγραφεί στο site. Μπορεί να επιλέξει την περιοχή που τον ενδιαφέρει και μια από τις πολλές κατηγορίες προσφορών που δίνονται από το site. Υπάρχει η επιλογή προβολής προσφορών που λήγουν άμεσα, best sellers προσφορών, αλλά και όσων έχουν βαθμολογηθεί υψηλά από τους χρήστες που τις έχουν ήδη αγοράσει, εφόσον δίνεται επιπλέον η δυνατότητα βαθμολόγησης. Σε κάθε προσφορά, αναγράφεται το πλήθος αγορών, η βαθμολογία τους κτλ. Επιπλέον, υπάρχει η επιλογή εύρους τιμής προσφοράς που επιθυμεί να αγοράσει ο χρήστης. Για παράδειγμα μπορεί να επιλέξει να του εμφανιστούν προσφορές που κοστίζουν λιγότερο από 50 ευρώ κτλ. Η διαφορά αυτού του site από τα υπόλοιπα που παρουσιάστηκαν είναι ότι, συγκεντρώνει όλες τις προσφορές που υπάρχουν ήδη σε άλλα site προσφορών. Οπότε όταν ο χρήστης επιλέξει κάποια από τις προσφορές που παρουσιάζονται ανακατευθύνεται στο site που παρέχει εξ αρχής την προσφορά. Οπότε ο ρόλος του είναι να τις συγκεντρώνει όλες και να τις παρουσιάζει.

Εικόνα 5. Στιγμιότυπο του OLADeals (<https://www.oladeals.gr/>)

GoodDeals

Η GoodDeals αποτελεί μια διαδικτυακή πλατφόρμα προσφορών διαφόρων κατηγοριών. Για παράδειγμα υπάρχουν προσφορές για θέατρο, κινηματογράφο, εστιατόρια, ζαχαροπλαστική κτλ. Λειτουργεί με αρκετά παρόμοιο τρόπο με τις υπόλοιπες πλατφόρμες προσφορών που περιγράφηκαν παραπάνω. Δηλαδή, δίνει τη δυνατότητα αναζήτησης προσφοράς από το πεδίο *Αναζήτηση* και η αγορά μπορεί να γίνει εφόσον έχει εγγραφεί ο χρήστης. Ακόμη, υπάρχει δυνατότητα πληρωμής είτε μέσω χρεωστικής/πιστωτικής κάρτας, είτε μέσω PayPal και το κουπόνι αποστέλλεται μέσω mail. Ο χρήστης χρειάζεται να εκτυπώσει το κουπόνι που θα του σταλεί, ώστε να λάβει την προσφορά που έχει αγοράσει.



Εικόνα 6. Στιγμιότυπο του GoodDeals (<https://www.gooddeals.gr/>)

1.5 Σύγκριση Συστημάτων

Στο διαδίκτυο υπάρχουν δεκάδες εφαρμογές που παρέχουν προσφορές και δε θα μπορούσαν να αναφερθούν εδώ όλες. Τα συστήματα τα οποία περιγράφηκαν παραπάνω, είναι τα γνωστότερα site προσφορών που υπάρχουν αυτή τη στιγμή στην Ελλάδα. Όπως παρατηρήθηκε, όλα είναι κάτι που μπορεί να θεωρηθεί ως θετικό στοιχείο για αυτά τα site. Από την άλλη πλευρά όμως, κάτι τέτοιο μπορεί να κουράσει το χρήστη, ιδιαίτερα όταν του αποστέλλονται συνεχώς ενημερώσεις για προσφορές για προσφορές κατηγοριών που δεν τον ενδιαφέρουν και σίγουρα δε δίνει συγκεκριμένο χαρακτήρα στο site προσφορών. Από τις προαναφερθείσες εφαρμογές ξεχωρίζουν αυτή της Regroup.gr καθώς και το Deals365.

Η Regroup.gr έχει μεγάλο πλήθος προσφορών και δίνει τη δυνατότητα στο χρήστη να αγοράσει την προσφορά που επιθυμεί με τους περισσότερους δυνατούς τρόπους, όπως χρήση πιστωτικής ή χρεωστικής κάρτας, ενώ δέχεται πολλούς διαφορετικούς τύπους καρτών όπως MasterCard, Diners Club, American Express, Cirrus, MasterPass, Maestro, Visa. Επιπλέον, δίνει τη δυνατότητα πληρωμής μέσω VivaWallet, καθώς έχει και συνεργασία και με την τράπεζα Eurobank. Ο χρήστης που κάνει την πληρωμή του με κάρτα Eurobank στη Regroup, έχει επιστροφή 3% μέσω της εφαρμογής «Επιστροφή» της Eurobank.

Η διαδικτυακή πλατφόρμα προσφορών Deals365, εκτός του ότι είναι προσιτή σχεδιαστικά στο χρήστη, δίνει τη δυνατότητα στον χρήστη να κάνει και τηλεφωνική παραγγελία. Επιπλέον, σε περίπτωση που δεν έχει εκτυπωμένο κουπόνι, μπορεί με τον κωδικό και το ονοματεπώνυμό του να λάβει την προσφορά που έχει αγοράσει.

2. Τεχνολογίες διαδικτυακών εφαρμογών

Τα θεμέλια του παγκόσμιου ιστού αποτελούνται από ένα παγκόσμιο σύστημα υπερκειμένων, τα βασικά συστατικά του οποίου είναι:

URL: Ένα παγκόσμιο μέσο διευθυνσιοδότησης πληροφοριακών οντοτήτων.

HTTP: Ένα πρωτόκολλο για επικοινωνίες πελάτη – διακομιστή χωρίς καταγραφή της κατάστασης (stateless).

HTML: Γλώσσα σήμανσης για την παρουσίαση περιεχομένου υπερκειμένου σε συσκευές διαφόρων τύπων.

Το πρωτόκολλο HTTP πελάτη-διακομιστή, εφαρμόζεται μεταξύ των πελατών ιστού (WebBrowsers) και διαδικτυακών διακομιστών (WebServers). Οι διευθύνσεις URL

χρησιμοποιούνται από το πρωτόκολλο HTTP για τον εντοπισμό πόρων ή διαδικασιών που πρέπει να εκτελεστούν από τον διακομιστή Web, προκειμένου να εντοπιστεί το επιθυμητό περιεχόμενο που πρέπει να διοχετευτεί στον πελάτη. Τα αιτήματα HTTP κωδικοποιούν και μεταφέρουν τις παραμέτρους για τις διαδικασίες που πρέπει να εκτελεστούν και επιστρέφουν τα αποτελέσματα στον πελάτη. Τα αποτελέσματα μορφοποιούνται στο μορφότυπο HTML για παρουσίαση στον πελάτη. Ο φυλλομετρητής κατανοεί τις ετικέτες δομής και παρουσίασης της HTML και τις χρησιμοποιεί για παρουσίαση εγγράφων σε συσκευές διαφορετικών τύπων. Η επεξεργασία των εισερχόμενων αιτημάτων HTTP στον διαδικτυακό διακομιστή πραγματοποιείται συνήθως από Servlets (συνήθως γραμμένα σε Java), τα οποία επεξεργάζονται από την Servlet Engine του διακομιστή. Η παραγόμενη έξοδος HTML που θα αποσταλεί στο φυλλομετρητή για παρουσίαση, γίνεται συνήθως με τη μορφή JSP (Javaσελίδες διακομιστή) από μια μηχανή JSP στον διακομιστή. Υπάρχουν επίσης εναλλακτικές τεχνολογίες που μπορούν να χρησιμοποιηθούν στο διακομιστή Web για προετοιμασία των δεδομένων που θα τροφοδοτήσουν το φυλλομετρητή.

Το 1997 παρουσιάστηκε η έκδοση 4 της HTML και επίσης ο μηχανισμός Cascading Style Sheets (CSS). Με τη χρήση CSS έγινε δυνατός ο διαχωρισμός των πληροφοριών παρουσίασης (κανόνες στυλ των CSS) από τη δομή του περιεχομένου ιστού που ορίζεται από τις ετικέτες HTML.

Η εμφάνιση της γλώσσας JavaScript και του HTML Document Object Model (DOM), δημιούργησαν τη δυνατότητα οι σελίδες που προβάλλονται στο φυλλομετρητή να αντιδρούν σε γεγονότα. Το DOM είναι μια προγραμματιστική διεπαφή (API) σε ένα έγγραφο HTML το οποίο έχει φορτωθεί στο φυλλομετρητή και χρησιμοποιείται για να χειρίζεται την HTML μέσα από εντολές JavaScript. Ο κώδικας JavaScript μπορεί να χρησιμοποιηθεί για παράδειγμα για να ελέγξει για σφάλματα στο φυλλομετρητή όταν ο χρήστης εισάγει τιμές σε ένα πεδίο της σελίδας, αντί να στείλει τις τιμές που πρέπει να ελεγχθούν στον διακομιστή Web. Εάν εντοπιστεί ένα σφάλμα (για παράδειγμα αν εντοπιστεί σφάλμα όταν ο χρήστης εισάγει αλφαβητικούς χαρακτήρες σε ένα πεδίο τηλεφωνικού αριθμού της σελίδας), το σφάλμα εντοπίζεται από τον κώδικα JavaScript στον Browser και η HTML της σελίδας τροποποιείται ώστε να παρουσιάσει ένα μήνυμα σφάλματος. Επίσης, μπορεί να εκτελεστεί μια ενέργεια επί των δεδομένων χωρίς να σταλούν τα λανθασμένα δεδομένα στο διακομιστή, να γίνει ο εντοπισμός του σφάλματος σε αυτόν και να παραχθεί μια νέα σελίδα με το μήνυμα σφάλματος και τη διορθωτική ενέργεια που πρέπει να εκτελεστεί, η οποία πρέπει να σταλεί εκ νέου στο φυλλομετρητή.

Αρχικά, οι ιστότοποι παρείχαν κυρίως περιεχόμενο. Μέχρι το 2004, η διάκριση μεταξύ περιεχομένου και εφαρμογών δεν ήταν σαφώς καθορισμένη. Για παράδειγμα, οι κατάλογοι αγορών παρείχαν ιστότοπους δημοπρασιών σε πραγματικό χρόνο, επιτρέποντας στους χρήστες να κάνουν αγορές και πωλήσεις. Οι διαδικτυακές εφαρμογές έγιναν πολύ σημαντικές για τη βιομηχανία λογισμικού. Συνήθως περιλαμβάνουν σύνθετη επιχειρησιακή λογική, η οποία υποστηρίζεται από σύνθετες δομές πλοήγησης στη βάση δεδομένων για την παραγωγή πολλαπλών εξόδων και προβολών παρουσίασης σε διαφορετικές συσκευές. Επίσης υποστηρίζουν τη συμμετοχή και τη συνεργασία των χρηστών στο πλαίσιο της διαδικτυακής εφαρμογής. Πολλές διαδικτυακές εφαρμογές έχουν χαρακτήρα δοσοληψιών και η λογική της εφαρμογής και η διεπαφή πρέπει να υποστηρίζουν την εκτέλεση δοσοληψιών (για παράδειγμα, τα κουμπιά αποχώρησης από μία σελίδα («Πίσω») είναι ανενεργά κατά τη διάρκεια εκτέλεσης της δοσοληψίας. Μια εφαρμογή Ιστού μπορεί να χρησιμοποιεί δεδομένα από διαφορετικές βάσεις δεδομένων οι οποίες βρίσκονται σε διαφορετικές τοποθεσίες. Η λογική διεκπεραίωσης δοσοληψιών θα πρέπει να εγγυάται τη μετατροπή των δεδομένων σε μορφές κοινά αντιληπτές από τα μέρη που επικοινωνούν, καθώς και την ορθότητα των πληροφοριών στο καταναεμημένο περιβάλλον. Σε αυτό το μεταβαλλόμενο περιβάλλον, τα HTML, DOM και JavaScript, αποτελούν το βασικό μοντέλο προγραμματισμού των διαδικτυακών εφαρμογών.

Οι συμβατικές διαδικτυακές εφαρμογές είναι ουσιαστικά Εφαρμογές Επεξεργασίας Φορμών δεδομένων (Form Processing Applications). Η αλληλεπίδραση του χρήστη συνήθως οδηγεί σε αίτημα στον διακομιστή διαδικτύου, ώστε να αλλάξει ολόκληρη τη σελίδα που βλέπει ο χρήστης. Στο μοντέλο δυναμικής HTML, αυτό συνήθως περιλαμβάνει την ανάκτηση δεδομένων από τη βάση δεδομένων με τα οποία συμπληρώνονται τα διάφορα πεδία φόρμας και τη δημιουργία μιας νέας σελίδας HTML, που αποστέλλεται στο φυλλομετρητή του χρήστη. Δεν υπάρχει δυνατότητα αντικατάστασης μόνο των μερών της φόρμας. Ένα μειονέκτημα του μοντέλου είναι ότι δεν υποστηρίζει επεξεργασία πολυμέσων εγγενώς στο φυλλομετρητή.

Πιο πρόσφατα αναπτύχθηκε το μοντέλο Εμπλουτισμένων Εφαρμογών εφαρμογών Διαδικτύου (Rich Internet Applications), για την κάλυψη αυτής της ανάγκης. Οι εμπλουτισμένες διαδικτυακές εφαρμογές χρησιμοποιούν μερικές ειδικές προσθήκες που υπάρχουν εκτός του προγράμματος περιήγησης στο διαδίκτυο (όπως το Flash και άλλα), προκειμένου να επιτευχθεί αυτή η αναπαραγωγή και διαχείριση πολυμέσων. Ωστόσο, τα πρόσθετα εκτός του προγράμματος περιήγησης εμφανίζουν συχνά προβλήματα επειδή δεν υποστηρίζονται από τα πρότυπα και δημιουργούν ζητήματα ασφάλειας. Επιπλέον, δεδομένου ότι αυτά τα πρόσθετα δεν βασίζονται σε πρότυπα, αλλά αποτελούν υλοποιήσεις των εταιρειών που τα υποστηρίζουν, δεν είναι εγγυημένη η ύπαρξή τους ως λειτουργικότητα και υποστήριξη σε όλες τις λειτουργικές πλατφόρμες. Για παράδειγμα, η Adobe ανακοίνωσε ότι το Flash δεν πρόκειται να αναπτυχθεί περαιτέρω και να υποστηριχθεί στο μέλλον (Adobe, 2011). Παρ' όλα αυτά, αυτά τα πρόσθετα επέτρεψαν τη δημιουργία κάποιων εξειδικευμένων, διαδικτυακών εφαρμογών, παρόλο που αυτές ήταν εξαρτημένες από τα πρόσθετα αυτά.

Πρόσφατα, στο μοντέλο εμπλουτισμένων διαδικτυακών εφαρμογών, προωθούνται οι τεχνολογίες AJAX για να αποφευχθεί η ανανέωση ολόκληρης της σελίδας από το διακομιστή. Το AJAX βασίζεται σε ένα ασύγχρονο πρωτόκολλο επικοινωνίας του πελάτη με το διακομιστή. Στις διαδικτυακές εφαρμογές, το πρόγραμμα-πελάτης (φυλλομετρητής) μπορεί να αποστέλλει ασύγχρονο αίτημα για αντικατάσταση τμημάτων της σελίδας στον διακομιστή, ενώ ο χρήστης συνεχίζει να εργάζεται στην ίδια σελίδα. Όταν τα αποτελέσματα φτάσουν πίσω, ενσωματώνονται στο συγκεκριμένο τμήμα της σελίδας που αντιστοιχούν. Το AJAX χρησιμοποιεί XML και ασύγχρονο πρωτόκολλο HTTP για την υλοποίηση ασύγχρονων διαδραστικών εφαρμογών βασισμένων σε JavaScript. Το πρωτόκολλο XML Http Request δεν είναι ακόμα πρότυπο, παρόλο που η ένωση World Wide Web (W3C) κυκλοφόρησε το πρώτο πρότυπο προδιαγραφής για αυτό, προσπαθώντας να το επιβάλλει ως πρότυπο στον παγκόσμιο ιστό. Η χρήση τεχνολογιών AJAX, από τα προγράμματα περιήγησης υποστηρίζει τη μερική ανανέωση της σελίδας στο περιβάλλον εργασίας του χρήστη. Η τεχνολογία αυτή καθιστά εφικτή την εμφάνιση διαδραστικών κινούμενων εικόνων σε τμήματα μιας σελίδας. Όταν χρησιμοποιείται η XML για μετάδοση δεδομένων, χρησιμοποιείται το XSLT για τον μετασχηματισμό των δεδομένων αυτών. Ωστόσο, δεν είναι απαραίτητο τα δεδομένα να μεταδίδονται σε μορφή ογκωδών XML αρχείων. Τα δεδομένα προς το φυλλομετρητή πλέον μεταδίδονται με τον πιο συμπαγή μορφότυπο JSON (JavaScript Object Notation). Οι τεχνολογίες AJAX εμφανίζουν το μειονέκτημα ότι ενδέχεται να δημιουργήσουν μεγάλο αριθμό αιτημάτων προς το διακομιστή. Ένα άλλο μειονέκτημα είναι ότι οι crawlers δεν μπορούν να χρησιμοποιήσουν τη JavaScript για να κάνουν ευρετηρίαση του περιεχομένου των σελίδων.

2.1 Αρχιτεκτονικά μοντέλα στη σχεδίαση διαδικτυακών εφαρμογών και διαδικτυακών υπηρεσιών

Οι διαδικτυακές εφαρμογές είναι πολύπλοκα συστήματα λογισμικού τα οποία πρέπει να σχεδιαστούν σύμφωνα με συγκεκριμένη Αρχιτεκτονική Λογισμικού² (St. Laurent, 2014), ώστε να είναι σε θέση να υλοποιήσουν όλες τις προβλεπόμενες λειτουργίες και να ελαχιστοποιηθούν οι πόροι που είναι απαραίτητοι για τη συντήρησή τους στο μέλλον. Ο ορισμός της Αρχιτεκτονικής Λογισμικού που βασίζεται στο βιβλίο των Mary Shaw και David Garlan και άλλων είναι: Η Αρχιτεκτονική Λογισμικού περιλαμβάνει τη συγκρότηση μιας δέσμης σημαντικών αποφάσεων για την οργάνωση ενός συστήματος λογισμικού που περιλαμβάνει³ (David & Mary, 1994):

- την επιλογή των δομικών στοιχείων και των διεπαφών από τα οποία αποτελείται το σύστημα.
- τη συνεργασία μεταξύ αυτών των στοιχείων
- τη σύνθεση αυτών των δομικών και λειτουργικών στοιχείων σε μεγαλύτερα υποσυστήματα
- ένα αρχιτεκτονικό πρότυπο, το οποίο καθοδηγεί αυτή την οργάνωση.

Η αρχιτεκτονική λογισμικού περιλαμβάνει επίσης λειτουργικότητα, ευχρηστία, ανθεκτικότητα, απόδοση, επαναχρησιμοποίηση, κατανοητότητα, οικονομικούς και τεχνολογικούς περιορισμούς, συμβιβασμούς και αισθητικές «ανησυχίες».

² <https://www.oreilly.com/ideas/web-application-development-is-different-and-better>

³ https://www.cs.cmu.edu/afs/cs/project/vit/ftp/pdf/intro_softarch.pdf

Ένα αρχιτεκτονικό πρότυπο ή αρχιτεκτονικό μοτίβο, είναι ένα σύνολο αρχών το οποίο δημιουργεί ένα αφηρημένο πλαίσιο το οποίο μπορεί να εφαρμοστεί σε μια οικογένεια συστημάτων. Ένα αρχιτεκτονικό μοτίβο δημιουργεί τις συνθήκες για την κατάτμηση ενός συστήματος και ενθαρρύνει την επαναχρησιμοποίηση σχεδιαστικών λύσεων. Επιταχύνοντας έτσι τη λύση σε επαναλαμβανόμενα προβλήματα.

Τα Αρχιτεκτονικά Μοτίβα μπορεί να θεωρηθούν σύνολα αρχών, τα οποία διαμορφώνουν μια εφαρμογή. Οι Garlan και Shaw ⁴ορίζουν ένα αρχιτεκτονικό μοντέλο ως: «μια οικογένεια συστημάτων από την άποψη της δομικής οργάνωσης». Πιο συγκεκριμένα, ένα αρχιτεκτονικό μοντέλο καθορίζει το λεξιλόγιο των δομικών στοιχείων και των διασυνδέσεων που μπορούν να χρησιμοποιηθούν σε περιπτώσεις εφαρμογής του, μαζί με ένα σύνολο περιορισμών για το πώς αυτά τα δομικά στοιχεία μπορούν να συνδυαστούν.

Το Layered Architectural Pattern [Microsoft 2009] επικεντρώνεται στην ομαδοποίηση της συναφούς λειτουργικότητας μιας εφαρμογής σε ξεχωριστά στρώματα. Η λειτουργικότητα σε κάθε στρώμα συνδέεται με έναν κοινό ρόλο ή έργο. Τα κύρια πλεονεκτήματα του Layered Architectural Pattern περιλαμβάνουν:

- **Αφαίρεση:** Τα επίπεδα επιτρέπουν στις αλλαγές να πραγματοποιούνται στο αφηρημένο επίπεδο. Το επίπεδο αφαίρεσης που χρησιμοποιείται σε κάθε στρώμα της ιεραρχικής στοίβας μπορεί να αυξηθεί ή να μειωθεί.
- **Απομόνωση:** Επιτρέπει την απομόνωση τεχνολογικών αναβαθμίσεων σε επιμέρους στρώματα, ώστε να μειωθούν οι κίνδυνοι και να ελαχιστοποιηθούν οι συνέπειες στο συνολικό σύστημα.
- **Διαχειρισιμότητα:** Ο διαχωρισμός των βασικών λειτουργιών βοηθά στον προσδιορισμό των εξαρτήσεων και την οργάνωση του κώδικα σε πιο διαχειρίσιμα τμήματα.
- **Απόδοση:** Η κατανομή των επιπέδων του μοντέλου σε πολλαπλά φυσικά στρώματα, μπορεί να βελτιώσει την επεκτασιμότητα, την ανεκτικότητα σε σφάλματα και την απόδοση του συνολικού συστήματος.
- **Επαναχρησιμοποίηση:** Ένα φυσικό στρώμα μπορεί να επαναχρησιμοποιηθεί σε άλλες συναφείς υλοποιήσεις, οι οποίες ενδεχομένως απαιτούν διαφορετικές προβολές των ίδιων δεδομένων και λειτουργιών.
- **Ευκολία δοκιμών:** Η ευκολία δοκιμών του συστήματος οφείλεται στις σαφώς προσδιορισμένες λειτουργίες κάθε στρώματος. Τα ξεχωριστά πρότυπα παρουσίασης των δεδομένων επιτρέπουν τη δημιουργία αντικείμενων που μιμούνται τη συμπεριφορά των πραγματικών.

Ενδεικτικά, μια πολυεπίπεδη αρχιτεκτονική Client Server περιλαμβάνει δύο επίπεδα: πελάτη και διακομιστή. Σε εφαρμογές βάσεων δεδομένων, αυτή η αρχιτεκτονική χρησιμοποιείται για να διαχωρίσει τη διεπαφή από τις διεργασίες διαχείρισης των δεδομένων. Χρησιμοποιείται επίσης για την κατανομή της λειτουργικότητας του συστήματος σε διαφορετικές υπολογιστικές μηχανές.

Για τις διαδικτυακές εφαρμογές, ο διαχωρισμός σε δύο επίπεδα δεν είναι επαρκής. Στο πεδίο αυτό έχουν προταθεί πολυστρωματικές αρχιτεκτονικές. Ενδεικτικά, η αρχιτεκτονική τριών επιπέδων χωρίζει το στρώμα διαχείρισης δεδομένων που είναι υπεύθυνο για τη διαχείριση των δεδομένων σε μόνιμη αποθήκευση από το στρώμα επιχειρησιακής λογικής, το οποίο είναι υπεύθυνο για την υλοποίηση των επιχειρησιακών διαδικασιών (εφαρμογή επιχειρηματικών κανόνων και πολιτικών και αλλαγή της κατάστασης των επιχειρηματικών δεδομένων). Το τρίτο στρώμα είναι το επίπεδο διεπαφής χρήστη. Αυτός ο διαχωρισμός επιτρέπει ουσιαστικά στο στρώμα Επιχειρησιακής Λογικής να επικοινωνεί με περισσότερες από μία βάσεις δεδομένων και επίσης να εφαρμόζει αποτελεσματικές στρατηγικές για την κλιμάκωση (διαχείριση διακομιστών – ειδύλων (replication servers) και εφαρμογή στρατηγικών εξισορρόπησης φορτίου).

⁴ https://www.cs.cmu.edu/afs/cs/project/able/ftp/intro_softarch/intro_softarch.pdf

Στη βιβλιογραφία έχουν περιγραφεί πολλά αρχιτεκτονικά πρότυπα για διαδικτυακές εφαρμογές⁵ (Martin&Michael , 2017), (Microsoft, 2018). Ένα πολύ υψηλού επιπέδου αρχιτεκτονικό μοτίβο που καλύπτει τη συνολική αρχιτεκτονική των Διαδικτυακών Εφαρμογών, είναι το πρότυπο BCED (οριοθέτηση, έλεγχος, οντότητα, διεπαφή βάσης δεδομένων). Είναι μια ειδική περίπτωση του διαστρωματωμένου αρχιτεκτονικού μοντέλου, (Layered Architectural Pattern). Το πρότυπο BCED προωθεί τη δόμηση των εφαρμογών διαδικτύου σε τέσσερα επίπεδα:

- Το **Επίπεδο Ορίου** (Boundary Layer) διαχειρίζεται τη διεπαφή του χρήστη με τη διαδικτυακή εφαρμογή.
- Το **Επίπεδο Ελέγχου** (ControlLayer) είναι υπεύθυνο για την διαχείριση της εισόδου του χρήστη και τη μεταφορά του ελέγχου στα κατάλληλα τμήματα της εφαρμογής, τα οποία θα εφαρμόσουν την επιχειρησιακή λογική.
- Το **Επίπεδο Οντοτήτων** (EntityLayer) περιλαμβάνει τις οντότητες (αντικείμενα) που θα εκτελέσουν την επιχειρησιακή λογική και θα αλλάξουν τη μακροπρόθεσμη κατάσταση του συστήματος. Τα αντικείμενα του Επιπέδου Οντοτήτων ανακτώνται από ένα χώρο μόνιμης αποθήκευσης και οι τροποποιημένες τιμές αποθηκεύονται επίσης σε χώρο μόνιμης αποθήκευσης μετά την εκτέλεση των μεθόδων του αντικειμένου.
- Το **Επίπεδο Διεπαφής Βάσης Δεδομένων** (DataBaselInterfaceLayer) είναι υπεύθυνο για την επικοινωνία με το χώρο μόνιμης αποθήκευσης δεδομένων (βάση δεδομένων), με σκοπό την ανάκτηση και την αποθήκευση δεδομένων πίσω σε αυτόν. Τα δεδομένα στις βάσεις δεδομένων είναι συνήθως σε μορφή σχεσιακού μοντέλου (σχέσεις, πλειάδες, κλπ.) και μετατρέπονται σε μορφή αντικειμενοστρεφούς μορφής για χρήση τους από το Επίπεδο Οντοτήτων.

Η βάση δεδομένων καθαυτή, δεν περιγράφεται από το πρότυπο BCED. Οι Βάσεις Δεδομένων ενδέχεται να προϋπάρχουν και η διαδικτυακή εφαρμογή ενδέχεται να συγκεντρώνει και να μετατρέπει δεδομένα από διάφορες βάσεις δεδομένων για χρήση στις επιχειρησιακές διαδικασίες που υλοποιεί το σύστημα. Αυτή η λειτουργία διεκπεριώνεται στο Επίπεδο Διεπαφής Βάσης Δεδομένων, το οποίο είναι σαφώς διακριτό από το επίπεδο οντοτήτων. Το επίπεδο οντοτήτων αντιπροσωπεύει και διαχειρίζεται το περιεχόμενο των αντικειμένων της διαδικτυακής εφαρμογής κατά το χρόνο εκτέλεσης. Το Επίπεδο Οντοτήτων και το Επίπεδο Ελέγχου, εφαρμόζουν την επιχειρησιακή λογική της εφαρμογής.

Υπάρχουν πολλά άλλα πρότυπα που έχουν περιγραφεί στη βιβλιογραφία και τα οποία εφαρμόζονται για το σχεδιασμό τμημάτων της συνολικής αρχιτεκτονικής και όχι της συνολικής αρχιτεκτονικής.

Το μοντέλο **Model – View - Controller (MVC)** είναι ένα από τα πολλά μοντέλα που προτείνουν το διαχωρισμό της διεπαφής από τη διαχείριση του περιεχομένου. Περιγράφει τρία στοιχεία της αρχιτεκτονικής, το Μοντέλο (Model) με ευθύνη τη διαχείριση των δεδομένων, την Παρουσίαση (View) με ευθύνη την παρουσίαση των δεδομένων στη διεπαφή και τον Ελεγκτή (Controller), ο οποίος είναι υπεύθυνος να ειδοποιήσει το μοντέλο για τυχόν αλλαγές που προκαλούνται στα δεδομένα ως αποτέλεσμα της αλληλεπίδρασης του χρήστη με το περιβάλλον εργασίας. Το σημαντικό σημείο εδώ είναι, ότι το Μοντέλο δεν έχει επαφή ούτε με την Προβολή ούτε με τον Ελεγκτή. Αυτό επιτρέπει στο Μοντέλο να είναι ανεξάρτητο από την Προβολή και τον Ελεγκτή. Αν αυτό δεν ληφθεί υπόψη και το Μοντέλο καλεί την Προβολή για να της υπομνήσει αλλαγές στο Μοντέλο, κάθε φορά που πρέπει να προστεθεί μια νέα συσκευή πελάτη, ο κώδικας του Μοντέλου θα πρέπει να αλλάζει για να υποστηρίξει τη νέα συσκευή.

Τα περισσότερα μοντέλα διεπαφής χρήστη βασίζονται στο πρότυπο επεξεργασίας συμβάντων (Events). Οι αλληλεπιδράσεις των χρηστών με τη διεπαφή παράγουν συμβάντα διαφορετικών τύπων τα οποία επεξεργάζονται από τον Ελεγκτή. Για παράδειγμα, ένας τύπος συμβάντος μπορεί να υποδεικνύει ότι το ποντίκι πέρασε πάνω από ένα κουμπί στο περιβάλλον του φυλλομετρητή. Ένα άλλο συμβάν μπορεί να υποδεικνύει ότι ο χρήστης πάτησε το πλήκτρο. Τα συμβάντα της διεπαφής χρήστη στο μοντέλο MVC επεξεργάζονται από τον Ελεγκτή. Η επεξεργασία μπορεί να έχει ως αποτέλεσμα ο Ελεγκτής να ζητήσει την τροποποίηση της

⁵ <https://docs.microsoft.com/en-us/dotnet/standard/modern-web-apps-azure-architecture/common-web-application-architectures>

διεπαφής χρήστη (για παράδειγμα για να υποδείξει ότι το κουμπί είναι πατημένο). Μπορεί επίσης να έχει ως αποτέλεσμα ο Ελεγκτής να καλέσει το μοντέλο επειδή ο χρήστης έχει προσθέσει ή διαγράψει κάποια δεδομένα.

Το μοντέλο (**MVP**) δεν λειτουργεί με κυκλικό τρόπο όπως το μοντέλο MVC. Στηρίζεται στον Παρουσιαστή (Presenter), για να συντονίσει τα πάντα, να λάβει δεδομένα από τα Μοντέλα και να δημιουργήσει τις Προβολές. Όταν πραγματοποιείται μια ενέργεια στην Προβολή, ο Παρουσιαστής παρακολουθεί τη δράση και συντονίζει την εργασία με άλλες υπηρεσίες, στέλνοντας τις αλλαγές πίσω στο Μοντέλο. Όταν το Μοντέλο είναι ενημερωμένο, ο Παρουσιαστής λαμβάνει τις αλλαγές σε αυτό και πραγματοποιεί τις διορθώσεις στην Προβολή.

Το μοντέλο **Publish Subscribe** σχετίζεται επίσης με το σχεδιασμό της διεπαφής χρήστη και το μοτίβο MVC. Το μοντέλο προτείνει μια λύση για τις τροποποιήσεις που συμβαίνουν στο Μοντέλο και πώς αυτές μεταδίδονται στην Προβολή. Δεδομένου ότι το Μοντέλο δεν μπορεί να καλέσει την Προβολή, η Προβολή μπορεί να καλεί περιοδικά το Μοντέλο για να διαπιστώσει εάν υπάρχουν αλλαγές που πρέπει να αντικατοπτρίζονται στη διεπαφή. Αυτό θα δημιουργούσε μια τεράστια κίνηση στο Μοντέλο, ακόμη και αν δε συμβαίνουν αλλαγές σ' αυτό. Μια λύση για την αποφυγή περιττής κυκλοφορίας, είναι η δημιουργία μιας λίστας συνδρομητών στο Μοντέλο, όπου συνδρομητές είναι οι πελάτες που ενδιαφέρονται να ειδοποιηθούν για μια αλλαγή συγκεκριμένου τύπου σε αυτό. Κάθε φορά που υπάρχουν αλλαγές στα δεδομένα του Μοντέλου, ένα γεγονός αλλαγής συγκεκριμένου τύπου αποστέλλεται στη λίστα συνδρομητών και ειδοποιούνται όλοι οι συνδρομητές που έχουν εγγραφεί για το συγκεκριμένο τύπο συμβάντος. Ο τύπος συμβάντος μπορεί να προσδιοριστεί από τα δεδομένα που βλέπει ο χρήστης στη διεπαφή του. Το μοντέλο Publish Subscribe, εφαρμόζεται σε πολλές διαφορετικά αρχιτεκτονικά επίπεδα του σχεδιασμού, και όχι μόνο στο περιβάλλον χρήστη. Βασίζεται στην αρχή του Χόλιγουντ: «Μη μας τηλεφωνείτε. Θα σας καλέσουμε». Ορισμένες φορές, χρησιμοποιείται ο όρος Observer Pattern για την περιγραφή του μηχανισμού αυτού.

Υπάρχουν διάφορα μοντέλα σχεδίασης που σχετίζονται με τον Ελεγκτή του μοντέλου BCED. Ένα από τα πιο σημαντικά είναι το **Pattern Controller Application**. Υποστηρίζει ότι πρέπει να υπάρχει ένα κεντρικό σημείο για το χειρισμό της πλοήγησης στην οθόνη και της ροής της διαδικτυακής εφαρμογής. Το μοντέλο Model View Controller (MVC) μπορεί να υλοποιήσει ορισμένες από αυτές τις αποφάσεις. Συνήθως όμως ένας Ελεγκτής του μοντέλου MVC δρα για μία προβολή (μία σελίδα) και έτσι δημιουργούνται πολλά αντίγραφα του Ελεγκτή. Όταν η εφαρμογή είναι περίπλοκη, αυτό μπορεί να οδηγήσει σε επανάληψη κοινών τμημάτων κώδικα, επειδή αρκετοί ελεγκτές πρέπει να γνωρίζουν πώς πρέπει να αντιδράσουν σε συγκεκριμένες συνθήκες. Επιπλέον, όταν η λογική ελέγχου είναι κατανεμημένη, είναι δύσκολο να δημιουργείται μια συνεκτική άποψη για το πώς συμπεριφέρεται η εφαρμογή. Επίσης η συντήρηση και η επέκταση της εφαρμογής γίνεται πολύ δύσκολη. Το μοντέλο Application Controller τοποθετεί όλη τη λογική ροής σε έναν Ελεγκτή Εφαρμογής, αποφεύγοντας έτσι όλα τα παραπάνω ζητήματα. Σε ορισμένες περιπτώσεις ο Ελεγκτής Εφαρμογής μπορεί να γίνει πολύ σύνθετος. Σε αυτές τις περιπτώσεις ο Ελεγκτής Εφαρμογής μπορεί να διεκπεραιώνει μόνο τον κεντρικό έλεγχο και στη συνέχεια να ενεργοποιεί εξειδικευμένους Ελεγκτές Περίπτωσης Χρήσης (Use Case Controllers), για να διεκπεραιώνει τις εργασίες των συγκεκριμένων περιπτώσεων. Επιπλέον, ο Ελεγκτής Εφαρμογών εκτελεί συχνά λειτουργίες οι οποίες είναι σχετικές με το αίτημα του πελάτη. Τέτοιες λειτουργίες σχετίζονται με την επικύρωση εάν ο πελάτης έχει εγκαθιδρύσει μια έγκυρη σύνοδο (session), εάν υποστηρίζεται ο τύπος του προγράμματος περιήγησης του πελάτη, το είδος της κωδικοποίησης δεδομένων, οι μετατροπές δεδομένων, η καταγραφή της κυκλοφορίας κλπ. Οι λειτουργίες αυτές είναι ανεξάρτητες μεταξύ τους και η σειρά με την οποία εκτελούνται είναι επίσης ακαθόριστη.

Με την έλευση τεχνολογιών όπως εφαρμογές REST, WebServices, Widgets και εφαρμογές Mashup, η οργάνωση της λειτουργικότητας που προσφέρεται είναι σημαντική, ώστε αυτή να παρουσιάζεται σε υψηλό επίπεδο σε μορφή προγραμματιστικής διεπαφής (API) και όχι σε δεκάδες αντικείμενα που απαρτίζουν την εφαρμογή. Αυτό το API θα χρησιμοποιηθεί στη συνέχεια για να υποστηρίξει την παραγωγή νέων εφαρμογών πάνω στην υποδομή που είναι ήδη διαθέσιμη από το υπάρχον σύστημα. Η διάθεση αυτής της λειτουργικότητας σε άλλα συστήματα και εφαρμογές μπορεί να υλοποιηθεί γίνει μέσω διασυνδέσεων Web Services ή REST.

Τα γενικά API που διαθέτουν την επιχειρησιακή λογική της εφαρμογής ώστε να μπορούν να επαναχρησιμοποιηθούν περιγράφονται από το Application Façade Pattern. Το

Application Façade Pattern, είναι ένα προαιρετικό στοιχείο το οποίο παρουσιάζει μια απλοποιημένη διεπαφή με τα συστατικά της επιχειρησιακής λογικής, συχνά συνδυάζοντας πολλαπλές επιχειρησιακές λειτουργίες σε μία μόνο λειτουργία.

Υπάρχουν επίσης αρκετά μοντέλα που περιγράφονται για το Στρώμα Διεπαφής Βάσης Δεδομένων (Data Base Interface Layer) του μοντέλου BCED. Το Πρότυπο Data Access Object (DAO) αναθέτει τις λειτουργίες αποθήκευσης και ανάκτησης από και προς το χώρο μόνιμης αποθήκευσης των πληροφοριακών οντοτήτων σε Data Access Objects, αντί να ενσωματώνει όλες τις λειτουργίες αποθήκευσης και ανάκτησης (που μπορεί να αφορούν μετατροπές κλπ.) στο αντικείμενο της οντότητας καθαυτό. Ένα άλλο σχετικό μοντέλο είναι το Data Transfer Object, το οποίο μπορεί να χρησιμοποιεί το Data Access Object για τη λήψη δεδομένων για την περαιτέρω κατασκευή ενός σύνθετου αντικειμένου μεταφοράς, το οποίο είναι αναγκαίο να δημιουργηθεί και να αποσταλεί στον πελάτη. Το Data Transfer Object είναι ένα χρήσιμο μοντέλο επειδή διαφορετικά τα υψηλότερα στρώματα της αρχιτεκτονικής τα οποία χρειάζονται λίστες δεδομένων από χαμηλότερα στρώματα μπορούν να χρησιμοποιούν πολλαπλές κλήσεις μεθόδων σε διαφορετικά επίπεδα.

Μια Web Service είναι μια μέθοδος επικοινωνίας μεταξύ δύο μηχανών μέσω του Ιστού. Τα τελευταία χρόνια δόθηκε μεγάλη έμφαση στον καθορισμό προτύπων και αρχιτεκτονικών για υπηρεσίες Web. Οι όροι που χρησιμοποιούνται είναι Web Services ή SOAP based Services ή Service Oriented Architectures. Η βάση τους για την ανταλλαγή δεδομένων είναι η XML και δίνεται έμφαση στην επικοινωνία μεταξύ των μηχανών με βάση το Simple Object Access Protocol (SOAP). Τα μηνύματα SOAP μεταφέρουν δεδομένα XML και συχνά χρησιμοποιούνται για την υλοποίηση πρωτοκόλλου RPC ανεξάρτητου από την πλατφόρμα. Από τη φύση της είναι έντονα τυποποιημένη προσέγγιση για την ανάπτυξη καταμεμημένου λογισμικού. Πολλά πρότυπα έχουν αναπτυχθεί και αναπτύσσονται για την υποστήριξη διαφόρων λειτουργιών των καταμεμημένων συστημάτων όπως η διάθεση προγραμματιστικών διεπαφών για κλήσεις λειτουργιών (WSDL), ασφάλειας, εξουσιοδοτήσεων, ενορχήστρωσης των υπηρεσιών, σύνθεσης των υπηρεσιών⁶ κλπ. (Alonso, et al., 2004).

Το μοντέλο σχεδιασμού και υλοποίησης υπηρεσιών της Representational State Transfer (**REST**) έχει ευρεία αποδοχή λόγω της απλότητας χρήσης, της μικρότερης καμπύλης εκμάθησης και των ελαφρότερων εφαρμογών⁷ (Fielding&Taylor, 2002). Το μοντέλο REST για υλοποίηση των διαδικτυακών υπηρεσιών βασίζεται στις ακόλουθες αρχές:

- **Αναγνώριση πόρων και υπηρεσιών (Resource and Service Identification)** μέσω URI. Μια υπηρεσία RESTful διαθέτει τόσο δεδομένα όσο και υπηρεσίες με ομοιόμορφο τρόπο.
- **Ομοιόμορφη διασύνδεση:** Ο χειρισμός των πόρων γίνεται με χρήση των μεθόδων PUT, GET, POST και DELETE. Η μέθοδος GET ανακτά την τρέχουσα κατάσταση του πόρου. Η μέθοδος POST μεταφέρει τη νέα κατάσταση στον πόρο.
- **Αυτοπεριγραφικά μηνύματα (Self Descriptive Messages):** Οι πόροι αποσυνδέονται από την αναπαράστασή τους. Το περιεχόμενο είναι προσβάσιμο σε διάφορες μορφές (HTML, XML, PDF, κ.ο.κ.). Τα μεταδεδομένα σχετικά με τον πόρο είναι διαθέσιμα και χρησιμοποιούνται για τον έλεγχο της προσωρινής αποθήκευσης, για τη διαπραγμάτευση της μορφής αναπαράστασης και την πραγματοποίηση ελέγχου ταυτότητας.
- **Statefull Interactions:** Τα μηνύματα αιτημάτων προς την υπηρεσία είναι αυτοδύναμα. Ο πόρος είναι stateless. Αυτό σημαίνει ρητή μεταφορά της κατάστασης, όπου η πληροφορία της κατάστασης είναι ενσωματωμένη στο μήνυμα.

Τα μειονεκτήματα του μοντέλου REST, είναι ότι παρέχει λιγότερη ασφάλεια σε σύγκριση με τα αναπτυγμένα πρότυπα ασφαλείας που υλοποιούνται στις υπηρεσίες SOAP και ότι ο

⁶ https://books.google.gr/books?id=W-_c-mEs0fgC&pg=PA292&lpg=PA292&dq=Alonso,+et+al.,+2004+wsdl&source=bl&ots=cbiV4T3nbl&sig=ACfU3U2g9xLI4B1MOpTYuU1wkQiB-_dkjw&hl=el&sa=X&ved=2ahUKewj8uOqt4OvgAhWs8aYKHfTICOAQ6AEwAHoECAAAQ#v=onepage&q=Alonso%2C%20et%20al.%2C%202004%20wsdl&f=false

⁷ <https://www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf>

χειρισμός τύπων δεδομένων είναι καλύτερος στο μοντέλο SOAP. Ωστόσο, το γεγονός ότι στο REST ο διακομιστής είναι stateless, επιτρέπει την κατασκευή ιεραρχικών υπηρεσιών σε οποιοδήποτε βάθος⁸ (Pautasso, et al., 2008).

Οι εφαρμογές REST χρησιμοποιούν συχνά το μορφότυπο JSON για τη μεταφορά δεδομένων αντί για τη μορφή XML που είναι πιο σύνθετη και ογκώδης.

Ο μορφότυπος JavaScript Notation Object Notation (JSON) είναι ένα πρότυπο βασισμένο σε κείμενο που έχει σχεδιαστεί για ανταλλαγή δεδομένων, με δυνατότητα κατανόησης του περιεχομένου και από τον άνθρωπο. Προέρχεται από τη γλώσσα JavaScript για την απεικόνιση απλών δομών δεδομένων και πινάκων δεδομένων που ονομάζονται αντικείμενα. Ο μορφότυπος είναι ανεξάρτητος από τη γλώσσα και υπάρχουν αναλυτές διαθέσιμοι για πολλές γλώσσες προγραμματισμού (JAVA, PHP) και όχι μόνο για JavaScript.

2.2 Διαδικτυακές εφαρμογές εμπλουτισμένου περιεχομένου και HTML5

Οι εφαρμογές για κινητές συσκευές, πρέπει να έχουν ιδιαίτερα χαρακτηριστικά λόγω της φύσης του κινητού, η οποία επιτρέπει τη χρήση του οπουδήποτε και οποτεδήποτε⁹ (Gualtieri, 2012). Οι χρήστες παίρνουν τις κινητές τους συσκευές παντού μαζί τους και αποτελούν ένα προσωπικό τους αντικείμενο. Το πλαίσιο της κινητής τηλεφωνίας, είναι αυτό που καθορίζει το σχεδιασμό εφαρμογών για τις έξυπνες συσκευές κινητών τηλεφώνων¹⁰ (Beyer&Holtzblatt, 1998). Το πλαίσιο αυτό, περιγράφει το περιβάλλον στο οποίο οι χρήστες μπορούν να χρησιμοποιήσουν την εφαρμογή και τις λειτουργίες που προσφέρονται σε αυτό το περιβάλλον. Ο σχεδιασμός εφαρμογών για κινητά, πρέπει να λαμβάνει υπόψη τη θέση (όπου βρίσκεται) και τη μετακίνηση (ανάγκες που έχει ο χρήστης κατά τη διάρκεια της κίνησης, για παράδειγμα περπάτημα, μετακίνηση με αυτοκίνητο, κλπ.). Μια άλλη βέλτιστη πρακτική για το σχεδιασμό εφαρμογών κινητής τηλεφωνίας είναι η αμεσότητα, δηλαδή η δυνατότητα να είναι διαθέσιμες οι κατάλληλες λειτουργίες όταν ο χρήστης τις χρειάζεται ή ακόμα και να προβλέπουν τις ανάγκες των χρηστών με βάση την τοποθεσία και τη μετακίνησή τους. Ο σχεδιασμός εφαρμογών για κινητά πρέπει επιπλέον, να λαμβάνει υπόψη τις προσωπικές επιλογές του εκάστοτε χρήστη. Ορισμένοι χρήστες για παράδειγμα, μπορεί να θέλουν να ενημερωθούν για την παρουσία ειδικής προσφοράς σε ένα κατάστημα που βρίσκεται κοντά, ενώ άλλοι μπορεί να την αγνοούν. Τέλος, οι φορητές συσκευές έχουν διαφορετικά χαρακτηριστικά, δυνατότητες και μεγέθη οθόνης. Η διεπαφή θα πρέπει να προσαρμόζεται στις διαφορές αυτές και να προσφέρει την καλύτερη δυνατή παρουσίαση της πληροφορίας και αντίστοιχη ευκολία χρήσης.

Με σκοπό τη δημιουργία ευχάριστου και λειτουργικού περιβάλλοντος χρήσης, ο σχεδιαστής θα πρέπει να δημιουργήσει και να επεξεργαστεί τις διαφορετικές προσωπικότητες των δυναμικών χρηστών μιας εφαρμογής (Buxton, 2007), (Moggridge, 2007). Οι προσωπικότητες αυτές, πρέπει να δοκιμαστούν σε πραγματικές συνθήκες, ώστε να παρατηρηθούν ο τρόπος ζωής, οι αντιδράσεις και τα συναισθήματά τους, ανάλογα με την εκάστοτε κατάσταση. Ακόμη, ο σχεδιασμός θα πρέπει να επικεντρωθεί στη δημιουργία λογισμικού το οποίο προσαρμόζεται στο μεταβαλλόμενο περιβάλλον (λόγω της μετακίνησης του χρήστη). Ο σχεδιαστής πρέπει να επικυρώσει τη σχεδίαση στο πραγματικό πλαίσιο και περιβάλλον λειτουργίας της εφαρμογής και όχι σε συνθήκες ενός γραφείου, χωρίς επαφή με τις πραγματικές συνθήκες λειτουργίας της (Beyer&Holtzblatt, 1998).

Καθώς η ισχύς των κινητών συσκευών και η διάρκεια ζωής της μπαταρίας αυξάνεται, οι κινητές διαδικτυακές εφαρμογές αποκτούν ιδιαίτερη σημασία για τον κάτοχο της συσκευής. Οι εφαρμογές κινητής τηλεφωνίας είναι εγγενώς συνδεδεμένες με το φυσικό περιβάλλον τους και πιο προσωπικές από τις παραδοσιακές εφαρμογές στο διαδίκτυο. Οι χρήστες μεταφέρουν τα smartphones τους παντού και ανά πάσα στιγμή μπορούν να τα χρησιμοποιήσουν τόσο για ψυχαγωγία όσο και για εργασία. Πολύ συχνά η χρήση των εφαρμογών εξαρτάται από το περιβάλλον γύρω από τον χρήστη. Ο χρήστης μπορεί να θέλει να μεταβεί σε μια συγκεκριμένη τοποθεσία όπου πραγματοποιείται μια συνάντηση που σχετίζεται με την εργασία του και χρειάζεται συγκεκριμένες ειδικές οδηγίες και οπτικές υποδείξεις για να βρεθεί στον προορισμό

⁸https://www.researchgate.net/publication/221022734_RESTful_Web_Services_vs_Big_Web_Services_-_Making_the_Right_Architectural_Decisions

⁹ Design Mobile Apps From The Outside In

¹⁰ <http://wtf.tw/ref/holtzblatt.pdf>

του. Σε μια άλλη περίπτωση, ο χρήστης περιηγείται απλώς σε μια πόλη την οποία επισκέπτεται και ενδιαφέρεται να διερευνήσει σημεία ενδιαφέροντος που βρίσκονται κοντά. Σε πολλές εφαρμογές για κινητά, η έννοια του περιβάλλοντος περιλαμβάνει όχι μόνο τον τόπο, αλλά και το χρόνο. Ένα τυπικό παράδειγμα είναι η παρουσίαση γεγονότων που λαμβάνουν χώρα σε συγκεκριμένο τόπο και χρόνο. Έτσι, οι κινητές εφαρμογές θα πρέπει να υποστηρίζουν εμπλουτισμένες διαδικτυακές εφαρμογές με διαδραστικά γραφικά, εικόνες, βίντεο, ήχο, κινούμενα σχέδια κλπ. Τα έξυπνα τηλέφωνα είναι ένα βασικό μέσο κοινωνικής επικοινωνίας και αλληλεπίδρασης (Keirren, 2011). Σε αυτό το παράδειγμα, το περιβάλλον απαρτίζεται από τους φίλους του χρήστη του έξυπνου τηλεφώνου. Οι φίλοι θα πρέπει να ενημερώνονται για τη διαθεσιμότητά του για επικοινωνία (αν αυτός το επιθυμεί), για την τοποθεσία του (αν το επιθυμεί) και θα πρέπει να μπορούν να του ζητήσουν συστάσεις ή συμβουλές εάν το χρειάζονται. Από την άλλη πλευρά, ο ιδιοκτήτης του έξυπνου τηλεφώνου θα πρέπει να είναι σε θέση να προσαρμόζει την εφαρμογή σύμφωνα με τα ενδιαφέροντά του, να επιλέγει για τι είδους γεγονότα ενδιαφέρεται να γνωρίζει, από ποιους φίλους θέλει να λαμβάνει ενημερώσεις κτλ. Αυτά είναι χαρακτηριστικά παραδείγματα της ανάγκης για την εξατομίκευση των κινητών διαδικτυακών εφαρμογών σε ένα περιβάλλον ασύγχρονης επικοινωνίας.

Οι κινητές εφαρμογές που κατασκευάζονται σήμερα, είναι εφαρμογές αναπτυγμένες για συγκεκριμένο τύπο συσκευής (για παράδειγμα iPhone). Οι εφαρμογές εκμεταλλεύονται τη λειτουργικότητα που προσφέρεται από το λειτουργικό σύστημα του κινητού για την παροχή της επιθυμητής λειτουργικότητας στο χρήστη. Για παράδειγμα, εάν ο χρήστης χρειάζεται πρόσβαση σε τοπικό αποθηκευτικό χώρο, οι προγραμματιστικές κλήσεις θα είναι συγκεκριμένες για το λειτουργικό σύστημα της συσκευής. Το ίδιο ισχύει όταν οι εφαρμογές ενσωματώνουν λειτουργικότητα εμπλουτισμένων μέσων με χρήση πρόσθετων (plugins) που σχετίζονται με την συγκεκριμένη πλατφόρμα ή χρησιμοποιούν μηχανισμούς πρόσβασης σε πληροφορίες περιβάλλοντος (GPS, κατεύθυνση κλπ.) μέσω της συσκευής. Επιπλέον, τα μεγέθη συσκευών ποικίλλουν δραστικά ανάλογα με τον κατασκευαστή και το μοντέλο της συσκευής. Αυτό αποτελεί πολύ σημαντικό πρόβλημα για τους κατασκευαστές λογισμικού για εφαρμογές εμπλουτισμένων μέσων. Το πρόβλημα αυτό πολλές φορές αντιμετωπίζεται με την ανάπτυξη διαφορετικών εφαρμογών για διαφορετικούς τύπους συσκευών, γεγονός που κάνει την ανάπτυξη και συντήρηση της εφαρμογής δαπανηρή και χρονοβόρα. Επίσης ο χρόνος ανάπτυξης επιμηκύνεται με αποτέλεσμα την καθυστέρηση της εισαγωγής της εφαρμογής σε μεγάλα τμήματα της αγοράς.

Η **HTML5** είναι ένα πρότυπο ανάπτυξης διαδικτυακού περιεχομένου, που στοχεύει στην ικανοποίηση των απαιτήσεων για ανάπτυξη εφαρμογών εμπλουτισμένων μέσων, οι οποίες θα αποφεύγουν όλα τα παραπάνω προβλήματα¹¹ (Freeman&Robson, 2011). Ορισμένες από τις λειτουργίες που προτείνει, είναι πιο σταθερές από τις παλαιότερες προσεγγίσεις των plugins και ως εκ τούτου οι βασικότεροι κατασκευαστές συσκευών και λειτουργικών συστημάτων το υποστηρίζουν ήδη. Στη συνέχεια καταγράφονται ορισμένες από τις βασικότερες και πιο σημαντικές λειτουργίες που ενσωματώνονται στο πρότυπο.

Το στοιχείο **καμβά**: Το στοιχείο καμβά είναι ένας καμβάς εικονοστοιχείων που εξαρτάται από την ανάλυση της συσκευής και ο οποίος μπορεί να χρησιμοποιηθεί για την απεικόνιση γραφημάτων, γραφικών παιχνιδιών ή εικόνων σε κίνηση. Ο προγραμματιστής έχει στη διάθεσή του μια ορθογώνια περιοχή στην οποία μπορεί να σχεδιάσει. Στην πραγματικότητα η λειτουργία της σχεδίασης αποθηκεύει εντολές για σχεδίαση. Η JavaScript είναι η γλώσσα η οποία χρησιμοποιείται για τη σχεδίαση των γραφικών αντικειμένων. Οι εντολές JavaScript επιτρέπουν στον προγραμματιστή να σχεδιάζει αντικείμενα από σημείο σε σημείο. Το Canvas API είναι ένα σύνολο λειτουργιών που ορίζονται στο πρότυπο HTML5 για την σχεδίαση σχημάτων, τον καθορισμό διαδρομών, κλπ. Το στοιχείο καμβά είναι ένας άμεσος τρόπος απεικόνισης γραφικών στο περιβάλλον της εφαρμογής. Δεν αποθηκεύει το σύνολο των αντικειμένων που απεικονίζονται στο περιβάλλον χρήσης, παρά αυτά αναδημιουργούνται με την εκτέλεση των αντίστοιχων εντολών της γλώσσας JavaScript. Για να αναδημιουργηθεί ένα γραφικό αντικείμενο, πρέπει να εκτελεστούν όλες οι εντολές σχεδίασης που αντιστοιχούν σε αυτό. Εάν ο προγραμματιστής θέλει απλώς να μετακινήσει το αντικείμενο σε άλλη τοποθεσία, πρέπει να ξανασχεδιάσει το αντικείμενο. Εννοιολογικά ο καμβάς είναι ένα API κατώτερου επιπέδου, πάνω

¹¹ <http://www.etpe.gr/custom/pdf/etpe1871.pdf>

στο οποίο δημιουργούνται άλλα API για την απλουστευμένη δημιουργία των γραφικών στοιχείων, όπως το SVG.

Το στοιχείο SVG: Το στοιχείο SVG επιτρέπει το σχεδιασμό διανυσματικών γραφικών στοιχείων. Μια εικόνα περιγράφεται ως μια συλλογή αντικειμένων, τα οποία αναπαρίστανται στη γλώσσα XML. Τα γραφικά ενός σκηνοθέτη σε περιγραφή SVG επιτρέπουν στους χειριστές συμβάντων να συσχετίζονται με συγκεκριμένα αντικείμενα και να αντιδρούν σε ενέργειες του χρήστη σε αυτά. Αυτός ο χειρισμός είναι πιο δύσκολο να υλοποιηθεί στον καμβά της HTML5.

Το στοιχείο βίντεο: Το στοιχείο βίντεο επιτρέπει την ενσωμάτωση βίντεο μέσα σε μια ιστοσελίδα χωρίς τη χρήση εξωτερικών προσθηκών (όπως flash και Quicktime). Οι μορφές βίντεο που χρησιμοποιούνται είναι πολλές και δεν υπάρχει καθολικά αποδεκτό πρότυπο. Το βίντεο σε προτυποποίηση H264 και το Web M (ανοικτού κώδικα) είναι οι κύριες μορφές που υποστηρίζονται από διαφορετικές πλατφόρμες.

Γεωγραφικός εντοπισμός: Η HTML5 υποστηρίζει ένα API γεωγραφικού εντοπισμού. Υπάρχουν διαθέσιμοι περισσότεροι από ένας τρόποι για να υπολογιστεί η θέση μιας κινητής συσκευής. Η διεύθυνση IP, η σύνδεση σε ένα σημείο ασύρματης πρόσβασης, η κεραία κινητής τηλεφωνίας με την οποία επικοινωνεί η συσκευή, η θέση GPS που υπολογίζεται μέσω δορυφορικών πληροφοριών, είναι μηχανισμοί που επιτρέπουν τον εντοπισμό της θέσης της συσκευής. Ο εξοπλισμός GPS μιας κινητής συσκευής καταναλώνει πολλή ενέργεια από τη συσκευή, γι' αυτό και οι χρήστες πολλές φορές απενεργοποιούν τη δυνατότητα αυτή μέχρι να τη χρειαστούν. Η διαδικασία εντοπισμού περιλαμβάνει κάποια βήματα. Οι χάρτες Google αρχικά χρησιμοποιούν την κεραία κινητής τηλεφωνίας ή τη θέση του ασύρματου σημείου πρόσβασης και δίνουν μια προσέγγιση της θέσης (ένα κύκλο μεγάλης ακτίνας), και στη συνέχεια χρησιμοποιούν τριγωνοποίηση με άλλες κεραίες τηλεφωνίας, ώστε ο αλγόριθμος να δώσει ακριβέστερα αποτελέσματα. Στη συνέχεια, υπεισέρχεται το GPS ώστε να υπολογίσει το ακριβές σημείο της συσκευής.

Η HTML5 περιλαμβάνει νέες ετικέτες περιγραφής στοιχείων, συμπεριλαμβανομένων των: <MathML>, <section>, <header>, <new>, κλπ.

Υποστήριξη για εφαρμογές εκτός δικτύου (Off-line Web Applications):

Χρησιμοποιώντας την HTML5, μπορούν να δημιουργηθούν διαδικτυακές εφαρμογές που λειτουργούν σε κατάσταση εκτός δικτύου. Είναι σαφές ότι αυτή είναι μια πολύ σημαντική λειτουργία για τα περιβάλλοντα εφαρμογών για κινητές συσκευές, δεδομένου ότι η κινητή συσκευή μπορεί να μην έχει πρόσβαση στο δίκτυο ή η σύνδεση να είναι πολύ δαπανηρή ή να καταναλώνει ενέργεια για πρόσβαση στα δεδομένα μέσω του δικτύου από την τρέχουσα θέση του χρήστη. Μια εφαρμογή εκτός σύνδεσης ξεκινά ως μια on-line διαδικτυακή εφαρμογή. Την πρώτη στιγμή της πρόσβασης του διαδικτυακού διακομιστή, αυτός ενημερώνει τον φυλλομετρητή για τα αρχεία που χρειάζονται για να λειτουργήσει εκτός σύνδεσης. Τα αρχεία μπορούν να είναι HTML, JavaScript, εικόνες, βίντεο κ.λπ. Μόλις φορτωθούν, η εφαρμογή μπορεί να λειτουργήσει χρησιμοποιώντας τα, ακόμη και αν ο φυλλομετρητής δεν είναι συνδεδεμένος στο διαδίκτυο. Ο φυλλομετρητής είναι σε θέση να ανιχνεύσει εάν υπάρχει σύνδεση στο διαδίκτυο. Όταν ο φυλλομετρητής συνδεθεί ξανά στο διαδίκτυο, όλες οι αλλαγές που έχουν τυχόν γίνει από το χρήστη στην εφαρμογή, μεταφέρονται αυτόματα στον διαδικτυακό διακομιστή στο παρασκήνιο.

Προσωρινή αποθήκευση πόρων (Resource caching): Όλοι οι περιηγητές ιστού αποθηκεύουν πόρους όπως HTML, CSS, εικόνες και JavaScript κατά την πρώτη φόρτωση της σελίδας, αλλά ο προγραμματιστής και ο χρήστης δεν έχουν κανένα έλεγχο για το χρονικό διάστημα κατά το οποίο το πρόγραμμα περιήγησης θα συνεχίσει να αποθηκεύει προσωρινά τους πόρους. Το HTML5 επιτρέπει την προσωρινή αποθήκευση πόρων με το Appcache. Ο προγραμματιστής δημιουργεί ένα αρχείο οδηγιών, το οποίο ουσιαστικά απαριθμεί όλες τις διευθύνσεις URL των πόρων που το πρόγραμμα περιήγησης θα πρέπει να αποθηκεύσει στο χώρο προσωρινής αποθήκευσης. Ο φυλλομετρητής διαβάζει το αρχείο αυτό, κατεβάζει τους πόρους και τους αποθηκεύει σε τοπικό επίπεδο. Από τη στιγμή που αυτοί αποθηκευτούν στην προσωρινή μνήμη, ενημερώνονται μόνο όταν το αρχείο δήλωσης τροποποιηθεί στον διακομιστή και όχι όταν τροποποιηθούν τα ίδια τα αρχεία. Η εφαρμογή κάνει πάντα πρόσβαση στους πόρους που είναι αποθηκευμένοι σε τοπικό επίπεδο, άσχετα εάν υπάρχει διαδικτυακή σύνδεση ή όχι.

WebStorage: Οι διαδικτυακές εφαρμογές πρέπει να έχουν πρόσβαση και να παράγουν δεδομένα ακόμη και όταν είναι εκτός σύνδεσης. Για την αποθήκευση δεδομένων που δημιουργούνται χρειάζονται τοπική αποθήκευση στο πρόγραμμα - πελάτη. Οι εγγενείς εφαρμογές (εφαρμογές που διαχειρίζεται το συγκεκριμένο λειτουργικό σύστημα του φυλλομετρητή) χρησιμοποιούν τις λειτουργίες του λειτουργικού συστήματος για την αποθήκευση και την ανάκτηση δεδομένων της εφαρμογής. Αυτό τις καθιστά εξαρτώμενες από το λειτουργικό σύστημα και οι λύσεις τους δε μπορούν να χρησιμοποιηθούν γενικευμένα, αφού δεν υποστηρίζονται από πρότυπα που ακολουθούν οι φυλλομετρητές. Η HTML5 παρέχει τυπικές λειτουργίες για αποθήκευση και ανάκτηση ζευγών δεδομένων τα οποία περιγράφονται από ζεύγη κλειδίων / τιμών τοπικά στο πρόγραμμα περιήγησης. Τα δεδομένα δεν μεταδίδονται στον διακομιστή διαδικτύου. Δεδομένου ότι αυτή η λειτουργικότητα υλοποιείται μέσα στο πρόγραμμα περιήγησης, μπορεί να εκτελεστεί σε οποιαδήποτε πλατφόρμα. Η JavaScript χρησιμοποιείται για την πρόσβαση στα δεδομένα που έχουν αποθηκευτεί σε ένα τοπικό αντικείμενο αποθήκευσης. Ένα τοπικό αντικείμενο αποθήκευσης μπορεί να αντιμετωπιστεί ως ένας πίνακας δεδομένων. Τα δεδομένα μπορούν να είναι οποιουδήποτε τύπου που υποστηρίζεται από τη JavaScript, συμπεριλαμβανομένων των συμβολοσειρών, λογικών, ακέραιων και πραγματικών αριθμών. Τα συμβάντα αποθήκευσης μπορούν να χρησιμοποιηθούν για την παρακολούθηση των αλλαγών που έγιναν στα αντικείμενα στην τοπική αποθήκευση χρησιμοποιείται ένα ειδικό αντικείμενο, το οποίο ονομάζεται `storage event object`. Το αντικείμενο αυτό αποθηκεύει το αναγνωριστικό της παλαιάς τιμής, τη νέα τιμή και τη διεύθυνση URL της σελίδας που προκάλεσε την αλλαγή. Από προεπιλογή, κάθε εφαρμογή μπορεί να χρησιμοποιήσει 5 Mbytes τοπικού αποθηκευτικού χώρου. Οι αλλαγές που πραγματοποιούνται κατά τη διάρκεια που η εφαρμογή λειτουργεί σε κατάσταση αποσύνδεσης από το δίκτυο, παραμένουν αποθηκευμένες τοπικά. Όταν όμως η συσκευή συνδεθεί εκ νέου στο δίκτυο επιτρέπεται αυτόματα η αποθήκευση στο διακομιστή. Τα δεδομένα στον τοπικό αποθηκευτικό χώρο μπορούν να διαβαστούν και να τροποποιηθούν μόνο από τον ιστότοπο που τα έχει δημιουργήσει.

Βάση δεδομένων WebSQL: ΗHTML5 παρέχει ένα API το οποίο δημιουργεί διεπαφή με μια βάση δεδομένων τύπου SQL, η οποία μπορεί να προσπελαστεί από τη JavaScript.

IndexedDB: Η HTML5 παρέχει ένα API για τη χρήση ενός χώρου αποθήκευσης αντικειμένων. Το μοντέλο περιλαμβάνει βάσεις δεδομένων, εγγραφές και πεδία ενώ παρέχονται ειδικές μέθοδοι πρόσβασης στο περιεχόμενο.

Web Workers: Η HTML5 παρέχει μια πρότυπη μέθοδο ώστε οι φυλλομετρητές να εκτελούν εντολές JavaScript στο παρασκήνιο. Επιτρέπει την ταυτόχρονη εκτέλεση πολλαπλών νημάτων. Τα νήματα μπορούν να κάνουν ελέγχους, μαθηματικούς υπολογισμούς, πρόσβαση σε τοπικό αποθηκευτικό χώρο και να αποκρίνονται στις ενέργειες των χρηστών.

Οι εφαρμογές εμπλουτισμένων μέσων είναι μια πολύ απαιτητική αλλά και πολλά υποσχόμενη κλάση εφαρμογών στο διαδίκτυο, η οποία αναμένεται να δημιουργήσει μια τεράστια αγορά στο εγγύς μέλλον. Η τρέχουσα κατάσταση στο πεδίο των εφαρμογών κινητής τηλεφωνίας είναι ότι πρόκειται για εγγενείς εφαρμογές, οι οποίες έχουν αναπτυχθεί για μια συγκεκριμένη πλατφόρμα λειτουργικού συστήματος. Έτσι, πρέπει να δημιουργούνται πολλές διαφορετικές εφαρμογές και να συντηρούνται στο μέλλον, για να υποστηρίξουν παρόμοια λειτουργικότητα σε διαφορετικές πλατφόρμες. Η HTML 5 είναι ένα πρότυπο ανάπτυξης εφαρμογών που επιτρέπει την ανάπτυξη εφαρμογών εμπλουτισμένων μέσων ανεξάρτητων από την πλατφόρμα, οι οποίες υποστηρίζουν αλληλεπιδράσεις χρηστών με πολυμέσα, σε ένα μεταβαλλόμενο και εξατομικευμένο φυσικό περιβάλλον της κινητής συσκευής.

Η παρούσα εργασία, επιδεικνύει τη δυνατότητα ανάπτυξης τέτοιων συστημάτων ανεξάρτητων από πλατφόρμα και προτείνει αρχιτεκτονική συστημάτων, αρχιτεκτονικά πρότυπα, προσφερόμενες λειτουργίες, διεπαφές χρήστη, κατάλληλα εργαλεία και μεθοδολογίες για την κατασκευή τέτοιων συστημάτων.

3. Κινητές συσκευές και πλατφόρμες

Οι συσκευές έξυπνων κινητών, παρέχουν τη δυνατότητα συνεχούς πρόσβασης στο διαδίκτυο (mobile internet). Η ιδιότητα αυτή, έχει αλλάξει τα δεδομένα τόσο στην πληροφόρηση όσο και στη διασκέδαση. Η χρήση τους αντικαθιστά σταδιακά τους φορητούς υπολογιστές (laptop) και

τις συσκευές αναπαραγωγής μουσικής¹² (mp3 players) (Deloitte, 2011), καθώς προσφέρουν ένα ευρύ σύνολο υπηρεσιών, διαθέσιμο στον κάτοχό τους οποιαδήποτε χρονική στιγμή και σε οποιοδήποτε γεωγραφικό σημείο. Οι υπηρεσίες που προσφέρουν:

- Ψυχαγωγία (μουσική, videos, παιχνίδια).
- Πρόσβαση σε ενημέρωση και ειδησεογραφία.
- Επικοινωνία και κοινωνική δικτύωση.
- Αναζήτηση προϊόντων και προσφορών.
- Αναζήτηση καταστημάτων πώλησης, καθώς και οδηγιών προς αυτά.
- Αξιολόγηση και σχολιασμό γεγονότων, προϊόντων και υπηρεσιών.
- Αγορά προϊόντων και υπηρεσιών.

3.1 Τεχνικά χαρακτηριστικά και λειτουργιών των έξυπνων κινητών τηλεφώνων
Οι έξυπνες φορητές συσκευές έχουν αποκτήσει πληθώρα χαρακτηριστικών αλλά και τέτοια υπολογιστική ισχύ, που είναι σε θέση να υποκαταστήσουν πολλές εφαρμογές και χρήσεις των κλασικών υπολογιστών ή φορητών υπολογιστών. Ο συνδυασμός της μείωσης του κόστους μια τέτοιας συσκευής με την πληθώρα των δυνατοτήτων τους, δημιουργεί τις προϋποθέσεις γι' αυτό. Παρακάτω θα αναλυθούν οι πιο διαδεδομένες λειτουργίες και τα χαρακτηριστικά των έξυπνων κινητών συσκευών που απαντώνται στα σημερινά κινητά τηλέφωνα και tablets¹³ (Ilyas & Ahson, 2006).

- **Οθόνη αφής**

Οι οθόνες των έξυπνων συσκευών χαρακτηρίζονται από τη δυνατότητα που δίνουν για αλληλεπίδραση με αφή. Στα έξυπνα κινητά τηλέφωνα η διαγώνιος της οθόνης αποκτά σημασία: εκεί φιλοξενείται το εικονικό πληκτρολόγιο (αφή), εκεί αναπαράγονται τα πολυμέσα, οι λίστες με τα μηνύματα, οι χάρτες και τα παράθυρα του φυλλομετρητή. Οι οθόνες στα κινητά τηλέφωνα ποικίλουν και μπορούν να είναι από 2,8 ίντσες μέχρι και 10 ίντσες αναλόγως τη συσκευή. Επίσης, ιδιαίτερη σημασία έχει η υποστηριζόμενη ανάλυση (συνήθως είναι 720x1200pixels). Η οθόνη δεν σημαίνει μόνο αναπαραγωγή περιεχομένου αλλά και είσοδο με την αφή. Τεχνολογίες όπως το multi-touch (εκτέλεση λειτουργίας με παραπάνω από ένα ταυτόχρονο άγγιγμα της οθόνης) δημιουργούν νέες δυνατότητες και λειτουργίες σε ένα έξυπνο κινητό τηλέφωνο.

- **Πρόσβαση στο διαδίκτυο -Ασύρματη δικτύωση Wi-Fi**

Τα έξυπνα τηλέφωνα προσφέρουν ευκολία πρόσβασης στο διαδίκτυο είτε μέσω τηλεπικοινωνιακών δικτύων (3G, 4G) είτε μέσω ασύρματων δικτύων Wi-Fi. Οι δυνατότητες περιήγησης στο διαδίκτυο δεν περιορίζονται μόνο μέσω του φυλλομετρητή που διαθέτει η συσκευή αλλά επεκτείνονται με τη χρήση εξειδικευμένων εφαρμογών για πρόσβαση σε όλες τις γνωστές σελίδες αναπαραγωγής πολυμέσων, σελίδες κοινωνικής δικτύωσης κ.λπ.

- **Διεπαφή**

Σε μία τέτοια συσκευή, οι περισσότερες λειτουργίες γίνονται με το άγγιγμα, τα πλήκτρα της συσκευής έχουν καταργηθεί σχεδόν εξ ολοκλήρου (με την εξαίρεση υποτυπωδών πλήκτρων, όπως αυτό που οδηγεί στην αρχική οθόνη του κινητού, πλήκτρα αυξομείωσης της έντασης του ήχου ή άλλες βασικές επιλογές).

- **Φωτογραφική μηχανή**

Η ενσωματωμένη ψηφιακή φωτογραφική μηχανή ενός έξυπνου κινητού τηλεφώνου συχνά ξεπερνά σε δυνατότητες μια απλή ψηφιακή φωτογραφική μηχανή. Τα σύγχρονα έξυπνα τηλέφωνα έχουν δυνατότητες λήψης φωτογραφιών υψηλής ανάλυσης και ποιότητας, βίντεο υψηλής ευκρίνειας. Επιπλέον ενσωματώνουν εφαρμογές επεξεργασίας βίντεο και εικόνας.

- **Επεξεργαστής**

Τα περισσότερα έξυπνα κινητά τηλέφωνα διαθέτουν ιδιαίτερα ισχυρούς επεξεργαστές ώστε να εξυπηρετούν ολοένα και περισσότερες ταυτόχρονες λειτουργίες (multitasking). Στις περισσότερες κινητές συσκευές συναντώνται επεξεργαστές διπλού και τετραπλού πυρήνα.

¹² Revolutions 2010 Survey: Mobile Apps and e-readers Transform Consumer Behavior, 2011

¹³ Smartphones Research Report, 2006

- **Χάρτες- πλοήγηση & λειτουργίες εύρεσης θέσης**

Παράλληλα με άλλες λειτουργίες τα περισσότερα έξυπνα κινητά τηλέφωνα υποστηρίζουν λειτουργίες πλοήγησης με χάρτες (GoogleMaps) σε συνεργασία με σύστημα εντοπισμού θέσης GPS, είτε με A-GPS (σύστημα εύρεσης θέσης με την βοήθεια δικτύου κινητής τηλεφωνίας).

- **Συνδεσιμότητα**

Μια άλλη σημαντική παράμετρος που διαφοροποιεί τις έξυπνες κινητές συσκευές αυτές είναι η ύπαρξη πολλαπλών διεπαφών υλικού, για τη διασύνδεση με άλλες ομοειδείς συσκευές που τις υποστηρίζουν. Στο πλαίσιο αυτό οι έξυπνες συσκευές διαθέτουν Bluetooth, καλώδιο σύνδεσης USB ή και έξοδο σε τηλεοπτικό δέκτη (TV-out), γεγονός που διευκολύνει τη διαχείριση και αναπαραγωγή του περιεχομένου τους.

- **Μνήμη**

Ο όγκος των δεδομένων που διαχειρίζονται σήμερα οι κινητές συσκευές και οι πιο σύνθετες λειτουργίες που εκτελούν στα δεδομένα αυτά, δημιουργούν την ανάγκη για περισσότερη μνήμη αποθήκευσης. Η μνήμη αυτή ποικίλει από συσκευή σε συσκευή και μπορεί να φτάσει σε μεγέθη αντίστοιχα ενός επιτραπέζιου ή φορητού υπολογιστή. Εφαρμογές οι οποίες έχουν ανάγκη την ύπαρξη επαρκούς και μεγάλης μνήμης είναι η αναπαραγωγή πολυμέσων αλλά και η επεξεργασία των περισσότερων μορφών δεδομένων (αρχεία κειμένου, εικόνας, ήχου).

- **Πληθώρα εφαρμογών ανάλογα με το λειτουργικό σύστημα**

Μια ακόμη σημαντική παράμετρος είναι η ύπαρξη εφαρμογών για το λειτουργικό σύστημα, όπως παιχνίδια και εξειδικευμένες υπηρεσίες. Το πλέγμα εταιρειών παραγωγής των έξυπνων τηλεφωνικών συσκευών περιλαμβάνει εκτός από εταιρείες κατασκευής και ανάπτυξης του λογισμικού συστήματος, μια πλειάδα εταιρειών παραγωγής και ανάπτυξης εφαρμογών (application developers) για smartphones. Οι εταιρίες παραγωγής του λειτουργικού συστήματος των κινητών τηλεφώνων έχουν αναπτύξει ηλεκτρονικά καταστήματα εφαρμογών, όπως το AppStore για τα κινητά που βασίζονται στο iPhone Operating System της Apple και το Play Store για κινητά με λειτουργικό Android.

3.2 Εφαρμογές για κινητές πλατφόρμες

Τα νέα λειτουργικά συστήματα έχουν αλλάξει τον τρόπο με τον οποίο χρησιμοποιούνται οι κινητές συσκευές τα τελευταία χρόνια. Ένα από τα πιο σημαντικά επιτεύγματα, είναι η δημιουργία μέσω αυτών, πολλών νέων και εντυπωσιακών εφαρμογών. Οι εφαρμογές για κινητές πλατφόρμες είναι ένα μέρος της παγκόσμιας αγοράς κινητών συσκευών που μεγαλώνει και αναπτύσσεται ραγδαία. Αποτελούνται από λογισμικό που «τρέχει» σε μια κινητή πλατφόρμα και εκτελεί συγκεκριμένες λειτουργίες για τον χρήστη του. Οι κινητές εφαρμογές χρησιμοποιούνται σε πλήθος μοντέλων κινητών τηλεφώνων, ακόμα και σε συσκευές χαμηλού κόστους στην αγορά. Στα νέα λειτουργικά συστήματα, μπορεί κάποιος να τις προμηθευτεί κατεβάζοντας τις από συγκεκριμένα ηλεκτρονικά καταστήματα εφαρμογών. Η αποδοχή και η ευρεία χρησιμοποίησή τους επιτεύχθηκε λόγω των πολλών λειτουργιών που μπορούν να πραγματοποιούν, που περιλαμβάνει από απλά περιβάλλοντα χρήσης για βασικές υπηρεσίες τηλεφωνίας και μηνυμάτων, μέχρι εξειδικευμένες υπηρεσίες όπως τα βιντεοπαιχνίδια και εφαρμογές πολυμέσων. Οι κατηγορίες των εφαρμογών αυτών είναι πολλές. Εφαρμογές σαν αυτές που χρησιμοποιούνται για την αποστολή και λήψη SMS/MMS, προγράμματα περιήγησης στο διαδίκτυο και εφαρμογές αναπαραγωγής πολυμέσων, όπως mp3 players, έρχονται εγκατεστημένες στα λειτουργικά συστήματα των συσκευών ενώ οι υπόλοιπες μπορούν να εγκατασταθούν μετά την αγορά της συσκευής. Για παράδειγμα, ο χρήστης μπορεί να «κατεβάσει» εφαρμογές μέσω του ασύρματου δικτύου και να τις εγκαταστήσει ο ίδιος ή μπορεί να τις μεταφορτώσει και εγκαταστήσει από το ηλεκτρονικό κατάστημα που είναι συμβεβλημένο με το συγκεκριμένο λειτουργικό σύστημα. Ανεξάρτητα με τον τρόπο που οι εφαρμογές καταλήγουν στον χρήστη, οι εφαρμογές για κινητές πλατφόρμες είναι ήδη ένα μεγάλο και συνεχώς αυξανόμενο μέρος της αγοράς λογισμικού και όπως είναι επόμενο, ο αριθμός των εταιριών ανάπτυξης τέτοιων εφαρμογών αυξάνεται.

Από τεχνικής άποψης, μπορούμε να τις χωρίσουμε σε κατηγορίες σε σχέση με το προγραμματιστικό περιβάλλον στο οποίο εκτελούνται:

- Εφαρμογές που τρέχουν στο περιβάλλον του λειτουργικού συστήματος όπως εφαρμογές που τρέχουν σε iOS, Android, Symbian OS, Windows Phone και Blackberry OS.

- Εφαρμογές που τρέχουν σε web browser περιβάλλον όπως τα Webkit, Mozilla/Firefox, Opera Mini και RIM.
- Άλλες πλατφόρμες και εικονικά συστήματα όπως τα Java/J2ME, BREW, FlashLite και Silverlight.

Από άποψη λειτουργιών, μπορούμε να διακρίνουμε τις εφαρμογές για κινητές πλατφόρμες ως εξής:

- Εφαρμογές επικοινωνιών όπως e-mail, μηνυμάτων, περιήγησης στο διαδίκτυο, ειδησεογραφίας και πληροφοριών και κοινωνικής δικτύωσης.
- Παραγωγικές εφαρμογές: ημερολόγια, αριθμομηχανές, σημειώσεων, υπενθυμίσεων, επεξεργασίας κειμένων, λογιστικών φύλλων, υπηρεσιών GPS και τραπεζικών υπηρεσιών.
- Εφαρμογές πολυμέσων όπως γραφικών και εικόνας, παρουσίασης, αναπαραγωγής βίντεο, αναπαραγωγής ήχου και ροής δεδομένων ήχου και εικόνας.
- Εφαρμογές παιχνιδιών όπως πάζλ και στρατηγικής, τράπουλας και καζίνο, δράσης και περιπέτειας, αθλητικές και χόμπι.

Τα τελευταία χρόνια οι εφαρμογές για κινητές πλατφόρμες έχουν εξελιχθεί ως ένα σημείο, που προσφέρουν στον χρήστη ένα ευρύ σύνολο δυνατοτήτων αλλά και ταχύτητας χρήσης. Από την άποψη αυτή, τέτοιου είδους εφαρμογές έχουν χαρακτηριστικές διάφορες από την συνήθη πλοήγηση σε ιστοσελίδες φτιαγμένες για κινητά συστήματα (Mobile Web) όπου ακόμα χαρακτηρίζονται από προβλήματα πρόσβασης αλλά και χαμηλές ταχύτητες στο δίκτυο κινητής τηλεφωνίας.

4. Android έναντι HTML5

Ο ρυθμός τεχνολογικών εξελίξεων και ανάπτυξης στην αγορά της κινητής τηλεφωνίας, δημιουργεί δυσκολίες στους προγραμματιστές εφαρμογών να σχεδιάσουν μια εφαρμογή για χρήση σε κινητές συσκευές, όχι μόνο από τεχνική άποψη, αλλά και επειδή τα μερίδια αγοράς μεταβάλλονται ταχύτατα μεταξύ διαφορετικών συστημάτων.

Στο ξεκίνημα της επανάστασης των έξυπνων τηλεφώνων, το iPhone iOS κυριάρχησε στην αγορά κινητής τηλεφωνίας. Εξελικτικά το Google Android, έχει ξεπεράσει το iPhone από πλευράς μεριδίου αγοράς, εξαιτίας εν μέρει της ισχύος του εμπορικού σήματος της Google και εν μέρει της ανοικτότητας της πλατφόρμας. Άλλα λειτουργικά συστήματα κινητής τηλεφωνίας περιλαμβάνουν το BlackBerry και το Windows 10 Mobile.

Όταν μια επιχείρηση εξετάζει τον τρόπο ενσωμάτωσης της τεχνολογίας για κινητές συσκευές στο υπάρχον επιχειρηματικό μοντέλο, το κύριο ζήτημα τόσο για τους πελάτες όσο και για τους προγραμματιστές, είναι η επιλογή μεταξύ του είδους των εφαρμογών που θα αναπτύξει. Οι κυριότεροι τύποι εφαρμογών που μπορούν να χρησιμοποιηθούν από κινητές συσκευές είναι: οι εγγενείς εφαρμογές, οι εφαρμογές ιστού και οι υβριδικές εφαρμογές.

Άλλα θέματα που ανακύπτουν στις επιλογές που θα γίνουν κατά την ανάπτυξη εφαρμογών για κινητές συσκευές περιλαμβάνουν:

- Οι εγγενείς περιορισμοί στο χρόνο φόρτωσης, προσωρινής αποθήκευσης και καθυστέρησης των κινητών συσκευών.
- Τα περιβάλλοντα αλληλεπίδρασης στις κινητές μονάδες, τα οποία βασίζονται περισσότερο σε γραφικά και συμβολισμούς, παρά σε κείμενο.
- Οι ταχύτητες μεταφοράς δεδομένων στα δίκτυα κινητής τηλεφωνίας και οι ενδεχόμενες ογκοχρεώσεις που επιβάλλουν οι πάροχοι των δικτύων.

Πρέπει επίσης να ληφθεί υπόψη ότι ο όρος "κινητό" δεν περιγράφει πλέον μόνο μια συσκευή κινητού τηλεφώνου, αλλά μια σειρά άλλων τύπων συσκευών που έχουν αυξήσει τη δημοτικότητά τους, όπως τα tablets ή άλλες υπολογιστικές συσκευές που τοποθετούνται μεταξύ ενός ελαφρού laptop και ένα smartphone, αλλά και συσκευές οι οποίες μπορούν να φορεθούν (γυαλιά ή ρολόγια).

Η ανάπτυξη των τεχνολογιών κινητής τηλεφωνίας σήμαινε ότι οι επιχειρήσεις ορισμένων τομέων, λαμβάνουν ακόμη και το μεγαλύτερο μέρος της διαδικτυακής κυκλοφορίας τους από χρήστες που περιηγούνται σε περιβάλλοντα κινητής τηλεφωνίας.

Η επιλογή πλατφόρμας για ανάπτυξη κινητών εφαρμογών αποτελεί μια στρατηγική απόφαση για την επιχείρηση. Όταν εξετάζεται μια στρατηγική για ανάπτυξη και διάθεση εφαρμογών για κινητές συσκευές, μια από τις μεγάλες αποφάσεις για πολλούς πελάτες είναι εάν θα επικεντρωθούν στην ανάπτυξη μιας εφαρμογής σε περιβάλλον ιστού (web) ή σε εγγενείς (native) εφαρμογές που απευθύνονται και είναι συμβατές με συγκεκριμένες κινητές πλατφόρμες. Δεν υπάρχει λύση "ενός μεγέθους για όλους" σε αυτό το ζήτημα, λόγω του αριθμού των λειτουργικών πλατφορμών και η καλύτερη προσέγγιση για μια επιχείρηση μπορεί να είναι εντελώς διαφορετική από αυτήν για άλλη.

Ένας επιπλέον παράγοντας είναι ο έντονος ανταγωνισμός στον τομέα του λογισμικού για κινητές συσκευές, ο οποίος τροφοδοτεί τις επιταχυνόμενες εξελίξεις σε κάθε μία από τις λειτουργικές πλατφόρμες κινητών. Τα Windows Mobile αντικαταστάθηκαν από το Windows Phone (Windows 10 Mobile). Τόσο το Apple iOS όσο και το Android ενσωματώνουν σημαντικές αλλαγές σε κάθε έκδοσή τους, καθιστώντας το έργο της υποστήριξης των χρηστών επίτιμο και κοστοβόρο για μια μεγάλη επιχείρηση, ακόμη και αν αυτή έχει υιοθετήσει ένα ενιαίο λειτουργικό σύστημα για τις εφαρμογές της.

Ορισμένες από τις πλατφόρμες –και ειδικότερα η Android - αναπτύσσονται για συσκευές που κατασκευάζονται από ένα μεγάλο πλήθος κατασκευαστών, ο καθένας από τους οποίους επιθυμεί να τροποποιήσει το λειτουργικό σύστημα στις ιδιαίτερες απαιτήσεις του.

Στα πρώτα στάδια εμφάνισης των smartphone, η τεχνολογία αιχμής ήταν ουσιαστικά διαθέσιμη μόνο για τους καταναλωτές οι οποίοι είχαν τη δυνατότητα να αποκτήσουν τα πιο ακριβά συστήματα. Σήμερα, τα έξυπνα τηλέφωνα έχουν γίνει πιο προσιτά από άποψη κόστους, και περίπου 85 τοις εκατό των ανθρώπων στο Ηνωμένο Βασίλειο χρησιμοποιούν σήμερα έξυπνο τηλέφωνο ή πρόσβαση σε αυτό¹⁴ (Deloitte, 2017).

Το αποτέλεσμα των εξελίξεων αυτών είναι ότι οι προγραμματιστές πρέπει να αναπτύξουν έχοντας κατά νου μια μεγάλη γκάμα μεγεθών οθόνης, προδιαγραφές και διαμορφώσεις του υλικού - και τελικά μια σειρά θεμελιωδών διαφορετικών μοντέλων κινητών συσκευών.

Υπάρχουν τρεις προσεγγίσεις στην ανάπτυξη εφαρμογών για κινητές συσκευές: (α) η ανάπτυξη εγγενών (native) εφαρμογών, κατασκευασμένων ειδικά για το σκοπό αυτό και οι οποίες λειτουργούν σε ένα λειτουργικό σύστημα, (β) η ανάπτυξη μιας διαδικτυακής εφαρμογής, η οποία όμως είναι προσαρμοσμένη για χρήση από κινητές συσκευές και (γ) η ανάπτυξη υβριδικής εφαρμογής¹⁵ (Cognizant, 2014).

4.1 Διαδικτυακές εφαρμογές

Πρόκειται για εφαρμογές οι οποίες χρησιμοποιούν κώδικα HTML (HTML, CSS και JavaScript) και χρησιμοποιούνται μέσα από ένα φυλλομετρητή σε οποιονδήποτε τύπο συσκευής. Αυτού του είδους οι εφαρμογές δεν μεταφορτώνονται από μια τοποθεσία, όπως γίνεται στις εφαρμογές για κινητές συσκευές.

Μια εφαρμογή HTML5 η οποία αναπτύσσεται με αποκριτικές μεθόδους σχεδίασης (responsive design) προσαρμόζει την εμφάνισή της κατάλληλα ώστε τα στοιχεία του περιβάλλοντος εργασίας της να εμφανίζονται σε λειτουργικές θέσεις της οθόνης της κινητής συσκευής.

Σε περίπτωση αναβάθμισης μιας διαδικτυακής εφαρμογής, αρκεί να ολοκληρωθεί η ανάπτυξη και η δοκιμαστική της λειτουργία. Αφού μεταφορτωθεί στο διαδικτυακό διακομιστή που τη φιλοξενεί, όλες οι αναβαθμίσεις θα είναι διαθέσιμες σε όλους τους χρήστες που τη χρησιμοποιούν.

Οι εφαρμογές αυτού του είδους έχουν σημαντικά πλεονεκτήματα. Οι διαδικτυακές εφαρμογές στη γενικότητά τους, απαιτούν λιγότερους πόρους και κόστος για να αναπτυχθούν. Επίσης, λόγω της ιδιότητας της HTML5 να λειτουργεί πανομοιότυπα σε όλες τις συσκευές και τα λειτουργικά συστήματα, καθιστά τις διαδικτυακές εφαρμογές συμβατές με αυτά. Όπως ήδη

¹⁴ https://www2.deloitte.com/gr/en/pages/strategy-operations/articles/2017_shared_services_survey_results_deloitte_greece.html

¹⁵ <https://www.cognizant.com/whitepapers/Hybrid-Mobile-Application-Analysis-and-Guidelines-codex888.pdf>

αναφέρθηκε, η συντήρηση και αναβάθμιση τέτοιου είδους εφαρμογών είναι πιο εύκολη σε σχέση με τις εγγενείς εφαρμογές.

Όσον αφορά τα μειονεκτήματα των διαδικτυακών εφαρμογών, αυτές δεν επωφελούνται από τις αναβαθμίσεις του λειτουργικού συστήματος και τις πρόσθετες δυνατότητές τους. Επίσης, η δημιουργία απαιτητικών εφαρμογών (όπως παιχνιδιών) είναι πιο δύσκολη από τις εγγενείς εφαρμογές. Σημαντικό μειονέκτημα των διαδικτυακών εφαρμογών είναι ότι δε μπορούν να χρησιμοποιηθούν χωρίς σύνδεση στο διαδίκτυο.

4.2 Εγγενείς εφαρμογές

Οι εγγενείς εφαρμογές αναπτύσσονται χρησιμοποιώντας μια γλώσσα προγραμματισμού η οποία επικοινωνεί απευθείας με το λειτουργικό σύστημα της κινητής συσκευής. Για παράδειγμα, το λειτουργικό σύστημα Android, λειτουργεί σε γλώσσα Java και οι εγγενείς του εφαρμογές απαιτείται να έχουν γραφεί σε Java. Για το λειτουργικό σύστημα iOS η γλώσσα προγραμματισμού είναι η Objective C ή η Swift.

Κάθε πλατφόρμα διαθέτει ένα διαφορετικό σύνολο εργαλείων για τους προγραμματιστές των εγγενών εφαρμογών, συμπεριλαμβανομένου του μηχανισμού μεταφόρτωσης της εφαρμογής στο ηλεκτρονικό κατάστημα διάθεσης των εφαρμογών. Αυτό σημαίνει ότι για να είναι δυνατή η εγκατάσταση μιας εφαρμογής, πρέπει ο χρήστης να την κατεβάσει από το ηλεκτρονικό κατάστημα.

Οι εγγενείς εφαρμογές κινητών προσφέρουν την καλύτερη εμπειρία χρήσης, διαθέτοντας περισσότερα λειτουργικά χαρακτηριστικά, διότι είναι πιο ευέλικτες και προσαρμόσιμες στις εξειδικευμένες δυνατότητες κάθε συσκευής. Ο χρόνος φόρτωσής τους είναι μικρότερος σε σχέση με τις διαδικτυακές εφαρμογές, ενώ και οι επιδόσεις τους είναι καλύτερες τόσο σε κατάσταση σύνδεσης, όσο και χωρίς σύνδεση.

Σε επιχειρηματικό επίπεδο, υπάρχει η πεποίθηση ότι η ανάπτυξη και διάθεση εγγενών εφαρμογών στους συνεργάτες και τους πελάτες δημιουργεί καλύτερη εταιρική εικόνα, ενώ εγγυάται και υψηλότερα επίπεδα ασφάλειας σε σχέση με τις εφαρμογές HTML5. Το κόστος ανάπτυξης μιας εγγενούς εφαρμογής είναι υψηλότερη σε σχέση με μια αντίστοιχη HTML5, δεδομένου ότι η ίδια εφαρμογή θα πρέπει να ξαναγραφεί για καθεμιά από τις πλατφόρμες στις οποίες απευθύνεται, ενώ και ο χρόνος ανάπτυξης είναι μεγαλύτερος για την εφαρμογή καθαυτή.

Στα πλεονεκτήματα των εγγενών εφαρμογών συγκαταλέγεται η ευκολία να μεταφορτώνονται οι εφαρμογές από ένα ηλεκτρονικό κατάστημα εφαρμογών και να είναι διαθέσιμες για χρήση αμέσως μετά την εγκατάστασή τους. Η διαχείριση των εφαρμογών είναι απλή διαδικασία, τόσο για τους προγραμματιστές, όσο και για τους τελικούς χρήστες.

Ένα από τα πιο σημαντικά πλεονεκτήματά τους είναι η αξιοποίηση δυνατοτήτων της κινητής συσκευής, όπως η κάμερα, ο γεωγραφικός εντοπισμός, το βιβλίο διευθύνσεων, κ.λπ. Επίσης μια εγγενής εφαρμογή μπορεί να συνεργαστεί με άλλες εγγενείς εφαρμογές, ώστε να επωφεληθεί από τα χαρακτηριστικά των συνεργαζόμενων εφαρμογών. Επιπλέον, έχουν δημιουργηθεί πολυπληθείς κοινότητες προγραμματιστών εγγενών εφαρμογών, στις οποίες οι προγραμματιστές δημιουργούν μια κοινή γνωσιακή βάση για επίλυση προγραμματιστικών θεμάτων, συντείνοντας με αυτό τον τρόπο στην επιτάχυνση του κύκλου ανάπτυξης μιας εφαρμογής. Τέλος, δεδομένου ότι οι εγγενείς αναπτύσσονται με τη γλώσσα του λειτουργικού συστήματος και επικοινωνούν άμεσα με αυτό, επωφελούνται από τις εκάστοτε αναβαθμίσεις του και ενσωματώνουν καλύτερα χαρακτηριστικά ασφάλειας.

Ένα από τα κύρια μειονεκτήματα των εγγενών εφαρμογών είναι η εξάρτησή τους από το συγκεκριμένο λειτουργικό σύστημα. Σε επιχειρηματικό επίπεδο, η ανάπτυξη εγγενών εφαρμογών απαιτεί μεγαλύτερες επενδύσεις, δεδομένου ότι χρειάζονται ξεχωριστές ομάδες για τη σχεδίαση, ανάπτυξη και συντήρηση της εφαρμογής για καθεμιά πλατφόρμα στην οποία απευθύνεται. Τέλος η διαχείριση των εγγενών εφαρμογών είναι μια δύσκολη διαδικασία, δεδομένου ότι οποιαδήποτε αλλαγή θα πρέπει να προγραμματιστεί για όλες τις λειτουργικές πλατφόρμες της εφαρμογής ξεχωριστά.

4.3 Υβριδικές εφαρμογές κινητού

Οι υβριδικές εφαρμογές χρησιμοποιούν συνδυασμό μεθοδολογιών ανάπτυξης και χρησιμοποιούν τόσο την HTML5, όσο και τα εγγενή εργαλεία του λειτουργικού συστήματος,

ώστε να ενσωματώσουν τα πλεονεκτήματα των δύο τεχνολογιών. Αυτή η προσέγγιση δίνει τη δυνατότητα υλοποίησης των κύριων λειτουργιών της εφαρμογής με χρήση web τεχνολογιών και παράλληλα να πρόσβασης σε χαρακτηριστικά της συσκευής που είναι διαθέσιμα μόνο για εγγενείς εφαρμογές, μέσω των εγγενών προγραμματιστικών διεπαφών (APIs) που λειτουργούν ως γέφυρα ανάμεσα στις κύριες λειτουργίες της εφαρμογής και τα χαρακτηριστικά της συσκευής. Με τον τρόπο αυτό επιτυγχάνεται η επαναχρησιμοποίηση ενοτήτων της εφαρμογής που υλοποιήθηκε σε web τεχνολογίες, ενώ το μόνο που χρειάζεται να τροποποιηθεί για να λειτουργήσει η εφαρμογή και σε άλλα λειτουργικά είναι η αλλαγή των native APIs που χρησιμοποιούνται για τις λειτουργίες της συσκευής (LionbridgeWhitepaper, 2012).

Δεδομένου ότι οι χρήστες συχνά απαιτούν την ίδια εφαρμογή σε διαφορετικές πλατφόρμες, οι επιχειρήσεις υιοθετούν αυτή τη μέθοδο για ανάπτυξη των εφαρμογών τους. Οι υβριδικές εφαρμογές αναπτύσσονται πιο γρήγορα, ενώ ταυτόχρονα δίνουν στο χρήστη ικανοποιητική εμπειρία χρήσης.

Μεταξύ των πλεονεκτημάτων των υβριδικών εφαρμογών, είναι ότι η αναβάθμισή τους δεν απαιτεί την έγκριση ή την παρέμβαση του χρήστη, δεδομένου ότι αυτή γίνεται ενώ ο χρήστης χρησιμοποιεί την εφαρμογή. Επίσης, μια υβριδική εφαρμογή έχει ομοιόμορφη όψη και εργονομία σε όλες τις λειτουργικές πλατφόρμες, ενώ μπορούν να χρησιμοποιηθούν κοινά εργαλεία παρακολούθησης της χρήσης τους σε όλες τις πλατφόρμες.

Όσον αφορά στα μειονεκτήματα των υβριδικών εφαρμογών, αυτές δε μπορούν να υλοποιήσουν εύκολα απαιτητικές εφαρμογές, όπως τα παιχνίδια. Επίσης, το επίπεδο εμπειρίας του χρήστη δεν είναι όμοιο με αυτό των εγγενών εφαρμογών.

4.4 Σύγκριση

Η απόφαση για τη δημιουργία ενός τύπου εφαρμογής επιβάλλεται από πολλούς παράγοντες και δεν είναι μονοσήμαντη. Πρέπει να αξιολογηθούν τα πλεονεκτήματα και τα μειονεκτήματα κάθε τύπου εφαρμογής και να ληφθεί ανάλογα η απόφαση.

Οι εφαρμογές HTML5 δημιουργούν γρήγορα αλληλεπιδραστικές εφαρμογές. Όμως δεν έχουν τη δυνατότητα να εκμεταλλευτούν πλήρως τις δυνατότητες της συσκευής και να ενσωματώσουν τα χαρακτηριστικά των διεπαφών της εκάστοτε πλατφόρμας. Αντίθετα, οι εγγενείς εφαρμογές δίνουν στους χρήστες την ευχέρεια να χρησιμοποιούν τις ευκολίες της συσκευής, έχουν όμως μεγαλύτερο χρόνο ανάπτυξης και το συνολικό κόστος της εφαρμογής αυξάνεται δεδομένου ότι πρέπει να προγραμματίζεται για όλες τις λειτουργικές πλατφόρμες

Ένα από τα βασικότερα πλεονεκτήματα των διαδικτυακών εφαρμογών είναι ότι με τη χρήση HTML5 αρκεί η ανάπτυξη μιας και μόνης εκδοχής της εφαρμογής για όλες τις συσκευές, γεγονός το οποίο εξοικονομεί χρόνο και πόρους ανάπτυξης. Επιπλέον, μία εφαρμογή βασισμένη αποκλειστικά σε τεχνολογίες HTML, CSS και JavaScript μπορεί να μετατραπεί σε εφαρμογή κινητού με τη χρήση ενός mobile wrapper (όπως το Phone Gap), επιτρέποντάς της να διατίθεται από τα ηλεκτρονικά καταστήματα εφαρμογών των λειτουργικών πλατφορμών. Αυτού του είδους οι εφαρμογές ονομάζονται υβριδικές και συνδυάζουν χαρακτηριστικά τόσο HTML5 όσο και εγγενών εφαρμογών.

Οι επιχειρήσεις αποκτούν ένα μεγάλο πλεονέκτημα, δεδομένου ότι περιορίζουν δραστικά το κόστος ανάπτυξης, ενώ ταυτόχρονα μειώνουν το χρόνο που απαιτείται έως την εμπορική διάθεση της εφαρμογής. Σημαντικό χαρακτηριστικό τους επίσης είναι, ότι περιορίζεται η πιθανότητα εμφάνισης σφαλμάτων μεταξύ των διαφορετικών πλατφορμών. Με δεδομένο ότι η ανάπτυξη μιας εφαρμογής HTML5 απαιτεί τον έλεγχο του κώδικά της μία φορά, ο χρόνος των δοκιμών της εφαρμογής μειώνεται ανάλογα, επιτρέποντας την ταχεία ανάπτυξη εμπορικά διαθέσιμων εφαρμογών.

Οι εφαρμογές HTML5 ταιριάζουν σε περιβάλλοντα τα οποία επεκτείνουν υπάρχοντα πληροφοριακά συστήματα. Η HTML5 μπορεί να ανταποκριθεί με επάρκεια σε ελαφρές εφαρμογές, οι οποίες δεν έχουν αυξημένες απαιτήσεις ασφάλειας ούτε πρόσβαση σε λειτουργικότητα της συσκευής.

Σε περιπτώσεις τέλος στις οποίες το περιεχόμενο της εφαρμογής τροποποιείται συχνά, η HTML5 αποτελεί επίσης την πιο πρόσφορη επιλογή.

5. Location Based Services

Τα συστήματα τοποθεσίας, έχουν αναπτυχθεί για τον προσδιορισμό και την παρακολούθηση της θέσης των χρηστών κινητών συσκευών όπως τα τηλέφωνα, με χρήση των παγκόσμιων συστημάτων εντοπισμού θέσης (GPS), καθώς και διάφορα συστήματα τριγωνισμού που χρησιμοποιούν σήματα κυψελοειδούς τηλεφώνου, τηλεοπτικά σήματα εκπομπής ή αντίστοιχες τεχνικές. Η διαρκής επέκταση των χρήσεων των συστημάτων εντοπισμού δημιουργεί έντονες ανάγκες για βελτιωμένα συστήματα για διάφορους σκοπούς.

Η διάδοση των έξυπνων κινητών τηλεφώνων έχει εισάγει τα δεδομένα θέσης σε πραγματικό χρόνο σε πολλές πτυχές της ζωής του ανθρώπου. Ορισμένες υπηρεσίες κινητής τηλεφωνίας, χρησιμοποιούν τη θέση του smartphone για να προσφέρουν στον χρήστη κατευθύνσεις, στοχοθετημένες συστάσεις ή άλλες πληροφορίες για τη συγκεκριμένη τοποθεσία που βρίσκεται. Άλλες υπηρεσίες ενσωματώνουν ένα "στρώμα" (layer) τοποθεσιών σε άλλους τύπους λειτουργιών. Επίσης έχουν αναπτυχθεί υπηρεσίες κοινοποίησης της θέσης του χρήστη με τους φίλους ή το ευρύ κοινό. Σήμερα διαμορφώνονται δύο τύποι υπηρεσιών εντοπισμού: (α) εκείνες που χρησιμοποιούν τη θέση του ατόμου για να παρέχουν πληροφορίες με γεωγραφική στόχευση, όπως οδηγίες ή συστάσεις, και (β) γεωσκοπικές υπηρεσίες που επιτρέπουν στους χρήστες να δηλώνουν την παρουσία τους σε συγκεκριμένες τοποθεσίες ή να κοινοποιούν την τοποθεσία τους στους φίλους τους. Ωστόσο, γίνεται ολοένα και πιο δύσκολο να οριοθετηθούν αυτές οι δύο κατηγορίες, καθώς αφενός οι υπηρεσίες κοινωνικών μέσων δικτύωσης ενσωματώνουν την τοποθεσία ως στοιχείο της δραστηριότητας των χρηστών, αφετέρου οι υπηρεσίες εντοπισμού περιλαμβάνουν ποικίλους βαθμούς κοινωνικής λειτουργικότητας.

Η **Foursquare**, μια γεωχωρική υπηρεσία που αρχικά επικεντρώθηκε στην κοινοποίηση της θέσης του χρήστη της, ο οποίος κέρδιζε κάποιους εικονικούς «πόντους» κάθε φορά που κοινοποιούσε τη θέση του, τώρα αρχίζει να υποβαθμίζει το σύστημα των «πόντων» και της κοινοποίησης της θέσης στα κοινωνικά δίκτυα¹⁶ (Vindu & , 2013). Αντί για αυτό, χρησιμοποιεί τα δεδομένα που συγκέντρωσε από την έως τώρα λειτουργία της, για να προσφέρει στους χρήστες προτάσεις σε πραγματικό χρόνο, οι οποίες όμως προσαρμόζονται στην τοποθεσία που βρίσκεται ο χρήστης, ενώ ταυτόχρονα προβάλλει δραστηριότητες και συστάσεις φίλων και άλλων χρηστών που σχετίζονται με την εν λόγω τοποθεσία¹⁷ (Foursquare , 2013).

Εν τω μεταξύ, οι υπηρεσίες κοινωνικών δικτύων όπως το Facebook, το Instagram και το Twitter έχουν προσθέσει ένα προαιρετικό επίπεδο αναγνώρισης τοποθεσίας, ώστε οι χρήστες να μπορούν να δηλώσουν τη θέση που βρίσκονται όταν δημοσιεύουν αντίστοιχο υλικό στους ιστότοπους κοινωνικής δικτύωσης. Επίσης, πολλές υπηρεσίες που εστιάζονται στην παροχή πληροφοριών (όπως εφαρμογές παρακολούθησης δραστηριότητας και ιστότοποι αξιολόγησης προϊόντων και υπηρεσιών όπως το Yelp), ενσωματώνουν την κοινή χρήση της τοποθεσίας στις άλλες κοινωνικές πτυχές που ενσωματώνουν.

5.1 Το επιστημονικό πεδίο των υπηρεσιών εντοπισμού

Σε αντίθεση με άλλα παραδοσιακά συστήματα γεωγραφικών πληροφοριών (GIS) και εφαρμογές χαρτογράφησης στο διαδίκτυο, ο γεωγραφικός εντοπισμός αναγνωρίζει το πλαίσιο στο οποίο βρίσκονται οι χρήστες μιας αντίστοιχης εφαρμογής και μπορεί να προσαρμόσει ανάλογα το περιεχόμενο και την παρουσίασή του (Steiniger, et al., 2006). Ένα άλλο ιδιαίτερο χαρακτηριστικό είναι ότι οι εφαρμογές εντοπισμού χρησιμοποιούνται συχνά σε ένα δυναμικό και περιβάλλον κινητών επικοινωνιών (Huang&Gao, 2018). Αυτά τα ξεχωριστά χαρακτηριστικά καθιστούν την ανάπτυξη εφαρμογών εντοπισμού διαφορετική από άλλες εφαρμογές τύπου GIS και ανοίγουν πολλά ερευνητικά ερωτήματα πέραν του επιστημονικού πεδίου της επιστήμης γεωγραφικών πληροφοριών (GIScience).

Γενικά, η έρευνα που διεξάγεται στο επιστημονικό πεδίο του γεωγραφικού εντοπισμού, μπορεί να ταξινομηθεί σε επτά περιοχές: τοποθέτηση, μοντελοποίηση, επικοινωνία, αξιολόγηση, εφαρμογές, ανάλυση δεδομένων που παράγονται από εντοπισμό, κοινωνικές και συμπεριφορικές επιπτώσεις. Οι τρεις πρώτοι τομείς αντιπροσωπεύουν τον πυρήνα του εντοπισμού, δεδομένου πως κάθε εφαρμογή εντοπισμού πρέπει να διεκπεραιώνει τα κύρια

¹⁶ With New App, Foursquare Strives to Be 'Magic' in Your Pocket

¹⁷ <http://blog.foursquare.com/2013/08/29/a-smarter-foursquare-so-you-dont-miss-a-thing/>

καθήκοντα της εύρεσης της τοποθεσίας, της μοντελοποίησης δεδομένων και της επικοινωνίας δεδομένων.

Εντοπισμός τοποθεσίας (positioning): Όπως δηλώνει το όνομα της υπηρεσίας, ο εντοπισμός πρέπει να είναι σε θέση να καθορίσει τη θέση του χρήστη. Ως εκ τούτου, η τοποθέτηση ή προσδιορισμός θέσης είναι ένα κρίσιμο κομμάτι του εντοπισμού. Σε υπαίθρια περιβάλλοντα, τα παγκόσμια δορυφορικά συστήματα πλοήγησης (GNSS), όπως τα παγκόσμια συστήματα εντοπισμού θέσης (GPS), έχουν απλουστεύσει τη λύση αυτού του προβλήματος. Ωστόσο, σε άλλου είδους περιβάλλοντα, όπως σε πυκνά αστικά περιβάλλοντα, σε εσωτερικούς και υπόγειους χώρους, η ακριβής και αξιόπιστη εύρεση της θέσης εξακολουθεί να αποτελεί σημαντική τεχνολογική πρόκληση. Η έρευνα σε αυτό το θέμα επικεντρώνεται κυρίως στον αέναο εντοπισμό θέσης, με στόχο την ακριβή και έγκαιρη εκτίμηση της θέσης ενός χρήστη ή ενός αντικειμένου οποτεδήποτε και οπουδήποτε.

Μοντελοποίηση: Οι χρήστες έχουν κεντρική θέση στις διαδικασίες εντοπισμού. Για την υποστήριξη χρηστών, ο εντοπισμός θα πρέπει να μοντελοποιεί τη θέση, το χωρικό πλαίσιο, τα χαρακτηριστικά και τις ανάγκες ενός χρήστη κινητής τηλεφωνίας και να παρέχει υπηρεσίες προσαρμοσμένες σε αυτούς. Ταυτόχρονα, ο γεωγραφικός χώρος και οι θέσεις που σχετίζονται με τις εφαρμογές εντοπισμού θα πρέπει επίσης να μοντελοποιούνται αποτελεσματικά. Η έρευνα σε αυτή την πτυχή επικεντρώνεται κυρίως στον τρόπο με τον οποίο αυτά τα είδη πληροφοριών μπορούν να μοντελοποιηθούν στο πλαίσιο του εντοπισμού και πώς μπορούν να χρησιμοποιηθούν για την παροχή εξατομικευμένων υπηρεσιών.

Επικοινωνία: Από την πλευρά του χρήστη, οι εφαρμογές εντοπισμού του παρέχουν πληροφορίες σχετικές με την τρέχουσα θέση του μέσω κινητών συσκευών, ώστε να υποστηρίξουν τη λήψη αποφάσεων και τις δραστηριότητές του στο συγκεκριμένο γεωγραφικό χώρο. Αυτό μπορεί να θεωρηθεί ως μια διαδικασία επικοινωνίας, στην οποία οι σχετικές (χωροχρονικές) πληροφορίες μεταφέρονται από τις εφαρμογές εντοπισμού στους χρήστες. Η έρευνα σε αυτή την πτυχή επικεντρώνεται κυρίως σε δύο βασικά ερωτήματα: Ποιες πληροφορίες πρέπει να κοινοποιούνται στον χρήστη και σε ποιες μορφές παρουσίασης (π.χ. κινητοί χάρτες, επαυξημένη πραγματικότητα ή προφορική καθοδήγηση).

Εφαρμογές: Η ταχύτατη πρόοδος στις παραπάνω πτυχές (εντοπισμός, μοντελοποίηση και επικοινωνία) πυροδότησαν την ανάπτυξη πολλών καινοτόμων εφαρμογών εντοπισμού, βρίσκοντας χρήσεις της συγκεκριμένης τεχνολογίας σε διάφορους τομείς όπως η πλοήγηση και η αναζήτηση οδών, ο τουρισμός, τα κοινωνικά δίκτυα, η ψυχαγωγία, η υγειονομική περιθάλψη και οι μεταφορές.

Αξιολόγηση: Για να διασφαλιστεί ότι μια εφαρμογή εντοπισμού ανταποκρίνεται στις ανάγκες του χρήστη, είναι απαραίτητη η αξιολόγηση αυτών των υπηρεσιών, όσον αφορά τη χρηστικότητα και τη χρησιμότητά τους. Λόγω του ότι οι εφαρμογές εντοπισμού χρησιμοποιούνται συχνά ενώ οι άνθρωποι βρίσκονται σε κίνηση, πρέπει να εξεταστούν περαιτέρω οι πτυχές της λήψης αποφάσεων για κινητές συσκευές σε πραγματικό χρόνο. Αυτή η απαίτηση δημιουργεί πολλές μεθοδολογικές προκλήσεις για τους ερευνητές του πεδίου του εντοπισμού και τους προγραμματιστές.

Ανάλυση των δεδομένων που παράγονται από τον εντοπισμό: Οι εφαρμογές εντοπισμού όχι μόνο βοηθούν στη διευκόλυνση των καθημερινών δραστηριοτήτων και λήψης αποφάσεων σε συνάρτηση με το γεωγραφικό χώρο, αλλά παράγουν και πολλά στοιχεία για το πώς οι άνθρωποι ζουν, ταξιδεύουν και αλληλεπιδρούν μεταξύ τους στο περιβάλλον. Ως εκ τούτου, ένας κλάδος έρευνας στο επιστημονικό πεδίο του εντοπισμού επικεντρώνεται στην ανάλυση των δεδομένων εντοπισμού βάσει τοποθεσίας, δεδομένων κοινωνικών μέσων και γεωγραφικών πληροφοριών, με σκοπό την καλύτερη κατανόηση της συμπεριφοράς των ανθρώπων σε διαφορετικά περιβάλλοντα. Η εξόρυξη αυτών των (μεγάλων) χωρικών δεδομένων επιτρέπει την ανάπτυξη καινοτόμων εφαρμογών σε τομείς όπως οι μεταφορές, ο πολεοδομικός σχεδιασμός, οι έξυπνες πόλεις και οι κοινωνικές επιστήμες, ενώ μπορεί να δώσει κατευθύνσεις για την περαιτέρω βελτίωση των εφαρμογών εντοπισμού, οι οποίες παράγουν αυτά τα δεδομένα.

Κοινωνικές και συμπεριφορικές επιπτώσεις: Τα ζητήματα ιδιωτικού απορρήτου αποτελούν πρόκληση για την τεχνολογία εντοπισμού επί μακρόν. Τα τελευταία χρόνια, η αυξανόμενη χρήση του εντοπισμού, καθώς και η τεχνολογία εντοπισμού θέσης / δραστηριότητας, προκάλεσαν περαιτέρω ζητήματα σχετικά με το ιδιωτικό απόρρητο, καθώς και

κοινωνικά, νομικά και δεοντολογικά ζητήματα. Πολλές βασικές ερωτήσεις αντιμετωπίζονται συχνά σε αυτό το πλαίσιο, για παράδειγμα, "Ποια είναι τα προσωπικά και ηθικά ζητήματα του εντοπισμού;" και "Πώς μπορούμε να αντιμετωπίσουμε καλύτερα την ιδιωτικότητα και τις δεοντολογικές ανησυχίες των χρηστών μιας εφαρμογής εντοπισμού;".

6. Μεθοδολογία υλοποίησης

Στο κεφάλαιο αυτό θα αναφερθούμε στις τεχνολογίες και αρχιτεκτονικές που χρησιμοποιήθηκαν για την υλοποίηση αυτής εργασίας.

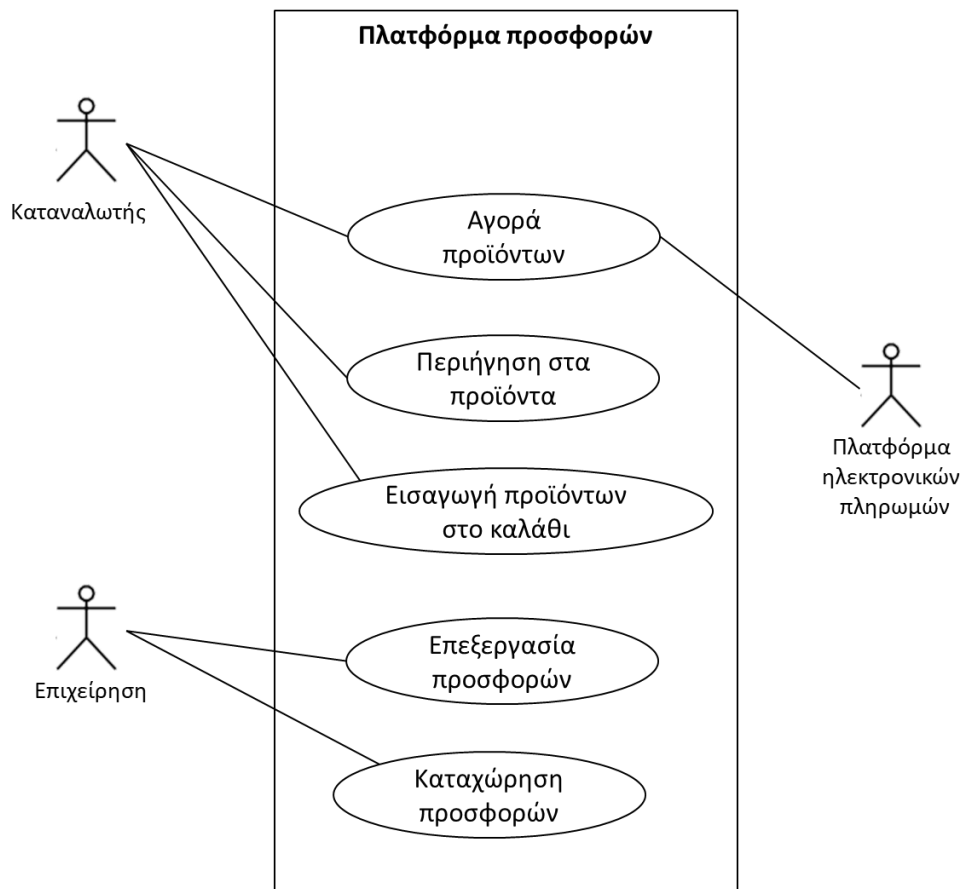
6.1 Λογικός σχεδιασμός

Ο σχεδιασμός της εφαρμογής παράγει προδιαγραφές σε λογικό και φυσικό επίπεδο. Ο λογικός σχεδιασμός προσδιορίζει τις οντότητες του συστήματος και τις σχέσεις μεταξύ τους όπως αυτές θα γίνουν αντιληπτές από τους χρήστες. Περιγράφει τις εισόδους και τις εξόδους, τις λειτουργίες επεξεργασίας που θα εκτελούνται, τις επιχειρησιακές διαδικασίες, τα μοντέλα δεδομένων και τους ελέγχους.

Στο επιχειρησιακό μοντέλο της εφαρμογής εμπλέκονται δύο οντότητες:

- Οι επιχειρήσεις που κάνουν προσφορές και συμμετέχουν στο σύστημα.
- Οι χρήστες που έχουν χρησιμοποιούν την εφαρμογή στη φορητή τους συσκευή και αναζητούν προσφορές.

Το μοντέλο αλληλεπίδρασης των οντοτήτων με το σύστημα περιγράφεται στο use-case διάγραμμα που ακολουθεί.



Διάγραμμα 1. Use-Case

Όπως φαίνεται από το Διάγραμμα 1, η εφαρμογή υποστηρίζει σχέσεις αλληλεπίδρασης τόσο με επιχειρήσεις όσο και με καταναλωτές. Οι επιχειρήσεις χρησιμοποιούν την εφαρμογή για να καταχωρούν προσφορές και να ρυθμίζουν παραμέτρους δημοσίευσής τους (χρόνο έναρξης και λήξης, κατάσταση προσφοράς). Οι καταναλωτές χρησιμοποιούν την εφαρμογή για να βρίσκουν προσφορές, οι οποίες βρίσκονται σε δεδομένη ακτίνα γύρω από το σημείο στο οποίο βρίσκονται και να αγοράζουν από το αντίστοιχο κατάστημα. Το λογισμικό υποστηρίζει πληρωμή μέσω της πλατφόρμας ηλεκτρονικών πληρωμών PayPal.

6.1.1 Λειτουργία Business-To-Business

Οι επιχειρήσεις που έχουν συμβληθεί με το σύστημα, έχουν τη δυνατότητα να εισάγουν στο σύστημα τα δεδομένα που αφορούν στην προσφορά που διαθέτει το αντίστοιχο κατάστημα. Για την καταχώρηση μιας νέας προσφοράς απαραίτητη είναι η σύνδεση στο διαδίκτυο μέσα από έναν φυλλομετρητή και η αντίστοιχη εξουσιοδότηση για πρόσβαση στη σελίδα εισαγωγής των δεδομένων της νέας προσφοράς.

Σημείωση: Τα δικαιώματα πρόσβασης κάθε χρήστη προσδιορίζονται από το διαχειριστή του συστήματος, μετά την εγγραφή του χρήστη.

Αυτή η προσέγγιση προσφέρει έναν πολύ απλό τρόπο ενημέρωσης προσφορών, χωρίς να είναι απαραίτητη κάποια εξειδικευμένη εφαρμογή από την πλευρά της επιχείρησης.

6.1.2 Λειτουργία Business-To-Customer

Τα βήματα λειτουργίας της εφαρμογής είναι τα ακόλουθα:

1. Ο χρήστης συνδέεται στον ιστότοπο της εφαρμογής είτε από τον υπολογιστή είτε από την φορητή συσκευή.
 - Η εφαρμογή (έχοντας ενεργοποιημένη τη δυνατότητα GPS της συσκευής) αναγνωρίζει τις τρέχουσες συντεταγμένες του χρήστη.
 - Η εφαρμογή ενεργοποιεί μια RESTful υπηρεσία για την ανάκτηση των προσφορών, οι οποίες πληρούν τα κριτήρια που έχει εισάγει ο χρήστης στο περιβάλλον εργασίας του. Η RESTful υπηρεσία επιστρέφει ένα αρχείο σε μορφότυπο JSON, το οποίο περιέχει τα αποτελέσματα με τα σημεία πώλησης που έχουν καταχωρήσει προσφορές.
 - Ο χρήστης μπορεί να ορίσει φίλτρο απόστασης και να αναζητά προσφορές σε σημεία μέσα στη δεδομένη ακτίνα.
 - Η εφαρμογή μεταφράζει το μορφότυπο JSON και εμφανίζει σε λίστα αλλά και στο χάρτη τα διαθέσιμα σημεία.
 - Οι προσφορές εμφανίζονται σε χάρτη Google με marker και info window. Το info window περιλαμβάνει πληροφορίες όπως το όνομα καταστήματος, η κατηγορία της προσφοράς, ο τίτλος της και η τιμή).
 - Η θέση του χρήστη επισημαίνεται στο χάρτη με τη χρήση διαφορετικού marker.
 - Κάτω από την περιοχή εμφάνισης του χάρτη υπάρχει κατάλογος με τις προσφορές και τα λεπτομερή τους στοιχεία, ταξινομημένος χρονικά, από την προσφορά που δημοσιεύτηκε πιο πρόσφατα προς αυτή που δημοσιεύτηκε παλαιότερα.
 - Ο κατάλογος προσφορών εμφανίζει τα εξής στοιχεία για την προσφορά: το όνομα του καταστήματος, την περιγραφή της προσφοράς, το προϊόν της προσφοράς, το χρόνο που αυτή ισχύει, και την απόσταση του από το σημείο που βρίσκεται ο χρήστης.
 - Ο χρήστης επιλέγει μία προσφορά, Η σελίδα ενημερώνεται με λεπτομέρειες της προσφοράς. Εμφανίζεται το πλήρες κείμενο της προσφοράς ενώ υπάρχει πεδίο για τον προσδιορισμό του πλήθους των προϊόντων που ο χρήστης θέλει να παραγγείλει. Για να καταχωρηθεί η παραγγελία, χρήστης πατάει το αντίστοιχο κουμπί.
 - Ο χρήστης συνεχίζει την περιήγησή του στις προσφορές που βρίσκονται δημοσιευμένες και κατά βούληση προσθέτει στο καλάθι νέες προσφορές.

- Όταν ολοκληρώσει τις επιλογές του, εμφανίζει τη σελίδα του καλαθιού. Από τη σελίδα αυτή έχει δυνατότητα να αφαιρέσει κάποια προσφορά ή να προχωρήσει στην ηλεκτρονική πληρωμή.
- Η πληρωμή γίνεται μέσω κλήσης RESTful υπηρεσίας της πλατφόρμας PayPal. Η πληρωμή ολοκληρώνεται εικονικά και επιστρέφεται JSONαρχείο με το αποτέλεσμα της.
- Η εφαρμογή αναλύει το αρχείο JSONκαι εμφανίζει μήνυμα για την επιτυχή ή όχι ολοκλήρωση της πληρωμής. Εάν η πληρωμή έγινε κανονικά, το καλάθι αδειάζει.

6.2 Αρχιτεκτονική

Σε αυτήν την ενότητα θα αναφερθούμε στην αρχιτεκτονική που ακολουθήθηκε για την ανάπτυξη της εφαρμογής.

Η αρχιτεκτονική της εφαρμογής βασίζεται στο μοντέλο Model – View – Controller, το οποίο κατανέμει τις διαφορετικές λειτουργίες της εφαρμογής σε διακριτά στρώματα λογισμικού.

Οι λειτουργίες διακρίνονται σε τρεις κατηγορίες:

1. Επικοινωνία με τη βάση δεδομένων για υποβολή επερωτήσεων SQL.
2. Επεξεργασία των στοιχείων και έλεγχοι τιμών.
3. Εμφάνιση και μορφοποίηση των δεδομένων.

6.2.1 Model

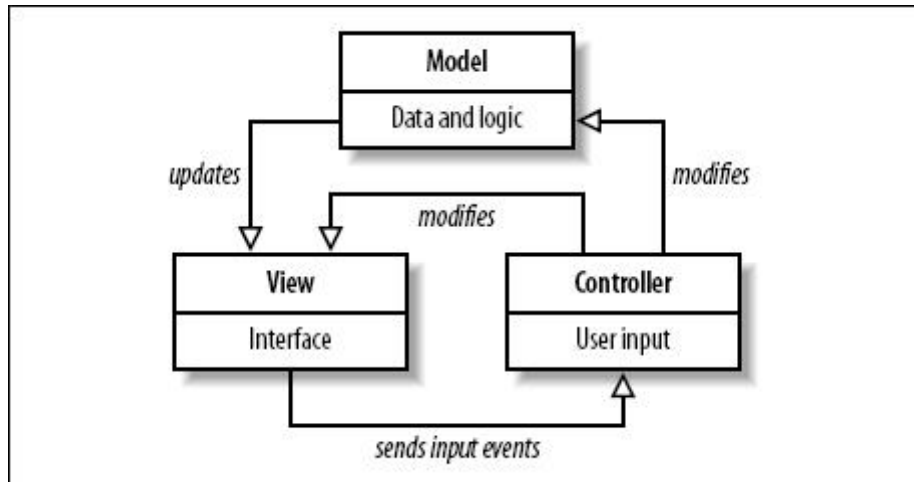
Στο στρώμα Model τοποθετούνται οι λειτουργίες της εφαρμογής που σχετίζονται με την πρόσβαση στη βάση δεδομένων. Οι λειτουργίες αυτές γράφονται με τη μορφή συναρτήσεων. Είναι κάποιες συναρτήσεις με τις οποίες εκτελούνται λειτουργίες διαχείρισης των δεδομένων που λαμβάνουμε από τη βάση. Για παράδειγμα για να εμφανιστεί η σελίδα των προσφορών τα οποία βρίσκονται στη βάση δεδομένων, το πρώτο βήμα στο Model των προσφορών είναι να υπάρχει η function “get_products()” η οποία περιέχει κώδικα που επικοινωνεί με τη βάση δεδομένων και αντλεί τα επιθυμητά δεδομένα.

6.2.2 View

Μέσα στη View υπάρχει το πρότυπο της HTML της σελίδας της εφαρμογής στο οποίο συγχωνεύονται τα δυναμικά δεδομένα που λαμβάνονται από το στρώμα Model και έτσι μορφοποιείται ό,τι βλέπει ο χρήστης στο φυλλομετρητή του. Τις περισσότερες φορές μία View επικοινωνεί με έναν Controller και αφού ο Controller κάνει τις διάφορες επεξεργασίες των δεδομένων, στέλνει στη View συγκεκριμένα δεδομένα για να προβληθούν.

6.2.3 Controller

Ο Controller είναι ο διαμεσολαβητής μεταξύ του Model και της View. Ελέγχει τον τρόπο που εκτελείται η εφαρμογή. Ελέγχει επιχειρησιακούς κανόνες, κανόνες πρόσβασης και εκτελεί ελέγχους ορθότητας. Επικοινωνεί με το Model, αντλεί τα δεδομένα που ζητά και εν συνεχεία αφού τα επεξεργαστεί, τα στέλνει πίσω στη View για απεικόνιση. Πιο κάτω θα δούμε μια συνολική απεικόνιση του MVC.



Εικόνα 7. Ένα παράδειγμα MVC

Στο παράδειγμα της παραπάνω εικόνας συμβαίνουν τα εξής:

- Ο χρήστης δίνει κάποια στοιχεία σε μία HTML φόρμα. Για παράδειγμα συμπληρώνει τη φόρμα εισαγωγής νέας προσφοράς και πατάει το κουμπί submit.
- Στη συνέχεια ο Controller έχοντας λάβει το input του χρήστη επικοινωνεί με το Model χρησιμοποιώντας το input του χρήστη σαν μεταβλητή.
- Το Model διαβιβάζει τα στοιχεία στη βάση δεδομένων και ελέγχει δεδομένα από το Model τα οποία, όταν τα λαμβάνει τα προσαρμόζει ανάλογα με αυτό που ζήτησε ο χρήστης.
- Στη συνέχεια ο Controller ενημερώνει τη View με ένα κατάλληλο μήνυμα.

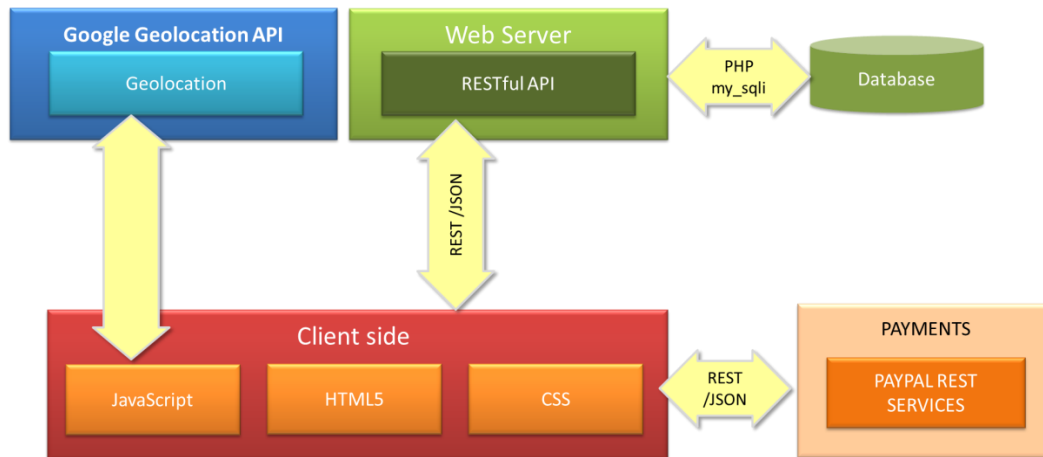
Για παράδειγμα στην εφαρμογή, όταν ο χρήστης ζητήσει να δει το καλάθι αγορών του, ο αντίστοιχος Controller θα επικοινωνήσει με τη συνάρτηση `get_basket()` του Model που θα ζητάει όλα τα περιεχόμενα του πίνακα BASKET με παράμετρο το ID του χρήστη. Το Model θα βρίσκει τις προσφορές αυτά και μέσω της χρήσης της αντίστοιχης συνάρτησης στον Controller, αυτός θα τις εμφανίζει στην οθόνη.

6.3 Φυσικός σχεδιασμός

Φυσικός σχεδιασμός είναι η διαδικασία μετάφρασης του λογικού μοντέλου του υπό ανάπτυξη συστήματος, σε συγκεκριμένο σχεδιασμό των ενοτήτων και της αρχιτεκτονικής συνεργασίας τους στο νέο σύστημα. Παράγει τις προδιαγραφές υλικού, λογισμικού, φυσικών βάσεων δεδομένων και μέσω των εισόδου δεδομένων στο σύστημα και εξόδων από αυτό. Ο φυσικός σχεδιασμός δημιουργεί τις προδιαγραφές για το μετασχηματισμό του λογικού σχεδίου σε ένα λειτουργικό σύστημα ανθρώπων, υλικού και λογισμικού.

6.4 Αρχιτεκτονική λογισμικού

Για την υλοποίηση της εφαρμογής, εφαρμόστηκε το αρχιτεκτονικό μοντέλο **REST** για την παροχή υπηρεσιών οι οποίες φιλοξενούνται και εκτελούνται σε έναν διαδικτυακό διακομιστή. Οι ενόπτες λογισμικού που απαρτίζουν το σύστημα και συνεργάζονται για την υλοποίηση των στόχων του, όπως και οι αλληλεπιδράσεις μεταξύ τους απεικονίζονται στην Εικόνα 8.



Εικόνα 8. Η αρχιτεκτονική του λογισμικού

6.4.1 Διακομιστής διαδικτύου

Για την ανάπτυξη του λογισμικού του Snack Mart χρησιμοποιήθηκε το εργαλείο XAMPP. Το XAMPP είναι ένα ελεύθερο λογισμικό το οποίο περιέχει ένα εξυπηρετητή ιστοσελίδων, το οποίο μπορεί να εξυπηρετεί και δυναμικές ιστοσελίδες τεχνολογίας PHP. Είναι ανεξάρτητο πλατφόρμας και τρέχει σε Microsoft Windows, Linux, Solaris, MacOSX και χρησιμοποιείται ως πλατφόρμα για την σχεδίαση και ανάπτυξη ιστοσελίδων με την τεχνολογίες όπως PHP, JSP και Servlets. Επίσης οι σχεδιαστές του XAMPP, προόριζαν το λογισμικό ως εργαλείο ανάπτυξης και δοκιμής ιστοσελίδων τοπικά στον υπολογιστή χωρίς να είναι απαραίτητη η σύνδεση στο διαδίκτυο. Για να είναι δυνατή η χρήση του, πολλές σημαντικές λειτουργίες ασφάλειας έχουν απενεργοποιηθεί. Στην πράξη το XAMPP ορισμένες φορές χρησιμοποιείται και για την φιλοξενία ιστοσελίδων. Υπάρχει ειδικό εργαλείο το οποίο περιέχεται στο XAMPP για την προστασία με κωδικό των σημαντικών μερών. Το XAMPP υποστηρίζει την δημιουργία και διαχείριση βάσεων δεδομένων τύπου MySQL και SQLite.

Όταν το XAMPP εγκατασταθεί στον τοπικό υπολογιστή διαχειρίζεται τον localhost ως ένα απομακρυσμένο κόμβο, ο οποίος συνδέεται με το πρωτόκολλο μεταφοράς αρχείων FTP.

Με το XAMPP δημιουργούμε δυναμικές σελίδες, δηλαδή το περιεχόμενο τροποποιείται ανάλογα με την είσοδο του χρήστη ή κάποια άλλη πληροφορία όπως για παράδειγμα μια πληροφορία από τη βάση δεδομένων.

Επίσης, το XAMPP υποστηρίζει server side javascript μέσω Node.js και πρότυπες σελίδες HTML5, ενώ εκμεταλλεύεται τις τεχνολογίες CSS3 και HTML5 και βιβλιοθήκες Javascript (όπως η JQuery).

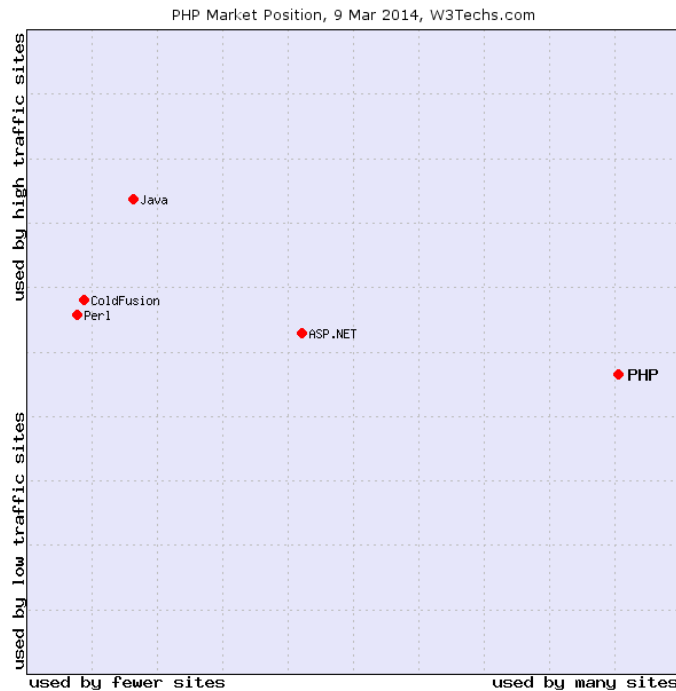
6.4.2 PHP

Το επίπεδο επιχειρησιακής λογικής για την παρούσα εργασία, υλοποιήθηκε μέσω ενός συνόλου σελίδων PHP, οι οποίες εκτελούνται στον application server του συνολικού συστήματος. Ο μηχανισμός της τεχνολογίας PHP προσφέρει την εκτέλεση του κώδικα στο διακομιστή εφαρμογών και τη δυναμική διαμόρφωση των HTML σελίδων που διατίθενται στον φυλλομετρητή του χρήστη της εφαρμογής. Το πλεονέκτημα αυτής της προσέγγισης είναι ότι η PHP είναι τεχνολογία διακομιστή, με αποτέλεσμα οι επεξεργαστικές διεργασίες να διεκπεραιώνονται στην πλευρά του διακομιστή. Επίσης η πλευρά του χρήστη δεν απαιτεί κανένα πρόσθετο λογισμικό (π.χ. Plugin ή runtime όπως η java) για να εκτελεστεί η εφαρμογή.

Επίσης η PHP είναι μια απλοποιημένη γλώσσα με πολύ σύντομο κύκλο εκμάθησης, γεγονός που διευκολύνει και επιταχύνει την ανάπτυξη της εφαρμογής. Επίσης προσφέρει ολοκληρωμένο σύνολο εντολών διεπαφής με διακομιστές βάσεων δεδομένων, γεγονός που διευκολύνει ιδιαίτερα στην περίπτωση του ταξιδιωτικού οδηγού, όπου θα υπάρχει έντονη συναλλαγή με το σύστημα διαχείρισης βάσεων δεδομένων.

Ακόμη, η PHP είναι ανεξάρτητη από το λειτουργικό σύστημα που χρησιμοποιείται στο διακομιστή της εγκατάστασης: λειτουργεί σε UNIX, Macs και Windows servers.

Η PHP δεν απαιτεί αυξημένους υπολογιστικούς πόρους του διακομιστή. Αυτό την καθιστά μια γρήγορη γλώσσα στην εκτέλεση του προγράμματος χωρίς να επιβραδύνει τις υπόλοιπες διεργασίες του διακομιστή. Επίσης η PHP είναι ευρύτατα αποδεκτή ως σταθερή γλώσσα η οποία χρησιμοποιείται κατά κόρον για ανάπτυξη εφαρμογών βάσεων δεδομένων. Η Εικόνα 9, αποτυπώνει τη θέση της PHP στο σύνολο των προγραμματιστικών περιβαλλόντων διακομιστή που διατίθενται σήμερα.



Εικόνα 9. Η θέση της PHP μεταξύ των γλωσσών ανάπτυξης εφαρμογών διακομιστή (server-side technologies) (Πηγή: W3Techs – Web Technology Surveys)

Ένα σημαντικό χαρακτηριστικό της PHP είναι ότι υλοποιεί πολλαπλά επίπεδα ασφάλειας για την προστασία των εφαρμογών από κακόβουλες επιθέσεις.

Επίσης η PHP προσφέρει σημαντικές δυνατότητες διασύνδεσης εφαρμογών. Χρησιμοποιεί ένα αρθρωτό σύστημα επεκτάσεων μέσω βιβλιοθηκών για το χειρισμό γραφικών, δεδομένων XML, κρυπτογραφίας, κ.κ.

Ιδιαίτερα σημαντικό σε σχέση με το έργο ανάπτυξης της πλατφόρμας προσφορών είναι το χαρακτηριστικό που δίνει η PHP για διασύνδεση με συστήματα βάσεων δεδομένων. Διατίθενται διεπαφές για συστήματα βάσεων δεδομένων MySQL, Microsoft SQL Server, Oracle, Informix, Postgres, κ.α. Επίσης η PHP είναι συμβατή και εκτελείται στην πλειονότητα των application servers (Microsoft Internet Information Server, Apache, THHTTPD, AOL Server, κ.α.).

6.4.3 JSON

Το JSON (JavaScript Object Notation) είναι ένα ελαφρύ πρότυπο ανταλλαγής δεδομένων. Είναι σχετικά απλό να αναλυθεί (parse) και να παραχθεί (generate) από εφαρμογές λογισμικού. Είναι βασισμένο πάνω σε ένα υποσύνολο της γλώσσας προγραμματισμού JavaScript, Standard ECMA-262.

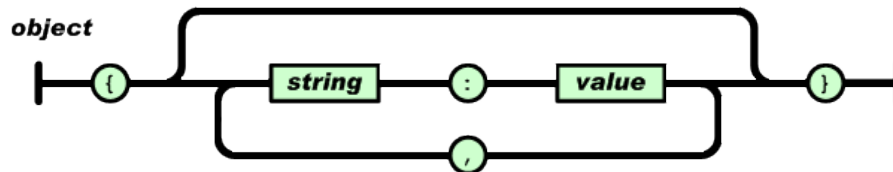
Το JSON είναι ένα πρότυπο κειμένου το οποίο είναι τελείως ανεξάρτητο από γλώσσες προγραμματισμού αλλά χρησιμοποιεί πρακτικές (conventions), οι οποίες είναι γνωστές στους προγραμματιστές της οικογένειας προγραμματισμού C, συμπεριλαμβανομένων των C, C++, C#, Java, JavaScript, Perl, Python, και πολλών άλλων. Αυτές οι ιδιότητες κάνουν το JSON μια ιδανική γλώσσα προγραμματισμού ανταλλαγής δεδομένων.

Το JSON είναι χτισμένο σε δύο δομές:

- Σε μια συλλογή από ζευγάρια ονομάτων/τιμών. Σε διάφορες γλώσσες προγραμματισμού, αυτό εκλαμβάνεται ως ένα αντικείμενο (object), πίνακας hash (hashtable), λίστα κλειδιών ή associative πίνακας.
- Σε μια ταξινομημένη λίστα τιμών. Στις περισσότερες γλώσσες προγραμματισμού, αυτό εκλαμβάνεται ως ένας πίνακας (array), δάνυσμα, λίστα, ή ακολουθία.

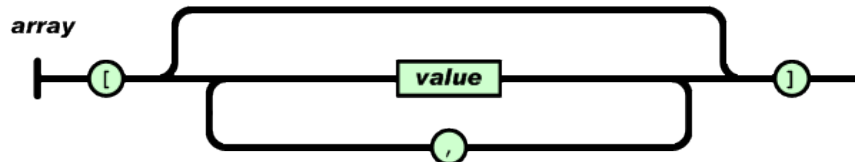
Η δομή μιας ροής (stream) JSON είναι:

- Ένα αντικείμενο (object) είναι ένα σύνολο από ζευγάρια ονομάτων / τιμών. Ένα αντικείμενο (object) ξεκινάει με { (αριστερό άγκιστρο) και τελειώνει με } (δεξιό άγκιστρο). Κάθε όνομα ακολουθείται από : (άνω - κάτω τελεία) και τα ζευγάρια ονόματος/τιμής χωρίζονται από , (κόμμα).



Εικόνα 10. Η μορφή ενός object σε JSON

- Ένας πίνακας (array) είναι μια συλλογή από τιμές σε σειρά. Ένας πίνακας (array) ξεκινάει με [(αριστερή αγκύλη) και τελειώνει με] (δεξιά αγκύλη). Οι τιμές χωρίζονται με , (κόμμα).



Εικόνα 11. Η μορφή ενός array σε JSON

Μια τιμή μπορεί να είναι string μέσα σε διπλά quotes, ή αριθμός (number), ή true ή false ή null, ή αντικείμενο (object) ή πίνακας (array).

Η πρόσβαση σε δεδομένα μορφής JSON παρέχεται από το JSON API. Η χρήση JSON παρέχει σημαντικά πλεονεκτήματα:

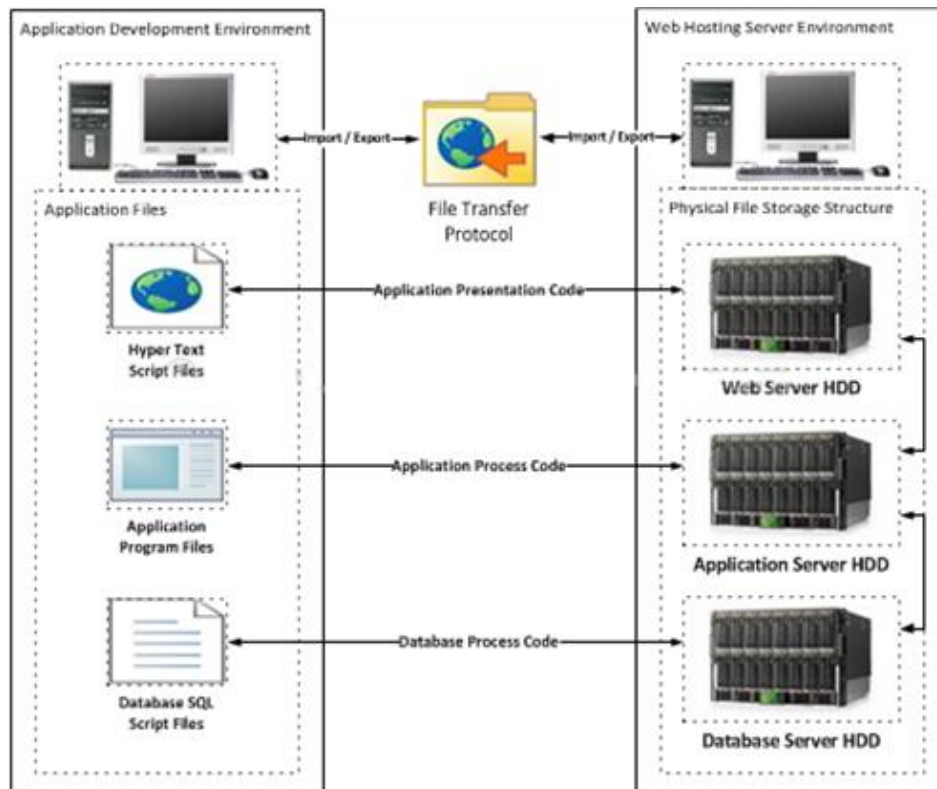
4. Το μέγεθος των αρχείων JSON είναι πολύ μικρότερο από τα αρχεία XML για τα ίδια δεδομένα.
5. Η ανάλυση (parsing) και η επεξεργασία ενός αρχείου XML είναι πολύ πιο περίπλοκη και απαιτεί περισσότερους υπολογιστικούς πόρους σε σχέση με το JSON.
6. Οι ρουτίνες ανάλυσης JSON είναι εγγενείς στη JavaScript ενώ αυτές για την ανάλυση XML χρησιμοποιούνται εξωτερικές βιβλιοθήκες. Αυτό έχει σαν αποτέλεσμα η ανάλυση δεδομένων JSON να είναι τουλάχιστον 5 φορές γρηγορότερη απαιτώντας το 1/3 των πόρων με την χρήση JSON απ' ό,τι των αντίστοιχων δεδομένων XML.

6.4.4 Επικοινωνία με GPS

Η επικοινωνία με το GPS της φορητής συσκευής γίνεται μέσω του geolocation API της Google. Το geolocation κατά την αρχικοποίησή του ρωτά το χρήστη της εφαρμογής να εκχωρήσει το δικαίωμα στην σελίδα να εντοπίσει την τοποθεσία που βρίσκεται με βάση την IP του και να εμφανίσει μια επισήμανση στην τρέχουσα θέση του.

7. Υλοποίηση Συστήματος

Για την υλοποίηση του συστήματός μας εφαρμόστηκε αρχιτεκτονική 3 επιπέδων (3-tier architecture). Το αρχιτεκτονικό μοντέλο περιγράφεται αναλυτικά στην εικόνα που ακολουθεί.



Εικόνα 12. Η αρχιτεκτονική τριών επιπέδων της πλατφόρμας προσφορών

Η αρχιτεκτονική αυτή προσφέρει ένα σύνολο πλεονεκτημάτων:

1. Η διαχείριση των δεδομένων είναι ανεξάρτητη από τον φυσικό τρόπο αποθήκευσης των δεδομένων (συστήματα βάσεων δεδομένων).
2. Ανεξαρτησία από το γραφικό περιβάλλον διεπαφής. Είναι σχετικά απλή η διαδικασία μετάβασης σε επιπλέον πλατφόρμες παρουσίασης των εφαρμογών (κινητά τηλέφωνα, tablets, σύγχρονες τηλεοπτικές συσκευές, κ.λπ.) Επιπρόσθετα η χρήση cascading stylesheets (CSS) επιτρέπει την εύκολη τροποποίηση της εμφάνισης των σελίδων του οδηγού, χωρίς να απαιτείται η παραμικρή αλλαγή στον κώδικα των σελίδων.
3. Ευελιξία στην τροποποίηση του επιπέδου παρουσίασης των πληροφοριών, ενώ τα επίπεδα επιχειρησιακής λογικής και διαχείρισης δεδομένων παραμένουν αμετάβλητα.
4. Είναι δυνατή η απασχόληση ομάδων εργασίας με τελείως διαφορετικές εξειδικεύσεις κατά τη διάρκεια της ανάπτυξης του συστήματος (για παράδειγμα ειδικοί στις βάσεις δεδομένων αναπτύσσουν το σχήμα της βάσης, ενώ ταυτόχρονα σχεδιαστές εφαρμογών διαδικτύου σχεδιάζουν και υλοποιούν τις σελίδες των διεπαφών).
5. Τα αποθηκευμένα δεδομένα είναι πιο ασφαλή, δεδομένου ότι η πρόσβαση σε αυτά γίνεται έμμεσα μέσω συνόδων HTTP (οι οποίες έχουν την ιδιότητα να είναι connectionless).
6. Κλιμάκωση. Κάθε επίπεδο της αρχιτεκτονικής μπορεί να αναβαθμίζεται ανεξάρτητα από τα υπόλοιπα ανάλογα με το μοντέλο χρήσης των εφαρμογών και τον αντίστοιχο φόρτο που προκαλείται.
7. Βελτιωμένες επιδόσεις. Το επίπεδο παρουσίασης (presentation layer) προσφέρει μηχανισμούς caching για την ελαχιστοποίηση χρήσης του δικτύου για μεταφορά δεδομένων που χρησιμοποιούνται επαναλαμβανόμενα. Επίσης παρέχεται η δυνατότητα ανεξάρτητης αναβάθμισης των άλλων δύο επιπέδων της αρχιτεκτονικής με περισσότερους διακομιστές και εφαρμογή πολιτικών κατανομής υπολογιστικού φορτίου (loadbalancing).
8. Εύκολη εγκατάσταση των εφαρμογών στις εγκαταστάσεις του παρόχου.
9. Αξιοποίηση των πολιτικών ασφάλειας που προσφέρει ο πάροχος των υπηρεσιών φιλοξενίας των εφαρμογών.

Στη συνέχεια περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της πλατφόρμας προσφορών.

7.1 Βάση δεδομένων

Το επίπεδο δεδομένων περιλαμβάνει το διακομιστή βάσεων δεδομένων, όπου αποθηκεύονται οι πληροφορίες των προσφορών, των χρηστών και του ηλεκτρονικού καταστήτου καθώς και τις συναρτήσεις υποβολής επερωτήσεων προς τη βάση δεδομένων.

Τα δεδομένα αποθηκεύονται και διατηρούνται στους διακομιστές του επιπέδου δεδομένων ανεξάρτητα από τους διακομιστές εφαρμογών ή τις εφαρμογές υλοποίησης της επιχειρησιακής λογικής. Το επίπεδο δεδομένων τυπικά αποθηκεύει τα δεδομένα της εφαρμογής με ασφάλεια και διεκπεραιώνει τις δοσοληψίες αναζήτησης και διάθεσης των πληροφοριών στα άλλα επίπεδα του μοντέλου τριών επιπέδων που χρησιμοποιείται.

Στην περίπτωση της πλατφόρμας προσφορών, το επίπεδο δεδομένων υλοποιήθηκε στο σύστημα MySQL RDBMS. Η διεπαφή της εφαρμογής με το σχεσιακό σύστημα γίνεται μέσω κώδικα php και το mysqliAPI το οποίο βρίσκεται ενσωματωμένο στη γλώσσα PHP.

7.2 Συναρτήσεις διεπαφής με βάση δεδομένων

Το λογισμικό που υλοποιεί τον Controller υλοποιείται στο αρχείο api.php. Συνοπτικά οι συναρτήσεις αυτές είναι:

set_new_product	Δέχεται το POST array των δεδομένων της προσφοράς και αποθηκεύει τα δεδομένα στη βάση δεδομένων. Επίσης μεταφορτώνει τη φωτογραφία της προσφοράς στο φάκελο images του web server.
update_product	Δέχεται POST array με τα ενημερωμένα δεδομένα της προσφοράς και ενημερώνει την εγγραφή με το συγκεκριμένο id προσφοράς.
get_products	Η συνάρτηση get_products επιστρέφει ένα array των προσφορών ταξινομημένων κατά όνομα προσφοράς. Επιστρέφονται μόνο οι προσφορές που βρίσκονται σε κατάσταση «ΔΗΜΟΣΙΕΥΜΕΝΗ».
get_product	Επιστρέφει τα στοιχεία μιας προσφοράς βάσει του id της στη βάση δεδομένων.
get_allproducts	Η συνάρτηση get_allproducts επιστρέφει ένα array των προσφορών ταξινομημένων κατά όνομα προσφοράς. Επιστρέφονται όλες οι προσφορές, ανεξάρτητα με την κατάσταση δημοσίευσης που βρίσκονται.
in_basket	Συνάρτηση η οποία καταχωρεί μία προσφορά στο καλάθι αγορών του χρήστη που έχει συνδεθεί.
remove_from_basket	Διαγράφει μία γραμμή από το καλάθι παραγγελίας του συνδεδεμένου χρήστη.
empty_basket	Διαγράφει όλα τα περιεχόμενα των παραγγελιών του συνδεδεμένου χρήστη.
get_basket	Η συνάρτηση get_basket δέχεται σαν παράμετρο το ID ενός πελάτη και επιστρέφει array των προσφορών που έχει επιλέξει για αγορά.
do_login	Δέχεται δεδομένα του POST array και επικυρώνει το χρήστη στη βάση δεδομένων (επιβεβαιώνει ότι υπάρχει το ζευγάρι username και password καταχωρημένο στη βάση).
set_new_user	Δέχεται POST array των δεδομένων ενός χρήστη ο οποίος κάνει εγγραφή στην πλατφόρμα και δημιουργεί μια νέα εγγραφή στον πίνακα users
update_user	Δέχεται POST array των ενημερωμένων δεδομένων ενός χρήστη

	ο οποίος επεξεργάζεται το προφίλ του στην πλατφόρμα και ενημερώνει τα στοιχεία της εγγραφής με το αντίστοιχο id στον πίνακα users.
get_comments	Επιστρέφει τα σχόλια που έχουν γίνει για μια προσφορά με συγκεκριμένο id προσφοράς.
get_user	Επιστρέφει τα στοιχεία χρήστη με συγκεκριμένο id.
get_users	Επιστρέφει τα στοιχεία όλων των χρηστών του συστήματος ταξινομημένα ανά ρόλο και όνομα χρήστη.

7.3 Επίπεδο επιχειρησιακής λογικής (Controller)

Το επίπεδο επιχειρησιακής λογικής υλοποιήθηκε μέσω ενός συνόλου συναρτήσεων, οι οποίες εκτελούνται στο διακομιστή. Ο κώδικας που υλοποιεί τον Controller βρίσκεται στο αρχείο api.php και κατά κύριο λόγο δέχεται τα αποτελέσματα του Model, τα κωδικοποιεί σε μορφότυπο JSON, ενεργοποιεί την κατάλληλη View και διαβιβάζει σ' αυτή τα δεδομένα για εμφάνιση.

Ο Controller υλοποιείται από τον κώδικα που ακολουθεί και ο οποίος ανάλογα με την ενέργεια που κάνει ο χρήστης και η οποία διαβιβάζεται στο api.php μέσω του URI (GETmethod):

Κώδικας 1. Ενεργοποίηση συνάρτησης του API μετά από ανάλυση των παραμέτρων της GET method κλήσης της από τον φυλλομετρητή.

```
// Η παρακάτω πρόταση ελέγχει το URI (μέσω του arrayGET),
// ώστε να προσδιοριστεί η εντολή που θα εκτελεστεί.
// Ελέγχει το arraypossible_url και αν η εντολή ανήκει σε αυτό, εκτελείται
// με τις κατάλληλες παραμέτρους
if (isset($_GET["action"]) && in_array($_GET["action"], $possible_url))
{
switch ($_GET["action"])
{
// Δημιουργία νέας προσφοράς
case "set_new_product":
$value = set_new_product();
header("Location: listproducts.php");
break;
// Σύνδεση και δημιουργία session cookie
case "do_login":
$value = do_login();
echo "RESULT: ".$value;
break;
// Αποσύνδεση και καταστροφή session cookie
case "do_logoff":
$value = do_logoff();
// Εμφάνιση της αρχικής σελίδας για εισαγωγή στοιχείων σύνδεσης
header("Location: index.php");
break;
// Επιστροφή array όλων των προσφορών
case "get_products":
$value = get_products();
break;
```

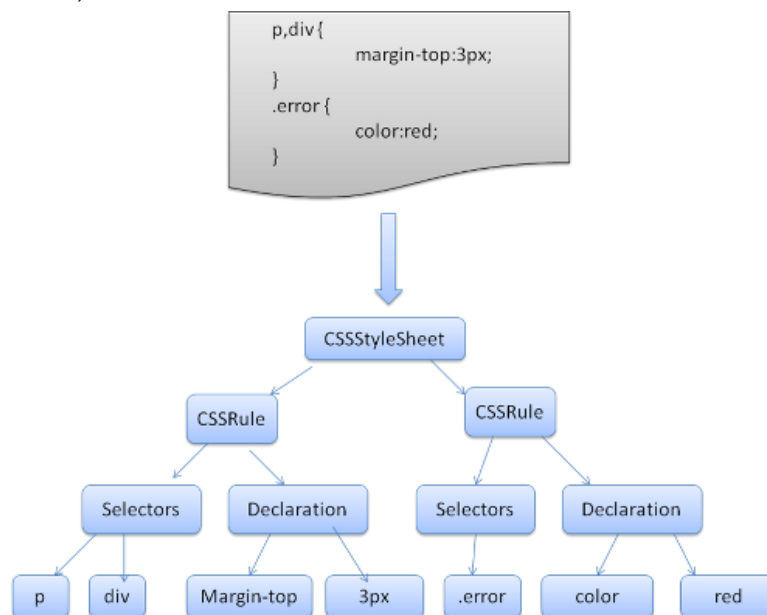
```
// Εισαγωγή επιλεγμένης προσφοράς στο καλάθι αγορών
case "in_basket":
    $value = in_basket();
    // Εμφάνιση καλάθιού
header("Location: listbasket.php");
break;
// Επιστροφή array με τα περιεχόμενα του καλάθιού για το χρήση τη με κωδικό uid
case "get_basket":
if (isset($_GET["uid"]))
    $value = get_basket($_GET["uid"]);
break;
}
```

Η τεχνολογία PHP εκτελεί τον κώδικα στο διακομιστή εφαρμογών και διαμορφώνει δυναμικά τις HTML σελίδες που διατίθενται στον φυλλομετρητή του χρήστη της εφαρμογής. Το πλεονέκτημα αυτής της προσέγγισης είναι ότι η PHP είναι τεχνολογία διακομιστή, με αποτέλεσμα οι επεξεργαστικές διεργασίες να διεκπεραιώνονται στην πλευρά του διακομιστή. Επίσης η πλευρά του χρήστη δεν απαιτεί κανένα πρόσθετο λογισμικό (π.χ. plugin ή runtime (όπως η java)) για να εκτελεστεί η εφαρμογή.

7.4 Επίπεδο παρουσίασης (View)

Για την παρουσίαση των τελικών HTML σελίδων που δημιουργούνται μέσω του κώδικα PHP χρησιμοποιείται ο Apache HTTP server.

Η εφαρμογή χρησιμοποιεί HTML5 για την περιγραφή του περιεχομένου, ενώ για τη μορφοποίηση των σελίδων που διαβιβάζονται στο φυλλομετρητή για εμφάνιση χρησιμοποιείται τεχνολογία Cascading Stylesheets. Χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή, για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός δικτυακού τόπου. Η CSS είναι μια γλώσσα προορισμένη να προσδιορίζει τα χαρακτηριστικά εμφάνισης μιας ιστοσελίδας, διαμορφώνοντας χαρακτηριστικά όπως γραμματοσειρές, χρώματα, στοίχιση, τοποθέτηση στοιχείων στη σελίδα, κλπ.



Εικόνα 13. Επεξεργασία tags της HTML μέσω κανόνων CSS

7.4.1 Ενεργοποίηση και χρήση Geolocation API

Για την ενεργοποίηση του geolocation απαιτείται μια εμφωλευμένη εντολή του API (Κώδικας 1) στη ρουτίνα αρχικοποίησης του χάρτη (ρουτίνα `map()`), η οποία επιστρέφει τις (x,y) συντεταγμένες του χρήστη και εμφανίζει (προαιρετικά) έναν `marker` εκεί.

Κώδικας 1

```
google.maps.event.addListenerOnce(map, 'tilesloaded', function() {
watchID = navigator.geolocation.watchPosition(gotPosition, null, {maximumAge: Infinity,
timeout: 60000, enableHighAccuracy: true});
```

Για να γίνει κλήση στο geolocationAPI, χρησιμοποιείται η μέθοδος `watchPosition`. Η παράμετρος `enableHighAccuracy` αρχικοποιείται σε τιμή `true`. Όταν η τιμή της είναι `true`, για το γεωγραφικό εντοπισμό χρησιμοποιείται όχι μόνο το GPS της φορητής συσκευής, αλλά και τα ασύρματα δίκτυα που υπάρχουν στην περιοχή του χρήστη και με τα οποία έρχεται αυτός σε επαφή.

Για να ενεργοποιηθεί η μέθοδος `watchPosition`, πρέπει να δημιουργηθεί ένας listener με όνομα "tilesloaded", ο οποίος αντιλαμβάνεται πότε έχει ολοκληρωθεί η ασύγχρονη φόρτωση του χάρτη. Όταν η φόρτωση του χάρτη ολοκληρωθεί, καλείται η μέθοδος `watchPosition` (Κώδικας 1).

Ένθετα στοιχεία χαρτών

Στους χάρτες της εφαρμογής υλοποιούνται δύο τύποι ένθετων στοιχείων (overlays):

- **Markers.** Χρησιμοποιούνται για την απεικόνιση της θέσης του χρήστη και της θέσης των προσφορών στο χάρτη.
- **Shape.** Χρησιμοποιείται για τη σχεδίαση του κύκλου προσδιορισμού της περιοχής αναζήτησης προσφορών.

Η δημιουργία του `marker` της συσκευής και του κύκλου με ακτίνα εμφάνισης προσφορών γίνεται μέσω της ρουτίνας "gotPosition" (Κώδικας **Error! Reference source not found.**).

Κώδικας 3

```
// Εύρεση συντεταγμένων της συσκευής χρήστη και κεντραρίσματος του χάρτη
// στις συντεταγμένες αυτές
// Δημιουργία marker συσκευής και κύκλου
// προσδιορισμού της περιοχής αναζήτησης προσφορών

functiongotPosition(position) {
map.setCenter(new google.maps.LatLng(position.coords.latitude, position.coords.longitude));
var point = new google.maps.LatLng(position.coords.latitude, position.coords.longitude);

if (!marker) {
// Δημιουργία marker της συσκευής
marker = new google.maps.Marker({
position: point,
map: map,
title: "I am here...",
icon: 'img/personal.png' // αλλαγή του εικονιδίου
});

// Κύκλος που ορίζεται από την ακτίνα αναζήτησης
varCircleRadius = parseInt (document.getElementById("maxDistance").value);
var circle = new google.maps.Circle({
```

```

map: map,
radius: CircleRadius,
fillColor: '#AA0000'
});
circle.bindTo('center', marker, 'position');
Circles.push (circle);

} else {
// Μετακίνηση και τοποθέτηση του marker στην τρέχουσα θέση
marker.setPosition(point);
}
}

```

Για τη δημιουργία των ένθετων στοιχείων στο χάρτη χρησιμοποιούνται οι μέθοδοι:

```
google.maps.Marker()
```

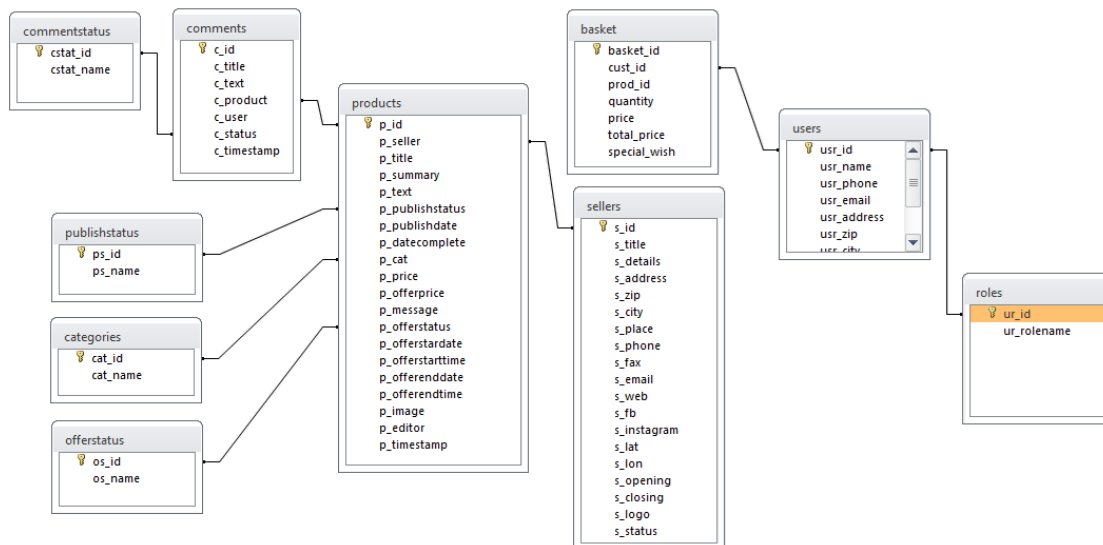
και

```
google.maps.Circle()
```

Οι μέθοδοι αυτές εμφανίζουν τα ένθετα στοιχεία στο χάρτη της περιοχής που βρίσκεται ο χρήστης (Κώδικας **Error! Reference source not found.**).

7.5 Μοντέλο οντοτήτων – συσχετίσεων

Η ανάλυση της πλατφόρμας προσφορών κατέληξε σε ένα σύνολο οντοτήτων, οι οποίες οργανώνονται, όπως στο σχεσιακό μοντέλο του Διαγράμματος 2, που φαίνεται πιο κάτω:



Διάγραμμα 2. Το σχεσιακό μοντέλο του συστήματος και οι συνδέσεις των πινάκων

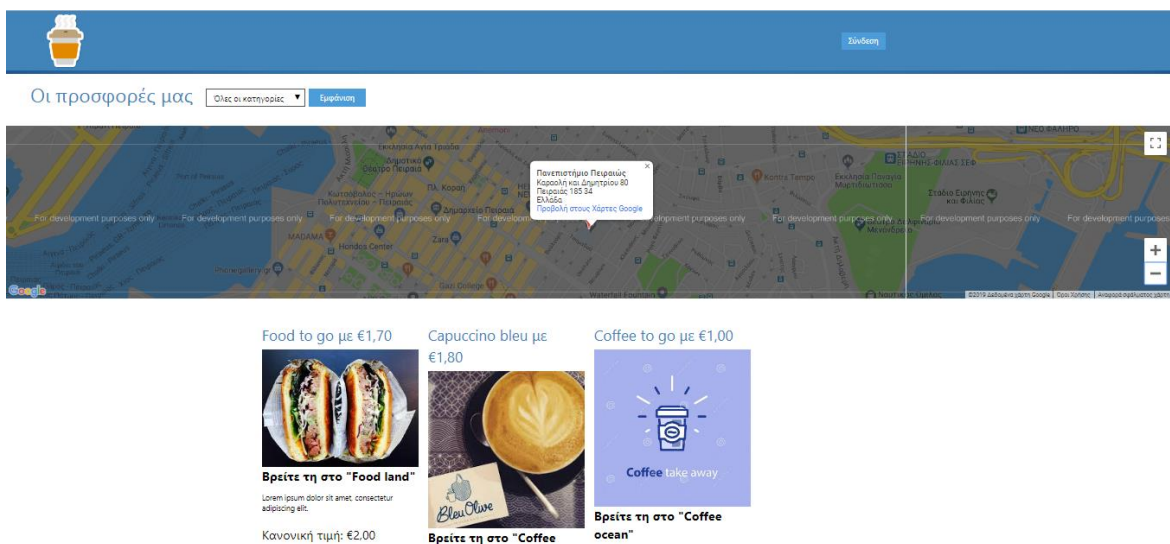
7.6 Δομή της εφαρμογής

Πρόκειται για την ενότητα η οποία αναλαμβάνει την εμφάνιση των ιστοσελίδων στο φυλλομετρητή. Για την υλοποίηση του view, δημιουργήθηκαν ρηρ σελίδες, οι οποίες δέχονται δεδομένα από το model σε μορφή JSON array, τα αποκωδικοποιούν και τα μορφοποιούν για εμφάνιση στην οθόνη. Οι σελίδες του view είναι οι εξής:

1. basketview.php
2. error.php
3. header.php
4. index.php

5. messages.php
6. newcomment.php
7. payment.php
8. productedit.php
9. productinfo.php
10. productlist.php
11. productnew.php
12. success.php
13. useredit.php
14. userlogin.php
15. usernew.php
16. userslist.php

Πιο κάτω φαίνεται αναλυτικά μέσα από στιγμιότυπα, η λειτουργία της εφαρμογής.



Εικόνα 14. Στιγμιότυπο από την αρχική εικόνα της εφαρμογής

Αρχικά, όταν συνδέεται ο χρήστης στην εφαρμογή εμφανίζεται η παραπάνω οθόνη, με όλες τις διαθέσιμες προσφορές καθώς και έναν χάρτη που έχει ήδη εντοπίσει τη θέση του χρήστη. Για να μπορέσει να κάνει οποιαδήποτε ενέργεια, είτε να παραγγείλει είτε να συνδεθεί ως διαχειριστής, είτε ως κατάστημα που θέλει να προσθέσει κάποια νέα προσφορά, πρέπει να πατήσει πάνω δεξιά «Σύνδεση».

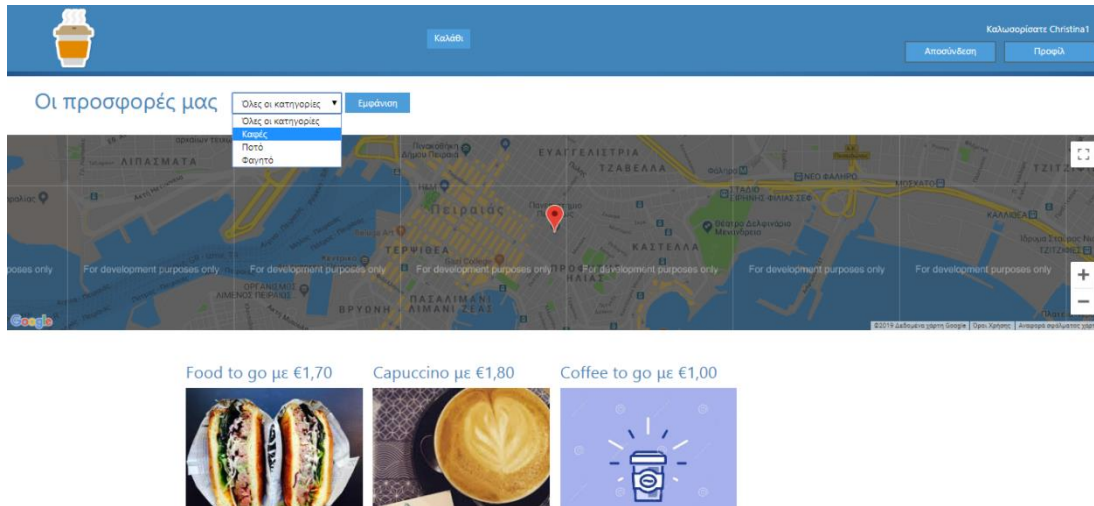


Εικόνα 15. Στιγμιότυπο από την φόρμα σύνδεσης στην εφαρμογή

Στην περίπτωση που θέλει να εγγραφεί ένας νέος χρήστης, θα πρέπει να επιλέξει την επιλογή, «Πιέστε εδώ εάν δεν έχετε λογαριασμό» και στη συνέχεια θα εμφανιστεί μια φόρμα εισαγωγής στοιχείων όπως φαίνεται και στην παρακάτω εικόνα.

Εικόνα 16. Στιγμιότυπο από την φόρμα εισαγωγής στοιχείων νέου χρήστη

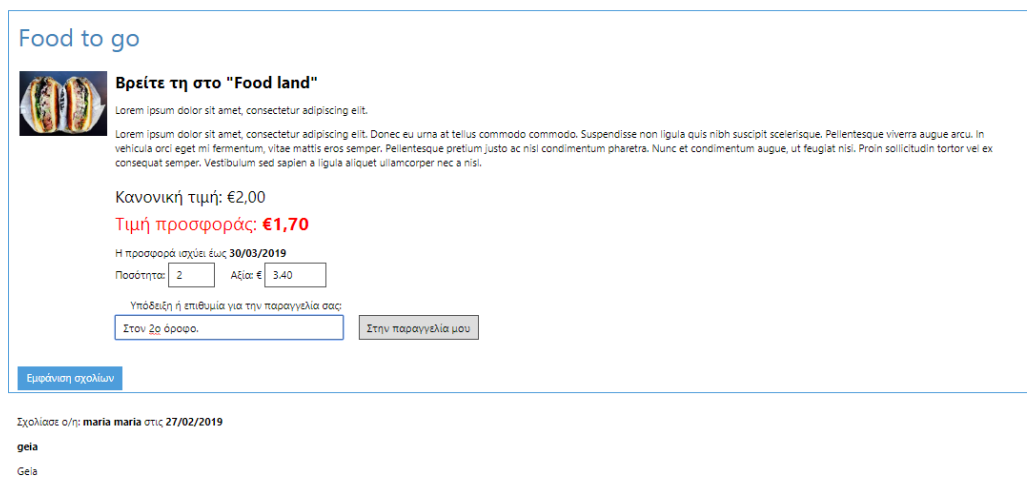
Έστω τώρα ότι έχουμε εγγραφεί και κάνουμε σύνδεση ως απλός χρήστης. Όπως φαίνεται στην πιο κάτω εικόνα μπορούμε να επιλέξουμε από το drop-down την κατηγορία των προϊόντων που μας ενδιαφέρει ώστε να εμφανιστούν είτε μόνο οι προσφορές για καφέ, είτε μόνο οι προσφορές για φαγητό.



Εικόνα 17. Στιγμιότυπο από την αρχική σελίδα του επισκέπτη

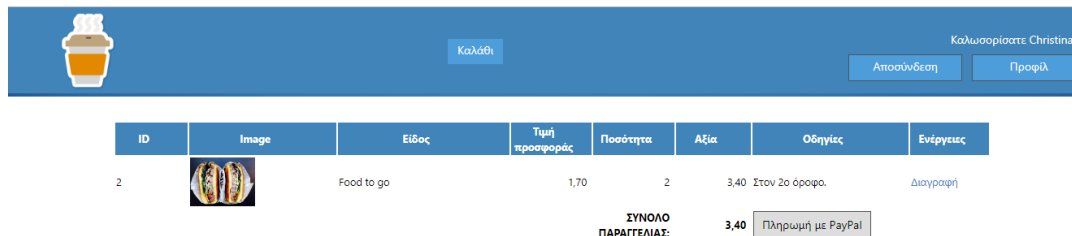
Ο χρήστης μπορεί να επιλέξει το προϊόν που τον ενδιαφέρει πατώντας κάτω από την προσφορά την επιλογή «Δες και παρήγγειλε». Στη σελίδα που θα του εμφανιστεί μπορεί να επιλέξει την ποσότητα που επιθυμεί ή να κάνει κάποια υπόδειξη σχετικά με την παραγγελία του και να καταχωρήσει την παραγγελία στο καλάθι του.

Η προσφορά αναλυτικά



Εικόνα 18. Στιγμιότυπο από την προσφορά που έχει επιλέξει ο χρήστης

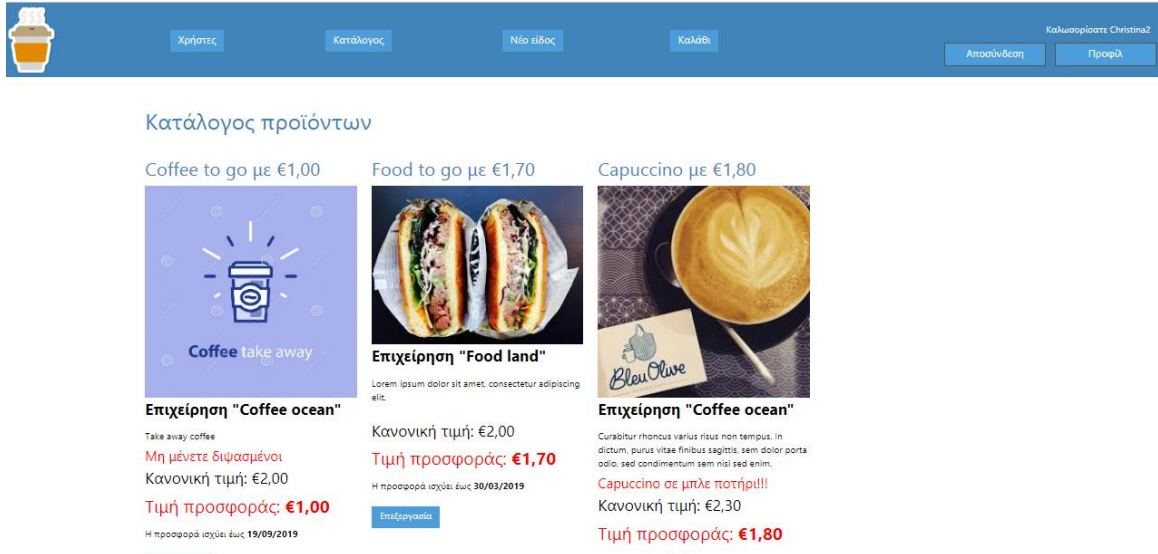
Πιο κάτω φαίνεται το καλάθι του χρήστη:



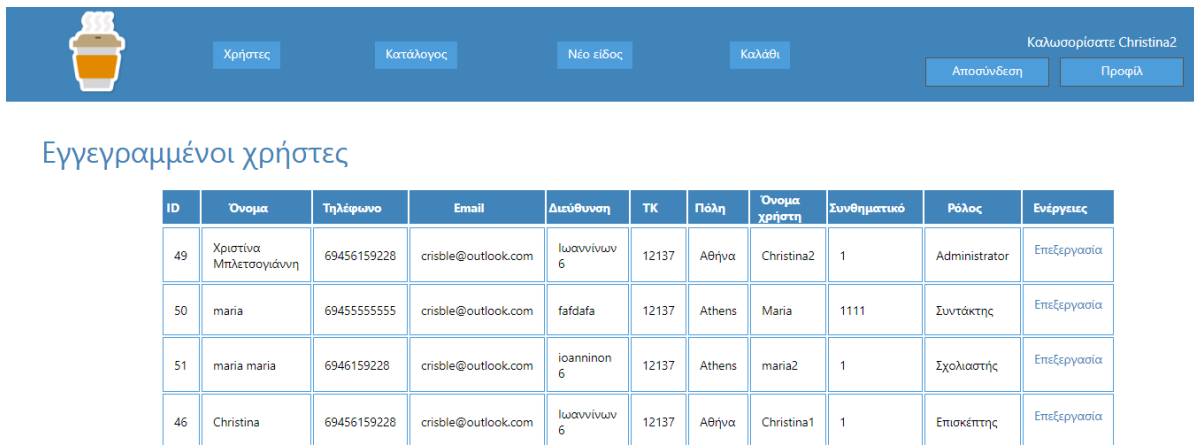
Εικόνα 19. Στιγμιότυπο από το καλάθι του χρήστη

Έστω ότι ο χρήστης που συνδέεται τώρα είναι ο administrator της σελίδας. Η αρχική του οθόνη θα είναι διαφορετική από αυτή του χρήστη. Μέσα από το διαχειριστικό του, του δίνεται η

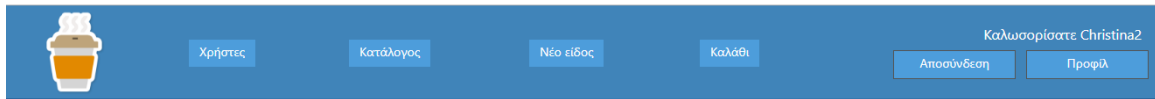
δυνατότητα να επεξεργαστεί τους εγγεγραμμένους χρήστες, να επεξεργαστεί τις δημοσιεύσεις των διαθέσιμων προϊόντων, καθώς και να εισάγει κάποιο νέο προϊόν.



Εικόνα 20. Στιγμιότυπο από την αρχική σελίδα του διαχειριστή



Εικόνα 21. Στιγμιότυπο από τους εγγεγραμμένους χρήστες όπως φαίνονται από το διαχειριστικό του administrator



Επεξεργασία στοιχείων είδους

* ID:

* Επιχείρηση:

* Κατηγορία:

* Ονομασία:

* Συνοπτική περιγραφή:

* Περιγραφή είδους:

* Αρχική τιμή είδους:

Εικόνα 22. Στιγμιότυπο από την επεξεργασία των ήδη διαθέσιμων προϊόντων από το διαχειριστικό του administrator



Νέο είδος

* Επιχείρηση:

* Κατηγορία:

* Ονομασία:

* Συνοπτική περιγραφή:

* Περιγραφή είδους:

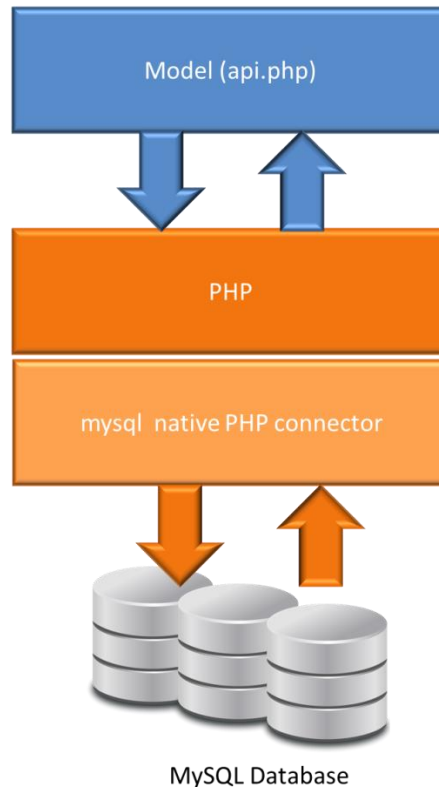
* Αρχική τιμή είδους:

Τιμή προσφοράς:

Εικόνα 23. Στιγμιότυπο από την εισαγωγή νέου προϊόντος από το διαχειριστικό του administrator

Επισήμανση Η φωτογραφία της προσφοράς αποθηκεύεται στο φάκελο images στον web server της πλατφόρμας προσφορών.

Η επεξεργασία των δεδομένων, η τήρηση των κανόνων ακεραιότητας και οι πολιτικές ασφάλειας γίνονται από το διακομιστή MySQL (Διάγραμμα 3).



Διάγραμμα 3. Αρχιτεκτονική επικοινωνίας της ενότητας διαχείρισης πληροφοριών με το σύστημα διαχείρισης βάσεων δεδομένων MySQL

8. Συμπεράσματα – Μελλοντικές επεκτάσεις

Στο συγκεκριμένο κεφάλαιο θα παρουσιαστούν τα συμπεράσματα που προέκυψαν από την υλοποίηση της παρούσας μεταπτυχιακής εργασίας. Θα γίνει αναφορά για την απόκτηση της τεχνογνωσίας που αποκτήσαμε και θα προταθούν μελλοντικές επεκτάσεις για την υπάρχουσα εφαρμογή.

8.1 Συμπεράσματα

Με τη ραγδαία ανάπτυξη της τεχνολογίας τα τελευταία χρόνια, ο χρήστης πλέον έχει εξοικειωθεί με τη χρήση της και έχει αποκτήσει περισσότερες απαιτήσεις ως προς τις εφαρμογές που του παρέχονται είτε διαδικτυακά είτε μέσω του κινητού τηλεφώνου του. Θέλει να του δίνεται η δυνατότητα να έχει πρόσβαση στην πληροφορία που επιθυμεί, όποτε την επιθυμεί. Επιπλέον, οι ρυθμοί της καθημερινότητας έχουν αυξήσει την ανάγκη του να μπορεί να κάνει αγορές τη στιγμή που θέλει, χωρίς να υπάρχει προϋπόθεση της φυσικής του παρουσίας. Αυτή την ανάγκη των χρηστών, έχουν καλύψει τα ηλεκτρονικά καταστήματα. Πλέον, οποιοδήποτε προϊόν ή υπηρεσία μπορεί να αγορασθεί από τον χρήστη εύκολα και γρήγορα μέσω του ηλεκτρονικού του υπολογιστή ή της φορητής του συσκευής.

Τα ηλεκτρονικά καταστήματα που υπάρχουν σήμερα στο διαδίκτυο είναι πολλά και παρέχουν ευκολία και ασφάλεια πληρωμών. Ο χρήστης μπορεί να αγοράσει από είδη ένδυσης, μέχρι πακέτα διακοπών και διάφορες υπηρεσίες.

Στην παρούσα μεταπτυχιακή εργασία σχεδιάστηκε μια τέτοια εφαρμογή. Πρόκειται για ένα ηλεκτρονικό κατάστημα στην πραγματικότητα, μέσω του οποίου, παρέχονται στον χρήστη διάφορες προσφορές σχετικά με ποτό/καφέ ή φαγητό. Μπορεί να επιλέξει κάποια από αυτές τις προσφορές που τον ενδιαφέρουν, έχοντας ακόμη και ως κριτήριο καταστήματα που βρίσκονται κοντά στην τοποθεσία του.

Στο πλαίσιο της ανάπτυξης της πλατφόρμας προσφορών, μελετήθηκαν υφιστάμενες υλοποιήσεις και έγινε προσπάθεια να σχεδιαστεί μια διαδικτυακή υπηρεσία, η οποία προσφέρει ένα απλό περιβάλλον χρήσης.

Επίσης, σημαντική προσπάθεια δόθηκε στον αρχιτεκτονικό της σχεδιασμό, ώστε να ακολουθούνται ορθές πρακτικές του μοντέλου MVC.

Η ανάπτυξη της διαδικτυακής υπηρεσίας δημοσίευσης προσφορών φαγητού και ποτού, έδωσε την ευκαιρία εξοικείωσης με τη γλώσσα PHP και την ενσωμάτωσή της στις σελίδες της εφαρμογής.

Ακόμη, η εντατική χρήση τεχνολογίας Cascading Stylesheets (CSS) έδωσε τη δυνατότητα πειραματισμού με πολλαπλές εκδοχές εμφάνισης των σελίδων, έως ότου προκύψει η τελική τους μορφή. Η ευελιξία δίνει τη δυνατότητα της σχετικά εύκολης αναπροσαρμογής της εμφάνισης της υπηρεσίας ανάλογα με τη συσκευή στην οποία χρησιμοποιείται.

Επιπλέον, αποκτήθηκε σημαντική εμπειρία στη διαχείριση και ανάπτυξη εφαρμογών με το σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων MySQL.

8.2 Μελλοντικές Επεκτάσεις

Οι δυνατότητες της εφαρμογής που υλοποιήσαμε μπορούν να επεκταθούν μελλοντικά προς αρκετές κατευθύνσεις. Μερικές ιδέες για μελλοντική επέκταση της εφαρμογής παρουσιάζονται παρακάτω:

- Βελτιστοποίηση του γραφικού περιβάλλοντος (UI).
- Δυνατότητα πληρωμής μέσω πιστωτικών/χρεωστικών καρτών.
- Δυνατότητα πληκτρολόγησης (με auto-complete) της περιοχής που ενδιαφέρει τον χρήστη, ώστε να του παρέχονται στοχευμένες προσφορές.
- Εγγραφή του χρήστη στο σύστημα μέσω κάποιο email για παράδειγμα Gmail ή με κάποιο λογαριασμό του στο Facebook/Twitter κλπ.
- Να δίνεται η δυνατότητα στο χρήστη μέσω του χάρτη, τοποθετώντας σε αυτόν μια πινέζα, να του εμφανιστεί η διαδρομή που πρέπει να ακολουθήσει ώστε να φτάσει στο κατάστημα που έχει επιλέξει.
- Η εφαρμογή αναπτύχθηκε με τέτοιο τρόπο, ώστε να είναι responsive, σαν μελλοντική όμως προοπτική θα μπορούσε να σχεδιαστεί και μέσω Android Studio και να μπορεί να την κατεβάσει ο χρήστης σαν εφαρμογή στο κινητό του.
- Μια ακόμη σημαντική βελτίωση της εφαρμογής θα ήταν να δίνεται η δυνατότητα στο χρήστη να προτείνει την εφαρμογή μέσω social media και με αυτό τον τρόπο να κερδίζει έκπτωση στην επόμενη αγορά του, κάθε φορά που κάποιος φίλος του κάνει μια αγορά από την εφαρμογή.
- Επιπλέον, να έρχεται στο κινητό του ένα μήνυμα με τον κωδικό αγοράς του τον οποίο θα μπορεί να επιδεικνύει στο κατάστημα από το οποίο αγοράστηκε η προσφορά.
- Ακόμη, μια ενδιαφέρουσα βελτίωση της εμπειρίας του χρήστη μέσω της εφαρμογής θα ήταν, η ύπαρξη μιας επιλογής (ας την ονομάσουμε για παράδειγμα «Wish») σε κάθε προσφορά, όπου ο χρήστης θα μπορεί να την επιλέγει αν τον ενδιαφέρει η προσφορά, να συμπληρώνει το πλήθος της ποσότητας της προσφοράς καθώς και την ημερομηνία που θα ήθελε να την εξαργυρώσει. Αυτή να καταχωρείται στο σύστημα και κάθε μήνα οι τρεις πρώτοι χρήστες που θα έχουν κάνει τις περισσότερες αγορές μέσω της εφαρμογής, θα κερδίζουν (δωρεάν) τη «Wish» που είχαν καταχωρήσει.
- Επιπλέον, θα μπορούσε κάθε μέρα να γίνεται μια «κλήρωση» από το σύστημα και ο χρήστης που θα έχει επιλεγεί τυχαία, να κερδίζει δωρεάν την αγορά που θα έχει κάνει.

Βιβλιογραφία

- Adobe. (2011, Νοέμβριος). *FLASH TO FOCUS ON PC BROWSING AND MOBILE APPS; ADOBE TO MORE AGGRESSIVELY CONTRIBUTE TO HTML5*. Ανάκτηση Αύγουστος 2018, από Adobe News: <https://blogs.adobe.com/conversations/2011/11/flash-focus.html>
- Alonso, G., Casati, F., Kuno, i., & Machiraju, V. (2004). Web Services. Στο G. Alonso, F. Casati, i. Kuno, & V. Machiraju, *Web Services. Data-Centric Systems and Applications* (σσ. 123-149). Berlin Heidelberg: Springer-Verlag.
- Beyer, H., & Holtzblatt, K. (1998). *Contextual Design: Defining Customer-Centered Systems*. Morgan Kaufmann.
- Buxton, B. (2007). *Sketching User Experiences: Getting the Design Right and the Right Design*. Morgan Kaufmann.
- Cognizant. (2014, Ιούλιος). *Hybrid Mobile Application Analysis and Guidelines*. Ανάκτηση Ιούλιος 2018, από Cognizant Whitepapers: <https://www.cognizant.com/whitepapers/Hybrid-Mobile-Application-Analysis-and-Guidelines-codex888.pdf>
- David, G., & Mary, S. (1994). *An Introduction to Software Architecture*. Pittsburgh, PA: Carnegie Mellon University.
- Deloitte. (2017). *Deloitte seventh annual Mobile Consumer Survey*. London: Deloitte.
- Deloitte. (2011). *Revolutions 2010 Survey: Mobile Apps and e-readers Transform Consumer Behavior*.
- Fielding, R., & Taylor, R. (2002). Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology* , 2 (2), σσ. 115–150.
- Foursquare . (2013, Αύγουστος). *A smarter Foursquare, so you don't miss a thing*. Ανάκτηση Αύγουστος 2018, από Foursquare blog: <http://blog.foursquare.com/2013/08/29/a-smarter-foursquare-so-you-dont-miss-a-thing/>
- Freeman, E., & Robson, E. (2011). *Head First HTML5 Programming building web apps with javascript*. O'Reilly.
- Gualtieri, M. (2012). *Design Mobile Apps From The Outside In*. cambridge, mA: Forrester Research.
- Hamerman, P. (2011). *Mobile Applications Will Empower Enterprise Business Processes*. Forrester Research.
- Huang, H., & Gao, S. (2018). Location-Based Services. Στο J. P. Wilson, *The Geographic Information Science & Technology Body of Knowledge (1st Quarter 2018 Edition)*. Ithaca, NY.
- Ilyas, M., & Ahson, S. (2006). *Smartphones Research Report*. International Engineering Consortium.
- Kerpen, D. (2011). *Likeable Social Media: How to Delight Your Customers, Create an Irresistible Brand, and Be Generally Amazing on Facebook (And Other Social Networks)*. McGraw Hill.

Lionbridge Whitepaper. (2012). *Mobile Web Apps vs Mobile Native Apps: How to Make the Right Choice*.

Martin, L. A., & Michael, T. F. (2017). *Scalability Rules: Principles for Scaling Web Sites, 2nd Edition*. Addison-Wesley Professional.

Microsoft. (2018, Αύγουστος). *Common web application architectures*. Ανάκτηση Σεπτέμβριος 2018, από Microsoft: <https://docs.microsoft.com/en-us/dotnet/standard/modern-web-apps-azure-architecture/common-web-application-architectures>

Moggridge, B. (2007). *Designing Interactions*. The MIT Press.

Pautasso, C., Zimmermann, O., & Leymann, F. (2008). RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision. *WWW 2008 / Refereed Track: Web Engineering - Web Service Deployment*. Beijing, China.

St. Laurent, S. (2014, Ιανουάριος). *Web application development is different (and better)*. Ανάκτηση Σεπτέμβριος 2018, από O'Reilly Web Programming: <https://www.oreilly.com/ideas/web-application-development-is-different-and-better>

Steiniger, S., Neun, M., & Edwardes, A. (2006). *Foundations of Location Based Services*. Zurich: University of Zurich.

Vindu, G., & . (2013). With New App, Foursquare Strives to Be 'Magic' in Your Pocket. *New York Times*.

<https://www.oreilly.com/ideas/web-application-development-is-different-and-better>

https://www.cs.cmu.edu/afs/cs/project/vit/ftp/pdf/intro_softarch.pdf

https://books.google.gr/books?id=W-c-mEs0fgC&pg=PA292&lpg=PA292&dq=Alonso,+et+al.,+2004+wsl&source=bl&ots=cbiV4T3nbl&sig=ACfU3U2g9xLI4B1MOpTYuU1wkQiB-_dkjw&hl=el&sa=X&ved=2ahUKEwj8uOqt4OvgAhWs8aYKHfTICOAQ6AEwAHoECAAQAQ#v=onepage&q=Alonso%2C%20et%20al.%2C%202004%20wsl&f=false

<https://www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf>

https://www.researchgate.net/publication/221022734_RESTful_Web_Services_vs_Big_Web_Services_-_Making_the_Right_Architectural_Decisions

<http://wtf.tw/ref/holtzblatt.pdf>

<http://www.etpe.gr/custom/pdf/etpe1871.pdf>

https://www2.deloitte.com/gr/en/pages/strategy-operations/articles/2017_shared_services_survey_results_deloitte_greece.html

<https://www.cognizant.com/whitepapers/Hybrid-Mobile-Application-Analysis-and-Guidelines-codex888.pdf>

<http://blog.foursquare.com/2013/08/29/a-smarter-foursquare-so-you-dont-miss-a-thing/>

