



## Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

### Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	<b>Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών.</b> <b>Analysis, design and implementation of a smart-phone application for interaction with geographic information systems via web services.</b>
Όνοματεπώνυμο Φοιτητή	<b>Σοφίος Κωνσταντίνος</b>
Πατρώνυμο	<b>Χαράλαμπος</b>
Αριθμός Μητρώου	<b>ΜΠΠΛ/10010</b>
Επιβλέπων	<b>Χρήστος Δουληγέρης, Καθηγητής</b>
Συνεπιβλέπων	<b>Δρ. Βασίλειος Μενεκλής, Ερευνητής</b>

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Ημερομηνία Παράδοσης **Μάρτιος 2013**

---

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

**Τριμελής Εξεταστική Επιτροπή**

(υπογραφή)

Χρήστος Δουληγέρης  
Καθηγητής

(υπογραφή)

Ιωάννης Θεοδωρίδης  
Καθηγητής

(υπογραφή)

Παναγιώτης  
Κοτζανικολάου  
Λέκτορας

## Περίληψη

Στην παρούσα μεταπτυχιακή διατριβή περιγράφεται η ανάλυση, η σχεδίαση και η υλοποίηση μιας εφαρμογής για κινητά τηλέφωνα με λειτουργικό σύστημα Android, η οποία βασίζεται στην πληροφορία της θέσης του χρήστη σε πραγματικό χρόνο. Η εφαρμογή αυτή επιτρέπει στο χρήστη να ανακαλύπτει σημεία ενδιαφέροντος που βρίσκονται κοντά στη γεωγραφική του θέση, λαμβάνοντας επιπλέον πληροφορίες για αυτά. Ο χρήστης έχει τη δυνατότητα να προσθέτει δικά του σημεία, καθώς επίσης και να ρυθμίζει την ακτίνα αναζήτησης αυτών σε περίπτωση που το επιθυμεί. Η ανάπτυξη της εφαρμογής βασίζεται στο μοντέλο αρχιτεκτονικής Πελάτη - Εξυπηρετητή (Client-Server) με τη χρήση των πιο σύγχρονων τεχνολογιών, όπως η πλατφόρμα PhoneGap η οποία επιτρέπει τη χρήση διαδικτυακών γλωσσών προγραμματισμού και σήμανσης (HTML5, Javascript, CSS3), καθώς και η χρήση της χωρικής βάσης δεδομένων PostgreSQL με την επέκταση PostGIS, η οποία αποτελεί ένα γεωγραφικό πληροφοριακό σύστημα.

Η σχεδίαση της εφαρμογής υλοποιήθηκε με τη χρήση της επαναληπτικής ενοποιημένης διαδικασίας (RUP) και της UML. Επιπλέον, στην εργασία αυτή πραγματοποιείται βιβλιογραφική έρευνα σχετικά με άλλες εργασίες και εφαρμογές που υπάρχουν και σχετίζονται είτε περισσότερο είτε λιγότερο με τη παρούσα μεταπτυχιακή διατριβή και επισημαίνονται οι ομοιότητες και οι διαφορές με τη κάθε μια. Τέλος, στην εργασία περιγράφεται κάθε τεχνολογία και υπηρεσία που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής.

## Abstract

The aim of this dissertation is to present the analysis, design and implementation of an application for Android Smartphones, that is based on user's location in real time. This application allows the user to explore points of interest located close to his position. The user has also the ability to add his own points of interest and to regulate the radius of search. The development of the application is based on the Client - Server model of architecture using the most modern technologies, such as the Phonegap platform which allows the usage of web programming languages and mark-up (eg. HTML5, Javascript, CSS3). In addition, in this dissertation the spatial database PostgreSQL with the PostGIS extension is used.

The design of the application has been illustrated by the Rational Unified Process with the usage of UML. Moreover, in this dissertation a bibliography research is performed, related to other projects and applications, examining to which extent they exist similarities or differences to each other. Finally, we describe all the technologies and services that have been used for the development of the current application.

## Ευχαριστίες

Η παρούσα μεταπτυχιακή διατριβή εκπονήθηκε στα πλαίσια της ολοκλήρωσης των σπουδών μου στο Μεταπτυχιακό Πρόγραμμα Σπουδών στην Πληροφορική του Πανεπιστημίου Πειραιώς.

Θέλω να ευχαριστήσω τον καθηγητή μου κ. Δουληγέρη που μου έδωσε την ευκαιρία να συνεργαστώ μαζί του για την εκπόνηση της μεταπτυχιακής μου διατριβής καθώς επίσης και τον Δρ. Βασίλη Μενεκλή για το θέμα της εργασίας που μου πρότεινε, για την υποστήριξη του και την πολύτιμη βοήθεια που μου προσέφερε σε όλα τα στάδια της εργασίας. Ευχαριστώ την οικογένεια μου, τη Βάσια, τον Γιώργο, τον Γιάννη, τον Σταύρο, τον Χρήστο, τον Νίκο που υπήρξαν πάντα το καλύτερο στήριγμα κατά τη διάρκεια της ακαδημαϊκής μου πορείας στο Πανεπιστήμιο Πειραιώς. Οι ευχαριστίες απευθύνονται επίσης προς τους διδάσκοντες του μεταπτυχιακού προγράμματος για την σημαντική δουλειά και γνώση που μου παρείχαν κατά τη διάρκεια των σπουδών μου.

Η εργασία αυτή αφιερώνεται στην οικογένεια μου, στη Βάσια, στο Γιώργο, στο Γιάννη, στο Σταύρο, στο Χρήστο και στο Νίκο.

Σοφίος Κωνσταντίνος,  
Αθήνα, Μάρτιος 2013

## Πίνακας περιεχομένων

ΚΕΦΑΛΑΙΟ 1.....	8
Εισαγωγή .....	8
Περιγραφή του θέματος.....	8
Περιγραφή της υλοποιηθείσας προσέγγισης.....	9
Δομή Μεταπτυχιακής Διατριβής .....	9
ΚΕΦΑΛΑΙΟ 2.....	11
Σχετική Βιβλιογραφία .....	11
ΚΕΦΑΛΑΙΟ 3.....	22
Τεχνολογίες υλοποίησης .....	22
Ορισμός Υπηρεσιών Εντοπισμού θέσης.....	22
Λειτουργικότητα Υπηρεσιών εντοπισμού θέσης.....	23
Λογισμικό Android .....	24
Χαρακτηριστικά Android .....	25
Αρχιτεκτονική Android .....	25
Eclipse & Android SDK Tools .....	28
Λογισμικά Open Source.....	28
PhoneGap.....	29
HTML5.....	30
Javascript .....	31
CSS.....	31
Jquery & Jquery Mobile .....	32
<i>Παραδείγματα</i> .....	32
JSON.....	33

<b>AJAX</b> .....	<b>34</b>
<b>ΚΕΦΑΛΑΙΟ 4</b> .....	Error! Bookmark not defined. <b>1</b>
<b>Αρχιτεκτονική</b> .....	<b>411</b>
<b>Ανάλυση Απαιτήσεων</b> .....	<b>411</b>
<b>Αρχιτεκτονική</b> .....	<b>422</b>
<b>Σχεδιασμός</b> .....	<b>433</b>
Διαγράμματα Περιπτώσεων Χρήσης (Use Cases) .....	444
Διαγράμματα Κλάσεων (Class Diagrams) .....	455
Διαγράμματα Συνεργασίας (Collaboration Diagrams) .....	466
Διαγράμματα Ακολουθίας (Sequence Diagrams).....	477
Διαγράμματα Δραστηριότητας (Activity Diagrams)....	499
Διαγράμματα Εξαρτημάτων .....	51
Διαγράμματα Διανομής.....	511
<b>Ανάπτυξη της εφαρμογής</b> .....	<b>522</b>
<b>ΚΕΦΑΛΑΙΟ 5</b> .....	Error! Bookmark not defined. <b>2</b>
<b>Συμπεράσματα</b> .....	<b>622</b>
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ</b> .....	<b>655</b>
<b>ΠΑΡΑΡΤΗΜΑ Α</b> .....	<b>677</b>
<b>Κώδικας Εφαρμογής</b> .....	<b>677</b>
<b>ΠΑΡΑΡΤΗΜΑ Β</b> .....	<b>955</b>
<b>Γλωσσάρι όρων</b> .....	<b>955</b>



## ΚΕΦΑΛΑΙΟ 1

### Εισαγωγή

Στο παρελθόν, η χρήση του κινητού τηλεφώνου περιοριζόταν αποκλειστικά στην πραγματοποίηση και λήψη κλήσεων, ωστόσο, τα τελευταία χρόνια η τεχνολογία της κινητής τηλεφωνίας έχει μεταβεί σε μια νέα εποχή όπου η χρήση τους επιτρέπει την επικοινωνία, την άμεση πληροφορία, τη διαδικτυακή κοινωνική δικτύωση και την πιο σύγχρονη χρήση πολυμέσων. Η νέα γενιά αυτή ακούει στο όνομα έξυπνων κινητών τηλεφώνων ή αλλιώς «Smart Phones».

Τεχνολογίες και δυνατότητες που ίσως φάνταζαν μακρινές πριν από λίγα χρόνια, όπως η αναπαραγωγή video και ήχου σε κινητές συσκευές είναι σήμερα πραγματικότητα. Η ανάπτυξη λοιπόν των κινητών συσκευών αλλά και των δικτύων κινητής τηλεφωνίας (γρηγορότερες συνδέσεις, μεγαλύτερες οθόνες με μεγαλύτερη ανάλυση, μεγαλύτερη μνήμη κ.ά.) που έχουν πλέον τη δυνατότητα να επεξεργαστούν μεγαλύτερη ποσότητα δεδομένων, έχει επιτρέψει την εισαγωγή νέων τεχνολογιών και υπηρεσιών στον τομέα της πληροφορίας που είναι διαθέσιμη στα έξυπνα κινητά τηλέφωνα. Οι υπηρεσίες αυτές υποστηρίζονται από εφαρμογές που επεκτείνονται και συνδυάζουν πολλούς τομείς όπως τις τηλεπικοινωνίες, την οικονομία, την ανάκτηση πληροφορίας, τη διασκέδαση, τα παιχνίδια κ.ά. Ο τελικός χρήστης έχει την απαίτηση να χρησιμοποιήσει αυτές τις εφαρμογές στο κινητό του τηλέφωνο. Οι κινητές συσκευές καλούνται άλλωστε να καλύψουν και τις απαιτήσεις για άμεση πρόσβαση στην πληροφορία οποιαδήποτε στιγμή και από οποιοδήποτε σημείο και αν βρίσκεται ο ενδιαφερόμενος.

Μια μεγάλη κατηγορία εφαρμογών άμεσα σχετιζόμενων με τα έξυπνα κινητά τηλέφωνα είναι η παρουσίαση της πληροφορίας σε πραγματικό χρόνο για τη θέση του χρήστη (π.χ. στο οδικό δίκτυο μιας χώρας) και ο συνδυασμός αυτός με πληροφορίες για την περιοχή. Πρόκειται για την υλοποίηση των Υπηρεσιών Θέσης (Location Based Services) (Chen & Kotz, 2000). Οι πληροφορίες θέσης πρέπει να μπορούν να συνδυαστούν με πληροφορίες για την περιοχή, το οδικό δίκτυο και τα σημεία ενδιαφέροντος αλλά και οδηγίες διαδρομής ή ακόμα και βέλτιστης μετακίνησης. Τον τομέα αυτό τον καλύπτουν συνήθως τα Γεωγραφικά Συστήματα Πληροφοριών, στα οποία η πληροφορία είναι οργανωμένη σε σχέση με τη θέση κάθε αντικειμένου. Αυτό αποτέλεσε και την έμπνευση για την πραγμάτωση της παρούσας εργασίας, εφόσον είναι άμεσα ορατή η χρησιμότητα ενός Γεωγραφικού Συστήματος Πληροφοριών για έξυπνα κινητά τηλέφωνα σε συνδυασμό με την παρεχόμενη υπηρεσία των Google Maps για την απεικόνιση της θέσης κάθε αντικειμένου.

### ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΘΕΜΑΤΟΣ

Στην παρούσα μεταπτυχιακή διατριβή περιγράφεται και μελετάται η διαδικασία σχεδίασης και υλοποίησης μιας εφαρμογής έξυπνων κινητών τηλεφώνων βασισμένη στις τεχνολογίες των Γεωγραφικών Πληροφοριακών συστημάτων (Geographic Information Systems – GIS ), καθώς επίσης και στην ανάλυση των τεχνολογιών που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής. Τα Γεωγραφικά Συστήματα Πληροφοριών είναι εξειδικευμένα συστήματα τα οποία χρησιμοποιούνται για τη γεωγραφική ανάλυση και χαρτογραφική απόδοση πληθώρας χωρικών ή χωροχρονικών κατανεμημένων φαινομένων, γεγονότων και δραστηριοτήτων. Αποτελούνται από υπολογιστές και λογισμικό, γεωγραφικά δεδομένα και διασφαλίζουν την αποτελεσματική συλλογή, αποθήκευση, ενημέρωση, διαχείριση, ανάλυση και απεικόνιση κάθε μορφής πληροφορίας με χωρική διάσταση.(Longey. Et al., 2011).

Κύριος σκοπός της εργασίας, είναι η ανάπτυξη μιας εφαρμογής σε Android που θα βασίζεται στο μοντέλο αρχιτεκτονικής Πελάτη - Εξυπηρετητή (Client Server) με τη χρήση των πιο σύγχρονων τεχνολογιών, η οποία έχει ως στόχο την εύρεση σημείων ενδιαφέροντος σε όλη την Ελλάδα που βρίσκονται κοντά στη θέση που είναι κάθε φορά ο χρήστης, σε πραγματικό χρόνο.

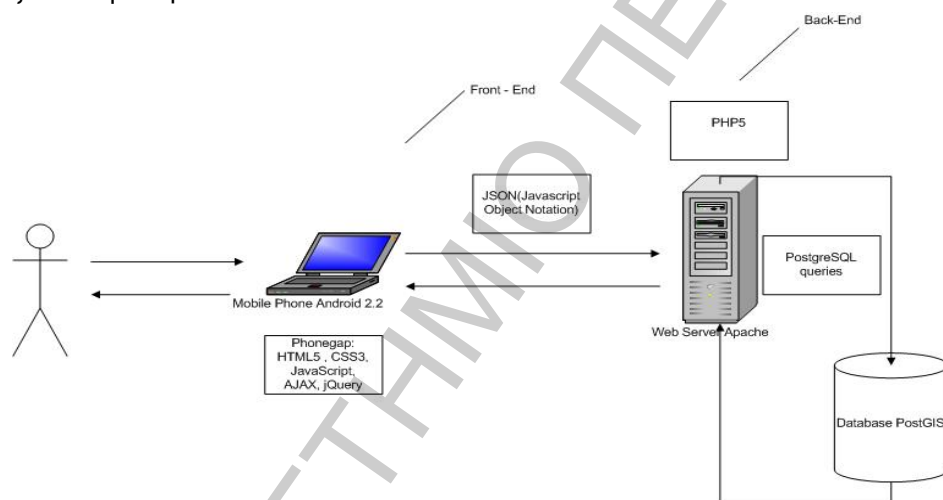
Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Επίσης, απαραίτητο τεχνολογικό συστατικό υλοποίησής της είναι να γίνει η χρήση της τεχνολογίας PhoneGap η οποία επιτρέπει τη χρήση γλωσσών προγραμματισμού και σήμανσης (HTML5, Javascript, CSS3), καθώς και η χρήση της χωρικής βάσης δεδομένων PostgreSQL με την επέκταση PostGIS και των αντίστοιχων query SQL.

## ΠΕΡΙΓΡΑΦΗ ΤΗΣ ΥΛΟΠΟΙΗΘΕΙΣΑΣ ΠΡΟΣΕΓΓΙΣΗΣ

Ο χρήστης «τρέχοντας» την εφαρμογή, έχει τη δυνατότητα να επιλέξει την κατηγορία των σημείων ενδιαφέροντος που επιθυμεί, όπως η κατηγορία υγείας (φαρμακεία, νοσοκομεία και ιατρικά κέντρα), φαγητό και διασκέδαση, δημόσιες υπηρεσίες (π.χ. στάσεις δημοσίων μέσων μεταφοράς), ανάγκες αυτοκινήτων (σημεία με καύσιμα και διαβάσεις πεζών), τουριστικά θέρετρα, κέντρα άθλησης, δραστηριότητες ελεύθερου χρόνου και ξενοδοχειακές μονάδες. Αφού έχει επιλέξει την κατηγορία που επιθυμεί, τότε εμφανίζονται σε εκείνον τα αποτελέσματα που επιθυμεί στον χάρτη που παρέχει η Google, τα Google maps. Επίσης, ο χρήστης έχει την δυνατότητα να πληροφορηθεί για τα σημεία ενδιαφέροντος που του εμφανίστηκαν στον χάρτη, όπως το όνομα, τη χιλιομετρική απόσταση από τη θέση του και την κατηγορία.

Η Αρχιτεκτονική που χρησιμοποιήθηκε βασίζεται στο μοντέλο Πελάτη – Εξυπηρετητή και απεικονίζεται στην παρακάτω εικόνα:



**Εικόνα 1** Αρχιτεκτονική εφαρμογής

Επισημαίνεται πως η εφαρμογή στηρίζεται σε λογισμικό και τεχνολογίες ανοικτού κώδικα (Open Source), δηλαδή σε λογισμικά που ο καθένας μπορεί ελεύθερα να χρησιμοποιεί, να διανέμει, να αντιγράψει και να τροποποιεί ανάλογα με τις ανάγκες του, χωρίς να απαιτείται η απόκτηση άδειας. Οι τεχνολογίες που χρησιμοποιήθηκαν είναι η πλατφόρμα ανοικτού κώδικα PhoneGap, η γλώσσα σήμανσης HTML5, η γλώσσα προγραμματισμού Javascript, πλαίσιο εργασίας JQuery και Mobile JQuery η υπηρεσία Google Maps, το πρότυπο ανταλλαγής δεδομένων JSON, ένας διαδίκτυακός εξυπηρετητής Apache, η γλώσσα προγραμματισμού PHP5 με χρήση του πλαισίου εργασίας CodeIgniter και η χωρική βάση δεδομένων PostgreSQL με την επέκταση PostGIS.

## ΔΟΜΗ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΔΙΑΤΡΙΒΗΣ

Το πρώτο κεφάλαιο αποτελεί το εισαγωγικό κομμάτι της μεταπτυχιακής διατριβής, στο οποίο αναφέρονται λίγα λόγια για την τεχνολογική εξέλιξη των κινητών τηλεφώνων και την ποικιλομορφία των εφαρμογών σήμερα. Περιγράφεται το πρόβλημα και το θέμα που έχει οριστεί να λύσει η εργασία αλλά και τι πραγματικά υλοποιήθηκε, περιγράφοντας την λειτουργία της

εφαρμογής, την αρχιτεκτονική της και αναφέροντας όλες τις τεχνολογίες που τελικά χρησιμοποιήθηκαν.

Το δεύτερο κεφάλαιο ασχολείται με τη σχετική βιβλιογραφία που υπάρχει και έχει δημοσιευτεί είτε σε επιστημονικά περιοδικά, είτε σε παγκόσμια ακαδημαϊκά συνέδρια είτε σαν διπλωματικές εργασίες φοιτητών σε διάφορα πανεπιστήμια, οι οποίες σχετίζονται με κάποιο από τα θέματα που μελετά η παρούσα μεταπτυχιακή διατριβή. Για κάθε μια από τις δημοσιευμένες επιστημονικές εργασίες γίνεται η περιγραφή της και μια σύντομη σύγκριση με την παρούσα.

Στο τρίτο κεφάλαιο αναφέρονται και περιγράφονται οι τεχνολογίες, τα εργαλεία και προγράμματα ανάπτυξης που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής.

Το τέταρτο κεφάλαιο ασχολείται εκτενώς με την αρχιτεκτονική της εφαρμογής. Περιγράφεται η πλατφόρμα και γίνεται σχεδιασμός με τη χρήση της αντικειμενοστρεφούς ανάπτυξης και σχεδίασης Λογισμικού με UML βασισμένη στη διαδικασία Rational Unified Process (RUP). Επίσης περιγράφεται ο τρόπος με τον οποίο εργάστηκε ο φοιτητής που εκπόνησε την παρούσα εργασία.

Τέλος, στο τελευταίο και πέμπτο κεφάλαιο περιέχονται τα συμπεράσματα της παρούσας μεταπτυχιακής διατριβής. Συμπεράσματα σχετικά με τις τεχνολογίες, την αρχιτεκτονική της εφαρμογής, τα σύγχρονα κινητά τηλέφωνα αλλά και τις κοινωνικές συνέπειες που έχουν, είτε θετικές είτε αρνητικές. Αναφέρονται οι περιορισμοί που έλαβαν χώρα στην ανάπτυξη της εργασίας και ο λόγος τους, καθώς και οι μελλοντικές επεκτάσεις που μπορεί να έχει η εφαρμογή αυτή.

## ΚΕΦΑΛΑΙΟ 2

### Σχετική Βιβλιογραφία

Στο κεφάλαιο αυτό γίνεται βιβλιογραφική αναφορά εργασιών και άρθρων που σχετίζονται με τα θέματα που ερευνά η παρούσα μεταπτυχιακή διατριβή. Οι εργασίες που εξετάζονται αποτελούν είτε διπλωματικές ή μεταπτυχιακές εργασίες πανεπιστημίων, είτε άρθρα και αναφορές που έχουν δημοσιευτεί.

Οι εργασίες και τα άρθρα που εξετάζονται είναι εννέα. Για κάθε μια από αυτές τις δημοσιευμένες εργασίες γίνεται μια περίληψη η οποία αποσκοπεί στο να εξετάσει το στόχο και το πρόβλημα της κάθε εργασίας που καλείται να λύσει ή να εξετάσει και τα αντικείμενα ή επιστημονικά πεδία που ερευνά. Επιπλέον, γίνεται σύγκριση με την παρούσα εργασία σχετικά με τις διαφορές που έχουν, τα πλεονεκτήματα και τα μειονεκτήματα της κάθε μίας.

Η πρώτη εργασία (Οικονόμου, 2010) αποτελεί διπλωματική η οποία αναφέρεται στην ανάπτυξη μιας εφαρμογής για κινητά τηλέφωνα η οποία βασίζεται στη γνώση της γεωγραφικής θέσης του χρήστη.

Αρχικά, γίνεται μια εισαγωγή σχετικά με τη συμβολή της χρήσης των κινητών τηλεφώνων στην αύξηση των χρηστών του διαδικτύου και τη συμβολή των έξυπνων κινητών τηλεφώνων στην ανάπτυξη νέων εφαρμογών. Στη συνέχεια εξετάζεται το ζήτημα της υπερπληροφόρησης στο διαδίκτυο, τη διαδραστικότητα μέσω της τεχνολογίας AJAX κάνοντας λόγο για τη νέα γενιά εφαρμογών. Επίσης, γίνονται ιστορικές αναφορές της εξέλιξης του διαδικτύου σε σχέση με τα κινητά τηλέφωνα από τη τεχνολογία του WAP έως τη σημερινή τεχνολογία των Smart phones και της νέας γενιάς του Διαδικτύου (WEB 2.0). Στη συνέχεια αναφέρεται στους χάρτες και το διαδίκτυο εξετάζοντας τη γνωστή σε όλους μας παρεχόμενη υπηρεσία/τεχνολογία της Google, τα Google Maps API. Εξετάζει επίσης, τις τεχνολογίες και τις εφαρμογές που παρέχουν στους χρήστες υπηρεσίες γεωγραφικών θέσεων όπως το GPS και το ασύρματο (Wi-Fi) σύστημα θέσης (position System). Στη συνέχεια εξετάζεται η εφαρμογή που αναπτύχθηκε με το όνομα Othesis.

Όπως αναφέρει ο Οικονόμου (2010), στόχος είναι ουσιαστικά η δημιουργία μιας εφαρμογής βασισμένης στην πληροφορία που ανακτάται από τη γεωγραφική θέση του χρήστη για τη συσκευή I-phone του περιβάλλοντος IOS. Η εφαρμογή δίνει την επιλογή στο χρήστη να αναζητήσει σε έναν εξωτερικό χώρο κατηγορίες τοποθεσιών και να τον παρακινήσει να παράγει περιεχόμενο παρέχοντας του μια συμμετοχική εφαρμογή. Η αρχιτεκτονική που χρησιμοποιεί είναι ένας web server και μια βάση δεδομένων MySQL. Τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν είναι η Python, το Django ένα διαδικτυακό πλαίσιο εργασίας (web framework) και το JSON. Η πλατφόρμα που αναπτύχθηκε η εφαρμογή, είναι το Titanium Appcelerator.

Οι ομοιότητες της παραπάνω διπλωματικής εργασίας που εξετάσαμε με την παρούσα μεταπτυχιακή διατριβή είναι ότι και οι δύο εξετάζουν μια εφαρμογή κινητού τηλεφώνου (smart phone) η οποία λαμβάνει κάθε φορά τη γεωγραφική θέση του χρήστη και του δίνει τη δυνατότητα να δει διάφορα σημεία ενδιαφέροντος που βρίσκονται σε μια συγκεκριμένη ακτίνα γύρω από αυτόν. Επίσης και οι δύο εφαρμογές βασίζονται στην αρχιτεκτονική πελάτη εξυπηρετητή και χρησιμοποιούν την τεχνολογία JSON (μορφή για μορφοποιήσεις δεδομένων σε javascript) για την ανταλλαγή δεδομένων.

Ωστόσο η κύρια διαφορά τους είναι η τεχνολογία που χρησιμοποιήθηκε για να αναπτυχθεί η κάθε μια. Η παρούσα εφαρμογή αναπτύχθηκε με την τεχνολογία Phonegap, η οποία δίνει τη δυνατότητα στον προγραμματιστή να χρησιμοποιήσει τις κλασσικές τεχνολογίες διαδικτυακού

προγραμματισμού όπως η HTML5, Javascript, AJAX, JQuery, JSON, CSS3, δίνοντας την δυνατότητα χρήσης του framework JQuery mobile. Επίσης στην πλευρά του εξυπηρετητή χρησιμοποιείται η γλώσσα προγραμματισμού PHP5, με τη χρήση του framework codeigniter και η Γεωγραφική βάση δεδομένων PostgreSQL. Από την άλλη, η εφαρμογή που περιγράφηκε αναπτύχθηκε με τη χρήση της γλώσσας Python και του πλαισίου εργασίας Django στην πλευρά του εξυπηρετητή. Να σημειωθεί και οι δύο προσεγγίσεις βασίζονται στο πρότυπο ανάπτυξης MVC. Δεν μπορεί να υπάρξει σύγκριση επειδή είναι δύο διαφορετικές γλώσσες προγραμματισμού. Αποτελούν προσωπική προτίμηση του κάθε προγραμματιστή. Επίσης, η εφαρμογή της παρούσας μεταπτυχιακής διατριβής, αναπτύχθηκε για περιβάλλον Android σε αντίθεση με αυτή που περιγράψαμε παραπάνω που αναπτύχθηκε για IOS. Επιπλέον, η διεπαφή των εφαρμογών είναι διαφορετική. Στην παρούσα εφαρμογή ο χρήστης πρέπει να επιλέξει την κατηγορία των σημείων ενδιαφέροντος που θέλει να δει. Και στις δύο εφαρμογές ο χρήστης έχει τη δυνατότητα να επιλέξει την ακτίνα που επιθυμεί να «ψάξει» το σύστημα και να του επιστρέψει τα ανάλογα αποτελέσματα.

Η τεχνολογία Phonegap που χρησιμοποιεί η παρούσα εφαρμογή επιτρέπει την ανάπτυξη εφαρμογών σε όλα τα λειτουργικά των έξυπνων κινητών τηλεφώνων που κυκλοφορούν (IOS, Android, Windows Phone, Blackberry, Symbian, WebOS, Bada). Επίσης, χρησιμοποιεί χωρική βάση δεδομένων PostgreSQL η οποία σαφώς υπερτερεί σε θέματα γεωγραφικών δεδομένων σε σχέση μια με μια κλασική βάση δεδομένων όπως η MySQL. Ωστόσο, υστερεί σε κάποιες λειτουργίες σε σχέση με την εφαρμογή που εξετάσαμε, όπως η δυνατότητα ταξινόμησης των πληροφοριών του κάθε σημείου ενδιαφέροντος ή η δυνατότητα αλλαγής του χάρτη. Η παρούσα εφαρμογή δίνει την δυνατότητα στο χρήστη να προσθέσει σημεία ενδιαφέροντος συμπληρώνοντας μια φόρμα η οποία αποστέλλεται στον διαχειριστή της εφαρμογής οποίος ελέγχει την ορθότητα των δεδομένων πριν τα εισάγει στην βάση δεδομένων. Επίσης στην παρούσα μεταπτυχιακή εργασία αποτυπώνεται ένας από τους πιο σύγχρονους τρόπους σχεδιασμού με τη χρήση UML και τη μέθοδο Rational Unified Process (RUP).

Η επόμενη εργασία (Παπαδόπουλος, 2011) αποτελεί μια διπλωματική εργασία η οποία αναφέρεται στη σχεδίαση και ανάπτυξη πλατφόρμας παροχής υπηρεσιών περιήγησης βασισμένη στη θέση των χρηστών.

Ο Παπαδόπουλος (2011), αναφέρει ότι σκοπός της διπλωματικής εργασίας είναι η μελέτη και η ανάπτυξη μιας εφαρμογής βασισμένης σε υπηρεσίες θέσης (Location Based Services). Η εφαρμογή περιλαμβάνει μια ιστοσελίδα στην οποία εξουσιοδοτημένοι χρήστες έχουν τη δυνατότητα να προσθέτουν και να επεξεργάζονται δεδομένα που έχουν να κάνουν με σημεία ενδιαφέροντος. Επίσης, περιλαμβάνεται και μια εφαρμογή για κινητά τηλέφωνα με λειτουργικό Android στην οποία οι χρήστες θα έχουν τη δυνατότητα να ξεναγούνται με τη βοήθεια χαρτών, κειμένου, ήχου σε σημεία ενδιαφέροντος που βρίσκονται στη βάση δεδομένων.

Στην εργασία του Παπαδόπουλου, αρχικά αναφέρονται τα βασικά χαρακτηριστικά και η αρχιτεκτονική μιας Υπηρεσίας βασισμένης στη θέση του χρήστη. Στη συνέχεια αναφέρονται τα χαρακτηριστικά, η αρχιτεκτονική και το μοντέλο εφαρμογών του λειτουργικού Android. Η ιδέα της εφαρμογής είναι να μπορεί ο χρήστης να ξεναγηθεί σε σημαντικά ορόσημα που βρίσκονται κοντά του με βάση πληροφορίες που έχουν εισαχθεί από ειδήμονες σχετικά με τα αυτά τα ορόσημα.

Για την εισαγωγή δεδομένων έχει δημιουργηθεί ένας ιστότοπος οποίος επιτρέπει με τη χρήση των Google Maps την δημιουργία σημείων ενδιαφέροντος. Για κάθε σημείο θα μπορεί να ανεβεί στον εξυπηρετητή υλικό το οποίο θα μπορεί μετά να κατεβάσει ο χρήστης στο κινητό του. Η εισαγωγή των δεδομένων θα γίνεται μόνο από εξουσιοδοτημένους χρήστες. Η αρχιτεκτονική που χρησιμοποιείται είναι του Πελάτη - Εξυπηρετητή με χρήση βάσης δεδομένων MySQL και διαδικτυακό εξυπηρετητή (web server) Apache. Η γλώσσα προγραμματισμού της πλευράς του εξυπηρετητή είναι η PHP. Στην πλευρά του πελάτη (Client) η εφαρμογή λειτουργεί σε Android όπως αναφέραμε και παραπάνω. Η εφαρμογή αρχικά εντοπίζει την γεωγραφική θέση του χρήστη κάνοντας χρήση GPS ή του δικτύου που βρίσκεται το ασύρματο τερματικό. Αφού

εντοπιστεί η θέση τότε η συσκευή στέλνει στον Server μέσω του διαδικτύου τη θέση του χρήστη και αυτός στέλνει πίσω τα σύνολα που έχουν σημεία ενδιαφέροντος κοντά στο χρήστη.

Οι ομοιότητες της παραπάνω διπλωματικής εργασίας που περιγράφηκε σε σχέση με την παρούσα μεταπτυχιακή διατριβή είναι ότι και οι δύο εξετάζουν μια εφαρμογή σε Android χρησιμοποιώντας την γεωγραφική θέση του χρήστη οπτικοποιώντας την σε Google Maps, δείχνοντας στο χρήστη τα κοντινά σημεία ενδιαφέροντος που υπάρχουν στη θέση που βρίσκεται κάθε φορά. Επίσης, υπάρχει ομοιότητα και στην γλώσσα προγραμματισμού που χρησιμοποιείται, την PHP. Οι δύο εφαρμογές βασίζονται στην αρχιτεκτονική πελάτη - εξυπηρετητή.

Ωστόσο υπάρχουν αρκετές διαφορές μεταξύ τους. Αρχικά, η βάση δεδομένων που αναπτύχθηκε η παρούσα εφαρμογή είναι η χωρική αντικειμενοστρεφή-σχεσιακή βάση δεδομένων PostgreSQL και η επέκταση PostGIS για υποστήριξη χωρικών λειτουργιών δίνοντας το πλεονέκτημα της απόδοσης σε χωρικά δεδομένα. Η PHP υλοποιήθηκε με χρήση του πλαισίου εργασίας Codeigniter το οποίο μας δίνει την δυνατότητα δημιουργίας τελειοποιημένων χαρακτηριστικών διαδικτυακών εφαρμογών. Μια ακόμα διαφορά ως προς τη τεχνολογία ανάπτυξης της εφαρμογής του κινητού είναι ότι η παρούσα εργασία υλοποιείται με Phonegap σε αντίθεση με αυτή που εξετάσαμε η οποία αναπτύχθηκε με JAVA. Η παρούσα εφαρμογή έχει ως στόχο να οπτικοποιήσει όλα τα σημεία ενδιαφέροντος ανά κατηγορίες που βρίσκονται σε μια συγκεκριμένη ακτίνα γύρω από τον χρήστη. Τα δεδομένα υπάρχουν ήδη στη βάση δεδομένων και δεν έχει δικαίωμα ο χρήστης να προσθέσει μόνος του δεδομένα. Μπορεί όμως να επικοινωνήσει με τον διαχειριστή στέλνοντας του μια φόρμα με στοιχεία ενός σημείου ενδιαφέροντος που θεωρεί ότι δεν υπάρχει στις ήδη υπάρχουσες εγγραφές. Αυτό συμβαίνει για τη διατήρηση της ασφάλειας και της ορθότητας των δεδομένων. Στην εφαρμογή που εξετάσαμε ο χρήστης μπορεί να εισαγάγει σημεία ενδιαφέροντος από τη διαδικτυακή ιστοσελίδα, με τη χρήση βέβαια λογαριασμού, για την ασφάλεια και την ορθότητα των δεδομένων που προσθέτονται. Στην παρούσα μεταπτυχιακή εργασία αποτυπώνεται ένας από τους πιο σύγχρονους τρόπους σχεδιασμού με τη χρήση UML και η μέθοδος Rational Unified Process (RUP).

Η επόμενη εργασία (Σουέρεφ & Αργυρόπουλος, 2012), αποτελεί πτυχιακή η οποία ασχολείται με την μελέτη της ανάπτυξης εφαρμογών για κινητά τηλέφωνα τα οποία χρησιμοποιούν το λειτουργικό σύστημα Android.

Οι Σουέρεφ & Αργυρόπουλος (2012), αναλύουν και εξετάζουν την ανάπτυξη εφαρμογών σε κινητά τηλέφωνα Android χρησιμοποιώντας τις κλασσικές τεχνολογίες του Διαδικτυακού προγραμματισμού όπως η HTML, CSS3 και Javascript.

Αρχικά, γίνεται μια αναφορά στην επεξεργασία της πλατφόρμας Android, παρέχοντας χρήσιμες πληροφορίες για την αρχιτεκτονική του, τα πλεονεκτήματά του, τα έξυπνα κινητά τηλέφωνα που την χρησιμοποιούν αλλά και οι ταμπλέτες, καθώς επίσης οδηγίες εγκατάστασης για τον προγραμματιστή και οδηγίες δημιουργίας εικονικής συσκευής Android. Στη συνέχεια, εξηγείται ο τρόπος δημιουργίας μιας απλής εφαρμογής Ιστού αρχικά, με την οποία θα καταχωρούνται λεπτομέρειες μελλοντικών εργασιών, τις οποίες θα διαχειρίζονται με τη χρήση του κινητού. Έπειτα γίνεται αναφορά στην αποθήκευση με την HTML5 και την αποθήκευση αντικειμένων Javascript με χρήση JSON καθώς επίσης υλοποιούνται και παραδείγματα λειτουργιών του CSS3 και της HTML5. Στη συνέχεια γίνεται λόγος στις βασισμένες-στην-τοποθεσία υπηρεσίες χαρτογράφησης και στην υλοποίηση ενός γεωκοινωνικού (geosocial) παιχνιδιού που χρησιμοποιεί στοιχεία από το γεωκοινωνικό δίκτυο (geosocial network) Gowalla. Τέλος γίνεται λόγος για την εφαρμογή PhoneGap και την προοπτική ενός κατασκευαστή ιστοσελίδων για την ανάπτυξη Διαδικτυακών εφαρμογών για κινητά.

Η παραπάνω εφαρμογή που περιγράφηκε με την εφαρμογή της παρούσας μεταπτυχιακής διατριβής διαφέρει σε πολύ μεγάλο βαθμό. Ουσιαστικά σκοπός της πτυχιακής εργασίας είναι η ανάλυση και η εξέταση της διαδικασίας ανάπτυξης εφαρμογών σε κινητά τηλέφωνα Android. Η παρούσα εφαρμογή αντίθετα, αποσκοπεί στην ανάλυση σχεδίαση και υλοποίησης μιας

εφαρμογής σε Android που έχει ως σκοπό να παίρνει το στίγμα του χρήστη και να επιστρέφει τα σημεία ενδιαφέροντος που βρίσκονται γύρω από αυτόν.

Ωστόσο υπάρχουν ομοιότητες στις τεχνολογίες που χρησιμοποιούν οι δύο αυτές εφαρμογές. Και οι δύο χρησιμοποιούν και αναλύουν τις κλασσικές διαδικτυακές τεχνολογίες (HTML5, CSS3, Javascript) καθώς επίσης και την τεχνική JSON. Επίσης, αναλύεται και περιγράφεται η τεχνολογία Phonegap καθώς επίσης και η πλατφόρμα Android. Στο σημείο αυτό πρέπει να τονίσουμε ότι η εργασία που εξετάσαμε μελετά και εξηγεί λεπτομερώς τις βασικές τεχνικές ανάπτυξης μιας εφαρμογής σε Android σε αντίθεση με την παρούσα εφαρμογή η οποία παρουσιάζει μια νέα γενιά εφαρμογών οι οποίες βασίζονται σε γεωγραφικά πληροφοριακά συστήματα και δεν περιέχει οδηγίες ανάπτυξης και προγραμματισμού. Επίσης στην παρούσα μεταπτυχιακή εργασία αποτυπώνεται ένας από τους πιο σύγχρονους τρόπους σχεδιασμού με τη χρήση UML και την μέθοδος Rational Unified Process (RUP).

Η επόμενη εργασία (Loi, 2010) αποτελεί διπλωματική η οποία ασχολείται με τη διερεύνηση της αρχιτεκτονικής Πληροφοριακών Συστημάτων για κινητά τηλέφωνα.

Όπως αναφέρεται η εργασία αυτή έχει ως στόχο την έρευνα και την ανάλυση της αρχιτεκτονικής των πληροφοριακών συστημάτων για σύγχρονα κινητά τηλέφωνα και της διαδικασίας ανάλυσης σχεδιασμού και υλοποίησης ενός τέτοιου συστήματος (Loi, 2010).

Αρχικά αναλύεται η πρόσβαση πληροφοριών από το διαδίκτυο σε κινητά τηλέφωνα, το διαδίκτυο των κινητών τηλεφώνων, η αρχιτεκτονική των εφαρμογών σε κινητά τηλέφωνα και ήδη υπάρχουσες αντίστοιχες εφαρμογές με αυτή της επιστημονικής εργασίας που εξετάζουμε. Στη συνέχεια, γίνεται λόγος για την διαδικασία ανάλυσης απαιτήσεων που πρέπει να γίνεται πριν από την διαδικασία ανάπτυξης και υλοποίησης μιας εφαρμογής. Έπειτα, αναλύεται η αρχιτεκτονική της εφαρμογής και η διαδικασία ενσωμάτωσης ενός ήδη υπάρχοντος συστήματος Agent. Στη συνέχεια περιγράφεται η διαδικασία της υλοποίησης της εφαρμογής χωρίς τη χρήση τεχνικής επεξήγησης κώδικα, καθώς και τα προβλήματα που προέκυψαν κατά τη διάρκεια της εφαρμογής, πώς αντιμετωπίστηκαν και τι επιπτώσεις είχαν αυτές οι αλλαγές για το σύστημα.

Η παραπάνω επιστημονική εργασία που περιγράφηκε αναφέρεται σε μια τελείως διαφορετική εφαρμογή από την παρούσα εργασία. Ωστόσο, υπάρχουν ομοιότητες διότι και οι δύο εργασίες αναφέρονται σε εφαρμογές κινητών τηλεφώνων. Επίσης, η αρχιτεκτονική που βασίζονται είναι η ίδια, «Πελάτη - Εξυπηρετητή» και η αναφορά χρήσης πρότυπου σχεδίου MVC στην γλώσσα προγραμματισμού PHP στη πλευρά του Εξυπηρετητή. Επίσης και οι δύο χρησιμοποιούν τεχνολογίες διαδικτυακού προγραμματισμού.

Η επόμενη εργασία (Κοντόπουλος, 2010) αποτελεί διπλωματική η οποία αναφέρεται στην ανάπτυξη διαδικτυακών εφαρμογών γεωγραφικών πληροφοριακών συστημάτων (GIS) με λογισμικό ανοικτού κώδικα Geoserver.

Ο Κοντόπουλος (2010), αναφέρει ότι στόχος της εργασίας είναι η διαχείριση της χωρικής πληροφορίας στο διαδίκτυο με τη χρήση λογισμικού ανοικτού κώδικα και πιο συγκεκριμένα η ανάπτυξη ενός γεωγραφικού πληροφοριακού συστήματος (GIS) που θα αξιοποιεί τις πληροφορίες ενός χωρικού εξυπηρετητή (Server) ο οποίος υλοποιείται με το λογισμικό Geoserver μέσω μίας Web εφαρμογής. Αρχικά, γίνεται μια αναφορά για τα Γεωγραφικά πληροφοριακά συστήματα(GIS) σχετικά με την ιστορία τους, τα χαρακτηριστικά τους την αναπαράσταση των χωρικών δεδομένων και τα συστήματα συντεταγμένων. Επίσης, γίνεται αναφορά για το ανοικτού κώδικα GIS περιβάλλον, την χωρική βάση δεδομένων PostgreSQL, τον Geoserver και την διαδικασία ερωτημάτων σε αυτόν, την τεχνολογία του MapServer και για το αποτέλεσμα σύγκρισης του MapServer και του Geoserver. Στη συνέχεια περιγράφεται η εφαρμογή QuantumGIS, ένα εργαλείο οπτικοποίησης χωρικών δεδομένων και διάφορες λειτουργίες και δυνατότητες του εργαλείου αυτού.

Στη συνέχεια η εργασία αναφέρεται στο τεχνολογικό υπόβαθρο που χρησιμοποιήθηκε για την υλοποίηση της εφαρμογής. Αρχικά δίνεται ένας ορισμός της XML και εξηγείται η δομή της. Έπειτα, αναφέρεται στο πρότυπο κωδικοποίησης της OpenGIS, τη GML(Geography Markup

Language). Στη συνέχεια αναλύει και εξηγεί τις τεχνολογίες διαδικτύου, HTML5, Javascript και JSON. Στο επόμενο κεφάλαιο ασχολείται με την υλοποίηση και το περίγραμμα εφαρμογής. Στη συνέχεια αναφέρεται στην αρχιτεκτονική της εφαρμογής η οποία βασίζεται στο μοντέλο «Πελάτη-Εξυπηρετητή». Στη πλευρά του εξυπηρετητή (Server) χρησιμοποιήθηκε ο χωρικός server Geoserver, η βάση δεδομένων PostgreSQL με την επέκταση PostGIS και ο διαδικτυακός εξυπηρετητής (Web Server) Apache με υποστήριξη της PHP. Παρακάτω αναλύεται ο σχεδιασμός της χωρικής βάσης δεδομένων. Στην πλευρά του πελάτη(Client), η εφαρμογή αναπτύχθηκε με τη χρήση HTML, CSS και Javascript. Τέλος, εξηγείται η συνολική διαδικασία υλοποίησης της εφαρμογής με την βοήθεια εικόνων καταλήγοντας σε συμπεράσματα και μελλοντικές επεκτάσεις της.

Η παραπάνω εφαρμογή που περιγράφηκε διαφέρει στο στόχο που έχει η εφαρμογή της παρούσας μεταπτυχιακής διατριβής. Ωστόσο, χρησιμοποιούν τις ίδιες σχεδόν διαδικτυακές τεχνολογίες όπως η HTML5, Javascript και CSS3. Επίσης, λειτουργούν κάτω από το ίδιο μοντέλο αρχιτεκτονικής, αυτό του «Πελάτη- Εξυπηρετητή» (Client-Server). Και οι δύο εφαρμογές κάνουν χρήση της χωρικής βάσης δεδομένων PostgreSQL με την επέκταση της PostGIS. Χρησιμοποιούν το διαδικτυακό εξυπηρετητή Apache με υποστήριξη της PHP.

Ωστόσο, η παρούσα εφαρμογή χρησιμοποιεί framework της PHP, το Codeigniter, το οποίο είναι ένα πλαίσιο εργασίας (framework) ανοιχτού λογισμικού για την κατασκευή δυναμικών ιστοσελίδων με PHP. Το πλαίσιο εργασίας Codeigniter χρησιμοποιεί το πρότυπο ανάπτυξης MVC (mode-view-controller) και είναι ένα από τα πιο γρήγορα πλαίσια εργασίας της αγοράς. Με την παραπάνω διπλωματική εργασία δεν μπορεί να γίνει περαιτέρω σύγκριση διότι διαφοροποιούνται αρκετά στον στόχο που έχει η κάθε εφαρμογή. Η σύγκριση έγινε στις ομοιότητες που παρατηρήθηκαν ανάμεσα τους.

Η επόμενη εργασία (Μπούρχας, 2012) αποτελεί διπλωματική η οποία ασχολείται με την ανάπτυξη δυναμικής παροχής υπηρεσιών βασισμένων στη θέση των χρηστών.

Όπως αναφέρει ο Μπούρχας (2012), σκοπός της συγκεκριμένης διπλωματικής εργασίας είναι η ανάλυση των Location-Based Services και η ανάπτυξη μιας πλατφόρμας παροχής υπηρεσιών σε χρήστες βάση της γεωγραφικής τους θέσης καθώς και των ενδιαφερόντων ή επιλογών τους. Επίσης η πλατφόρμα αυτή δίνει την δυνατότητα σε παρόχους υπηρεσιών να συνδέονται και να τοποθετούν προσφορές για προϊόντα ή υπηρεσίες τα οποία παρέχονται σε συγκεκριμένες γεωγραφικές θέσεις και για συγκεκριμένο χρονικό διάστημα.

Αρχικά, αναφέρεται στις υπηρεσίες εντοπισμού θέσης (Location-Based Services) αναλύοντας τον ορισμό τους, τα κύρια χαρακτηριστικά τους, τις κατηγορίες που ταξινομούνται και τα δομικά τους μέρη. Αναλύεται η ιστορική εξέλιξη της αρχιτεκτονικής των υπηρεσιών εντοπισμού θέσης και τα μοντέλα αρχιτεκτονικής που υπάρχουν αυτή την στιγμή. Στη συνέχεια εστιάζει στο τρόπο που λειτουργούν οι υπηρεσίες αυτές, χρησιμοποιώντας την τεχνολογία GPS, και στις τεχνολογίες που χρησιμοποιούνται για να προσδιοριστεί η γεωγραφική θέση του κάθε χρήστη. Επίσης, αναφέρονται οι νομοθετικές ρυθμίσεις περί προστασίας προσωπικών δεδομένων καθώς επίσης και τεχνικές προσέγγισης για την παροχή ασφάλειας σε αυτές τις υπηρεσίες. Στο δεύτερο μέρος της εργασίας περιγράφεται το λογισμικό ανοικτού κώδικα Android, η αρχιτεκτονική του και η σχέση του με τις Location-Based υπηρεσίες. Το τρίτο και τελευταίο μέρος της διπλωματικής αυτής εργασίας εστιάζει στην επεξήγηση της υλοποίησης της εφαρμογής και στην παρουσίαση της. Η αρχιτεκτονική που βασίζεται η εφαρμογή είναι αυτή του μοντέλου Πελάτη - Εξυπηρετητή (Client-Server). Από την πλευρά του Εξυπηρετητή (Server), η Βάση Δεδομένων που χρησιμοποιήθηκε είναι η MySQL σαν γλώσσα προγραμματισμού για την υποστήριξη του Server ήταν η PHP. Από την πλευρά του Πελάτη (Client), για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκε η αντικειμενοστραφής γλώσσα προγραμματισμού Java για λειτουργικό σύστημα Android. Επίσης, για την διαχείριση των απαντήσεων του Server προς τον Client χρησιμοποιήθηκαν πίνακες JSONArray με JSON αντικείμενα (objects). Τέλος, αναλύεται η διαδικτυακή εφαρμογή(ιστοσελίδα) με την οποία, όπως αναφέραμε και παραπάνω, οι πάροχοι υπηρεσιών έχουν την δυνατότητα να εισάγουν δεδομένα προσφορών σε συγκεκριμένες τοποθεσίες για ένα συγκεκριμένο χρονικό διάστημα.



Οι ομοιότητες της εργασίας που περιγράφηκε με την παρούσα μεταπτυχιακή εργασία είναι ότι υλοποιούν μια εφαρμογή κινητού τηλεφώνου (Smart phone) σε λειτουργικό Android που βασίζεται σε υπηρεσίες εντοπισμού θέσης (Location-Based Services) της γεωγραφικής θέσης του χρήστη και σύμφωνα με αυτή προβάλλονται σημεία ενδιαφέροντος. Επίσης και οι δύο εφαρμογές βασίζονται στο μοντέλο αρχιτεκτονικής Πελάτη - Εξυπηρετητή (Client-Server).

Ωστόσο διαφοροποιούνται στις τεχνολογίες ανάπτυξης της κάθε εφαρμογής αλλά και στον στόχο της κάθε μιας. Ως προς τον στόχο η παραπάνω εφαρμογή δίνει τη δυνατότητα στο χρήστη να καθορίσει τις προτιμήσεις αναζήτησης προσφορών. Επίσης οι πάροχοι ή οι καταστηματάρχες έχουν την δυνατότητα να εισαγάγουν δεδομένα προσφορών μέσω μιας απλής διαδικτυακής εφαρμογής. Η εφαρμογή λοιπόν εντοπίζει την γεωγραφική θέση του χρήστη, ο χρήστης επιλέγει τα κριτήρια αναζήτησης προσφορών, επιλέγει την απόσταση που θέλει και του εμφανίζονται τα αντίστοιχα αποτελέσματα. Από την άλλη μεριά, η εφαρμογή της παρούσας μεταπτυχιακής διατριβής είναι να εντοπίσει τα σημεία ενδιαφέροντος που βρίσκονται γύρω από τη θέση του χρήστη ταξινομώντας τα σε κατηγορίες. Επίσης ο χρήστης έχει το δικαίωμα να καθορίσει την ακτίνα αναζήτησης σημείων γύρω από αυτόν, αλλά και το δικαίωμα να προσθέσει σημεία ενδιαφέροντος που θεωρεί ότι δεν υπάρχουν στα αποτελέσματα μέσω συμπλήρωσης μιας φόρμας η οποία ελέγχεται από τον διαχειριστή του συστήματος.

Από την τεχνολογική σκοπιά των δύο εφαρμογών, αυτή που εξετάσαμε παραπάνω χρησιμοποιεί Java για την ανάπτυξη της εφαρμογής στο κινητό τηλέφωνο. Στην πλευρά του Εξυπηρετητή χρησιμοποιεί Web Server Apache με υποστήριξη PHP. Η βάση δεδομένων που χρησιμοποιείται είναι η MySQL. Από την άλλη, η παρούσα εφαρμογή χρησιμοποιεί το πακέτο Phonegap που δίνει την δυνατότητα χρήσης των κλασικών διαδικτυακών τεχνολογιών. Συνεπώς η ανάπτυξη από την πλευρά του πελάτη γίνεται με τη χρήση της HTML5, Javascript, το πλαίσιο εργασίας Jquery Mobile και CSS3. Στην πλευρά του Εξυπηρετητή, χρησιμοποιείται ο Web Server Apache με υποστήριξη PHP5 κάνοντας χρήση του framework Codeigniter. Η βάση δεδομένων που χρησιμοποιείται είναι η χωρική PostgreSQL κάνοντας χρήση της επέκτασης PostGIS, η οποία έχει την δυνατότητα να διαχειρίζεται χωρικά δεδομένα με μεγάλη απόδοση.

Η επόμενη εργασία αποτελεί πτυχιακή, η οποία μελετά ένα πληροφοριακό σύστημα για την καθοδήγηση χρηστών σε σχέση με σημεία ενδιαφέροντος, χρησιμοποιώντας γεωγραφικό πληροφοριακό σύστημα(GIS).

Αναφέρεται ότι στόχος της πτυχιακής αυτής εργασίας είναι η ο σχεδιασμός και η υλοποίηση ενός πληροφοριακού συστήματος για την καθοδήγηση των χρηστών με τη χρήση γεωγραφικού πληροφοριακού συστήματος (Μανδουράρη, 2011). Οι χρήστες της εφαρμογής έχουν τη δυνατότητα να βλέπουν που βρίσκονται τα σημεία ενδιαφέροντος (Points of interest) αναζητώντας τα στο χάρτη μέσω του ειδικού πεδίου αναζήτησης. Πιο συγκεκριμένα η αναζήτηση αφορά σημεία που βρίσκονται στο νομό Λασιθίου.

Αρχικά, η πτυχιακή εργασία αναφέρεται στις γλώσσες προγραμματισμού και εφαρμογές που χρησιμοποιήθηκαν, η PHP, Javascript, SQL, JSP και HTML. Στη συνέχεια περιγράφονται τα χαρακτηριστικά των Γεωγραφικών Πληροφοριακών Συστημάτων (GIS) κάνοντας μια ιστορική αναδρομή σε αυτά, αναφέροντας τα χαρακτηριστικά και τα πλεονεκτήματά τους. Επίσης, περιγράφεται η υπηρεσία Google Maps που μας βοηθάει να οπτικοποιούμε τα γεωγραφικά δεδομένα και αποτελέσματά μας. Η βάση δεδομένων που χρησιμοποιήθηκε για την εφαρμογή είναι η MySQL η οποία περιγράφεται στη συνέχεια της επιστημονικής εργασίας. Επίσης, περιγράφεται ο σχεδιασμός της Βάσης δεδομένων αλλά και οι πίνακες της εφαρμογής. Τέλος, αναφέρεται στην αρχιτεκτονική της εφαρμογής η οποία βασίζεται στο μοντέλο «Πελάτη – Εξυπηρετητή»(Client-server), την παρουσίαση της και τα συμπεράσματα της επιστημονικής αυτής εργασίας.

Οι δύο εργασίες που συγκρίνονται εξετάζουν κάποια κοινά ερευνητικά και τεχνολογικά πεδία, όπως τα γεωγραφικά πληροφορικά συστήματα(GIS) και τις διαδικτυακές τεχνολογίες. Η παραπάνω εφαρμογή, χρησιμοποιεί την ίδια αρχιτεκτονική με αυτή της παρούσας μεταπτυχιακής διατριβής. Και στις δύο εφαρμογές χρησιμοποιείται ο Web Server Apache με

υποστήριξη της PHP και χρήση πλαισίου εργασίας. Επίσης, χρησιμοποιούνται από κοινού οι τεχνολογίες διαδικτυακού προγραμματισμού, όπως η HTML, Javascript και CSS.

Ωστόσο οι δύο αυτές εφαρμογές έχουν σε αρκετά πεδία διαφορετικούς σκοπούς. Πρώτα από όλα η πρώτη αναφέρεται σε μια εφαρμογή έξυπνου κινητού τηλεφώνου (Smart phone) λειτουργικού Android η οποία έχει ως στόχο να βρει στο χρήστη, σύμφωνα με την εκάστοτε γεωγραφική του θέση, τα κοντινά σημεία ενδιαφέροντος σε αυτόν. Η δεύτερη είναι μια διαδικτυακή εφαρμογή η οποία έχει ως στόχο να απεικονίσει στο χάρτη τα σημεία ενδιαφέροντος του χρήστη. Να σημειωθεί ότι η αναζήτηση αφορά σημεία που βρίσκονται στο νομό Λασιθίου. Μια άλλη διαφορά μεταξύ των δύο επιστημονικών εργασιών και εφαρμογών είναι η χρήση διαφορετικής βάσης δεδομένων. Η παρούσα εφαρμογή χρησιμοποιεί τη χωρική βάση δεδομένων PostgreSQL η οποία έχει άριστη απόδοση σε περιπτώσεις που έχουμε να κάνουμε με χωρικά δεδομένα. Η εφαρμογή που περιγράψαμε παραπάνω χρησιμοποιεί την MySQL, ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων πολύ γνωστό σε όλους μας.

Η επόμενη εργασία, όπως αναφέρουν οι Erlacher et al.(2012), ασχολείται με την ανάπτυξη δύο εφαρμογών βασιζόμενων στη γεωγραφική θέση (location-based), μία στο διαδίκτυο και μία για κινητά τηλέφωνα (Smartphones) οι οποίες σχεδιάστηκαν και υλοποιήθηκαν από μια συνεργασία μαθητών της δευτεροβάθμιας εκπαίδευσης με το Τμήμα Γεωπληροφορικής του Πανεπιστημίου Εφαρμοσμένων Επιστημών στην Καρινθία (CUAS). Αναφέρεται μια διαδικτυακή εφαρμογή για τη συλλογή και την ανταλλαγή σημείων ενδιαφέροντος (POI), που εκπροσωπούν τα ενδιαφέροντα των νέων.

Αρχικά, γίνεται μια εισαγωγή στην οποία αναφέρεται η ερευνητική διαδικασία αυτής της επιστημονικής εργασίας, τονίζοντας ότι έγγραφο αυτό έχει ως στόχο να δείξει την συνεργατική ανάπτυξη ενός χωρικού μοντέλου δεδομένων, σύμφωνα με τις ανάγκες των νέων ανθρώπων και τις συνακόλουθες βάσει γεωγραφικής θέσης (location-based) εφαρμογές σε κινητά τηλέφωνα. Στη συνέχεια περιγράφεται η διαδικασία συλλογής δεδομένων των σημείων ενδιαφέροντος και η διαδικασία σχεδίασης και μοντελοποίησης της βάσης δεδομένων η οποία χρησιμοποιείται από τις εφαρμογές. Επίσης περιγράφεται η διαδικασία συλλογής δεδομένων σχετικά με γεωγραφικά σημεία που προκαλούν ενδιαφέρον στους νέους. Η συλλογή όπως αναφέρεται επιτεύχθηκε με τη χρήση ερωτηματολογίων. Παρακάτω, περιγράφονται οι εφαρμογές που προσφέρουν τα LBS (Located-Base Services). Στο τρίτο μέρος της εργασίας περιγράφεται η αρχιτεκτονική των εφαρμογών που υλοποιήθηκαν η οποία βασίζεται στο μοντέλο «Πελάτη - Εξυπηρετητή» χρησιμοποιώντας και δεδομένα που προέρχονται από εξωτερικές πηγές. Στη συνέχεια αναφέρεται στην υλοποίηση της εφαρμογής σε έξυπνο κινητό τηλέφωνο με λειτουργικό Android 2.2. Η εφαρμογή αυτή παρέχει τρεις λειτουργικότητες. Η πρώτη δίνει την δυνατότητα υπολογισμού της διαδρομής ενός ταξιδιού με επιστροφή στο εκάστοτε επιλεγμένο σημείο ενδιαφέροντος. Η δεύτερη λειτουργία της εφαρμογής είναι να παρέχει στο χρήστη, ανάλογα με την εκάστοτε γεωγραφική του θέση, τις πιο κοντινές σε αυτόν «παμπ» απεικονίζοντας τα αποτελέσματα στην οθόνη του κινητού. Να σημειωθεί, ότι η εφαρμογή στο κινητό χρησιμοποιεί το GPS. Η τρίτη και τελευταία λειτουργία της εφαρμογής είναι η εμφάνιση της συντομότερης διαδρομής μέσω του χάρτη Google Maps VIEW.

Η παραπάνω εργασία αναφέρεται σε υπηρεσίες βάσει της θέσης του χρήστη. Στις ίδιες υπηρεσίες βασίζεται και η εφαρμογή της παρούσας μεταπτυχιακής διατριβής. Αρχικά και οι δύο εφαρμογές βασίζονται στο μοντέλο αρχιτεκτονικής «Πελάτη- Εξυπηρετητή».

Ωστόσο η εφαρμογή που περιγράφηκε παραπάνω χρησιμοποιεί μίαν άλλη διαδικτυακή εφαρμογή, όχι σε κινητό τηλέφωνο, με την οποία επικοινωνεί για να πάρει τα δεδομένα των σημείων ενδιαφέροντος όπου εισάγονται και ανταλλάσσονται από τους χρήστες. Επίσης, για την συλλογή δεδομένων χρησιμοποιήθηκαν ερωτηματολόγια σε αντίθεση με την εφαρμογή της παρούσας μεταπτυχιακής διατριβής όπου τα δεδομένα που χρησιμοποιήθηκαν υπήρχαν έτοιμα σε έναν ιστότοπο ο οποίος παρέχει δεδομένα σημείων ενδιαφέροντος για ακαδημαϊκούς σκοπούς. Από τη σκοπιά της τεχνολογίας που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής που περιγράψαμε παραπάνω, δεν δόθηκαν πληροφορίες παρά μόνο ότι

χρησιμοποιήθηκε η χωρική βάση δεδομένων PostgreSQL, την ίδια δηλαδή βάση δεδομένων που χρησιμοποιεί η παρούσα εφαρμογή.

Η επόμενη εργασία αποτελεί μεταπτυχιακή διατριβή με θέμα την ανάπτυξη μιας διαδικτυακής τουριστικής εφαρμογής για κινητά τηλέφωνα.

Προτείνεται μια κινητή διαδικτυακή εφαρμογή που παρέχει πληροφορίες και οδηγίες πλοήγησης σε τουριστικές υπηρεσίες μέσω κινητών συσκευών στους τελικούς χρήστες (Καρατζάς, 2010). Ο στόχος αυτής της εφαρμογής, είναι να διερευνήσει βασικές αρχές της διαδικτυακής τεχνολογίας κινητών τηλεφώνων και επίσης, την ανάκτηση πληροφοριών και τεχνικών εξόρυξης για την οικοδόμηση μιας τέτοιας διαδικτυακής εφαρμογής σαν κινητός τουριστικός οδηγός. Το κύριο θέμα αυτής της εφαρμογής είναι να προσφέρουν στους τουρίστες-χρήστες τη δυνατότητα να προσαρμόσουν το περιεχόμενο που παρουσιάζεται σε μια προσωπική επιλογή που πρόκειται να φορτωθεί σε μια προσαρμοσμένη εφαρμογή.

Στην εργασία αυτή, αρχικά γίνεται μια εισαγωγή σχετικά με την χρησιμότητα του διαδικτύου στα έξυπνα κινητά τηλέφωνα και τη χρήση που έχουν οι υπηρεσίες βασισμένες στη θέση του χρήστη σε διάφορες εφαρμογές. Στο δεύτερο μέρος αναλύονται οι έννοιες του κινητού διαδικτύου (Mobile Web), η αλληλεπίδραση του χρήστη με το κινητό τηλέφωνο και θέματα χρηστικότητας, οι βασικές αρχές των βάσεων θέσης χρήστη υπηρεσιών σε συνδυασμό με προσωπικές διαδικτυακές υπηρεσίες. Επίσης, αναφέρεται στη χρήση τεχνικών δημιουργίας συστάσεων για τουριστικά μέρη, τεχνικές ανίχνευσης κοντινών στη θέση του χρήστη σημείων. Στο τρίτο μέρος περιγράφονται τεχνικές σχεδιασμού της εφαρμογής, οι λειτουργίες που παρέχει, η αρχιτεκτονική της και η ανάλυση αλγορίθμων εξαγωγής πληροφοριών. Η αρχιτεκτονική που βασίζεται η εφαρμογή αυτή είναι το μοντέλο «Πελάτη - Εξυπηρετητή». Όλες οι λειτουργίες έχουν υλοποιηθεί με τη χρήση διαδικτυακού προγραμματισμού και γλώσσες σήμανσης (Mark-up), όπως η JavaScript, Ajax, HTML5, CSS και PHP. Επίσης, η εφαρμογή έχει σχεδιαστεί για να υποστηρίζεται από κινητές συσκευές που διαθέτουν περιηγητές (Browsers) βασισμένοι στο WebKit υποστηρίζονται HTML5 και την λειτουργία της τοπικής αποθήκευσης (Local Storage). Επίσης, η εφαρμογή χρησιμοποιεί αλγόριθμους για την πληροφόρηση του ωραρίου λειτουργίας κάθε σημείου ενδιαφέροντος και σε κάποιες περιπτώσεις τις τιμές αγαθών που παρέχει το καθένα.

Οι ομοιότητες της παραπάνω εφαρμογής που εξετάσαμε με την παρούσα εφαρμογή είναι ότι και οι δύο βασίζονται στις υπηρεσίες βάση της γεωγραφικής θέσης του χρήστη και αναφέρονται σε έξυπνα κινητά τηλέφωνα και βασίζονται στο μοντέλο αρχιτεκτονικής «Πελάτη – Εξυπηρετητή». Σκοπός της παραπάνω εφαρμογής είναι να παρέχει πληροφορίες και οδηγίες πλοήγησης σε τουριστικές υπηρεσίες μέσω κινητών συσκευών στους τελικούς χρήστες, απεικονίζοντας σημεία ενδιαφέροντος. Από την άλλη σκοπός της παρούσας εφαρμογής είναι η εύρεση σημείων ενδιαφέροντος τα οποία ταξινομούνται ανά κατηγορία τα οποία βρίσκονται σε μια κοντινή απόσταση από την γεωγραφική θέση του χρήστη παρέχοντας την δυνατότητα να επιλεχτεί η ακτίνα αναζήτησης σημείων.

Ωστόσο η παραπάνω εφαρμογή κάνει χρήση αλγορίθμων οι οποίοι βοηθούν στην εύρεση των ωραρίων λειτουργίας που έχουν τα σημεία που βλέπει στον χάρτη ο χρήστης κάθε φορά καθώς και οι τιμές προϊόντων ή υπηρεσιών που παρέχουν.

Από την σκοπιά της τεχνολογίας και οι δύο εφαρμογές χρησιμοποιούν διαδικτυακό προγραμματισμό με τη χρήση HTML5, Javascript, AJAX, JSON, PHP. Ωστόσο η παρούσα εφαρμογή κάνει χρήση πλαισίου εργασίας στη Javascript (jQuery Mobile, jQuery Ajax) και στην PHP (Codeigniter). Η παραπάνω εφαρμογή κάνει χρήση τοπικής διαδικτυακής SQL βάσης δεδομένων, σε αντίθεση με της παρούσα εφαρμογή που χρησιμοποιεί την χωρική βάση δεδομένων PostgreSQL.

Παρακάτω παρουσιάζονται δύο συγκεντρωτικοί πίνακες με τις ομοιότητες και τις διαφορές όλων των εργασιών σε σχέση με τη σύγκριση που έγινε παραπάνω, με την παρούσα διπλωματική διατριβή.

<b>Ομοιότητες εργασιών</b>	
Οικονόμου, 2010	Εξετάζουν και οι δύο μια εφαρμογή σε κινητό τηλέφωνο η οποία βάσει της γεωγραφικής θέσης του χρήστη παρουσιάζει τα πιο κοντινά του σημεία ενδιαφέροντος γύρω από μια συγκεκριμένα ακτίνα. Επίσης, η αρχιτεκτονική και των δύο εφαρμογών βασίζεται στο μοντέλο Πελάτη - Εξυπηρετητή. Οι γλώσσες προγραμματισμού που χρησιμοποιούνται στη πλευρά του εξυπηρετητή βασίζονται στο πρότυπο MVC. Ο Χρησιμοποιούν και οι δύο μορφοποίηση δεδομένων σε Javascript, τη τεχνολογία JSON. Ο χρήστης έχει τη δυνατότητα να καθορίζει την ακτίνα αναζήτησης των σημείων ενδιαφέροντος καθώς επίσης μπορεί να δει πληροφορίες για το κάθε σημείο. Και στις δύο εφαρμογές τα δεδομένα οπτικοποιούνται με τη χρήση του Google Maps.
Παπαδόπουλος, 2011	Αποτελούν εφαρμογές σε κινητό τηλέφωνο με λειτουργικό Android. Χρησιμοποιούν τη γεωγραφική θέση του χρήστη για να αναζητήσουν κοντινά σημεία γύρω από αυτόν. Για την οπτικοποίηση των σημείων γίνεται χρήση του Google Maps. Οι δύο εφαρμογές βασίζονται στο μοντέλο αρχιτεκτονικής Πελάτη-Διακομιστή και χρησιμοποιούν την ίδια γλώσσα προγραμματισμού στη πλευρά του δεύτερου, την PHP.
Σουέρεφ & Αργυρόπουλος, 2012	Χρησιμοποιούνται και περιγράφονται τεχνολογίες διαδικτυακού προγραμματισμού. Γίνεται περιγραφή και χρήση του JSON. Αναλύεται και περιγράφεται η αρχιτεκτονική του Android.
Loi, 2010	Αποτελούν εφαρμογές κινητών τηλεφώνων. Βασίζονται στο μοντέλο αρχιτεκτονικής Πελάτη - Εξυπηρετητή. Στην πλευρά του εξυπηρετητή χρησιμοποιείται η γλώσσα προγραμματισμού PHP με χρήση πλαισίου εργασίας(framework) βασιζόμενο στο MVC. Επίσης, γίνεται χρήση τεχνολογιών διαδικτυακού προγραμματισμού.
Κοντόπουλος, 2010	Χρησιμοποιούνται και στις δύο εφαρμογές γεωγραφικά πληροφοριακά συστήματα. Χρησιμοποιούν τεχνολογίες διαδικτυακού προγραμματισμού. Η αρχιτεκτονική τους βασίζεται στο μοντέλο Πελάτη - Εξυπηρετητή. Και οι δύο εφαρμογές κάνουν χρήση της χωρικής βάσης δεδομένων PostgreSQL με επέκταση της PostGIS. Χρησιμοποιείται ο Apache Web Server με υποστήριξη της PHP.
Μπούρχας, 2012	Υλοποιούν μια εφαρμογή σε κινητό τηλέφωνο βασισμένη σε υπηρεσίες που βασίζονται στην γεωγραφική θέση του χρήστη, πραγματοποιώντας αναζήτηση των κοντινότερων σημείων ενδιαφέροντος. Και οι δυο εφαρμογές αναπτύχθηκαν για λειτουργικό Android και η αρχιτεκτονική τους βασίζεται στο μοντέλο Πελάτη - Εξυπηρετητή. Στη πλευρά του εξυπηρετητή χρησιμοποιείται ο εξυπηρετητής Apache με υποστήριξη της γλώσσας προγραμματισμού PHP.
Μανδουράρη, 2011	Χρησιμοποιούνται και στις δύο εφαρμογές γεωγραφικά πληροφοριακά συστήματα. Χρησιμοποιούνται τεχνολογίες διαδικτυακού προγραμματισμού όπως η Javascript, η γλώσσα σήμανσης HTML και CSS3. Η αρχιτεκτονική των δύο εφαρμογών βασίζεται στο μοντέλο Πελάτη - Εξυπηρετητή. Στη πλευρά του εξυπηρετητή λειτουργεί ο Web Server Apache με υποστήριξη της γλώσσας προγραμματισμού PHP η οποία χρησιμοποιεί πλαίσιο εργασίας(framework) στηριζόμενο στο MVC.
Erlacher et al., 2012	Βασίζονται και οι δύο εφαρμογές στις υπηρεσίες βάσει γεωγραφικής θέσης(location-based services) και υλοποιούνται σε κινητά τηλέφωνα με λειτουργικό Android. Χρησιμοποιούν τη χωρική βάση δεδομένων PostgreSQL.

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Καρατζάς, 2010	Βασίζονται και οι δύο εφαρμογές στις υπηρεσίες βάσει γεωγραφικής θέσης(location-based services) και υλοποιούνται σε κινητά τηλέφωνα. Χρησιμοποιούν και οι δύο εφαρμογές στις υπηρεσίες βάσει γεωγραφικής θέσης(location-based services) και υλοποιούνται σε κινητά τηλέφωνα. Η αρχιτεκτονική τους βασίζεται στο μοντέλο Πελάτη - Εξυπηρετητή. Χρησιμοποιούν τεχνολογίες διαδικτύου και γλώσσες διαδικτυακού προγραμματισμού. Στη πλευρά του Εξυπηρετητή χρησιμοποιείται η γλώσσα προγραμματισμού PHP.
----------------	---

Διαφορές εργασιών	
Οικονόμου, 2010	Η παρούσα εφαρμογή υλοποιήθηκε με τη χρήση του Phonegap και διαδικτυακών τεχνολογιών όπως η HTML, Javascript, CSS, JQuery AJAX και στη πλευρά του Εξυπηρετητή χρησιμοποιείται η γλώσσα προγραμματισμού PHP με το πλαίσιο εργασίας Codeigniter. Η εφαρμογή που περιγράφηκε υλοποιήθηκε με τη χρήση της γλώσσας Python και το πλαίσιο εργασίας Django. Η βάση δεδομένων που χρησιμοποιήθηκε στην πρώτη εφαρμογή είναι η χωρική Postgresql σε αντίθεση με τη δεύτερη η οποία χρησιμοποιεί τη MySQL. Επίσης, η πρώτη εφαρμογή υλοποιήθηκε για λειτουργικό Android σε αντίθεση με τη δεύτερη η οποία υλοποιήθηκε για λειτουργικό IOS. Η παρούσα εφαρμογή δίνει στο χρήστη τη δυνατότητα πρόσθεσης σημείων ενδιαφέροντος. Ωστόσο, η εφαρμογή που περιγράφηκε δίνει τη δυνατότητα στο χρήστη να ταξινομεί τις πληροφορίες των σημείων ενδιαφέροντος και τη δυνατότητα αλλαγής της μορφής του χάρτη. Στη πρώτη εφαρμογή έγινε ανάλυση και σχεδιασμός με χρήση UML ακολουθώντας τη μέθοδο RUP πριν την υλοποίησή της.
Παπαδόπουλος, 2011	Στην παρούσα εφαρμογή χρησιμοποιείται η χωρική βάση δεδομένων PostgreSQL σε αντίθεση με αυτή που περιγράφηκε η οποία χρησιμοποιεί τη MySQL. Η πρώτη εφαρμογή υλοποιήθηκε με πλαίσιο εργασίας(framework) στη PHP, ενώ η δεύτερη δεν χρησιμοποιεί κάποιο πλαίσιο εργασίας. Στη πλευρά του πελάτη η πρώτη εφαρμογή αναπτύχθηκε με τη χρήση του Phonegap και διαδικτυακών τεχνολογιών όπως η HTML, Javascript, JQuery, CSS3. Η δεύτερη εφαρμογή αναπτύχθηκε με τη γλώσσα προγραμματισμού Java. Και οι δύο εφαρμογές δίνουν τη δυνατότητα στο χρήστη να προσθέσει σημεία ενδιαφέροντος, ωστόσο στη πρώτη, ο χρήστης προσθέτει σημεία μέσω συμπλήρωσης μιας φόρμας στο κινητό τηλέφωνο, ενώ στη δεύτερη ο χρήστης έχει τη δυνατότητα να το κάνει μέσω ιστοσελίδας.
Σουρέφ & Αργυρόπουλος, 2012	Στόχος της παρούσας μεταπτυχιακής διατριβής είναι η ανάλυση, σχεδίαση και υλοποίηση μιας εφαρμογής σε κινητό τηλέφωνο με τη χρήση γεωγραφικού πληροφοριακού συστήματος. Η διπλωματική εργασία που περιγράφηκε παραπάνω έχει ως στόχο να αναλύσει και να εξετάσει τη διαδικασία ανάπτυξης μιας εφαρμογής σε Android.
Loi, 2010	Στόχος της παρούσας μεταπτυχιακής διατριβής είναι η ανάλυση, σχεδίαση και υλοποίηση μιας εφαρμογής σε κινητό τηλέφωνο με τη χρήση γεωγραφικού πληροφοριακού συστήματος, σε αντίθεση με αυτή που περιγράφηκε παραπάνω η οποία αναλύει την αρχιτεκτονική πληροφοριακών συστημάτων για σύγχρονα κινητά τηλέφωνα. Επίσης, η παρούσα εφαρμογή υλοποιήθηκε με τη χρήση Phonegap ενώ η δεύτερη με τη χρήση της αντικειμενοστρεφούς γλώσσας προγραμματισμού Java

Κοντόπουλος, 2010	Η παρούσα εργασία υλοποιεί μια εφαρμογή για κινητά τηλέφωνα Android σε αντίθεση με αυτή που περιγράφηκε παραπάνω, που υλοποιεί μια διαδικτυακή εφαρμογή. Και οι δύο εφαρμογές αναπτύχθηκαν με τη χρήση της γλώσσας PHP. Ωστόσο, στην παρούσα εφαρμογή χρησιμοποιήθηκε και το πλαίσιο εργασίας Codeigniter.
Μπούρχας, 2012	Στόχος της παρούσας εφαρμογής είναι η αναζήτηση σημείων ενδιαφέροντος γύρω από τη γεωγραφική θέση του χρήστη. Η εφαρμογή που περιγράφηκε παραπάνω έχει ως στόχο της τον εντοπισμό σημείων ενδιαφέροντος τα οποία αποτελούν καταστήματα και έχουν την εκάστοτε ημερομηνία προσφορές στα προϊόντα ή τις υπηρεσίες που πωλούν. Στη πρώτη εφαρμογή η προσθήκη σημείων γίνεται μέσω συμπλήρωσης φόρμας στο κινητό τηλέφωνο, ενώ στη δεύτερη, η προσθήκη σημείων γίνεται μέσω ιστοσελίδας. Η τεχνολογία με την οποία αναπτύχθηκε η πρώτη εφαρμογή στη πλευρά του Πελάτη είναι το Phonegap, ενώ η δεύτερη αναπτύχθηκε με τη χρήση της Java. Τέλος, η παρούσα εφαρμογή χρησιμοποιεί ως βάση δεδομένων τη PostgreSQL σε αντίθεση με αυτή που περιγράφηκε, η οποία χρησιμοποιεί τη MySQL.
Μανδουράρη, 2011	Στόχος της παρούσας εφαρμογής είναι η αναζήτηση σημείων ενδιαφέροντος γύρω από τη γεωγραφική θέση του χρήστη. Η εφαρμογή που περιγράφηκε παραπάνω υλοποιεί μια διαδικτυακή εφαρμογή η οποία απεικονίζει στο χρήστη σημεία ενδιαφέροντος που βρίσκονται στο νομό Λασιθίου. Η βάση δεδομένων που χρησιμοποιεί η πρώτη εφαρμογή είναι η PostgreSQL σε αντίθεση με τη δεύτερη η οποία χρησιμοποιεί τη MySQL.
Erlacher et al., 2012	Η εργασία που περιγράφηκε παραπάνω εκτός από μια εφαρμογή σε κινητό τηλέφωνο, υλοποιεί και μια διαδικτυακή εφαρμογή, στην οποία οι χρήστες προσθέτουν σημεία ενδιαφέροντος και τη χρησιμοποιεί η εφαρμογή του κινητού για να λάβει αυτά τα δεδομένα. Για τη συλλογή των δεδομένων στη παρούσα εφαρμογή χρησιμοποιήθηκε μια ιστοσελίδα η οποία παρέχει σημεία ενδιαφέροντος για κάθε χώρα και χρησιμοποιείται για ακαδημαϊκούς σκοπούς. Στη δεύτερη εφαρμογή, χρησιμοποιήθηκαν ερωτηματολόγια σε μαθητές της δευτεροβάθμιας εκπαίδευσης για τη συλλογή δεδομένων.
Καρατζάς, 2010	Στόχος της παρούσας εφαρμογής είναι η αναζήτηση σημείων ενδιαφέροντος γύρω από τη γεωγραφική θέση του χρήστη. Η εφαρμογή που περιγράφηκε παραπάνω έχει ως στόχο την παροχή πληροφοριών και οδηγιών πλοήγησης σε τουριστικές υπηρεσίες μέσω κινητού τηλεφώνου απεικονίζοντας τα σημεία ενδιαφέροντος. Επίσης, στη δεύτερη εφαρμογή χρησιμοποιούνται αλγόριθμοι που βοηθούν στην εύρεση των ωραρίων λειτουργίας των σημείων ενδιαφέροντος καθώς και τις τιμές των προϊόντων η υπηρεσιών που παρέχουν.

Σε αυτό το κεφάλαιο πραγματοποιήθηκε μια συγκριτική έρευνα για εργασίες που έχουν υλοποιηθεί οι οποίες σχετίζονται με τη παρούσα μεταπτυχιακή διατριβή. Το συμπέρασμα είναι ότι η παρούσα εφαρμογή περιέχει μια πρωτότυπη τεχνολογική υλοποίηση σε συνάρτηση με το στόχο που υλοποιεί αλλά και με την μέθοδο ανάλυσης και σχεδίασης που ακολουθεί. Στο σημείο αυτό πρέπει να σημειωθεί, ότι σε αυτό το κεφάλαιο αναφέρθηκαν αρκετές τεχνολογικές ορολογίες και γλώσσες προγραμματισμού σχετικά με την παρούσα εφαρμογή οι οποίες περιγράφονται και αναλύονται στο επόμενο κεφάλαιο, για να κατανοηθεί όχι μόνο ο ορισμός και ο σκοπός τους, αλλά και η τεχνολογική σημασία της χρήσης τους στην εφαρμογή.

## ΚΕΦΑΛΑΙΟ 3

### Τεχνολογίες υλοποίησης

Τα κινητά τηλέφωνα, ειδικότερα τα τελευταίας γενιάς (smartphones), σε συνδυασμό με το διαδίκτυο(Internet) έχουν φέρει επαναστατικές αλλαγές στην επικοινωνία και στην ζωή των ανθρώπων. Καθημερινά αυξάνονται ολοένα και περισσότερο οι χρήστες των Smart Phones, τα οποία δίνουν την δυνατότητα πρόσβασης στο Internet από όπου και αν βρίσκονται, οποιασδήποτε στιγμή το θελήσουν. Συνεπώς, τα τελευταία χρόνια χιλιάδες εφαρμογές έχουν αναπτυχθεί για τα νέα αυτά έξυπνα κινητά τηλέφωνα. Πλέον τα κινητά δεν αντιμετωπίζονται σαν απλές συσκευές τηλεφωνικής επικοινωνίας ή αποστολής γραπτών μηνυμάτων, αλλά ως εργαλεία παροχής πιο εξειδικευμένων υπηρεσιών εστιασμένων στις προσωπικές ανάγκες του κάθε χρήστη.

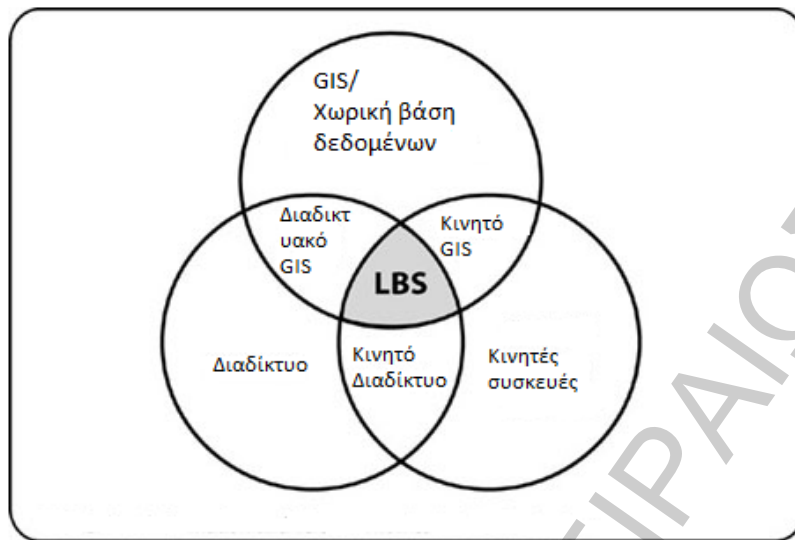
Με τη χρήση του Internet στα κινητά τηλέφωνα ο χρήστης έχει τη δυνατότητα να αποκτήσει πληροφορίες σχετικά με γεγονότα που συμβαίνουν κάθε στιγμή αλλά και για τοποθεσίες σημείων που τον ενδιαφέρουν μέσω της χρήσης χάρτη. Πάνω στην μεγάλη ανάγκη των χρηστών να αποκτήσουν τέτοιου είδους πληροφορίες αναπτύχθηκαν οι Υπηρεσίες εντοπισμού θέσης(Location-Based Services).

Η εφαρμογή που υλοποιήθηκε στα πλαίσια της παρούσας μεταπτυχιακής διατριβής βασίζεται στις υπηρεσίες εντοπισμού θέσης σε συνδυασμό με τη χρήση ενός Γεωγραφικού Πληροφοριακού Συστήματος(GIS). Στο κεφάλαιο αυτό θα αναλυθεί η κάθε τεχνολογία και υπηρεσία που χρησιμοποιήθηκε για να υλοποιηθεί η εφαρμογή αυτή.

### ΟΡΙΣΜΟΣ ΥΠΗΡΕΣΙΩΝ ΕΝΤΟΠΙΣΜΟΥ ΘΕΣΗΣ

Οι υπηρεσίες εντοπισμού θέσης (Location Based Services) αποτελούν ένα σύνολο εφαρμογών και υπηρεσιών που έχουν ως στόχο την παροχή εξατομικευμένων υπηρεσιών προς τελικούς χρήστες βάσει της γεωγραφικής τους θέσης. Κάθε άνθρωπος θέλει να γνωρίζει όσο το δυνατόν περισσότερα για τον χώρο στον οποίο βρίσκεται και κινείται. Επίσης ενδιαφέρεται να ξέρει για το που βρίσκονται τα αγαπημένα του άτομα. Όλες οι παραπάνω “ανάγκες” έρχονται να καλυφθούν από αυτές τις υπηρεσίες. Το μεγαλύτερο κίνητρο για την εισαγωγή τέτοιου είδους υπηρεσιών αποτέλεσε η νομική επιβολή στις ΗΠΑ έτσι ώστε μέχρι το τέλος του 2005 όλες οι καινούργιες συσκευές να μπορούν να παρέχουν πληροφορία θέσης όταν πραγματοποιείται μια επείγουσα κλήση. Πιο απλά για να κατανοήσουμε το στόχο των Υπηρεσιών αυτών ας λάβουμε υπόψη ένα παράδειγμα:

“Εστω ότι ένας χρήστης βρίσκεται στην Αθήνα, στην περιοχή του κέντρου και θέλει να βρει κάποιο φαρμακείο και δεν ξέρει που βρίσκεται το κοντινότερο του. Χρησιμοποιώντας μια εφαρμογή η οποία βασίζεται στα LBS έχει την δυνατότητα να βρει το κοντινότερο σε αυτόν φαρμακείο οπτικοποιημένο στον χάρτη, παρέχοντας του οδηγίες για το πώς θα φτάσει μέχρι εκεί. Πώς γίνεται αυτό; Οι υπηρεσίες αυτές δίνουν τη γεωγραφική θέση του κινητού άρα και τη θέση του χρήστη ακολουθώντας μια διαδικασία η εφαρμογή εντοπίζει σύμφωνα με την εκάστοτε θέση του τα σημεία που βρίσκονται κοντά του και τον ενδιαφέρουν (Points of Interest-POIs). Συμπεραίνουμε λοιπόν ότι οι υπηρεσίες εντοπισμού θέσης επεξεργάζονται δεδομένα σχετικά με τη τοποθεσία απαντώντας σε ερωτήματα όπως, που βρίσκομαι αυτή την στιγμή; Τι σημείο ενδιαφέροντος υπάρχει γύρω από τη θέση μου; Πώς θα μπορέσω να φτάσω μέχρι εκεί;



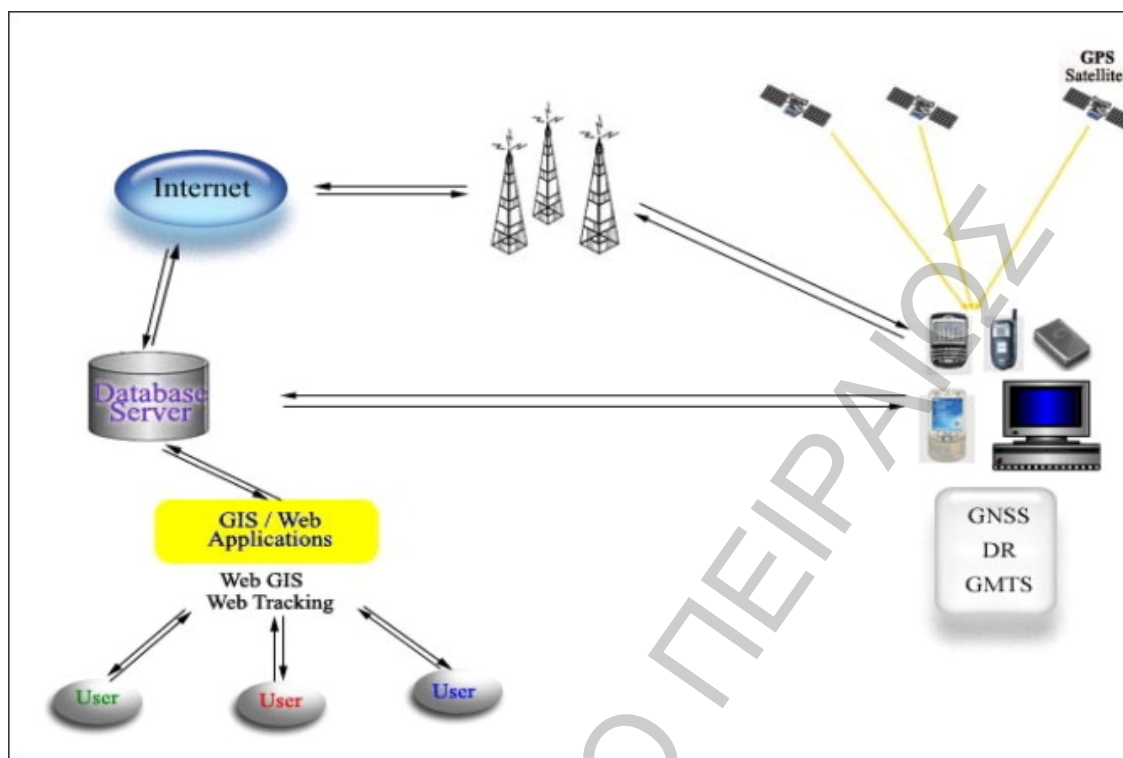
Εικόνα 2

### ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ ΥΠΗΡΕΣΙΩΝ ΕΝΤΟΠΙΣΜΟΥ ΘΕΣΗΣ

Όπως αναφέρθηκε και παραπάνω οι υπηρεσίες εντοπισμού θέσης λειτουργούν με τη χρήση GPS και το δίκτυο τηλεπικοινωνιών για να λάβουν πληροφορίες σχετικά με γεωγραφικές θέσεις, στέλνοντας τις στην πλευρά του Εξυπηρετητή (Server) για περαιτέρω επεξεργασία. Η παρούσα εφαρμογή, χρησιμοποιεί το GPS του κινητού τηλεφώνου αλλά και το Internet παρέχοντας τις πληροφορίες θέσης. Ποια είναι όμως τα συστατικά μέρη μιας υπηρεσίας εντοπισμού θέσης;

Αρχικά, το πρώτο συστατικό είναι μια ασύρματη συσκευή η οποία αποτελεί τον χρήστη (User). Το δεύτερο συστατικό είναι το δίκτυο κινητής τηλεφωνίας μέσω του οποίου γίνεται η μεταφορά των δεδομένων από το χρήστη και το αίτημα του από την ασύρματη συσκευή προς τον παροχέα υπηρεσίας και ακολούθως η αιτούμενη πληροφορία από τον παροχέα πίσω στο χρήστη. Το τρίτο συστατικό μέρος είναι η μονάδα εντοπισμού θέσης. Για να εντοπιστεί η θέση του χρήστη είτε πρέπει να γίνει χρήση του δικτύου κινητής τηλεφωνίας είτε το GPS της κινητής συσκευής του χρήστη είτε η χρήση WLAN στο οποίο είναι συνδεδεμένη η συσκευή. Το επόμενο συστατικό είναι ο παροχέας υπηρεσίας και εφαρμογής ο οποίος είναι υπεύθυνος για να προσφέρει τις διάφορες υπηρεσίες και λειτουργικότητες στο χρήστη και να επεξεργάζεται τα αιτήματά του. Παρακάτω το σχήμα απεικονίζει την αρχιτεκτονική μιας υπηρεσίας εντοπισμού θέσης (LBS) η οποία αποτελείται από τα συστατικά που περιγράψαμε παραπάνω.





Εικόνα 3 (Sadoun and Al-Bayari, 2007)

Στόχος κάθε υπηρεσίας εντοπισμού θέσης είναι η εύρεση των συντεταγμένων του χρήστη ώστε μετά από επεξεργασία της γεωγραφικής θέσης του, να έχει τα ανάλογα αποτελέσματα σχετικά με τα σημεία ενδιαφέροντος που υπάρχουν κοντά του. Αυτό πραγματοποιείται με τη γεωκωδικοποίηση (Geocoding), η οποία αποτελεί την ανάθεση συντεταγμένων θέσης σε κάποιο ορόσημο ή σημείο ενδιαφέροντος με βάση μια δοσμένη διεύθυνση. Ο σκοπός είναι να δοθεί μια λίστα με υποψήφιας θέσεις που η γεωγραφική τους τοποθεσία ταιριάζει ή είναι αρκετά κοντά στην γεωγραφική τοποθεσία της δοθείσας διεύθυνσης. Πολλαπλές διευθύνσεις μπορούν να υποβληθούν σε ένα ξεχωριστό έγγραφο αίτησης. Είναι εφικτό να υποδειχθεί ο αριθμός των σημείων που ταιριάζουν στην απόκριση για μια συγκεκριμένη διεύθυνση που εμπειρεχόταν στο αίτημα geocoding. Το geocoding μετατρέπει συνηθισμένες εγγραφές δεδομένων που περιέχουν διευθύνσεις σε γεωγραφικά αντικείμενα που μπορούν να προβληθούν σε ένα χάρτη.

## ΛΟΓΙΣΜΙΚΟ ANDROID

Το Android είναι ένα λειτουργικό σύστημα σχεδιασμένο ώστε να προσφέρει την μέγιστη παραμετροποίηση στα κινητά τηλέφωνα, ανάλογα με τις επιθυμίες του χρήστη. Προσφέρει στους καταναλωτές τη δυνατότητα να χρησιμοποιήσουν εξελιγμένες υπηρεσίες του Διαδικτύου, όπως ανταλλαγή μηνυμάτων ηλεκτρονικού ταχυδρομείου, πρόσβαση σε σελίδες κοινωνικής δικτύωσης και φυσικά πλοήγηση στον Παγκόσμιο Ιστό, χωρίς να αμελεί το τομέα ψυχαγωγία. Επίσης, υπόσχεται πληθώρα εφαρμογών που θα βελτιώσουν την εμπειρία της αναπαραγωγής μουσικής, της διαχείρισης αρχείων, της λήψης φωτογραφιών και βίντεο αλλά και της επικοινωνίας μέσω SMS, MMS και φωνητικών κλήσεων. Το Android βασίζεται στον πυρήνα του λειτουργικού συστήματος Linux και πλέον αναπτύσσεται από την Open Handset Alliance (OHA), μια σύμπραξη εταιρειών - κολοσσών στον τομέα της τεχνολογίας, όπως η Google, η Motorola, η HTC, η Samsung, η LG, η Qualcomm, η Sprint, η T-Mobile κ.α. οι οποίες το Νοέμβριο του 2007 έφεραν στο φως το νέο αυτό λειτουργικό σύστημα. Η Google δημοσίευσε το

μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού, γεγονός που επιτρέπει σε πλήθος προγραμματιστών να σχεδιάσουν εφαρμογές για το Android με τη χρήση της γλώσσας προγραμματισμού Java. Οι εφαρμογές αυτές μπορούν να μεταφορτωθούν στα κινητά τηλέφωνα από τον δικτυακό τόπο Android Market (Android).

## ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ANDROID

Τα χαρακτηριστικά του Android κατηγοριοποιούνται ως προς τις λειτουργίες της οθόνης, την αποθήκευση δεδομένων, τη συνδεσιμότητα, την περιήγηση στο διαδίκτυο, την υποστήριξη πολυμέσων και το περιβάλλον ανάπτυξης λογισμικού. Για την περιήγηση στον ιστό, το Android διαθέτει έναν φυλλομετρητή βασισμένο στην ανοιχτή τεχνολογία WebKit. Λογισμικό το οποίο είναι γραμμένο σε Java είναι δυνατόν να μεταγλωττιστεί και να εκτελεστεί στην εικονική μηχανή Dalvik, η οποία είναι μια εξειδικευμένη υλοποίηση εικονικής μηχανής, σχεδιασμένη για χρήση σε φορητές συσκευές, παρόλο που δεν είναι μια πρότυπη εικονική μηχανή Java. Το λειτουργικό Android υποστηρίζει διάφορες μορφές ήχου, στατικής και κινούμενης εικόνας όπως H.263, H.264 (σε 3GP ή MP4), MPEG-4SP, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, OGG-Vorbis, WAV, JPEG, PNG, GIF, BMP. Το λειτουργικό Android μπορεί να συνεργαστεί με κάμερες στατικής ή κινούμενης εικόνας, οθόνες αφής, GPS, αισθητήρες επιτάχυνσης, μαγνητόμετρα, δισδιάστατους καθώς και τρισδιάστατους επιταχυντές γραφικών. Ως προς το περιβάλλον ανάπτυξης λογισμικού περιλαμβάνει έναν προσομοιωτή συσκευής, εργαλεία για διόρθωση σφαλμάτων, μνήμη και εργαλεία ανάλυσης της απόδοσης του εκτελέσιμου λογισμικού καθώς και ένα επιπρόσθετο για το Eclipse IDE.

## ΑΡΧΙΤΕΚΤΟΝΙΚΗ ANDROID

Το παρακάτω σχήμα απεικονίζει το διάγραμμα της αρχιτεκτονικής Android. Το Android OS μπορεί να αναφέρεται ως μία στοίβα λογισμικού από διαφορετικές στιβάδες, όπου κάθε στρώμα είναι μία ομάδα αρκετών στοιχείων του προγράμματος. Μαζί περιλαμβάνουν το λειτουργικό σύστημα, ενδιάμεσο λογισμικό και σημαντικές εφαρμογές. Κάθε στρώμα στην αρχιτεκτονική παρέχει διαφορετικές υπηρεσίες στο στρώμα που βρίσκεται ακριβώς πάνω από το προηγούμενο.



Εικόνα 4

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Από ότι φαίνεται λοιπόν η αρχιτεκτονική του λειτουργικού συστήματος αποτελείται από 5 βασικά επίπεδα. Τον πυρήνα Linux (Linux Kernel), τις εγγενείς και τις προηγμένες βιβλιοθήκες (Libraries), την εικονική μηχανή Dalvik (Dalvik VM), το χρόνο εκτέλεσης (Android Runtime), το πλαίσιο εφαρμογής (Application Framework).

#### **Πυρήνας του Linux(Linux Kernel):**

Η βάση της στοίβας λογισμικού του Android είναι ο πυρήνας του Linux. Το σύνολο Android OS είναι χτισμένο στην κορυφή του πυρήνα Linux 2.6 με κάποιες περαιτέρω αρχιτεκτονικές αλλαγές που έγιναν από την Google. Αλληλεπιδρά με το υλικό και περιέχει όλα τα απαραίτητα προγράμματα οδήγησης υλικού. Οι οδηγοί είναι προγράμματα που ελέγχουν και επικοινωνούν με το υλικό. Ο πυρήνας του Linux λειτουργεί επίσης ως ένα επίπεδο αφαίρεσης μεταξύ του υλικού και άλλα στρώματα του λογισμικού. Το Android χρησιμοποιεί το Linux για όλες τις βασικές λειτουργίες του, όπως η διαχείριση μνήμης, διαχείριση διαδικασιών, τη δικτύωση, την ασφάλεια κ.λπ. Το γεγονός ότι το Android είναι χτισμένο σε ένα πιο δημοφιλές λειτουργικό σύστημα, έκανε το porting του Android με την ποικιλία του υλικού, ένα σχετικά ανώδυνο έργο.

#### **Βιβλιοθήκες:**

Στο δεύτερο επίπεδο της στοίβας υπάρχουν οι βιβλιοθήκες του Android. Είναι το στρώμα που επιτρέπει στη συσκευή να χειρίζεται διαφορετικούς τύπους δεδομένων. Αυτές οι βιβλιοθήκες είναι γραμμένες σε C ή C++ και είναι ειδικά για ένα συγκεκριμένο υλικό.

Μερικές από τις κύριες βιβλιοθήκες του Android είναι η Surface Manager η οποία διαχειρίζεται την πρόσβαση στο υποσύστημα προβολής, και συνθέτει απρόσκοπτα δισδιάστατα και τρισδιάστατα επίπεδα γραφικών τα οποία προέρχονται από πολλαπλές εφαρμογές. Το Media πλαίσιο εργασίας το οποίο προβλέπει διαφορετικούς κωδικοποιητές πολυμέσων που επιτρέπουν την καταγραφή και αναπαραγωγή των διαφόρων μορφών των μέσων ενημέρωσης όπως: MPEG4, H.264, MP3, AAC, AMR, JPG και PNG. Ο SQLite, αποτελεί το μηχανισμό διαχείρισης βάσεων δεδομένων που χρησιμοποιείται στο android για σκοπούς αποθήκευσης δεδομένων. Το WebKit αποτελεί την μηχανή περιήγησης που χρησιμοποιείται για την εμφάνιση περιεχομένου HTML. Η OpenGL, η γνωστή μηχανή δισδιάστατων(2D) και τρισδιάστατων(3D) γραφικών.

#### **Η εικονική μηχανή Dalvik:**

Σχεδόν το σύνολο των APIs του Android βασίζονται στη γλώσσα προγραμματισμού Java. Στην Java ως γνωστόν υπάρχει η λεγόμενη Java Virtual Machine στην οποία εκτελείται ο κώδικας bytecode των εφαρμογών. Στο Android υπάρχει κάτι παρόμοιο και δεν είναι άλλο από την εικονική μηχανή Dalvik.

Η Dalvik λοιπόν είναι η εικονική μηχανή μέσω της οποίας τρέχουν οι εφαρμογές του Android. Η κάθε εφαρμογή τρέχει μέσω τις δικής της εικονικής μηχανής στη δικιά της διεργασία και για αυτό το λόγο καμία εφαρμογή δεν έχει επαφή με την άλλη, ενώ εκτελούνται ταυτόχρονα. Η Dalvik δεν υποστηρίζει τον κώδικα bytecode, αντί αυτού οι κλάσεις της Java μεταγλωττίζονται σε αρχεία .dex ώστε να τρέξουν στην VM. Τα αρχεία dex ουσιαστικά αποτελούν συμπιεσμένα δεδομένα για εξοικονόμηση χώρου κατά την εκτέλεση. Η Dalvik VM επιτρέπει πολλαπλές χρήσεις της εικονικής μηχανής για να δημιουργηθούν ταυτόχρονα παροχές ασφάλειας, απομόνωση, διαχείριση μνήμης και threading υποστήριξη.

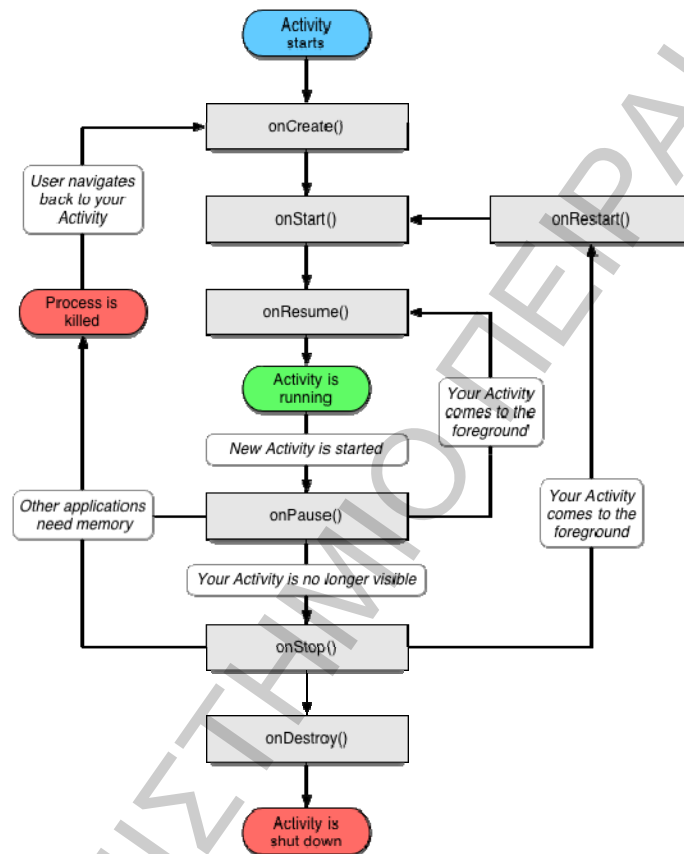
#### **Πλαίσιο Εφαρμογής (Application Framework):**

Τα προγράμματα αυτά διαχειρίζονται τις βασικές λειτουργίες του τηλεφώνου, όπως τη διαχείριση των πόρων, τη φωνητική διαχείριση κλήσεων κλπ. Οι προγραμματιστές έχουν στην διάθεση τους τη δυνατότητα ελέγχου του υλικού της συσκευής και μέσω αυτής μπορούν να αποκτήσουν πρόσβαση σε υπηρεσίες εντοπισμού, εκτέλεση διεργασιών παρασκήνιου, και πάρα πολλές ακόμη δυνατότητες οι οποίες βασίζονται στα APIs που είναι διαθέσιμα.

Σημαντικά τμήματα του πλαισίου εφαρμογής είναι ο διαχειριστής δραστηριότητας (Activity Manager) όπου διαχειρίζεται τον κύκλο ζωής των δραστηριοτήτων των εφαρμογών. Οι πάροχοι περιεχομένου (Content Providers) οι οποίοι διαχειρίζονται την ανταλλαγή δεδομένων μεταξύ

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

των εφαρμογών. Ο διαχειριστής τηλεφώνου (Telephony Manager) όπου διαχειρίζεται όλες τις φωνητικές κλήσεις. Ο διαχειριστής θέσης (Location Manager) όπου χρησιμοποιείται το GPS ή cell tower για τον εντοπισμό θέσης και γενικότερα για την διαχείριση της εκάστοτε γεωγραφικής τοποθεσίας. Ο διαχειριστής πόρων όπου παρέχει πρόσβαση σε υλικό το οποίο δεν είναι σε μορφή κώδικα όπως πχ, εικόνες, αρχεία XML, πίνακες χαρακτήρων, κλπ. Η εικόνα 5 απεικονίζει τον κύκλο ζωής δραστηριότητας Android



**Εικόνα 5**

Παραπάνω περιγράφηκε η αρχιτεκτονική και η τεχνολογία του λογισμικού Android. Αποτελεί το λειτουργικό σύστημα στο οποίο αναπτύχθηκε η εφαρμογή της παρούσας μεταπτυχιακής διατριβής. Ο λόγος που επιλέχθηκε είναι γιατί αποτελεί μια ανοιχτή και ελεύθερη πλατφόρμα ανάπτυξης. Διαθέτει αρχιτεκτονική βασισμένη σε δομικά στοιχεία τα οποία μπορούν να τροποποιηθούν, να ολοκληρωθούν και να προσαρμοστούν στις ανάγκες κάθε προγραμματιστή και κατά συνέπεια χρήστη. Επιτρέπει την αυτόματη διαχείριση του κύκλου ζωής μιας εφαρμογής, με πολλαπλές δικλίδες ασφαλείας ανάμεσα στα προγράμματα και προσφέρει βελτιστοποιήσεις στον τομέα διαχείρισης μνήμης και χαμηλής κατανάλωσης σε τέτοιο βαθμό που δεν έχει ξανασυναντηθεί σε άλλο έξυπνο κινητό τηλέφωνο.

## ECLIPSE & ANDROID SDK TOOLS

Το Eclipse αποτελεί ένα SDK (Software Development Kit), ένα ολοκληρωμένο δηλαδή περιβάλλον μέσα από το οποίο μπορούμε να γράψουμε και να εκτελέσουμε κώδικα. Δηλαδή είναι μία πλατφόρμα ανάπτυξης λογισμικών η οποία προσφέρει την δυνατότητα χρήσης πολλών γλωσσών, και το πιο κύριο χαρακτηριστικό του είναι το ότι συμπεριλαμβάνει συστήματα Integrated Development Environment (IDE) και πρόσθετα για επέκταση της υφιστάμενης πλατφόρμας από τους ίδιους τους χρήστες. Το Eclipse είναι γραμμένο σε Java χρησιμοποιείται για την ανάπτυξη εφαρμογών τόσο σε Java, όσο και σε άλλες γλώσσες, με την βοήθεια των πρόσθετων, όπως C, C++, Cobol, Python, Perl, PHP και άλλες. Έχει εκδοθεί σύμφωνα με τις παραμέτρους του Eclipse Public Licence που έχει ορίσει το Eclipse Foundation, είναι δωρεάν και προσφέρεται ως εφαρμογή ανοικτού κώδικα. Δημιουργήθηκε από το Free Software Community.

Το Android SDK σημαίνει «Android Software Development Kit» και είναι το επίσημο εργαλείο της Google για αυτούς που θέλουν να δημιουργήσουν εφαρμογές σε Android. Είναι το βασικό εργαλείο για τη δημιουργία εφαρμογών, την κατασκευή προσαρμοσμένης ROM και τη μεταγλώττιση πυρήνα για τη συσκευή μας.

Χρησιμοποιώντας το εργαλείο Eclipse και τις διάφορες βιβλιοθήκες που υπάρχουν στο Android SDK μπορούν να αναπτυχθούν από τις πιο απλές μέχρι και τις πιο πολύπλοκες εφαρμογές για Android συσκευές. Μεταγλωττίζοντας τον java κώδικα που αναπτύχθηκε για κάποια συγκεκριμένη εφαρμογή, θα παραχθεί ένα εκτελέσιμο αρχείο APK το οποίο μπορεί να μεταφερθεί σε μια συσκευή και να εγκατασταθεί. Έτσι λοιπόν η υλοποίηση νέων εφαρμογών είναι εύκολη, χρησιμοποιώντας το ανοιχτό λογισμικό-πρόγραμμα eclipse μαζί με το Android SDK. Ένα σημαντικό γεγονός που ευνοεί την ανάπτυξη εφαρμογών είναι πως το πακέτο Android SDK συνεργάζεται με το eclipse και συνεπώς ο προγραμματιστής μπορεί εύκολα και γρήγορα να βλέπει τις αλλαγές του κώδικα στον εξομοιωτή (emulator), που του παρέχει το Android SDK, χωρίς να χρειάζεται να εξαγει κάθε φορά την εφαρμογή και να την εγκαθιστά σε κινητό. Επίσης ο emulator είναι πολύ αξιόπιστος καθώς έχει ακριβώς την ίδια συμπεριφορά, αν η εφαρμογή εγκατασταθεί σε ένα κινητό τηλέφωνο Android. Τέλος, ένα άλλο σημαντικό πλεονέκτημα είναι το γεγονός πως σε αναβαθμίσεις του λογισμικού η εφαρμογή εξακολουθεί να δουλεύει χωρίς την ανάγκη επανασχεδιασμού κάποιων σημαντικών κομματιών του κώδικα που αφορά την αλληλεπίδραση της εφαρμογής με τα δομικά μέρη (hardware) του κινητού τηλεφώνου.

## ΛΟΓΙΣΜΙΚΑ OPEN SOURCE

Επισημάνεται πως η παρούσα εφαρμογή στηρίζεται σε λογισμικά και τεχνολογίες ανοικτού κώδικα, δηλαδή σε λογισμικά που ο καθένας μπορεί ελεύθερα να χρησιμοποιεί, να διανέμει, να αντιγράψει και να τροποποιεί ανάλογα με τις ανάγκες του, χωρίς να απαιτείται η απόκτηση άδειας. Η χρήση τέτοιων λογισμικών παρουσιάζει ορισμένα πλεονεκτήματα, όπως η παραγωγή καλύτερης ποιότητας λογισμικού σε επίπεδο αξιοπιστίας και ασφάλειας και συλλογική βελτίωση της τεχνογνωσίας μέσω της διαθεσιμότητας προς μελέτη του πηγαίου κώδικα. Πολύ μικρό κόστος απόκτησης και αναβάθμισης καθώς επίσης και μικρότερος χρόνος ανάπτυξης και βελτίωσης από τα εμπορικά πακέτα λόγω του πλήθους των προγραμματιστών που εργάζονται πάνω σε αυτά. Εύκολη απ' ευθείας επικοινωνία με τους προγραμματιστές του λογισμικού και ευρεία υποστήριξη μέσω forums και λίστες ηλεκτρονικού ταχυδρομείου (mailing lists) για επίλυση αποριών. Δυνατότητα επέμβασης στον κώδικα του λογισμικού και προσαρμογή στις ανάγκες του χρήστη. Δημιουργία κινήτρου για τις εταιρείες εμπορικού λογισμικού να βελτιώσουν τα προϊόντα τους ώστε να αντιμετωπίσουν τον ανταγωνισμό που προκύπτει από τα λογισμικά ανοικτού κώδικα.

Στη συνέχεια αναλύονται οι τεχνολογίες και τα λογισμικά ανοικτού κώδικα που έλαβαν μέρος στην αρχιτεκτονική της παρούσας εφαρμογής.

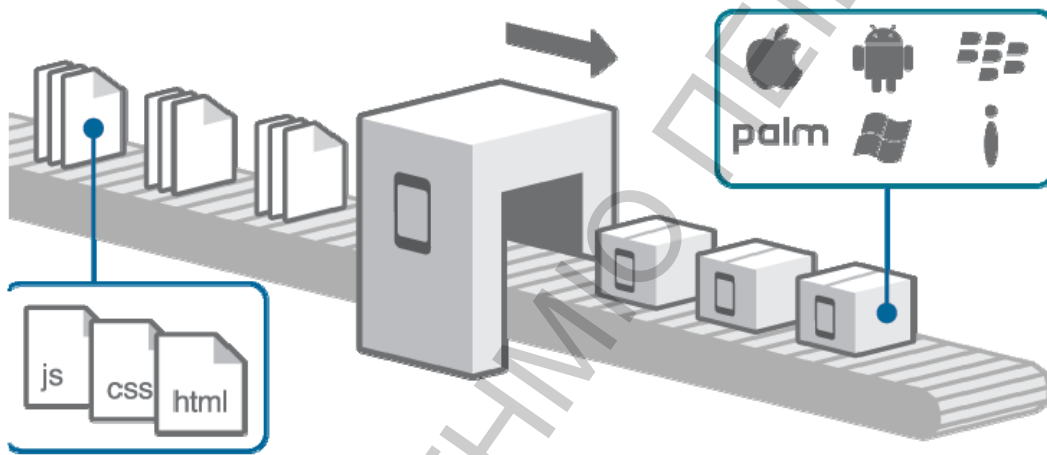
Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Τεχνολογίες που χρησιμοποιήθηκαν ονομαστικά είναι, το PhoneGap, HTML5, JavaScript (AJAX, JQuery Framework, JQuery Mobile), Google Maps API v3, JSON, CSS3, Web Server Apache, PHP5 (Codeigniter Framework) και PostgreSQL (με επέκταση PostGIS)

Παρακάτω περιγράφονται και αναλύονται όλες οι γλώσσες προγραμματισμού και σήμανσης που χρησιμοποιήθηκαν.

## PHONEGAP

Η δημιουργία εφαρμογών για κάθε συσκευή iPhone, Android, Windows Mobile και πολλά άλλα απαιτεί διαφορετικά πλαίσια και γλώσσες προγραμματισμού. Το Phonegap είναι μια πλατφόρμα ανοιχτού κώδικα (open source) που γεφυρώνει όλες τις εφαρμογές web και τις φορητές συσκευές. Εφαρμογές που δημιουργούνται, μπορούν να εκτελεστούν σε όλους τους τύπους λειτουργικών συστημάτων των κινητών τηλεφώνων (IOS, Android, Windows Phone, Symbian κλπ.) Το Phonegap βασίζεται στη χρήση HTML5 και JavaScript και CSS3.



**Εικόνα 6**

Επίσης, παρέχει μια διεπαφή προγραμματισμού εφαρμογών (API) που επιτρέπει την δυνατότητα απόκτησης πρόσβασης σε εγγενή λειτουργικότητα του λειτουργικού συστήματος με τη χρήση JavaScript. Οι ειδικές αρχιτεκτονικές εφαρμογές θα διαφέρουν μεταξύ των εφαρμογών που έχουν υλοποιηθεί σε Phonegap, ωστόσο οι περισσότερες βασιζόμενες σε εξυπηρετητή (Server-Based) εφαρμογές χρησιμοποιούν την ακόλουθη βασική αρχιτεκτονική.



**Εικόνα 7**

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Η εφαρμογή PhoneGap λειτουργεί ως πελάτης (Client) για το χρήστη αλληλεπιδρώντας με αυτόν. Ο PhoneGap πελάτης (Client) επικοινωνεί με έναν εξυπηρετητή (Server) εφαρμογών για τη λήψη δεδομένων. Ο εξυπηρετητής εφαρμογής διαχειρίζεται την επιχειρηματική λογική (Business Logic) και επικοινωνεί με μία βάση δεδομένων. Ο Εξυπηρετητής (Server) της εφαρμογής, είναι συνήθως ένας Web Server (Apache, IIS, κλπ.) και μπορεί να χρησιμοποιηθεί ως γλώσσα προγραμματισμού στη πλευρά αυτή η ColdFusion, η Java, η .NET, η PHP, κλπ. Το PhoneGap «δεν έχει άποψη» της τεχνολογίας που βρίσκεται από πίσω στο παρασκήνιο (back-end) και μπορεί να λειτουργήσει με κάθε Application Server που χρησιμοποιεί κλασικά πρωτόκολλα διαδικτύου. Ο Εξυπηρετητής εφαρμογής εκτελεί επιχειρηματική λογική και υπολογισμούς, και γενικά ανακτά δεδομένα από μία ξεχωριστή σχεσιακή βάση δεδομένων.

Οι εφαρμογές που έχουν υλοποιηθεί με Phonegap δεν μπορούν να επικοινωνήσουν απευθείας με μια βάση δεδομένων. Η επικοινωνία δρομολογείται μέσω ενός Application Server. Έτσι λοιπόν, η επικοινωνία μεταξύ πελάτη (Client) και Εξυπηρετητή (Server) μπορεί να βασίζεται σε πρότυπα αιτήματα HTTP για το περιεχόμενο HTML, REST-ful υπηρεσίες XML, JSON υπηρεσίες, ή SOAP (ή WebSockets αν το λειτουργικό που χρησιμοποιούμε το υποστηρίζει). Αυτές είναι οι ίδιες ακριβώς τεχνικές που θα *χρησιμοποιούνταν* σε μια κλασική διαδικτυακή εφαρμογή. Από την πλευρά του πελάτη, η λογική της εφαρμογής είναι μέσα σε μια ενιαία σελίδα HTML. Αυτή η σελίδα δεν εκφορτώνεται από τη μνήμη. Όλα τα δεδομένα θα εμφανίζονται με την ενημέρωση του HTML DOM. Όπως καταλαβαίνουμε τα δεδομένα ανακτώνται από το Εξυπηρετητή της εφαρμογής χρησιμοποιώντας τεχνικές AJAX και οι μεταβλητές διατηρούνται στη μνήμη με τη χρήση της JavaScript.

## HTML5

Όπως αναφέραμε παραπάνω για την ανάπτυξη της εφαρμογής της παρούσας μεταπτυχιακής διατριβής χρησιμοποιήθηκε η πλατφόρμα ανοικτού κώδικα PhoneGap με την οποία έχουμε την δυνατότητα να χρησιμοποιήσουμε διαδικτυακές γλώσσες. Σε αυτό το υποκεφάλαιο θα περιγράψουμε την HTML5.

Η HTML (HyperText Markup Language) αποτελεί την κύρια γλώσσα σήμανσης για την ανάπτυξη διαδικτυακών εφαρμογών και ιστοσελίδων. Η HTML γράφεται υπό την μορφή στοιχείων τα οποία αποτελούνται από ετικέτες οι οποίες περικλείονται μέσα σε σύμβολα λειτουργώντας ανά ζεύγη (π.χ. <html> </html>). Ο σκοπός ενός browser είναι να διαβάσει τα έγγραφα HTML και να τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Όταν ένας διαδικτυακός εξυπηρετητής ανοίγει ένα αρχείο HTML, τα στοιχεία (tags) μεταφράζονται σε κατάλληλα χαρακτηριστικά με αποτελέσματα στην εμφάνιση και στην λειτουργικότητα της συγκεκριμένης σελίδας. Η δημιουργία αρχείων HTML είναι πολύ απλή. Αρκεί να “τρέξουμε” έναν οποιοδήποτε διορθωτή κειμένου (text), όπως το Notepad των Windows. Να σημειωθεί ότι ένα από τα καλύτερα texts editors που χρησιμοποιούν οι προγραμματιστές είναι το Notepad++. Να γράψουμε τον κώδικα HTML που επιθυμούμε και να το αποθηκεύσουμε σαν αρχείο με κατάληξη «.html».

Η HTML5 αποτελεί ένα νέο πρότυπο ιστοσελίδων, προσφέρει νέες λειτουργίες για τον ορισμό αυτό που θα προσφέρει στο Διαδίκτυο, εφαρμογές πιο ισχυρές και πιο ευέλικτες. Συγκεκριμένα, επιτρέπει στους Browsers να υποστηρίξουν την αποθήκευση δεδομένων στην πλευρά του πελάτη (Client). Η προσωρινή ή μόνιμη αποθήκευση των δεδομένων αποτελεί ένα κρίσιμο και σημαντικό παράγοντα για τις διαδικτυακές εφαρμογές. Εκτός από τις παραδοσιακές μεθόδους αποθήκευσης δεδομένων, όπως τα cookies και τις βάσεις δεδομένων στην πλευρά του Server, μια νέα καινοτόμα υποδομή είναι διαθέσιμη για του Browsers παρέχοντας τοπική αποθήκευση. Με την αποθήκευση HTML5 Web API και Web SQL Database API γίνεται δυνατή η αποθήκευση των δεδομένων της εφαρμογής σαν κλειδί / τιμή, ή ακόμη και ως κανονικοποιημένο πίνακα σχεσιακής βάση δεδομένων, προσβάσιμο με SQL δηλώσεις από την πλευρά του πελάτη (Client), κάνοντας διαθέσιμη την αναζήτηση τους, ανάκτηση και ενημέρωση

ανά πάσα στιγμή. Επίσης, είναι δυνατή η αποθήκευση μεγάλου όγκου δεδομένων, χωρίς να επηρεάζεται η απόδοση της ιστοσελίδας ή εφαρμογής. Στη παρούσα εφαρμογή χρησιμοποιείται το αντικείμενο `localStorage` το οποίο μας δίνει τη δυνατότητα να αποθηκεύσουμε δεδομένα χωρίς ημερομηνία λήξης. Η HTML5 δεν έχει ακόμη εγκριθεί επίσημα, και δεν υποστηρίζεται σε όλα τα διαθέσιμα προγράμματα περιήγησης στο διαδίκτυο. Έτσι σήμερα, υποστηρίζεται εν μέρει από το Safari, Firefox και Chrome browser.

## JAVASCRIPT

Μια ακόμα διαδικτυακή γλώσσα προγραμματισμού που μας επιτρέπει η πλατφόρμα PhoneGap να χρησιμοποιήσουμε, είναι η Javascript.

Η γλώσσα προγραμματισμού JavaScript δημιουργήθηκε το 1995 από τον Brendan Eich της εταιρείας Netscape Communications με την βοήθεια της εταιρίας Sun Microsystems. Το αρχικό όνομα της γλώσσας ήταν Mocha και αργότερα το Mocha μετονομάστηκε σε LiveScript, και τελικά σε JavaScript, κυρίως επειδή η ανάπτυξη της επηρεάστηκε περισσότερο από τη γλώσσα προγραμματισμού Java. Ο τίτλος Javascript δημιουργεί κάποια σύγχυση επειδή παραπέμπει στην Java με την οποία δεν υπάρχει καμία σχέση.

Η Javascript αποτελεί μια γλώσσα σεναρίων και όχι σήμανσης, όπως η HTML που περιγράφηκε παραπάνω. Σκοπός της γλώσσας αυτής είναι να παρέχει την δυνατότητα παραγωγής δυναμικού περιεχομένου και την εκτέλεση κώδικα στην πλευρά του Πελάτη (Client). Αποτελεί μια εύκολη και μικρή στην εκμάθηση της γλώσσας σεναρίων, η οποία επεκτείνει την δράση της HTML. Με τη χρήση της οι ιστοσελίδες και οι διαδικτυακές εφαρμογές αποκτούν δυναμική εμφάνιση, γίνονται πιο αλληλεπιδραστικές με τον χρήστη, μπορούν να παρουσιάσουν ειδικά εφέ, να εκτελέσουν υπολογισμούς, να εμφανίσουν καλύτερα γραφικά ή λειτουργίες, να ελέγξουν δεδομένα από φόρμες και πολλά άλλα. Παρακάτω αναφέρονται οι δυνατότητες και οι περιορισμοί που έχει αυτή η γλώσσα. Η Javascript έχει δυνατότητες όπως ο έλεγχος στο περιεχόμενο και την εμφάνιση των διαδικτυακών εφαρμογών. Επίσης ο έλεγχος της συμπεριφοράς και των λειτουργιών του προγράμματος περιήγησης. Αλληλεπίδραση με φόρμες τις HTML και με τον χρήστη με τη βοήθεια των χειριστών γεγονότων(event handlers). Ανάγνωση ή καταγραφή της κατάστασης του υπολογιστή του χρήστη μέσω cookies. Επίσης, δυνατότητες όπως οι εναλλαγές εικόνων που δίνουν μια δυναμική αίσθηση στην ιστοσελίδα, αλληλεπίδραση με μικροεφαρμογές της Java και παροχή πληροφοριών για την οθόνη του χρήστη.

Ωστόσο, υπάρχουν και περιορισμοί σε αυτή τη γλώσσα σεναρίων. Για λόγους ασφάλειας, η Javascript δεν παρέχει τη δυνατότητα προσπέλασης χειρισμού αρχείων υπολογιστή του χρήστη. Η Javascript δεν υποστηρίζει δυνατότητες δικτύωσης οποιασδήποτε μορφής, ούτε διαθέτει εντολές δημιουργίας γραφικών ή σχεδίων. Ένα σενάριο Javascript εκτελείται στον υπολογιστή του χρήστη που επισκέπτεται μια διαδικτυακή εφαρμογή ή ιστοσελίδα, χωρίς να έχει τη δυνατότητα να ενημερώνει αρχεία που βρίσκονται σε ένα Web Server.

## CSS

Το CSS (Cascading Style Sheets) αποτελεί μια γλώσσα φύλλων στυλ και χρησιμοποιείται για τον έλεγχο της εμφάνισης μιας διαδικτυακής εφαρμογής. Ουσιαστικά ελέγχει την εμφάνιση που δημιουργεί η HTML. Επιτρέπει μεγαλύτερη ευελιξία στην διαμόρφωση μιας ιστοσελίδας, διαχωρίζοντας τον τρόπο μορφοποίησης των στοιχείων του εγγράφου (επικεφαλίδες, κείμενο, συνδέσμους κ.α.). Με την επισύναψη αρχείων CSS σε δομημένα έγγραφα (HTML, XML κτλ) ο χρήστης ή ο αναγνώστης έχει τη δυνατότητα να τροποποιήσει την παρουσίαση του εγγράφου χωρίς να κάνει αλλαγές στη δομή του, δηλαδή να προσθέσει κάποια ετικέτα κλπ.

Άλλος ένας σημαντικός λόγος για τη χρήση της γλώσσας αυτής στο διαδίκτυο είναι ο διαχωρισμός του περιεχομένου από την εμφάνιση του εγγράφου, δίνοντας έτσι ένα πλεονέκτημα σε θέματα που αφορούν την πρόσβαση στο περιεχόμενο, την καλύτερη δόμηση



του κώδικα, θέματα βελτιστοποίησης κλπ. Το κύριο πλεονέκτημα της CSS όπως αναφέρθηκε παραπάνω είναι το ότι μια μορφοποίηση μπορεί να εφαρμοστεί σε ένα σύνολο ιστοσελίδων. Αν απαιτούνται αλλαγές στη μορφοποίηση, τότε με μια επεξεργασία του CSS εγγράφου όλες οι σελίδες του ιστότοπου θα λάβουν την ίδια αλλαγή. Η τελευταία έκδοση του CSS είναι το CSS level 3 ή CSS3.

## JQUERY & JQUERY MOBILE

Το jQuery αποτελεί ένα πλαίσιο εργασίας, μια βιβλιοθήκη της Javascript η οποία μας δίνει τη δυνατότητα να δημιουργούμε διάφορα εφέ στη σελίδα μας, χωρίς όμως να χρειάζεται η ποσότητα κώδικα που θα χρειαζόταν εάν χρησιμοποιούσαμε την παραδοσιακή Javascript. Είναι δωρεάν και αποτελεί λογισμικό ανοικτού κώδικα, υπό την άδεια του MIT License. Με το JQuery μπορούμε να πλοηγηθούμε με ευκολία σε αρχεία HTML, να δημιουργήσουμε κινούμενα σχέδια, να χειριστούμε με ευκολία γεγονότα. Το μόνο που έχουμε να κάνουμε είναι να κατεβάσουμε το jquery.js αρχείο και το αντίστοιχο css του ή να δημιουργήσουμε μια σύνδεση μεταξύ του αρχείου html και της διεύθυνσης της βιβλιοθήκης(π.χ. <http://code.jquery.com/jquery-1.8.2.min.js>) και να το εφαρμόσουμε στη διαδικτυακή εφαρμογή. Για να το ενσωματώσουμε αρκεί να γράψουμε στο html αρχείο αυτό:

```
<script type="text/javascript" src="jquery.js"></script>
```

Τα εφέ που μπορούμε να δημιουργήσουμε είναι πολλά και διάφορα. Για παράδειγμα μπορούμε να εμφανίζουμε ή να εξαφανίζουμε παραγράφους, να αλλάζουμε χρώματα γραμματοσειράς, να ελέγχουμε δηλαδή με όλα τα html στοιχεία της σελίδας μας και να μπορούμε να ελέγχουμε ένα-ένα από αυτά ξεχωριστά. Το framework αυτό με ένα συγκεκριμένο τρόπο γραφής της Javascript δίνει τη δυνατότητα να προστεθούν εφέ στην εφαρμογή πολύ εύκολα καθώς και να εκτελεστούν ερωτήματα AJAX, τα οποία θα δούμε παρακάτω. Η προσέγγιση της jQuery βιβλιοθήκη επιτρέπει τη δημιουργία ισχυρών δυναμικών ιστοσελίδων και διαδικτυακών εφαρμογών.

Η σύνταξη είναι πολύ απλή:

### \$(επιλογή).δράση()

Το δολάριο είναι το σύμβολο που υποδηλώνει την jQuery. Ο (επιλογέας) είναι το ερώτημα (query) για το στοιχείο HTML που θέλουμε. Η δράση (action) είναι ότι θα συμβεί στο στοιχείο που επιλέξαμε.

#### ΠΑΡΑΔΕΙΓΜΑΤΑ

\$(this).hide() – Κρύβει το επιλεγμένο αντικείμενο

\$("p").hide() – Κρύβει όλες τις παραγράφους

\$("p.test").hide() – Κρύβει όλες τις παραγράφους με class="test"

\$("#test").hide() – Κρύβει όλα τα στοιχεία με id="test"

Επίσης, υπάρχει μια πολύ μεγάλη και ενεργή κοινότητα προγραμματιστών που έχουν δημιουργήσει διάφορα πρόσθετα (plugins) με τη χρήση του jQuery, τα οποία μπορεί ο καθένας να χρησιμοποιήσει, εκμηδενίζοντας έτσι τον χρόνο δημιουργίας του εφέ που επιθυμεί. Ακόμα, υπάρχουν ορισμένα λογισμικά (software), όπως για παράδειγμα το Glimmer, τα οποία μπορεί κάποιος να κατεβάσει στον υπολογιστή του και πανεύκολα να δημιουργήσει το εφέ ή την κίνηση που επιθυμεί και στη συνέχεια, το software αυτό να του δώσει έτοιμο τον κώδικα jQuery.

Όλα αυτά κάνουν το jQuery ένα πολύ δυνατό εργαλείο, με αποτέλεσμα όλο και περισσότερες ιστοσελίδες να επιλέγουν τη χρήση του. Σύμφωνα με το w3techs.com, το jQuery χρησιμοποιείται στο 79% των ιστοσελίδων που χρησιμοποιούν κάποια βιβλιοθήκη Javascript και στο 34.8% όλων των ιστοσελίδων παγκοσμίως. Δηλαδή, ακόμα και αν το jQuery σας φάνηκε

ξένο, σίγουρα το έχετε συναντήσει κάπου, εφόσον το χρησιμοποιούν πολλές δημοφιλείς ιστοσελίδες όπως τα Twitter.com, Msn.com, Amazon.com, Bbc.co.uk και πολλά άλλα.

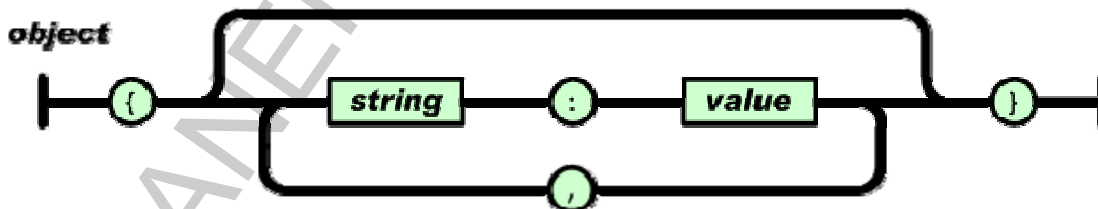
Η βιβλιοθήκη JQuery δημιούργησε και μια ειδική έκδοσή της για την εφαρμογή αυτού του framework στην ανάπτυξη εφαρμογών κινητών τηλεφώνων (Smart phones), το JQuery Mobile. Με αυτή την βιβλιοθήκη η οποία δουλεύει όπως το κλασσικό JQuery, μπορούμε να δημιουργήσουμε μοναδικές εφαρμογές υψηλού επιπέδου οι οποίες μπορούν να παίζουν σε όλα τα γνωστά λειτουργικά συστήματα κινητών αλλά και σε ταμπλέτες. Η τρέχουσα έκδοση alpha του JQuery Mobile υποστηρίζει iOS (iPhone, iPad, iPod Touch), Android, Blackberry OS6, webOS της HP, Fennec της Mozilla και Opera Mobile. Στο μέλλον, άλλες πλατφόρμες θα προστεθούν συμπεριλαμβανομένων των Windows Mobile, Symbian, MeeGo και περισσότερες. Οι κλασσικές μέθοδοι και ο τρόπος γραφής του JQuery είναι ίδιος και στην ανάπτυξη της εφαρμογής του κινητού. Ωστόσο το framework αυτό δίνει έξτρα δυνατότητες που χρησιμεύουν μόνο για εφαρμογές κινητών τηλεφώνων, αλλά και γραφικά τα οποία παραμετροποιούν την γραφική υπόσταση της εφαρμογής ώστε να προσαρμόζεται στην μικρή οθόνη ενός Smartphone ή tablet.

## JSON

Μια ακόμα τεχνολογία που χρησιμοποιήθηκε για την ανάπτυξη της παρούσας εφαρμογής είναι το JSON. Το JSON ή αλλιώς JavaScript Object Notation είναι ένα πρότυπο ανταλλαγής δεδομένων βασισμένο στην γλώσσα προγραμματισμού Javascript. Το JSON είναι μια μορφή κειμένου που είναι εντελώς ανεξάρτητη γλώσσα αλλά χρησιμοποιεί συμβάσεις που είναι γνωστές στους προγραμματιστές της οικογένειας γλώσσας προγραμματισμού C, συμπεριλαμβανομένων των C, C + +, C #, Java, JavaScript, Perl, Python, και πολλές άλλες. Αυτές οι ιδιότητες καθιστούν το JSON μια ιδανική γλώσσα ανταλλαγής δεδομένων. Χρησιμοποιείται ευρέως διότι είναι εύκολο από τους ανθρώπους να το διαβάσουν και ακόμη πιο εύκολο να ερμηνευτεί από τα προγράμματα, ώστε να μπορέσουν και να το παραγάγουν. Το πλεονέκτημα που έχει είναι η ανεξαρτησία του από τις γλώσσες προγραμματισμού. Στην επίσημη ιστοσελίδα του JSON μπορούμε να βρούμε πληροφορίες πως λειτουργεί για κάθε γλώσσα προγραμματισμού(<http://www.json.org/>).

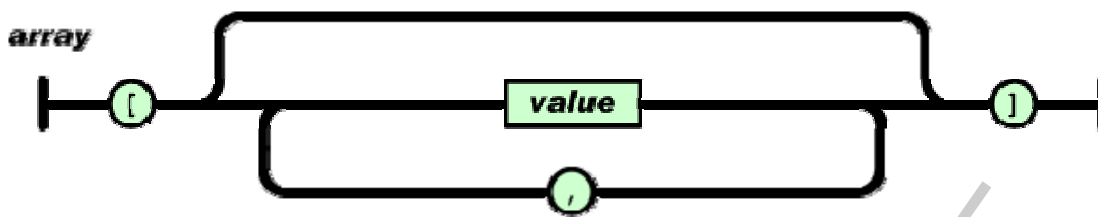
Το JSON είναι φτιαγμένο σε δύο δομές. Σε μια συλλογή ζευγών ονομάτων/τιμών, σε διάφορες γλώσσες αυτό είναι αντιληπτό σαν ένα αντικείμενο, μια εγγραφή, μια δομή έναν πίνακα κ.α. Και σε μια ταξινομημένη λίστα τιμών. Στις περισσότερες γλώσσες, αυτό γίνεται αντιληπτό ως ένα πίνακα, διάνυσμα, λίστα, ή ακολουθία.

Ένα αντικείμενο είναι ένα μη διατεταγμένο ζεύγος ονόματος/τιμής. Ένα αντικείμενο ξεκινάει με { (αριστερό άγκιστρο) και τελειώνει με } (δεξιό άγκιστρο). Κάθε όνομα ακολουθείται από: (άνω και κάτω τελεία) και τα όνομα / τιμή ζεύγη διαχωρίζονται από κόμμα. Φαίνεται και στο παρακάτω σχήμα:



Εικόνα 8

Σε ένα πίνακα μια σειρά ξεκινά με [ (αριστερή αγκύλη) και τελειώνει με ] (δεξιά αγκύλη). Οι τιμές χωρίζονται με, (κόμμα).



Εικόνα 9

Γιατί όμως προτιμάται πλέον η χρήση του JSON από την XML;

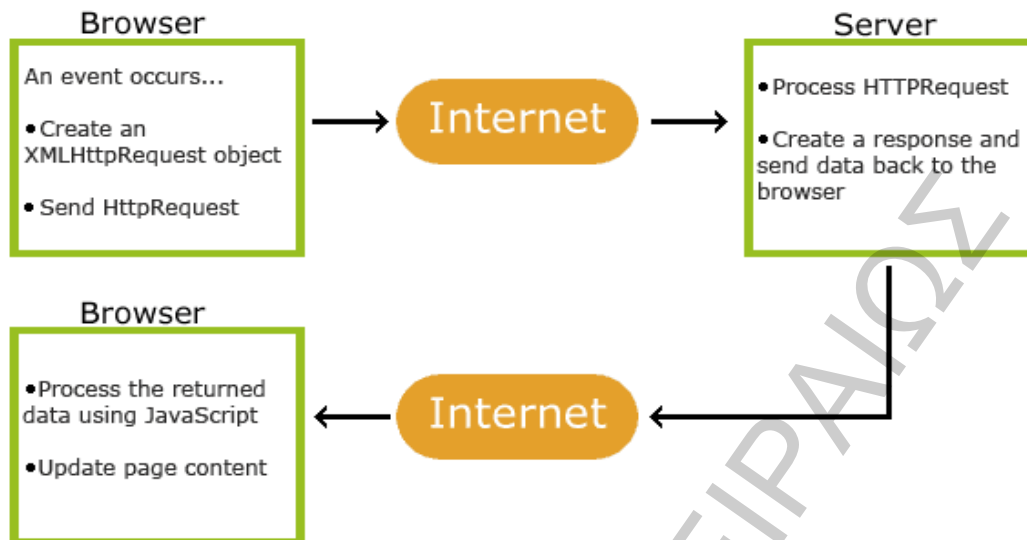
Το μέγεθος των αρχείων JSON είναι πολύ μικρότερο από τα αρχεία XML για τα ίδια δεδομένα. Το πέρασμα (parsing) και η επεξεργασία ενός αρχείου XML είναι πολύ πιο περίπλοκη και απαιτεί περισσότερους πόρους (resources) σε σχέση με το JSON. Επιπλέον, το json\_decode είναι έμφυτο (native) PHP συνάρτηση (function) ενώ για το XML πέρασμα (parsing) χρησιμοποιούνται PHP επεκτάσεις (extensions). Το τελικό αποτέλεσμα χρησιμοποιώντας το JSON είναι τουλάχιστον πέντε φορές γρηγορότερο και χρειάζεται το ένα τρίτο των απαιτήσεων σε πόρους σε σχέση με την τεχνολογία της XML.

## AJAX

Μια διαδικτυακή εφαρμογή ή ιστοσελίδα είναι ένα σύνολο html οδηγιών που στέλνει ο Εξυπηρετητής στο περιηγητή του χρήστη όταν αυτός αιτείται να "δει" το περιεχόμενο της ιστοσελίδας αυτής. Η αποστολή των οδηγιών html γίνεται μια φορά και αφού αυτή ολοκληρωθεί η επικοινωνία κλείνει Server-Browser. Έτσι κάθε φορά που ο διαχειριστής ή προγραμματιστής της διαδικτυακής εφαρμογής κάνει αλλαγές στο περιεχόμενο της, οι αλλαγές αυτές δεν είναι άμεσα ορατές από τον επισκέπτη εκτός και αν αυτός ανανεώσει την ιστοσελίδα οπότε και ξανά ανοίγει η επικοινωνία του Browser με τον server, παραλαμβάνοντας εκ νέου το ανανεωμένο περιεχόμενο.

Η χρήση της τεχνικής AJAX επιτρέπει σε μια διαδικτυακή εφαρμογή ή ιστοσελίδα να ανανεώνεται ασύγχρονα ανταλλάσσοντας στο παρασκήνιο δεδομένα, συνήθως μικρού όγκου, με τον server επιτρέποντας να ανανεώνονται μέρη της ιστοσελίδας, χωρίς να ανανεώνεται ολόκληρη η σελίδα.

Η τεχνική αυτή του AJAX χρησιμοποιείται πολύ συχνά σε εφαρμογές ή σελίδες που περιέχουν φόρμες εγγραφής σε κάποια ιστοσελίδα ή forum. Η AJAX δεν είναι γλώσσα προγραμματισμού όπως η JavaScript, αλλά ούτε και γλώσσα χαρακτηρισμού κειμένου όπως η HTML. Δεν θεωρείται καν γλώσσα. AJAX σημαίνει Asynchronous Javascript And XML και είναι η τεχνική με την οποία μπορούμε να δημιουργήσουμε πολύ πιο γρήγορες και δυναμικές ιστοσελίδες ή διαδικτυακές εφαρμογές περιορίζοντας τον όγκο δεδομένων που ανταλλάσσει ο εξυπηρετητής με τον περιηγητή του επισκέπτη. Η ιδιαίτερη τεχνική αυτή επιτρέπει την ανανέωση περιεχομένων μιας ιστοσελίδας χωρίς αυτή να ανανεωθεί ολόκληρη. Η υλοποίηση της τεχνικής AJAX γίνεται με το συνδυασμό του αντικειμένου (object) XMLHttpRequest (πραγματοποιεί την ασύγχρονη επικοινωνία με τον server), την Javascript/DOM (αλληλεπιδρά με τα δεδομένα και τα εμφανίζει), την CSS (μορφοποιεί τα προς εμφάνιση δεδομένα) και την XML (συχνά χρησιμοποιείται για την μεταφορά δεδομένων).



Εικόνα 10

Παρά το επαναστατικό αντίκτυπο του AJAX στο διαδίκτυο μπορεί να είναι δύσκολο να χρησιμοποιηθεί ακόμη και από πολύ έμπειρους προγραμματιστές. Οι άνθρωποι συνήθως δημιουργούν τις δικές τους προσαρμοσμένες λειτουργίες και συναρτήσεις για να χειριστούν το Ajax, έτσι ώστε η λειτουργικότητα να μπορεί να επαναχρησιμοποιηθεί σε μια διαδικτυακή εφαρμογή. Αυτό μπορεί να γίνει πολύ κουραστικό όταν μια διαδικτυακή εφαρμογή κάνει χρήση των διαφόρων τύπων του Ajax - όπως η ικανότητα να χειρίζεται τόσο σύγχρονες και ασύγχρονες κλήσεις - ή την ικανότητα να χειρίζεται διάφορες μορφές απάντησης (response) όπως HTML, CSV, XML, JSON, κλπ. Για αυτό το λόγο δημιουργήθηκε το JQuery AJAX το οποίο δίνει περισσότερες δυνατότητες στο χρήστη, μιας και ενσωματώνει όλες τις λειτουργίες που δίνεται από την βιβλιοθήκη της Javascript, το JQuery. Για να κατανοηθεί το πλεονέκτημα του JQuery AJAX έναντι στο κλασικό AJAX θα δούμε παρακάτω πώς συντάσσονται. Αιτήματα στο AJAX:

```
function aJaxRequest(id) {
  if (window.XMLHttpRequest) { // code for IE7+, FF, Chrome, Opera, Safari
    http=new XMLHttpRequest();
  }
  else { // code for IE6, IE5
    http=new ActiveXObject("Microsoft.XMLHTTP");
  }
  http.onreadystatechange=function() {
    if (http.readyState==4 && http.status==200) {
      response = http.responseText;
      // do something with response
    }
  }
  dest="servlet.php?id=" + id;
  http.open("GET", dest);
  http.send();
}
```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για εξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```
}
```

Και ένα JQuery AJAX Request:

```
$.Ajax({  
  type: "POST",  
  url: " servlet.php",  
  data: "id="+id,  
  success: function(id){  
    // do something with response  
    alert( "Data Saved: " + msg );  
  }  
});
```

Παρατηρούμε ότι η διαφορά είναι πολύ μεγάλη. Έτσι λοιπόν στην παρούσα εφαρμογή χρησιμοποιούνται ερωτήματα JQuery AJAX, τα οποία ουσιαστικά αποτελούν την γέφυρα επικοινωνίας της εφαρμογής του κινητού τηλεφώνου (Client ) με τον Εξυπηρετητή Server, μεταφέροντας μεταβλητές. Με την σειρά του ο Εξυπηρετητής απαντάει πίσω μεταφέροντας τα κατάλληλα δεδομένα σε μορφή JSON. Την επικοινωνία αυτή θα τη δούμε αναλυτικότερα στο επόμενο κεφάλαιο.

## PHP5 & Codeigniter

Η PHP είναι μια γλώσσα προγραμματισμού που σχεδιάστηκε για τη δημιουργία δυναμικών σελίδων στο διαδίκτυο και είναι επισήμως γνωστή ως HyperText Preprocessor. Στην παρούσα εφαρμογή είναι η γλώσσα που χρησιμοποιείται από την πλευρά του Εξυπηρετητή με την οποία διαχειρίζεται τη πληροφορία και παράγεται η κατάλληλη λειτουργικότητα.

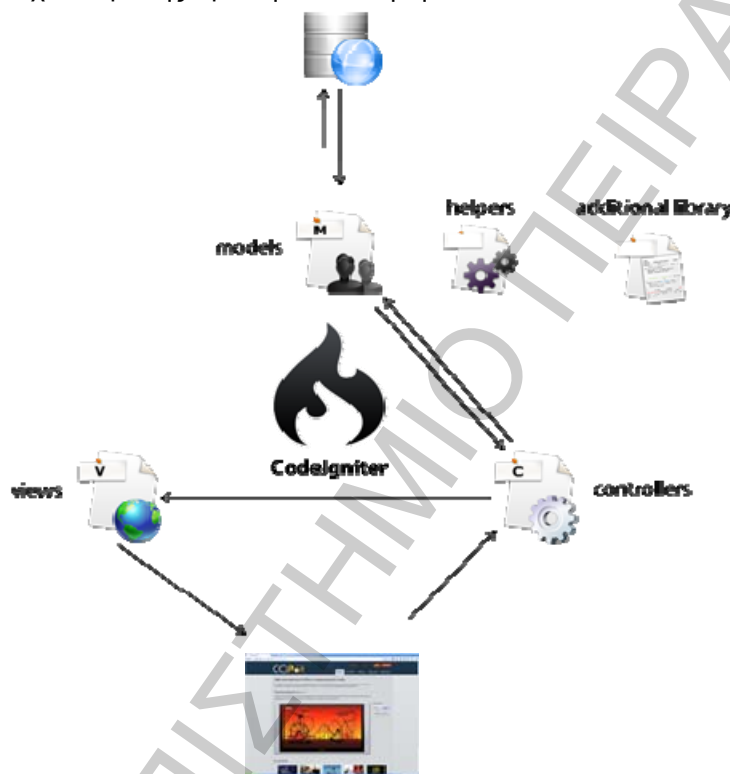
Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό Εξυπηρετητή του Παγκόσμιου Ιστού (π.χ. Web Server Apache) ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML. Αντίθετα από μια συνηθισμένη σελίδα HTML, η PHP δεν στέλνεται άμεσα σε έναν πελάτη (Client), αντί αυτού πρώτα αναλύεται και μετά αποστέλλεται το παραγόμενο αποτέλεσμα. Τα στοιχεία HTML στον πηγαίο κώδικα μένουν ως έχουν, αλλά ο κώδικας PHP ερμηνεύεται και εκτελείται. Ο κώδικας PHP μπορεί να θέσει ερωτήματα σε βάσεις δεδομένων, να δημιουργήσει εικόνες, να διαβάσει και να γράψει αρχεία, να συνδεθεί με απομακρυσμένους υπολογιστές, κτλ. Οι δυνατότητες που μας δίνει είναι απεριόριστες. Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. \*.php, \*.php4, \*.html κ.ά.). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης.html δεν θα λειτουργήσει και θα εμφανίσει στον περιηγητή τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στους τύπους MIME του server. Η έκδοση 5 αποτελεί την τελευταία έκδοση της γλώσσας PHP και περιλαμβάνονται νέα χαρακτηριστικά, όπως η βελτιωμένη υποστήριξη για αντικειμενοστρεφή προγραμματισμό, PHP Data Objects (ΠΟΠ) επέκταση (η οποία καθορίζει μια ελαφριά και συνεπή διεπαφή για πρόσβαση σε βάσεις δεδομένων), καθώς και πολλές βελτιώσεις επιδόσεων.

Η PHP επιλέχτηκε γιατί είναι μία εύκολη γλώσσα, είναι γρήγορη στη δημιουργία δυναμικών εφαρμογών, είναι αξιόπιστη και είναι δωρεάν.

Όπως οι περισσότερες γλώσσες προγραμματισμού και αυτή συνοδεύεται από πλαίσια εργασίας (frameworks). Το πλαίσιο εργασίας είναι το πλαίσιο εργασίας ή αλλιώς ο σκελετός ενός προγράμματος. Στην παρούσα διπλωματική χρησιμοποιήθηκε το Codeigniter.

Το Codeigniter είναι ένα πολύ δυνατό πλαίσιο εργασίας (framework) της PHP με πολύ μικρό αποτύπωμα. Έχει δημιουργηθεί για προγράμματα PHP, που χρειάζονται ένα απλό και εύχρηστο συνδυασμό εργαλείων για να δημιουργήσουν τελειοποιημένα χαρακτηριστικά διαδικτυακών εφαρμογών. Το Codeigniter είναι ιδανικό για έναν κατασκευαστή ιστοσελίδων, που ζει στον πραγματικό κόσμο, όπου μοιράζονται λογαριασμοί σε πομπούς και δέκτες με deadlines και τον έχει κουράσει η πολύ σκέψη και η χρήση εντελώς ατεκμηρίωτων πλαίσια εργασίας (frameworks).

Το πλαίσιο εργασίας Codeigniter χρησιμοποιεί το πρότυπο ανάπτυξης MVC (model – view - Controller) και είναι πολύ χρήσιμο επειδή κάνει τα σχέδια για να συνδεθούν οι εφαρμογές, προφανή, αλλά σε αντίθεση με άλλα πλαίσια MVC δεν είναι σχολαστικό. Αυτό θα το διαπιστώσουμε στο επόμενο κεφάλαιο το οποίο αναφέρεται στην αρχιτεκτονική της εφαρμογής αλλά και στον σχεδιασμό της πριν την υλοποίηση.



**Εικόνα 11**

Όπως παρατηρούμε το Codeigniter βασίζεται στο MVC (Model-View-Ελεγκτής(Controller)), ας δούμε λοιπόν τι κάνει το καθένα από αυτά και σε τι χρησιμεύουν:

Το Μοντέλο (Model) αποτελεί τις δομές δεδομένων. Δηλαδή με το μοντέλο δημιουργούμε συναρτήσεις οι οποίες περιέχουν λειτουργίες που βοηθάνε να ανακτήσουμε, παρεμβάλουμε και να ενημερώσουμε τα δεδομένα μας στην Βάση (Data base). Είναι συμβατό με όλες τις βάσεις δεδομένων, απλώς χρειάζεται να δηλώσουμε τη βάση και τα στοιχεία της (όνομα χρήστη, κωδικός, όνομα βάσης δεδομένων) σε ένα ειδικό αρχείο που ονομάζεται config/database.php

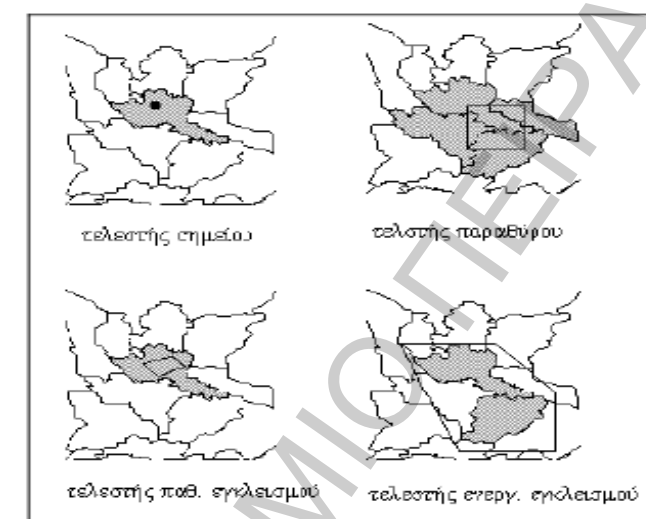
Η εμφάνιση (View) είναι οι πληροφορίες που παρουσιάζονται σε ένα χρήστη. Ένα View θα μπορούσε να είναι μέρος μιας ιστοσελίδας ή να αποτελεί μια ολοκληρωμένη ιστοσελίδα.

Ο Ελεγκτής (Controller) αποτελεί το μέσο επικοινωνίας μεταξύ του View και του Model και χρησιμεύει για να γίνει επεξεργασία ενός αιτήματος HTTP.

Στην παρούσα εφαρμογή τον ρόλο του View έχουν τα αρχεία HTML της πλευράς του πελάτη(Client) το οποίο επικοινωνεί με τον Ελεγκτή(Controller) με τη χρήση του jQuery AJAX. Στο επόμενο κεφάλαιο θα περιγραφεί ο τρόπος ανάπτυξης και λειτουργίας της εφαρμογής.

## PostgreSQL – PostGIS

Με τη πλέον αυστηρή έννοια, ο όρος Γεωγραφικά Πληροφοριακά Συστήματα (G.I.S. Geographic Information Systems) περιγράφει ένα σύστημα διαχείρισης χωρικών δεδομένων (spatial data). Ενσωματώνει, αποθηκεύει, επεξεργάζεται, αναλύει, διαμοιράζεται και απεικονίζει γεωγραφικές πληροφορίες. Σε ποιο γενική μορφή, Γεωγραφικά Πληροφοριακά Συστήματα είναι εργαλεία τα οποία επιτρέπουν στους χρήστες να δημιουργούν αλληλεπιδραστικά ερωτήματα για την ανάλυση χωρικών πληροφοριών, επεξεργασία δεδομένων, χαρτών και να προβάλουν τα αποτελέσματα αυτών των ενεργειών τους.



Εικόνα 12

Η PostgreSQL (<http://www.postgresql.org/>) είναι μία από τις δημοφιλέστερες, παγκοσμίως, σχεσιακή βάση δεδομένων ανοικτού κώδικα με ισχυρή υποστήριξη τύπων γεωγραφικών δεδομένων. Παρέχονται interfaces για γλώσσες προγραμματισμού όπως Perl, Python, C/C++, Embedded SQL, Delphi/ Kylix/ Pascal, VB, PHP, ASP, Java, και τις τεχνολογίες ODBC, JDBC. Η διαχείριση γίνεται κυρίως μέσω του PgAdmin III, αλλά και άλλες εφαρμογές τρίτων: (PgAccess, PhpPgAdmin, WinSQL). Υποστηρίζει χωρικά στοιχεία όπως γεωμετρικούς τύπους δεδομένων, χωρικούς δείκτες, γεωμετρικούς τελεστές, γεωμετρικές συναρτήσεις και μετατροπές γεωμετρικών στοιχείων. Παρακάτω παρουσιάζεται ένας πίνακας με τους γεωμετρικούς τύπους δεδομένων που υποστηρίζει η PostgreSQL:

Τύπος	Αποθήκευση	Περιγραφή
Point	16 bytes	Σημείο στο επίπεδο
Line	32 bytes	Ευθεία γραμμή
Lseg	32 bytes	Πεπερασμένο ευθύγραμμο τμήμα
Box	32 bytes	Ορθογώνιο
Path	16+16n bytes	Κλειστό μονοπάτι
Path	16+16n bytes	Ανοιχτό μονοπάτι
Polygon	40+16n bytes	Πολύγωνο

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Circle	24 bytes	Κύκλος
--------	----------	--------

Έχει σχεδιαστεί για να ικανοποιήσει τις ανάγκες των χρηστών, από απλό γράψιμο SQL ερωτημάτων έως την ανάπτυξη πολύπλοκων βάσεων δεδομένων. Περιέχει επεκτάσιμους τύπους δεδομένων και τελεστών που επιτρέπουν την εφαρμογή αναζήτησης πλήρους κειμένου, υψηλής απόδοσης γεωγραφικά πληροφοριακά συστήματα. Επίσης παρέχει χωρικά ευρετήρια (indexes) όπως R-δένδρα για πολυγωνικά δεδομένα και γενικευμένα δένδρα αναζήτησης (GiST). Τα γενικευμένα δένδρα αναζήτησης χρησιμοποιούνται για δεικτοδότηση όλων των χωρικών τύπων, για την υλοποίηση των R-δένδρων των ειδικών περιπτώσεων και υπάρχει η δυνατότητα παραμετροποίησης των χωρικών λειτουργιών. Η PostgreSQL υλοποιεί εξεζητημένα χαρακτηριστικά όπως Multi-Version Concurrency Control (MVCC), αποκατάσταση σημείου στο χρόνο (point in time recovery), θέσεις αποθήκευσης όπου τα πραγματικά στοιχεία στα οποία βασίστηκαν τα αντικείμενα βάσης δεδομένων μπορούν να διατηρηθούν table spaces, ασύγχρονη αναπαραγωγή (asynchronous replication), ένθετες συναλλαγές (nested transactions-saverpoints), αντίγραφα ασφαλείας σε απευθείας σύνδεση (online/hot backups), ένα εξελιγμένο ερώτημα (a sophisticated query planner/optimizer). Υποστηρίζει διεθνή σετ χαρακτήρων, κωδικοποίηση χαρακτήρων σε πολλά byte, Unicode καθώς και δυνατότητα ταξινόμησης δεδομένων ανεξάρτητα από τις τοπικές ρυθμίσεις (locale). Η PostgreSQL μπορεί να διαχειριστεί εύκολα μεγάλους αριθμούς ταυτόχρονων χρηστών καθώς και μεγάλο όγκο δεδομένων. Αν και υποστηρίζει από μόνη της χωρικούς τύπους δεδομένων, αυτοί δεν ακολουθούν ικανοποιητικά το πρότυπο OGC (Open GIS Consortium).

Η PostGIS (<http://postgis.refractor.net/>) είναι μία επέκταση της PostgreSQL για να υποστηρίζει χωρικά δεδομένα, σύμφωνα με το πρότυπο του OGC. Είναι σταθερή, γρήγορη, συμβατή με τα διεθνή πρότυπα, παρέχει εκατοντάδες χωρικές συναρτήσεις και είναι η πιο δημοφιλής αυτή τη στιγμή χωρική βάση δεδομένων ανοιχτού κώδικα. Η PostGIS χρησιμοποιείται από ποικίλους οργανισμούς ανά τον κόσμο, περιλαμβανομένων υψηλού ρίσκου κυβερνητικών υπηρεσιών και οργανισμών, αποθηκεύοντας terrabytes δεδομένων και εξυπηρετώντας εκατομμύρια διαδικτυακές κλήσεις ανά ημέρα. Η διαχείριση της βάσης δεδομένων γίνεται μεταξύ άλλων μέσω των pgAdmin και phpPgAdmin. Η είσοδος και έξοδος δεδομένων παρέχεται από πληθώρα εργαλείων μετατροπής (shp2pgsql, pgsql2shp, ogr2ogr, dxf2postgis). Και υπάρχουν πολλά λογισμικά GIS (desktop και διαδικτυακά) για επισκόπηση δεδομένων σε PostGIS. Επίσης, παρέχει ειδικούς τελεστές για τη σύνταξη ερωτημάτων, λειτουργίες συνάθροισης επάνω σε χωρικά δεδομένα καθώς και χωρικές συναρτήσεις. Επιτρέπει επίσης την ανάθεση προβολικών συστημάτων στα χωρικά δεδομένα.

### Google Maps API

Το Google Maps API (Application Programming Interface) αποτελεί μια διεπαφή προγραμματισμού εφαρμογών, το οποίο δίνει στο χρήστη τη δυνατότητα ενσωμάτωσης δυναμικού χάρτη σε εφαρμογές ή ιστοσελίδες κάνοντας χρήση της Javascript. Το Google Maps το παρέχει η Google, η οποία δίνει και έναν οδηγό χρήσης του API για τους προγραμματιστές. Το Google Maps API αποτελεί μια δωρεάν υπηρεσία διαθέσιμη για οποιαδήποτε εφαρμογή. Ο χρήστης θα πρέπει να δημιουργήσει έναν λογαριασμό στην Google και να του δοθεί ένα API key.

Ο προγραμματιστής έχει την δυνατότητα να παραμετροποιήσει τον χάρτη στις ανάγκες του όπως π.χ. να προσθέσει events, controls, overlays τα οποία μπορεί να είναι σημεία (points) ή πολύγωνα (polygons) ή γραμμές (polylines), μπορεί να προσθέσει markers και γενικότερα μέσω της Javascript να οπτικοποιήσει ότι αυτός θέλει. Τα events αποτελούν γεγονότα τα οποία εμφανίζονται μέσω των φυλλομετρητών ιστοσελίδων (web browsers) πάνω σε αντικείμενα του Google Maps. Τα γεγονότα (events) είναι διαφορετικά από τα σίγουρα γεγονότα πάνω στο Μοντέλο Αντικειμενικού Εγγράφου (DOM), το οποίο είναι μια προγραμματιστική διεπαφή που επιτρέπει σε προγράμματα και εκτελέσιμα σενάρια δράσης (scripts), την ενημέρωση του περιεχομένου και του στυλ όλου του εγγράφου με δυναμικό τρόπο. Παραδείγματα γεγονότων του Google Maps API είναι το κλικ του ποντικιού, το διπλό κλικ, η κίνηση του ποντικιού τα οποία

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών



σηματοδοτούν την έναρξη κάποιων διαδικασιών που περιγράφονται στον κώδικα Javascript. Τα overlays είναι αντικείμενα πάνω στο χάρτη που αντιστοιχούν σε συγκεκριμένες συντεταγμένες και γι αυτό μετακινούνται ανάλογα όταν μετακινείται ή αλλάζει εστίαση χάρτη. Χρησιμοποιούνται για το σχεδιασμό και την επίδειξη σημείων, γραμμών ή ολόκληρων περιοχών. Αυτή τη στιγμή η τρέχουσα έκδοση της υπηρεσίας αυτής είναι το Google Maps API v3.

### **BitNami WappStack**

Τέλος το BitNami είναι το ολοκληρωμένο πακέτο λογισμικού που χρησιμοποιήθηκε στην εφαρμογή και περιλαμβάνει όλα τα απαιτούμενα στοιχεία μιας διαδικτυακής εφαρμογής όπως web server, βάση δεδομένων, γλώσσα χρόνου εκτέλεσης. Το BitNami WAPP Stack απλοποιεί σε μεγάλο βαθμό την ανάπτυξη και εξάπλωση των εφαρμογών PHP. Περιλαμβάνει έτοιμες προς λειτουργία εκδόσεις του Apache, PostgreSQL, PHP και phpPgAdmin. Το BitNami WAPP Stack διανέμεται δωρεάν υπό την άδεια Apache 2.0 (<http://bitnami.org/stack/wappstack>). Αυτό το ολοκληρωμένο πακέτο περιλαμβάνει τη γλώσσα προγραμματισμού PHP5, τον Apache Web server, τη βάση δεδομένων MySQL, τη χωρική βάση δεδομένων PostgreSQL, το phpPgAdmin το οποίο όπως αναφέρθηκε παραπάνω είναι ένα διαδικτυακό εργαλείο διαχείρισης της PostgreSQL.

Σε αυτό το κεφάλαιο περιγράφηκαν όλες οι τεχνολογίες, οι υπηρεσίες και τα λογισμικά ανάπτυξης που χρησιμοποιήθηκαν για την υλοποίηση της παρούσας εφαρμογής. Όπως παρατηρήθηκε οι τεχνολογίες που χρησιμοποιήθηκαν ήταν δωρεάν και ανοικτού κώδικα (open source), οι οποίες αποτελούν έναν από τους πιο σύγχρονους τρόπους ανάπτυξης μιας τέτοιας είδους εφαρμογής. Αφού λοιπόν περιγράφηκαν οι διαδικτυακές τεχνολογίες σειρά έχει η ανάλυση των απαιτήσεων του χρήστη, η ανάλυση της αρχιτεκτονικής της εφαρμογής, ο σχεδιασμός της με τη χρήση της UML και η περιγραφή του τρόπου με τον οποίο αναπτύχθηκε η εφαρμογή από τη σκοπιά του προγραμματιστή. Όλα αυτά περιέχονται στο επόμενο κεφάλαιο.

## ΚΕΦΑΛΑΙΟ 4

### Αρχιτεκτονική

Στο κεφάλαιο αυτό θα ασχοληθούμε με την αρχιτεκτονική της εφαρμογής. Συγκεκριμένα, θα εξετάσουμε τις απαιτήσεις ενός χρήστη για μία «Location based» εφαρμογή σε έξυπνα κινητά τηλέφωνα με λειτουργικό Android. Στη συνέχεια θα ασχοληθούμε και θα αναλύσουμε την αρχιτεκτονική και τον σχεδιασμό της εφαρμογής οποίος πραγματοποιήθηκε με τη χρήση UML. Τέλος θα γίνει περιγραφή του τρόπου με τον οποίο έγινε η υλοποίηση της εφαρμογής δηλαδή με τον τρόπο που εργάστηκε ο προγραμματιστής.

### ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

Όπως αναφέραμε αρχικά θα αναλύσουμε τις απαιτήσεις ενός χρήστη για μια εφαρμογή βασισμένη στην γεωγραφική θέση του χρήστη, η οποία θα απευθύνεται σε έξυπνα κινητά (smart phones) Android. Γιατί θα πρέπει πριν την ανάπτυξη μιας εφαρμογής, να γίνει ανάλυση των απαιτήσεων του χρήστη;

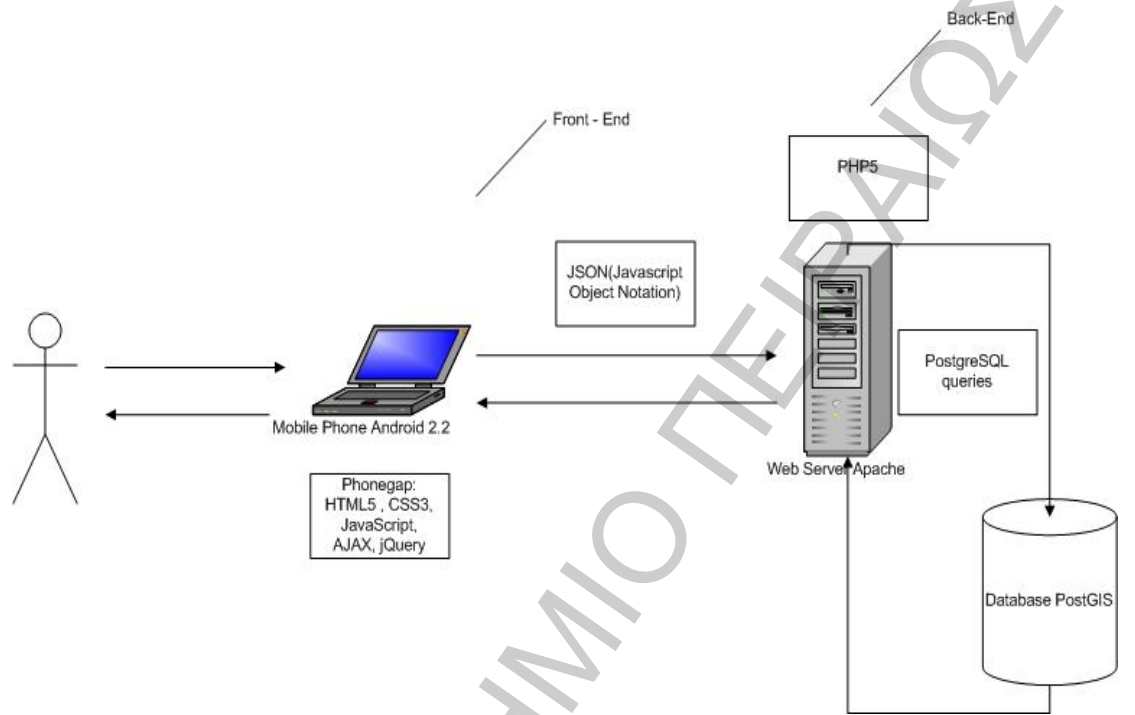
Είναι καλύτερο να προλαμβάνεις, παρά να θεραπεύεις. Είναι αποτελεσματικότερο να φτιάχνεις κάτι προσεκτικά σχεδιασμένο από την αρχή, παρά να το διορθώνεις εκ των υστέρων. Καλύτερα να προηγείσαι παρά να έπεις, να προλαμβάνεις και να προβλέπεις παρά να περιμένεις την έκβαση και κατόπιν να δράσεις. Καλύτερα να έχεις κατασταλαγμένες επιθυμίες και να ξέρεις τι ζητάς. Αυτές είναι εν πολλοίς οι βασικές αρχές που διέπουν την ανάλυση απαιτήσεων (requirements analysis), όσον αφορά στην πληροφορική και τα διάφορα έργα που περιλαμβάνει. Η ανάλυση απαιτήσεων είναι μια διαδικασία κατάρτισης μιας λίστας, όπου αναφέρονται οι προδιαγραφές που πρέπει να πληροί το προϊόν (η λύση) πληροφορικής που πρόκειται να δημιουργηθεί ή να εγκατασταθεί.

Η ιδέα για τη δημιουργία μιας τέτοιας εφαρμογής για έξυπνα κινητά τηλέφωνα γεννήθηκε από την θεώρηση πως η ανάγκη που δημιουργείται κυρίως στο χρήστη που βρίσκεται σε εξωτερικό περιβάλλον να αποκτήσει πληροφορίες για μέρη που βρίσκονται γύρω από τη γεωγραφική του θέση, ανάγκη η οποία γεννιέται λιγότερο σε ένα χρήστη που βρίσκεται σε ένα εσωτερικό περιβάλλον.

Ο χρήστης θέλει να έχει μια εφαρμογή στο κινητό του τηλέφωνο, η οποία να του δίνει τη δυνατότητα αναζήτησης των πιο κοντινών σε αυτόν σημεία ενδιαφέροντος. Τα αποτελέσματα αυτά του δίνουν τη δυνατότητα να τα βλέπει πάνω σε κάποιο χάρτη. Επίσης, να μπορεί να βλέπει για κάθε σημείο ενδιαφέροντος περαιτέρω πληροφορίες όπως τη χιλιομετρική απόσταση από αυτόν, το όνομα του ή το τηλέφωνο του κ.α. Ο χρήστης θα πρέπει να έχει τη δυνατότητα στοχευμένης αναζήτησης. Είναι σημαντικό τα σημεία ενδιαφέροντος να ταξινομούνται με κάποιο τρόπο όπως ανά κατηγορία(π.χ. εστιατόρια, υπηρεσίες υγείας, κυβερνητικές και δημόσιες υπηρεσίες κτλ.) Επίσης, είναι σημαντικό να μπορεί να καθορίζει τη χιλιομετρική απόσταση αναζήτησης των σημείων ενδιαφέροντος. Άλλωστε η εξατομίκευση αποτελεί παράγοντα αιχμής για όλες τις νέες διαδικτυακές εφαρμογές. Θα πρέπει επίσης να δίνεται στο χρήστη η δυνατότητα να προσθέτει ένα σημείο ενδιαφέροντος σε περίπτωση που θεωρεί ότι δεν υπάρχει ήδη σαν εγγραφή στη βάση δεδομένων. Κατ' αυτόν τον τρόπο η εμπειρία του χρήστη μπορεί να γίνεται πιο ευχάριστη και να ανανεώνεται ώστε να μπαίνει στον πειρασμό να ασχολείται με την εφαρμογή μεγαλύτερο χρονικό διάστημα κάθε φορά. Η στατικότητα και η μη κλιμάκωση της ποιότητας των προσφερόμενων υπηρεσιών αποτελούν κακό παράγοντα για του δημιουργούς διαδικτυακών εφαρμογών.

## ΑΡΧΙΤΕΚΤΟΝΙΚΗ

Αποφασίστηκε η ανάπτυξη και η υλοποίηση της εφαρμογής να γίνει βασιζόμενη στο μοντέλο «Πελάτη εξυπηρετητή» ή Client-Server. Τα κύρια τμήματα της εφαρμογής είναι ένας Web Server Apache που εξυπηρετεί τις αιτήσεις των χρηστών, μια χωρική βάση δεδομένων PostgreSQL με επέκταση PostGIS και η εφαρμογή στο κινητό τηλέφωνο. Η παρακάτω εικόνα απεικονίζει ακριβώς την αρχιτεκτονική που ακολουθεί η συγκεκριμένη εφαρμογή.



**Εικόνα 13**

Σύμφωνα με αυτή, ο χρήστης υποβάλλει το αίτημα του μέσω της εφαρμογής του κινητού τηλεφώνου. Αρχικά ο χρήστης επιλέγει από το κινητό την ακτίνα αναζήτησης. Αυτό είναι προαιρετικό βήμα ή απλώς μπορεί να αφήσει τη προεπιλεγμένη η οποία ορίζεται στα 500 μέτρα.

Η εφαρμογή επικοινωνεί μέσω της τεχνολογίας Ajax με τον εξυπηρετητή Web Server Apache. Ο εξυπηρετητής βασίζεται στο μοντέλο ή αρχιτεκτονική MVC(Model-View- Controller), την οποία περιγράψαμε παραπάνω. Το Ajax Request του Client «χτυπάει» στον Ελεγκτή (Controller) μεταφέροντας τέσσερις μεταβλητές. Ας δούμε το link του Request:

```
http://sofioskostas.dyndns-home.com/gmaps/index.php/php_genjson?lat=38.020779&lon=23.796558&category=Automotive&radius=1000
```

Το link αυτό χτυπάει στον Ελεγκτή(Controller) ο οποίος βρίσκεται στη πλευρά του Server με όνομα php\_genjson.php. Επίσης, μεταφέρει όπως βλέπουμε τέσσερις μεταβλητές το latitude και longitude της γεωγραφικής θέσης του χρήστη, την κατηγορία που επέλεξε και την ακτίνα που μπορεί επίσης να επέλεξε ή να είναι η προεπιλεγμένη που έχει οριστεί. Ο Ελεγκτής (Controller) με τη σειρά του, καλεί την αντίστοιχη συνάρτηση (public function) του μοντέλου όπου περνάει τις μεταβλητές που στάλθηκαν από το Ajax. Η συνάρτηση του μοντέλου κάνοντας χρήση της \$\_GET παίρνει τις μεταβλητές και τις αποθηκεύει σε μεταβλητές της PHP.

```
$lat = $_GET['lat'];
```

```
$lon = $_GET['lon'];
```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```
$category = $_GET['category'];
```

```
$radius = $_GET['radius'];
```

Στη συνέχεια η συνάρτηση του μοντέλου υλοποιεί ένα ερώτημα στη χωρική βάση δεδομένων.

```
$query = $this->db->query("SELECT gid, category,name,st_x(the_geom),
st_y(the_geom),ST_Distance((ST_Transform(ST_GeomFromText(the_geom,
4326),2100)),(ST_Transform(ST_GeomFromText('POINT($lon $lat)', 4326),2100)))/1000 as
dist
```

```
FROM greece_poi
```

```
WHERE ((ST_Transform(ST_GeomFromText(the_geom,
4326),2100))@ (ST_Buffer(ST_Transform(ST_GeomFromText('POINT($lon $lat)',
4326),2100),$radius)))
```

```
AND category='$category' ;");
```

Η βάση δεδομένων θα επιστρέφει τις συντεταγμένες της κατηγορίας των σημείων ενδιαφέροντος που ζητήθηκαν μαζί με το κλειδί τους την κατηγορία τους, το όνομα τους και την απόσταση από τη γεωγραφική θέση του χρήστη. Το αποτέλεσμα το επιστρέφει στον Ελεγκτή (Controller) υπό τη μορφή object array. Ο Ελεγκτής (Controller) με τη σειρά του το μετατρέπει τον πίνακα αυτό σε μορφή JSON(JavaScript Object Notation) κάνοντας χρήση της συνάρτησης json\_encode. Το JSON είναι και η απάντηση του Server προς τον Client. Με την ανάλογη επεξεργασία στη πλευρά του Client η Javascript τοποθετεί τις συντεταγμένες στα Google Maps.

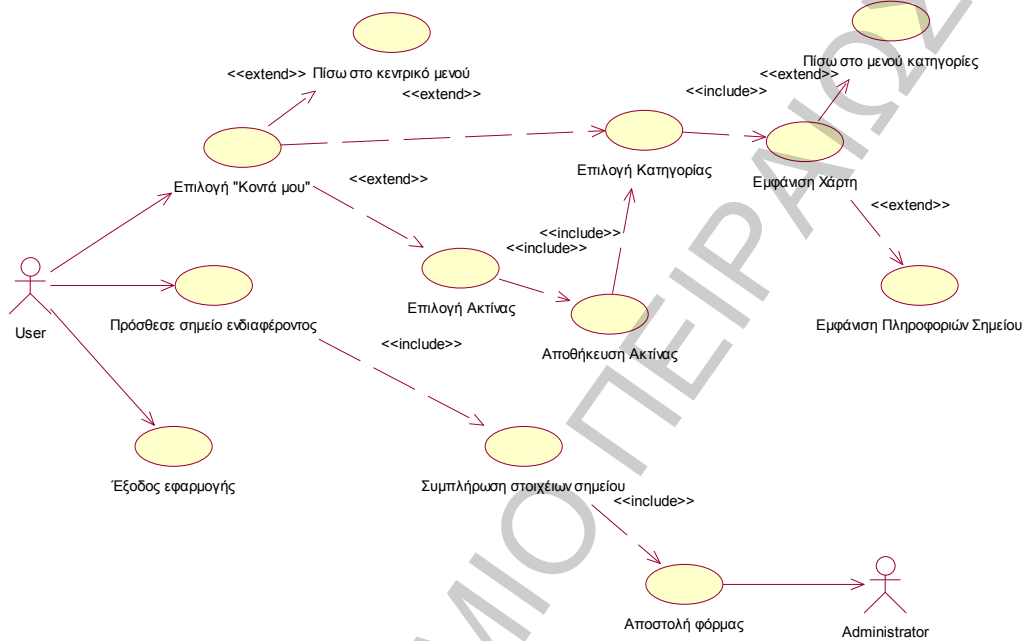
## ΣΧΕΔΙΑΣΜΟΣ

Στη συνέχεια θα αναλύσουμε τον σχεδιασμό της εφαρμογής με τη μέθοδο Rational Unified Process(Επαναληπτική Ενοποιημένη Διαδικασία) κάνοντας χρήση UML. Αρχικά να πούμε λίγα λόγια για τη μέθοδο αυτή :

Η RUP είναι μια μέθοδος διαχείρισης, σχεδιασμού και ανάπτυξης έργων λογισμικού μεγάλης κλίμακας. Ουσιαστικά επιτρέπει σε νέες εκδόσεις του έργου να υλοποιούν και νέες απαιτήσεις που εξελίσσονται όσο το σύστημα υλοποιείται. Καθοδηγείται από μελέτες χρήσης (use-cases) και χρησιμοποιεί τη UML σαν γλώσσα μοντελοποίησης. Το μοντέλο αυτό προσφέρει πλούσιο πλαίσιο υποστήριξης της διαδικασίας. Η RUP μια από τις πιο σύγχρονες και περιζήτητες διαδικασίες ανάλυσης, σχεδίασης και ανάπτυξης συστημάτων. Αναγνωρίζει και αντιμετωπίζει ρεαλιστικά τη καθημερινή πραγματικά της δυναμικής ανάγκης αλλαγής των απαιτήσεων που παρατηρείται σε μεγάλης κλίμακας έργα.

Η εμπειρία δείχνει ότι το 30-50% των πραγματικών απαιτήσεων εμφανίζεται μετά τη φάση της ανάλυσης και αφού έχει ξεκινήσει η διαδικασία υλοποίησης του έργου. Επίσης, παρέχει τη δυνατότητα ελέγχου και έγκαιρου εντοπισμού και περιορισμού του κινδύνου (risk management), διατέμνοντας το έργο σε επιμέρους υπό έργα και αφήνονται είτε όλη την ομάδα είτε μέρος αυτής να ασχοληθεί πρώτα και επισταμένα με τα τμήματα που ενέχουν μεγαλύτερο ρίσκο αποτυχίας. Επιτρέπει τη βήμα-βήμα ανάπτυξη του έργου κάνοντας συνδυασμένη σταδιακή (λίγο-λίγο) ανάλυση-σχεδιασμό-υλοποίηση. Από την αρχή του έργου και μέχρι το τέλος δίνει τη δυνατότητα σε όλες της ομάδες (ελέγχου, ανάπτυξης, ανάλυσης) να συμμετέχουν στο έργο και να συνεισφέρουν την εμπειρία τους. Η ύπαρξη πολλών επαναλήψεων δημιουργεί τη δυνατότητα της διόρθωσης και του επαναπροσδιορισμού των επιλογών και χτίζει εμπειρία στο σύνολο των ομάδων που συνεργάζονται. Εκ των πραγμάτων έχει τη φιλοσοφία της βήμα-προς-βήμα ανάλυσης και ανάπτυξης και άρα έχει σαν βασικό συστατικό την ανάπτυξη θεμελιωδών τμημάτων κώδικα (component). Έτσι αποφεύγει τις «τελικές λύσεις» τύπου Big-Bang που έχουν μεγάλο κίνδυνο αποτυχίας ενώ ταυτόχρονα εξασφαλίζει την επιθυμητή επεκτασιμότητα (modularity).

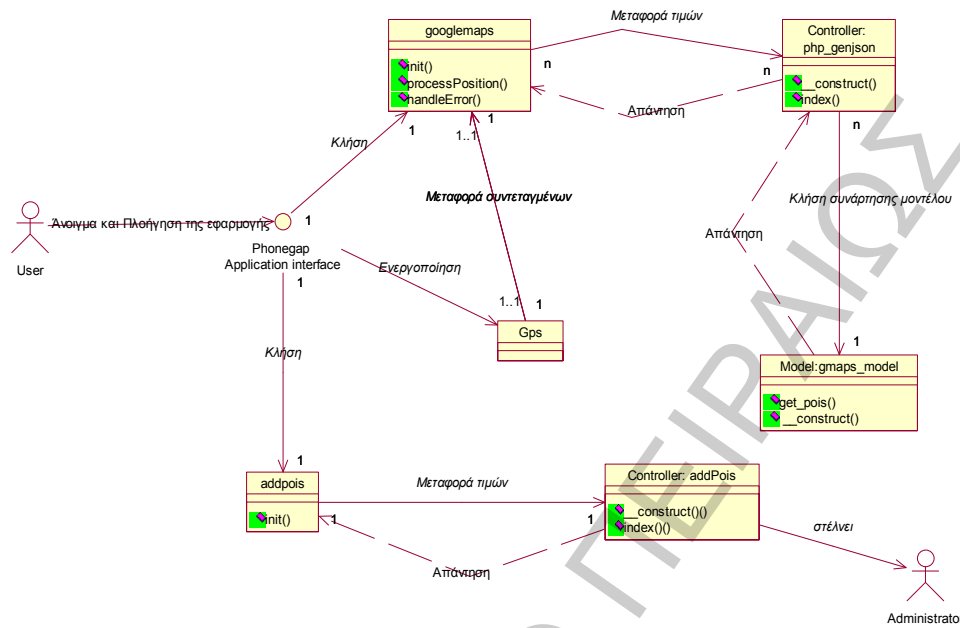
### Διαγράμματα Περιπτώσεων Χρήσης (Use Cases)



**Εικόνα 14**

Το παραπάνω διάγραμμα περιπτώσεων χρήσης συνοψίζει, σε γενικές γραμμές, την αρχική ιδέα της εφαρμογής που αναπτύχθηκε. Οι δράστες που ενεργοποιούν το σύστημα, όπως φαίνεται στο διάγραμμα, είναι ο χρήστης της εφαρμογής του κινητού και ο διαχειριστής της εφαρμογής. Ο διαχειριστής λαμβάνει με Email την φόρμα που συμπλήρωσε ο χρήστης για να προσθέσει κάποιο σημείο ενδιαφέροντος, την ελέγχει και στη περίπτωση που τα στοιχεία είναι ορθά τότε δημιουργεί την νέα εγγραφή στην βάση δεδομένων. Ο χρήστης από την άλλη είναι ο κύριος δράστης της εφαρμογής. Κάνοντας εκκίνηση την εφαρμογή αρχικά πρέπει να επιλέξει το «κοντά μου» για να βρει τα κοντινά σημεία που υπάρχουν γύρω από τη θέση του, και την επιλογή «πρόσθεσε σημείο» σε περίπτωση που θέλει να προσθέσει κάποιο σημείο ενδιαφέροντος. Επίσης στο αρχικό μενού έχει την επιλογή να κάνει έξοδο της εφαρμογής πατώντας το ανάλογο κουμπί. Αφού λοιπόν επιλέξει «κοντά μου», στη συνέχεια του εμφανίζεται ένα μενού με τις κατηγορίες των σημείων ενδιαφέροντος όπου θα πρέπει να επιλέξει αυτή που τον ενδιαφέρει. Επίσης έχει την δυνατότητα πριν επιλέξει την κατηγορία σημείου, να διαμορφώσει τα μέτρα της ακτίνας που επιθυμεί να γίνει η αναζήτηση. Επιλέγοντας την κατηγορία εμφανίζεται ο χάρτης (Google Maps) όπου φαίνεται η γεωγραφική θέση του χρήστη και τα σημεία ενδιαφέροντος γύρω από αυτό. Ο χρήστης έχει τη δυνατότητα να επιλέξει τις πληροφορίες όπου του εμφανίζονται πληροφορίες σχετικά με τα σημεία ενδιαφέροντος που του δείχνει ο χάρτης κάθε φορά. Τέλος ο χρήστης μπορεί να ακολουθήσει το κουμπί μια θέση πίσω που θα τον μεταφέρει στο μενού με τις κατηγορίες σημείων και από εκεί πίσω στο κεντρικό μενού και να κάνει έξοδο της εφαρμογής.

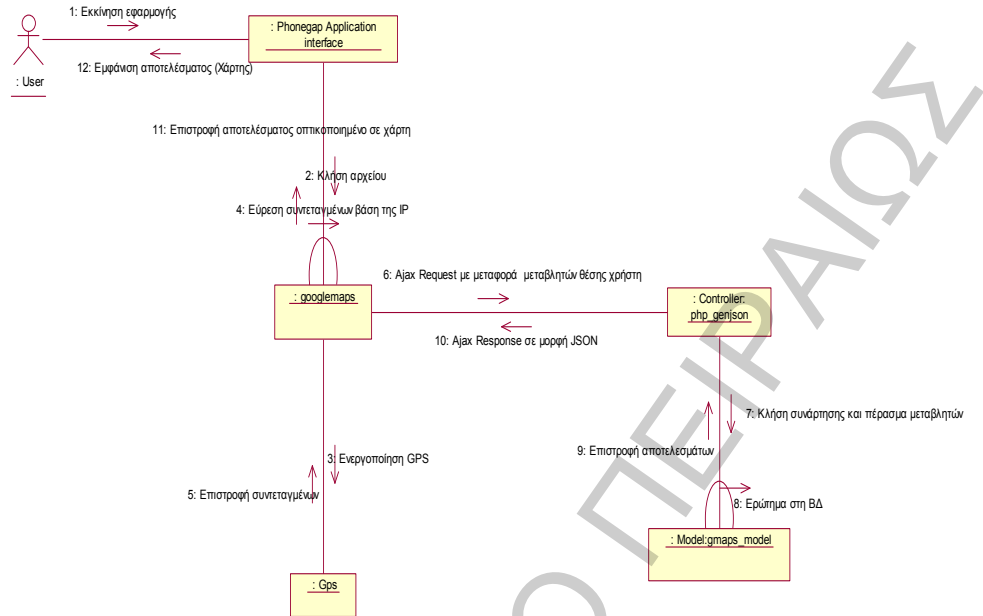
**Διαγράμματα κλάσεων (Class Diagrams)**



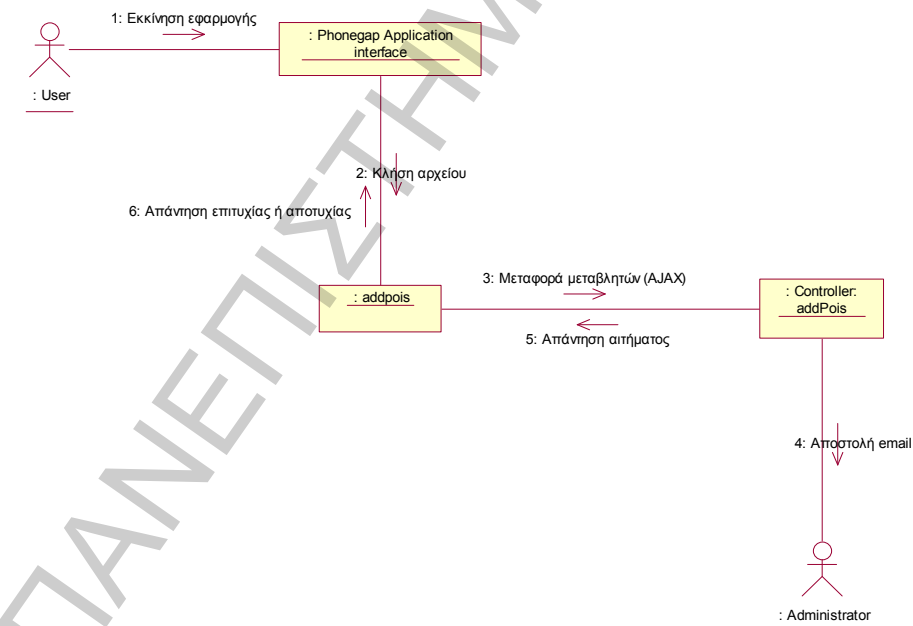
**Εικόνα 15**

Επόμενο βήμα είναι ο σχεδιασμός του διαγράμματος τάξεων (ή κλάσεων). Η αρχική ιδέα του σχεδιασμού περιλαμβάνει έξι κλάσεις και μια διεπαφή. Η διεπαφή ουσιαστικά περιλαμβάνει τα αρχεία HTML της εφαρμογής, δηλαδή την εμφάνιση που χρησιμοποιεί ο χρήστης για να πλοηγηθεί μέσα σε αυτή. Είναι υπεύθυνη στο να καλέσει τις κλάσεις googlemaps και addpois. Η κλάση googlemaps έχει σαν λειτουργία την εύρεση των συντεταγμένων του χρήστη, την μεταφορά μεταβλητών στη πλευρά του Server πραγματοποιώντας AJAX Request και την οπτικοποίηση, της απάντησης του Εξυπηρετητή, στο Google Maps. Η κλάση addpois έχει ως λειτουργία μέσω AJAX να στείλει τις μεταβλητές, οι οποίες προέκυψαν από συμπλήρωση φόρμας του χρήστη, στη κλάση Controller:addPois. Η κλάση Controller:php\_genjson πραγματοποιεί κλήση της ανάλογης συνάρτησης της κλάσης Model:gmaps\_model στην οποία μεταφέρει τις μεταβλητές που «περάστηκαν» μέσω AJAX από τη κλάση googlemaps. Η κλάση Model:gmaps\_model με τη σειρά της λαμβάνει τις μεταβλητές, πραγματοποιεί SQL ερώτημα στη Βάση δεδομένων και επιστρέφει ένα πίνακα (array) στη κλάση Controller:php\_genjson. Η τελευταία κωδικοποιεί τον πίνακα σε μορφή JSON και τον επιστρέφει πίσω στη κλάση googlemaps. Η κλάση Controller:addPois έχει ως λειτουργία να λάβει τις μεταβλητές που μεταφέρθηκαν από τη κλάση addpois και με την ανάλογη επεξεργασία να στείλει email στον διαχειριστή.

**Διαγράμματα Συνεργασίας (Collaboration Diagrams)**



**Εικόνα 16**



**Εικόνα 17**

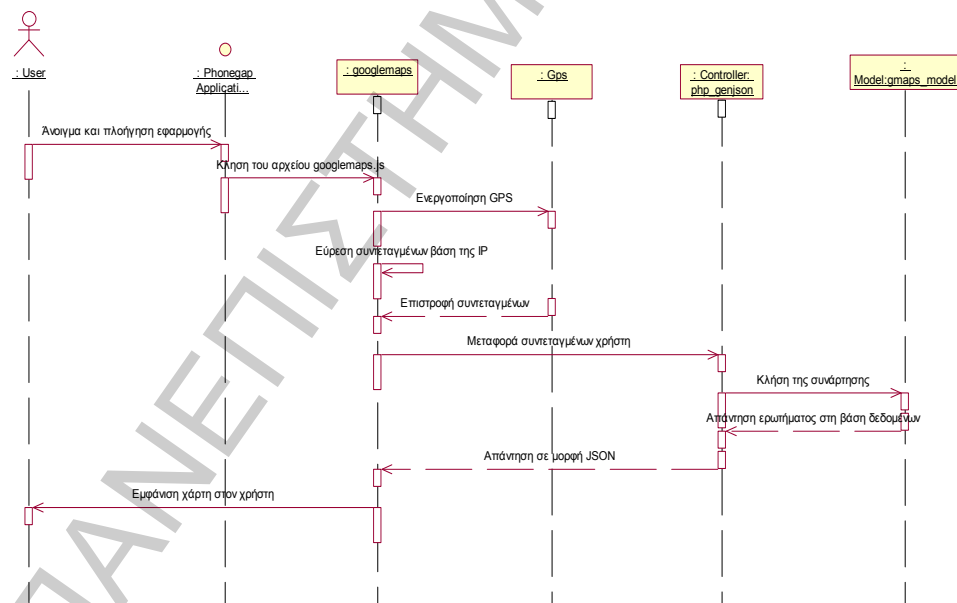
Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Τα διαγράμματα συνεργασίας είναι *διαγράμματα αλληλεπίδρασης* (συμπεριφοράς), που παρουσιάζουν τον τρόπο με τον οποίο διαφορετικά αντικείμενα σχετίζονται και ανταλλάσσουν μηνύματα μεταξύ τους. Ουσιαστικά δείχνουν μια αλληλεπίδραση αντικειμένων με έμφαση στη δομική τους οργάνωση. Περιλαμβάνουν αντικείμενα, σχέσεις μεταξύ αντικειμένων, αριθμημένα μηνύματα.

Στο πρώτο σχήμα παρουσιάζεται η διαδικασία εύρεσης των κοντινότερων σημείων ενδιαφέροντος γύρω από τη θέση του χρήστη. Αρχικά ο χρήστης ανοίγει την εφαρμογή. Στη συνέχεια επιλέγοντας τη κατηγορία σημείων ενδιαφέροντος γίνεται κλήση του αρχείου googlemaps.js. Με τη σειρά του αυτό επικοινωνεί με τον Ελεγκτή(Controller):php\_genjson στον οποίο μεταφέρει μεταβλητές τιμές. Ο τελευταίος καλεί τη συνάρτηση του Μοντέλου Model:gmaps\_model στο οποίο μεταφέρει τις μεταβλητές τιμές. Στο μοντέλο πραγματοποιείται ερώτημα στη βάση δεδομένων και επιστρέφει το αποτέλεσμα πίσω στον Controller:php\_genjson. Αυτός με τη σειρά του μεταφέρει πίσω στο googlemaps.js την απάντηση του μοντέλου. Επεξεργάζοντας την απάντηση, το αρχείο οπτικοποιεί στη διεπαφή το χάρτη, με τα αποτελέσματα δηλαδή με τα σημεία ενδιαφέροντος.

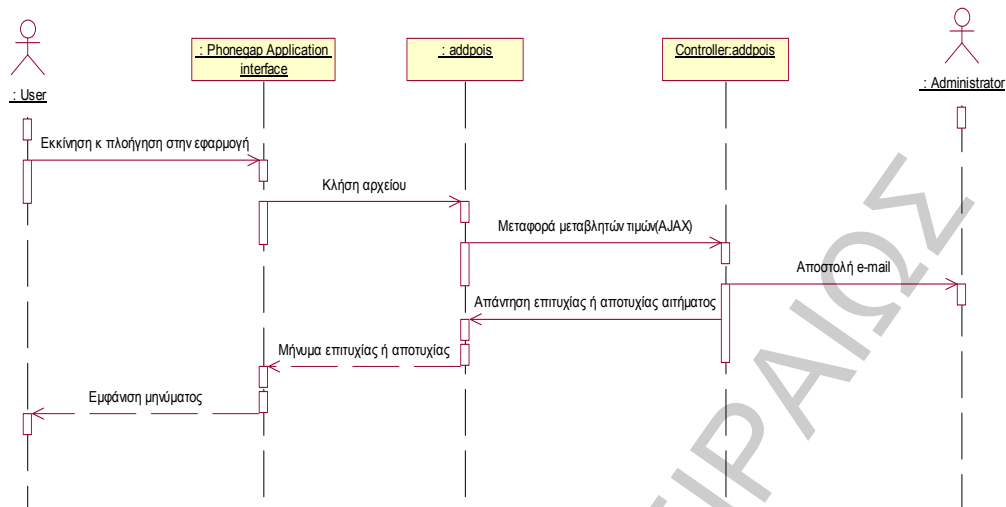
Στο δεύτερο σχήμα παρουσιάζεται η διαδικασία πρόσθεσης σημείων ενδιαφέροντος από το χρήστη. Αφού ο χρήστης ανοίξει την εφαρμογή ή βρεθεί στο αρχικό μενού, επιλέγει να προσθέσει σημεία. Συμπληρώνει τη φόρμα πρόσθεσης σημείων και πατάει υποβολή. Τότε καλείται το αρχείο addpois το οποίο μεταφέρει, τις μεταβλητές τιμές που συμπλήρωσε ο χρήστης στη φόρμα, στον Ελεγκτή (Controller) Ελεγκτή(Controller):addPois και αυτός με τη σειρά του μετά από μια ανάλογη επεξεργασία αποστέλλει e-mail στο διαχειριστή. Επίσης, απαντάει πίσω στο addpois για την επιτυχία ή αποτυχία του request. Το addpois απαντάει μέσω της διεπαφής στο χρήστη με μήνυμα επιτυχίας ή αποτυχίας.

### Διαγράμματα Ακολουθίας (Sequence Diagrams)



Εικόνα 18





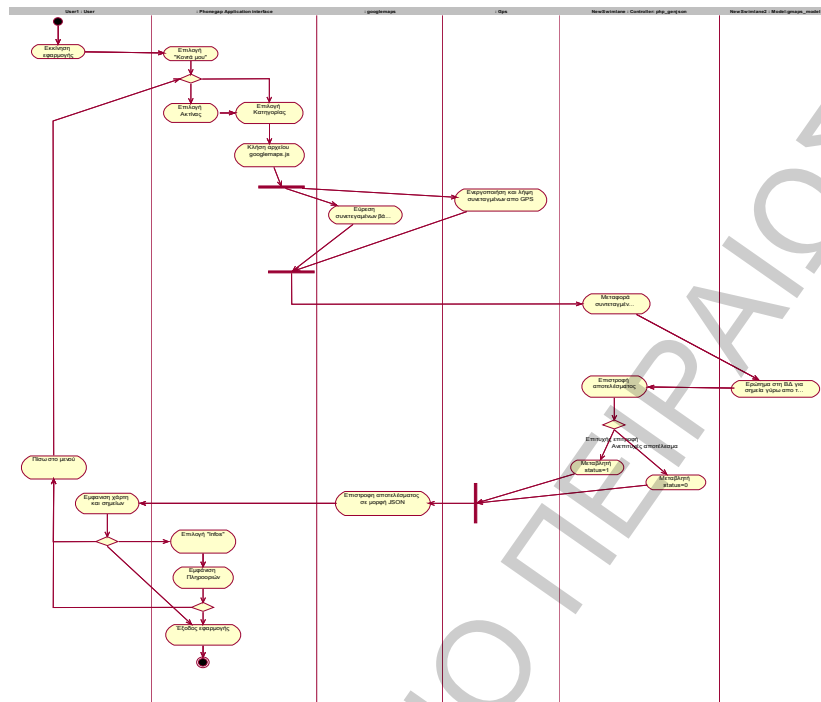
Εικόνα 19

Τα Διαγράμματα Ακολουθίας παρουσιάζουν μια αλληλεπίδραση αντικειμένων με έμφαση στη χρονική σειρά ανταλλαγής μηνυμάτων. Εστιάζουν στο πώς στέλνονται και λαμβάνονται μηνύματα ανάμεσα στα αντικείμενα. Έχουν δύο άξονες, τον κάθετο άξονα που δείχνει τον χρόνο κι τον οριζόντιο που φανερώνει ένα σύνολο αντικειμένων. Κάθε αντικείμενο του οριζόντιου άξονα παριστάνεται από ένα ορθογώνιο που έχει υπογραμμισμένο το όνομα της κλάσης. Επίσης μια διακεκομμένη κάθετη γραμμή ονομάζεται γραμμή ζωής του αντικειμένου(lifeline) και δείχνει την εκτέλεση του αντικειμένου κατά την διάρκεια της ακολουθίας. Η ανάγνωση του ακολουθιακού διαγράμματος γίνεται από πάνω προς τα κάτω, βλέποντας με αυτόν τον τρόπο, την ακολουθία των μηνυμάτων με την πάροδο του χρόνου.

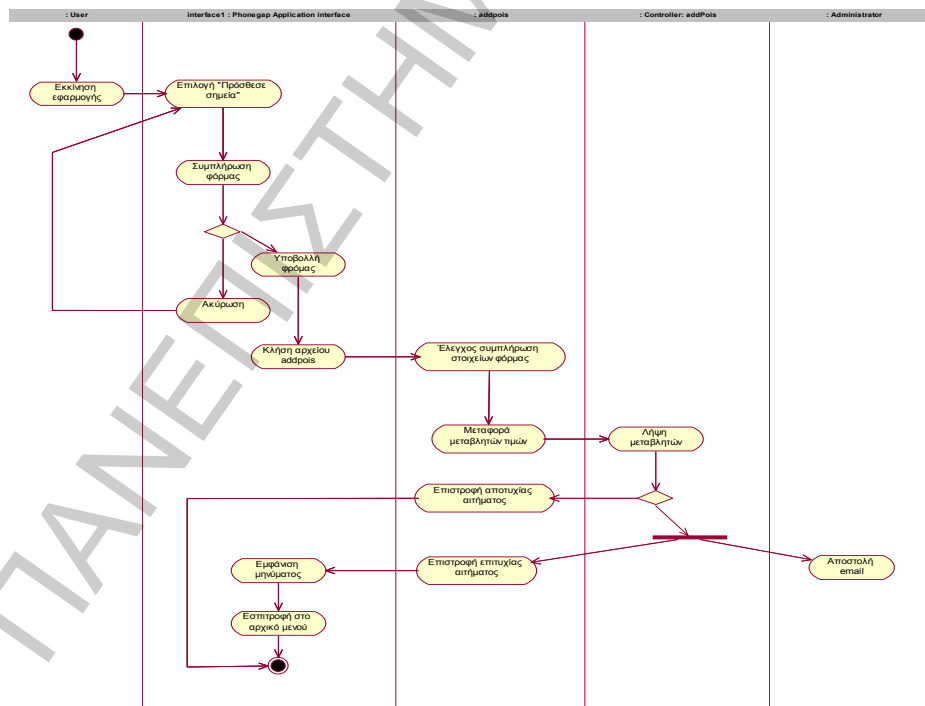
Στο πρώτο σχήμα απεικονίζεται η διαδικασία εύρεσης κοντινών σημείων γύρω από τη θέση του χρήστη. Αφού ο χρήστης ανοίγει την εφαρμογή και επιλέξει την κατηγορία σημείων ενδιαφέροντος που θέλει, η διεπαφή του κινητού καλεί το αρχείο googlemaps. Το τελευταίο ενεργοποιεί το GPS του κινητού και τρέχοντας μια συνάρτηση βρίσκει τις συντεταγμένες βάση της IP. Το GPS στέλνει συντεταγμένες βάση δορυφόρου πίσω στο googlemaps. Το αρχείο στέλνει τις συντεταγμένες μαζί με άλλες μεταβλητές στον Ελεγκτή(Controller):php\_genjson ο οποίος καλεί την ανάλογη συνάρτηση της κλάσης Model:gmaps\_model. Η κλάση Model:gmaps\_model εκτελεί ερώτηση στη Βάση δεδομένων και επιστρέφει το αποτέλεσμα στον Ελεγκτή(Controller):php\_genjson. Ο τελευταίος στέλνει πίσω στο googlemaps την απάντηση του μοντέλου, σε μορφή JSON και αυτό με τη σειρά του έπειτα από την ανάλογη επεξεργασία εμφανίζει το Google Map στην διεπαφή.

Στο δεύτερο σχήμα απεικονίζεται η διαδικασία πρόσθεσης σημείων ενδιαφέροντος. Αφού ο χρήστης ανοίξει την εφαρμογή ή βρεθεί στο αρχικό μενού και επιλέξει την πρόσθεση σημείων, τότε συμπληρώνει τη φόρμα και την υποβάλλει. Η διεπαφή καλεί το αρχείο addpois το οποίο στέλνει τις μεταβλητές των πεδίων της φόρμας στην κλάση Controller:addPois. Η τελευταία, αφού η αίτηση είναι επιτυχής και έπειτα από επεξεργασία, στέλνει e-mail στον διαχειριστή και παράλληλα απαντάει πίσω στην κλάση addpois την επιτυχία ή την αποτυχία της αίτησης. Το addpois με τη σειρά του επιστρέφει μήνυμα επιτυχία ή αποτυχίας στο χρήστη μέσω της διεπαφής.

**Διαγράμματα Δραστηριότητας(Activity Diagrams)**



**Εικόνα 20**



**Εικόνα 21**

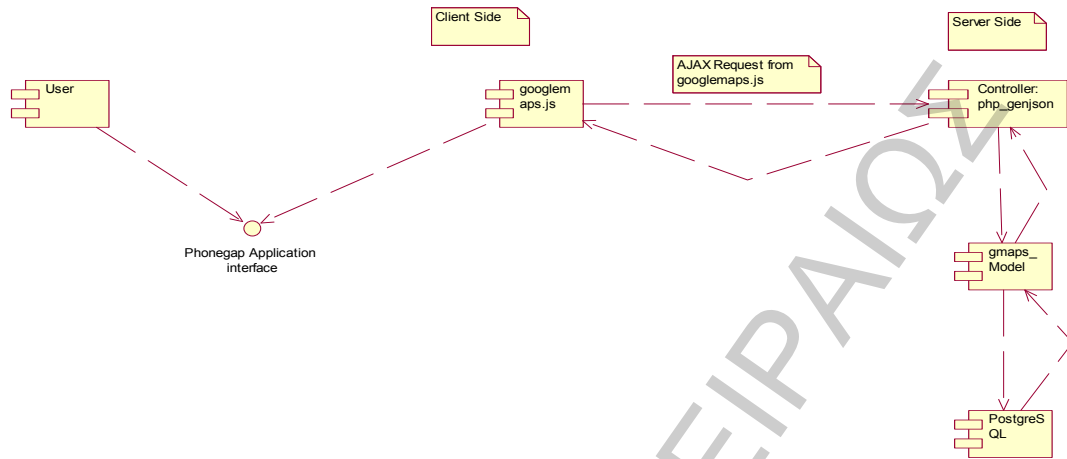
Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Τα Διαγράμματα Δραστηριοτήτων(Activity Diagrams) έχουν ως στόχο την καταγραφή των ενεργειών που εφαρμόζονται κατά την εκτέλεση μιας λειτουργίας (στιγμιότυπο της υλοποίησης της λειτουργίας). Περιγράφουν την εσωτερική λειτουργία ενός αντικειμένου και τις συσχετιζόμενες ενέργειες που εφαρμόζονται και πώς αυτές επηρεάζουν τα γύρω αντικείμενά τους. Υποδεικνύουν τον τρόπο που ένα στιγμιότυπο κάποιου use-case μπορεί να εφαρμοστεί στο πλαίσιο των αλλαγών της κατάστασης του αντικειμένου.

Η εικόνα 20 απεικονίζει τη διαδικασία εύρεσης των κοντινότερων σημείων, σε σχέση με τη γεωγραφική θέση του χρήστη. Αρχικά γίνεται εκκίνηση της εφαρμογής και ο χρήστης επιλέγει από το μενού «Κοντά μου». Στη συνέχεια του εμφανίζεται το επόμενο μενού με τις κατηγορίες των σημείων ενδιαφέροντος. Στο σημείο αυτό ο χρήστης έχει δύο επιλογές, ή να επιλέξει κατευθείαν την κατηγορία που επιθυμεί ή πρώτα να καθορίσει την ακτίνα αναζήτησης και ύστερα να επιλέξει την κατηγορία. Στη συνέχεια γίνεται κλήση του αρχείου googlemaps.js το οποίο ενεργοποιεί το GPS, βρίσκει τις συντεταγμένες του χρήστη βάση της IP του και μεταφέρει τις μεταβλητές τιμές του στίγματος του χρήστη, της ακτίνας και της κατηγορίας στη κλάση Controller:php\_genjson. Η τελευταία με τη σειρά της, καλεί την ανάλογη συνάρτηση της κλάσης Model:gmaps\_model στην οποία και μεταφέρει τις μεταβλητές τιμές. Η συνάρτηση της κλάσης πραγματοποιεί ένα ερώτημα στην Βάση δεδομένων και επιστρέφει πίσω στην Ελεγκτή (Controller):php\_genjson ένα πίνακα. Η κλάση Controller:php\_genjson ανάλογα με το αποτέλεσμα του ερωτήματος που έγινε προηγουμένως θέτει την μεταβλητή status ίσον με ένα σε περίπτωση που το αποτέλεσμα είναι επιτυχές και στην αντίθετη περίπτωση ίσον με μηδέν. Κωδικοποιεί τον πίνακα σε μορφή JSON και τον στέλνει πίσω στη κλάση googlemaps. Η τελευταία, ύστερα από επεξεργασία της απάντησης, εμφανίζει μέσω της διεπαφής τον χάρτη Google Map στο χρήστη. Στο σημείο αυτό ο χρήστης έχει δύο επιλογές, ή να πλοηγηθεί στο αρχικό μενού και να κάνει έξοδο από την εφαρμογή ή να επιλέξει να δει τις πληροφορίες για τα σημεία ενδιαφέροντος που του εμφανίστηκαν στον χάρτη και έπειτα, να πλοηγηθεί στην εφαρμογή μέχρι να κάνει έξοδο. Σε περίπτωση που το αποτέλεσμα της βάσης είναι κενό, σημαίνει ότι δεν έχουν βρεθεί σημεία ενδιαφέροντος. Τότε η κλάση googlemaps εμφανίζει ένα μήνυμα στο οποίο ρωτάει τον χρήστη εάν θέλει να αυξηθεί η ακτίνα μέχρι να επιστραφεί αποτέλεσμα. Στην περίπτωση αυτή επαναλαμβάνεται η διαδικασία από το σημείο που η τελευταία κλάση ξεκινάει την διαδικασία να στείλει τις μεταβλητές τιμές στη κλάση Controller:php\_genjson, με τη διαφορά ότι η τιμή της ακτίνας έχει αυξηθεί.

Στην εικόνα 21 απεικονίζεται η διαδικασία πρόσθεσης σημείου ενδιαφέροντος από τον χρήστη. Αρχικά, ο χρήστης επιλέγει από το μενού «Πρόσθεση σημείων». Στη συνέχεια, συμπληρώνει τα στοιχεία της φόρμας που του εμφανίζονται. Στο σημείο αυτό έχει δύο επιλογές, ή να υποβάλει τη φόρμα ή να την ακυρώσει και να επιστρέψει στο αρχικό μενού. Εάν υποβάλει την φόρμα τότε η διεπαφή καλεί το αρχείο addpois. Πραγματοποιείται έλεγχος εάν έχουν συμπληρωθεί τα στοιχεία της φόρμας και μεταφέρονται οι τιμές των πεδίων μέσω AJAX στη κλάση Controller:addPois, στην οποία γίνεται η λήψη των τιμών αυτών και καταχωρούνται σε αντίστοιχες μεταβλητές. Στο σημείο αυτό εάν το αίτημα γίνει επιτυχές τότε αποστέλλεται το e-mail στον διαχειριστή και παράλληλα επιστρέφεται πίσω απάντηση επιτυχίας και ο χρήστης επιστρέφει στο αρχικό μενού. Στην αντίθετη περίπτωση επιστρέφεται απάντηση αποτυχίας.

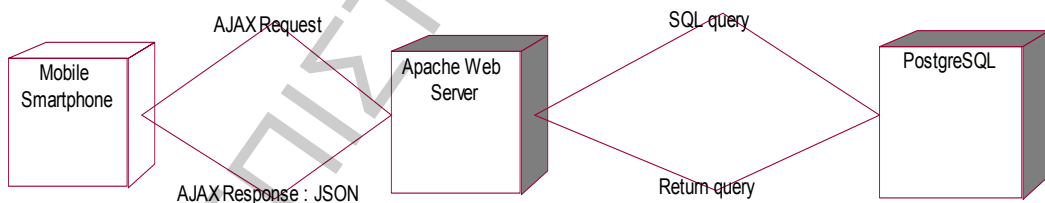
### Διαγράμματα Εξαρτημάτων



Εικόνα 22

Το διάγραμμα εξαρτημάτων είναι ένα *διάγραμμα δομής υλοποίησης* που χρησιμοποιείται για να μοντελοποιήσει πηγαίο κώδικα, εκτελέσιμες εκδόσεις, βάσεις δεδομένων και δυναμικά προσαρμοζόμενα συστήματα. Περιλαμβάνει εξαρτήματα, διεπαφές (*interfaces*) και σχέσεις εξάρτησης, γενίκευσης, σύνδεσης και υλοποίησης. Στο παραπάνω σχήμα γίνεται σαφές η πλευρά του πελάτη και του Εξυπηρετητή. Τα αρχεία της εφαρμογής αποτελούν τα αντικείμενα του διαγράμματος. Με τον τρόπο αυτό φαίνονται καθαρά τα αρχεία στην εφαρμογή και οι σχέσεις μεταξύ τους.

### Διαγράμματα Διανομής



Εικόνα 23

Τα διαγράμματα διανομής (*deployment diagrams*) παρουσιάζουν το πραγματικό περιβάλλον στο οποίο λειτουργεί το λογισμικό. Τα διαγράμματα αυτά παρουσιάζουν συσκευές του υλικού και τις φυσικές μονάδες λογισμικού που διανέμονται στο υλικό. Τα διαγράμματα διανομής αποτελούν ένα μέσο για την τεκμηρίωση της φυσικής αρχιτεκτονικής του συστήματος. Το βασικά στοιχεία ενός τέτοιου διαγράμματος είναι οι επεξεργαστές (*processors*), οι συσκευές (*devices*) καθώς και οι συνδέσεις μεταξύ τους. Το παραπάνω σχήμα δείχνει την αρχιτεκτονική της εφαρμογής που αναπτύχθηκε. Στον ρόλο της συσκευής είναι το κινητό τηλέφωνο του χρήστη, και αυτό γιατί δεν πραγματοποιείται κάποια επεξεργασία πληροφορίας ή δεδομένων. Η επεξεργασία πραγματοποιείται στην πλευρά του Web Server ο οποίος επικοινωνεί με τη βάση δεδομένων. Οπότε και οι δύο αποτελούν του επεξεργαστές της αρχιτεκτονικής της εφαρμογής. Επίσης, η σύνδεση μεταξύ της εφαρμογής στην πλευρά του κινητού (δηλαδή στην

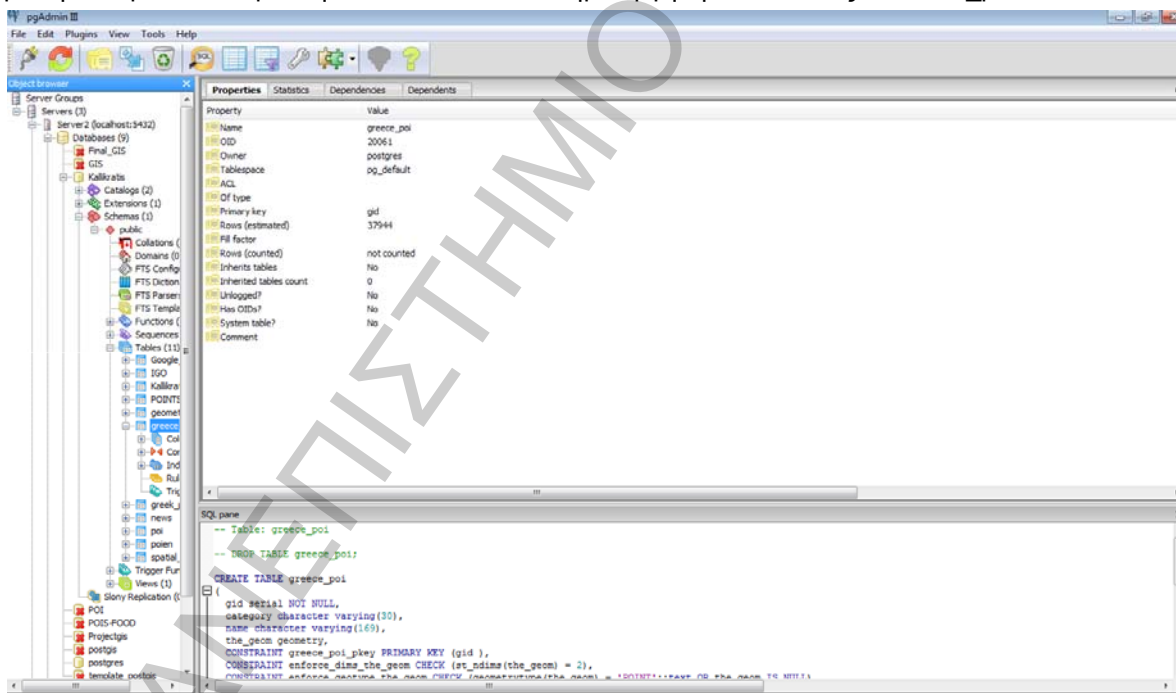
πλευρά του Πελάτη-Client) με τον Web Server ,γίνεται με τη χρήση AJAX. Η επικοινωνία μεταξύ του Server και της Βάσης δεδομένων γίνεται μέσω SQL ερωτημάτων(queries).

## ΑΝΑΠΤΥΞΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Στο σημείο αυτό θα αναλυθεί ο τρόπος που αναπτύχθηκε η εφαρμογή. Θα αναλυθεί ο τρόπος εργασίας και τα βήματα υλοποίησής της. Έως τώρα έχει γίνει η ανάλυση απαιτήσεων του χρήστη, ώστε να ξεκαθαριστούν οι ανάγκες του. Έχει γίνει σαφές το μοντέλο της αρχιτεκτονικής που θα βασιστεί η εφαρμογή καθώς επίσης και οι τεχνολογίες. Επίσης, σχεδιάστηκε ο τρόπος λειτουργίας του συστήματος, με τη χρήση της UML βασιζόμενοι στην Επαναληπτική Ενοποιημένη Διαδικασία(RUP).

Όπως αναφέραμε παραπάνω η αρχιτεκτονική που θα ακολουθήσει η εφαρμογή είναι το μοντέλο Πελάτη-Εξυπηρετητή (Client -Server). Ουσιαστικά η ανάπτυξη της εφαρμογής χωρίζεται σε τρία μέρη, στην πλευρά της χωρικής βάσης δεδομένων, στην πλευρά του Web Server Apache με υποστήριξη της PHP και στην πλευρά της εφαρμογής στο κινητό τηλέφωνο.

Αρχικά έγινε η εγκατάσταση της χωρικής βάσης δεδομένων PostgreSQL με την επέκταση της PostGIS. Για την εγκατάσταση της βάσης δεδομένων αποφασίστηκε η χρήση του πακέτου λογισμικού που περιγράψαμε στο προηγούμενο κεφάλαιο BitNami Wapp. Αφού εγκαταστήθηκε το πακέτο αυτό, ενσωματώθηκε και η επέκταση PostGIS η οποία υποστηρίζει χωρικά δεδομένα και μας δίνει την δυνατότητα χρήσης χωρικών συναρτήσεων. Στη συνέχεια δημιουργήθηκε η βάση δεδομένων με τη χρήση του εργαλείου διαχείρισης της βάσης δεδομένων pgAdminIII. Η βάση δεδομένων ονομάστηκε «Kallikratis» και δημιουργήθηκε ο πίνακας «Greece\_roi».



Εικόνα 24

Για την δημιουργία του πίνακα εκτελέστηκε το παρακάτω SQL ερώτημα:

```

CREATE TABLE greece_poi
(
gid serial NOT NULL,

```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```

category character varying(30),
name character varying(169),
the_geom geometry,
CONSTRAINT greece_poi_pkey PRIMARY KEY (gid ),
CONSTRAINT enforce_dims_the_geom CHECK (st_ndims(the_geom) = 2),
CONSTRAINT enforce_geotype_the_geom CHECK (geometrytype(the_geom) = 'POINT'::text
OR the_geom IS NULL),
CONSTRAINT enforce_srid_the_geom CHECK (st_srid(the_geom) = 2100)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE greece_poi
  OWNER TO postgres;

```

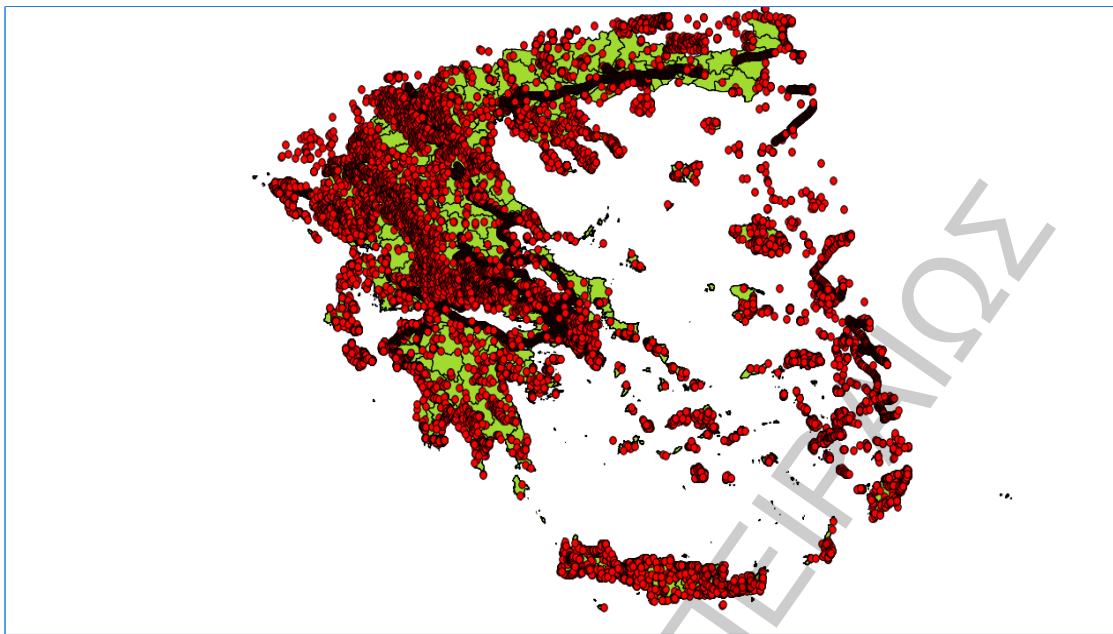
Στο σημείο αυτό έπρεπε να βρεθούν τα δεδομένα που θα καταχωρούνταν στην βάση δεδομένων. Τα δεδομένα βρέθηκαν από την ιστοσελίδα Cloudmade.com η οποία παρέχει γεωγραφικά δεδομένα για όλο τον κόσμο χωρισμένα ανά χώρες και ηπείρους. Το αρχείο που χρησιμοποιήθηκε ήταν σε μορφή shapfile (.shp), μια από τις μορφές που υποστηρίζει η χωρική βάση δεδομένων για απευθείας εισαγωγή(import). Με την απευθείας εισαγωγή ουσιαστικά δεν χρειάζεται να δημιουργήσουμε πίνακα, δημιουργείται αυτόματα μόνος του κατά την διαδικασία εισαγωγής του αρχείου shp. Οπότε, ο προηγούμενος πίνακας αντικαταστάθηκε από τον νέο πίνακα που δημιουργήθηκε κατά την εισαγωγή του αρχείου στη βάση δεδομένων. Για μεγαλύτερη απόδοση στη ταχύτητα της απάντησης της βάσης δεδομένων προστέθηκε ένα ευρετήριο(index) το οποίο χρησιμοποιεί τη στήλη the\_geom:

```

CREATE INDEX greece_poi_the_geom_gist
ON greece_poi
USING gist
(the_geom );

```

Στη συνέχεια, αφού υπήρχαν τα δεδομένα στη βάση έγινε μια ανάλυση των χωρικών δεδομένων και σχεδιάστηκε το ερώτημα το οποίο θα επέστρεφε τα σημεία που βρίσκονται μέσα σε έναν κύκλο με μια ακτίνα x με κέντρο μια γεωγραφική θέση. Για την καλύτερη κατανόηση των δεδομένων αποφασίστηκε η οπτικοποίησή τους. Για το λόγο αυτό εγκαταστάθηκε το λογισμικό Quantum GIS 1.8 το οποίο αποτελεί ένα ελεύθερο και ανοιχτού κώδικα πρόγραμμα που προσφέρει δυνατότητες οπτικοποίησης, τροποποίησης και ανάλυσης χωρικών δεδομένων. Επίσης αποφασίστηκε η χρήση του χάρτη της Ελλάδας για την καλύτερη κατανόηση των σημείων ενδιαφέροντος. Ο χάρτης της Ελλάδας χρησιμοποιήθηκε από τα δεδομένα δημόσιας διοίκησης της ιστοσελίδας geodata.gov.gr η οποία παρέχει ένα αρχείο μορφής shapfile με τον χάρτη της χώρας. Φορτώνοντας τα δεδομένα αυτά στο Quantum GIS 1.8 υπήρξε το παρακάτω αποτέλεσμα:



Εικόνα 25

Στη συνέχεια δημιουργήθηκε το χωρικό ερώτημα SQL για την εύρεση των κοντινότερων σημείων γύρω από μια συγκεκριμένη γεωγραφική θέση. Το ερώτημα που δημιουργήθηκε και χρησιμοποιήθηκε ήταν :

```
SELECT gid, category, name, st_x(the_geom),
st_y(the_geom), ST_Distance((ST_Transform(ST_GeomFromText(the_geom,
4326),2100)),(ST_Transform(ST_GeomFromText('POINT($lon $lat)', 4326),2100)))/1000 as dist
FROM greece_poi
WHERE ((ST_Transform(ST_GeomFromText(the_geom,
4326),2100))@(ST_Buffer(ST_Transform(ST_GeomFromText('POINT($lon $lat)',
4326),2100),$radius)))
AND category='$category'
```

Το παραπάνω ερώτημα ουσιαστικά ζητάει το κλειδί, το όνομα, την κατηγορία το Γεωγραφικό πλάτος (latitude), το Γεωγραφικό μήκος (longitude) και την απόσταση σε μέτρα των σημείων που βρίσκονται μέσα σε μια ακτίνα κύκλου x μέτρων από το σημείο με συντεταγμένες lat και lon. Η συνάρτηση ST\_Distance επιστρέφει την απόσταση δύο γεωμετριών στην δική μας περίπτωση δύο σημείων. Η συνάρτηση ST\_Buffer δημιουργεί έναν κύκλο γύρω από ένα σημείο με μια ακτίνα που καθορίζουμε εμείς. Παρατηρείται ότι οι πληροφορίες των συντεταγμένων της ακτίνας και της κατηγορίας είναι μεταβλητές, αυτό συμβαίνει γιατί ο χρήστης θα καθορίζει δυναμικά τις τιμές από την εφαρμογή στο κινητό του. Δηλαδή, οι συντεταγμένες θα δίνονται δυναμικά από την εφαρμογή του κινητού το οποίο θα βρίσκει τις συντεταγμένες της εκάστοτε θέσης του χρήστη. Επίσης ο χρήστης θα δίνει την ακτίνα αναζήτησης αλλά και την κατηγορία σημείων που επιθυμεί. Τρέχοντας το ερώτημα :

```
SELECT gid, category, name, st_x(the_geom),
st_y(the_geom), ST_Distance((ST_Transform(ST_GeomFromText(the_geom,
4326),2100)),(ST_Transform(ST_GeomFromText('POINT(23.796558 38.020779)',
4326),2100)))/1000 as dist
FROM greece_poi
```

```
WHERE ((ST_Transform(ST_GeomFromText(the_geom,
4326),2100))@ (ST_Buffer(ST_Transform(ST_GeomFromText('POINT(23.796558 38.020779)',
4326),2100),2000)))
```

```
AND category='Automotive'
```

έχουμε το παρακάτω αποτέλεσμα:

gid	integer	category	character varying(30)	name	character varying(169)	st_x	double precision	st_y	double precision	dist	double precision
1	5169	Automotive		Pedestrian Crossing		23.7750792		38.0045872		2.60433273948711	
2	5171	Automotive		Pedestrian Crossing		23.7752		38.0046128		2.59470126779847	
3	5172	Automotive		Pedestrian Crossing		23.7753606		38.0049278		2.56039768103293	
4	5173	Automotive		Pedestrian Crossing		23.7753868		38.0048674		2.56333821828161	
5	5174	Automotive		Pedestrian Crossing		23.7754451		38.0047277		2.57034683215026	
6	5175	Automotive		Pedestrian Crossing		23.7755051		38.0045932		2.57693058013303	
7	5176	Automotive		Pedestrian Crossing		23.7755265		38.0051125		2.53574018907283	
8	5178	Automotive		Pedestrian Crossing		23.7755918		38.0049885		2.54103736707223	
9	5179	Automotive		Pedestrian Crossing		23.7756129		38.0045934		2.57013697885375	
10	5180	Automotive		Pedestrian Crossing		23.7756362		38.0049195		2.54350685756697	
11	5181	Automotive		Pedestrian Crossing		23.7756929		38.0048254		2.5471606091585	
12	5226	Automotive		Pedestrian Crossing		23.7798988		38.0099433		1.89311034778238	
13	5228	Automotive		Pedestrian Crossing		23.780033		38.009868		1.88936861858633	
14	5229	Automotive		Pedestrian Crossing		23.7800999		38.0101618		1.86409186900481	
15	5230	Automotive		Pedestrian Crossing		23.780153		38.0097931		1.88665259133576	
16	5231	Automotive		Pedestrian Crossing		23.7801796		38.0098755		1.87897094411825	
17	5232	Automotive		Pedestrian Crossing		23.7802418		38.0100823		1.8600714539157	
18	5233	Automotive		Pedestrian Crossing		23.780252		38.0099557		1.86838030031499	
19	5235	Automotive		Pedestrian Crossing		23.7803895		38.0101093		1.8481864786451	
20	5244	Automotive		Parking		23.7809858597206		38.0354146		2.12244825982879	
21	5257	Automotive		Parking		23.7817601884863		38.03680875		2.20225483206682	
22	5275	Automotive		Pedestrian Crossing		23.7830847		38.0097638		1.70072138092769	
23	5278	Automotive		Pedestrian Crossing		23.783176		38.0099528		1.68010072496874	
24	5301	Automotive		Pedestrian Crossing		23.7845764		38.0151522		1.22304979148788	
25	5305	Automotive		Pedestrian Crossing		23.7846281		38.0154055		1.2049979611246	
26	5307	Automotive		Pedestrian Crossing		23.7847126		38.0150651		1.21778835571379	
27	5308	Automotive		Pedestrian Crossing		23.7847848		38.0153745		1.1947871230958	
28	5309	Automotive		Pedestrian Crossing		23.7849288		38.0152993		1.18811993845951	
29	5319	Automotive		Parking		23.786622507473		38.0076967		1.69338985093226	
30	5321	Automotive		Pedestrian Crossing		23.7868245		38.0078111		1.69268810214661	

## Εικόνα 26

Παραπάνω έτρεξε το χωρικό ερώτημα που δημιουργήθηκε με σταθερές τιμές και παρατηρείται ότι το αποτέλεσμα είναι ακριβώς αυτό που θέλαμε. Δηλαδή μας επιστρέφει ο κλειδί, το όνομα, την κατηγορία, το γεωγραφικό πλάτος, το γεωγραφικό μήκος και την απόσταση των σημείων ενδιαφέροντος σε μια ακτίνα 2000 μέτρων από τη μια συγκεκριμένη γεωγραφική θέση, πάντα μέσα στον ελλαδικό χώρο.

Στο σημείο αυτό ολοκληρώθηκε η εγκατάσταση της χωρικής βάσης δεδομένων, δημιουργήθηκε η βάση την οποία θα χρησιμοποιεί η εφαρμογή, καταχωρήθηκαν τα δεδομένα των σημείων και δημιουργήθηκε το χωρικό ερώτημα που χρειαζόμαστε για τη λειτουργία της εφαρμογής. Προχωράμε στη δημιουργία του δεύτερου μέρους της εφαρμογής, δηλαδή στην ανάπτυξη της λειτουργικότητας του Web Server Apache. Όπως αναφέραμε και στο προηγούμενο κεφάλαιο και στην περιγραφή της εφαρμογής στην πλευρά του εξυπηρετητή χρησιμοποιήθηκε γλώσσα προγραμματισμού PHP5 με τη χρήση του πλαισίου εργασίας ή αλλιώς framework Codeigniter το οποίο βασίζεται στο πρότυπο ανάπτυξης MVC (model-view-Ελεγκτή (Controller)). Το πακέτο BitNami που εγκαταστάθηκε περιέχει τον Apache και την PHP. Ωστόσο έπρεπε να εγκατασταθεί το framework. Η εγκατάσταση του ήταν απλή έπρεπε να αντιγραφεί ο αποσυμπίεσμένος φάκελος που κατέβηκε από την ιστοσελίδα του Codeigniter, μέσα στον φάκελο htdocs του πακέτου BitNami. Αφού εγκαταστάθηκε με επιτυχία το πλαίσιο εργασίας, πραγματοποιήθηκαν οι κατάλληλες ρυθμίσεις για την επικοινωνία με τη Βάση δεδομένων και για τον βασικό υπερσύνδεσμο που θα ανοίγει στον περιηγητή Ιστού.

Αρχικά, αποφασίστηκε να δημιουργηθεί το μοντέλο το οποίο είναι υπεύθυνο για την επικοινωνία με τη βάση δεδομένων. Στη συνέχεια δημιουργήθηκε ο Ελεγκτής (Controller) οποίος



θα καλεί την συνάρτηση του μοντέλου που πραγματοποιεί την χωρική ερώτηση στην Βάση δεδομένων που αναλύσαμε παραπάνω.

```
$data['greece_poi'] = $this->gmaps_model->get_pois();
```

Με την παραπάνω γραμμή καλείται η συνάρτηση `get_pois()` του μοντέλου καθώς επίσης επιστρέφει και το αποτέλεσμα από την τελευταία αποθηκεύοντας το σε έναν πίνακα. Επίσης, ο Ελεγκτής(Controller) θα πρέπει να κωδικοποιεί το αποτέλεσμα που παίρνει κάθε φορά από το μοντέλο σε μορφή JSON.

```
$result = array();
```

```
$result["data"]=$data["greece_poi"];
```

```
$result["status"]=1;
```

```
$result["message"]="Success!";
```

```
echo json_encode($result);
```

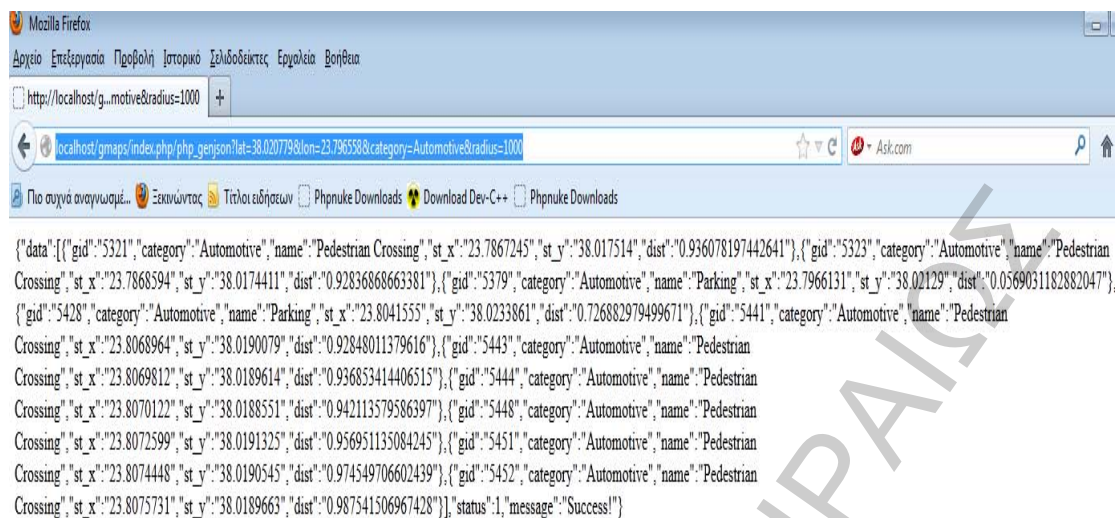
```
return TRUE;
```

Παρατηρούμαι ότι δημιουργείτε ένας νέος πίνακας που περιέχει τον πίνακα που επέστρεψε το μοντέλο, μια μεταβλητή «status» η οποία ανάλογα με την τιμή της δείχνει την επιτυχία η αποτυχία του ερωτήματος άρα και των δεδομένων που επέστρεψαν και ένα μήνυμα. Στην συνέχεια ο πίνακας `result` κωδικοποιείται σε μορφή JSON. Επίσης δημιουργήθηκε και ένας δεύτερος Ελεγκτής(Controller) για την αποστολή της φόρμας που συμπλήρωσε ο χρήστης για την προσθήκη σημείων ενδιαφέροντος. Το αρχείο αυτό με το όνομα `addrpols.php` δεν επικοινωνεί με κάποιο μοντέλο, η λειτουργία του είναι να λάβει τις τιμές των πεδίων της φόρμας που συμπλήρωσε ο χρήστης και να τις προωθήσει με e-mail στο διαχειριστή.

Αφού λοιπόν τελείωσε η ανάπτυξη της πλευράς του Apache Server, έπρεπε να γίνει έλεγχος ότι οι δύο πλευρές που αναπτύχθηκαν δουλεύουν με απόλυτη επιτυχία. Για τον έλεγχο χρησιμοποιήθηκε ένα πρόσθετο εργαλείο (plugin) του περιηγητή Mozilla, το Firebug. Δημιουργήθηκε ο υπέρ σύνδεσμος που θα τρέχει η εφαρμογή στην πλευρά του εξυπηρετητή.

```
http://localhost/gmaps/index.php/php\_genjson?lat=38.020779&lon=23.796558&category=Automotive&radius=1000
```

Παρατηρούμε ότι ο υπέρ σύνδεσμος καλεί το αρχείο `php_genjson.php` το οποίο έχει την ιδιότητα του Ελεγκτή(Controller). Επίσης παρατηρείται ότι το Link αυτό «περνάει» και κάποιες τιμές. Αυτές οι τιμές είναι αυτές οι μεταβλητές που αναφέραμε παραπάνω τις οποίες τις δηλώνει ο χρήστης μέσω των επιλογών του από την εφαρμογή και αυτή με τη σειρά της, τις στέλνει στον εξυπηρετητή. Τρέχοντας το link αυτό το αποτέλεσμα είναι το εξής:



### Εικόνα 27

Παρατηρούμε ότι ο εξυπηρετητής επιστρέφει αποτελέσματα σε μορφή JSON, άρα δουλεύει ακριβώς όπως είχε σχεδιαστεί.

Το τελευταίο μέρος της εφαρμογής είναι η ανάπτυξή της στην πλευρά του Πελάτη(Client) δηλαδή στο κινητό τηλέφωνο του χρήστη με τη χρήση της πλατφόρμας Phonegap. Αρχικά αναπτύχθηκε η συνάρτηση με την οποία θα λαμβάνεται το γεωγραφικό μήκος και πλάτος της θέσης του χρήστη βάσης της IP.

```

function init()
{
    if(navigator.geolocation)
    {
        var options = { frequency: 300 };
        navigator.geolocation.getCurrentPosition(processPosition,
        handleError,{enableHighAccuracy: true }, options);
    }
    else
    {
        navigator.notification.alert("Geolocation not supported in this browser");
    }
}

```

Η παραπάνω συνάρτηση στηρίζεται στο γεωεντοπισμό (geolocation), ο οποίος παρέχει πληροφορίες για τον εντοπισμό της συσκευής, όπως το γεωγραφικό πλάτος και μήκος. Πηγές πληροφορίας αποτελούν το GPS και η θέση μέσω του δικτύου, όπως η διεύθυνση IP, RFID, WiFi και Bluetooth διευθύνσεις MAC, και GSM / CDMA. Το geolocation.getCurrentPosition είναι μια ασύγχρονη συνάρτηση. Επιστρέφει τη τρέχουσα θέση της συσκευής στην επανάκληση του geolocationSuccess με ένα αντικείμενο, τη θέση, ως παράμετρο. Σε περίπτωση σφάλματος γίνεται η geolocationError επανάκληση. Με τη χρήση του {enableHighAccuracy: true }

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

ενεργοποιείται το GPS του κινητού. Το `processPosition` αποτελεί μια συνάρτηση η οποία παίρνει σαν παράμετρο τις συντεταγμένες:

```
var latlng = new google.maps.LatLng(pos.coords.latitude, pos.coords.longitude);
```

Η παραπάνω γραμμή κώδικα υλοποιείται μέσα στη τελευταία συνάρτηση που περιγράφηκε. Το γεωγραφικό μήκος και πλάτος αποθηκεύονται σε μία μεταβλητή η οποία χρησιμοποιείται για την εμφάνιση του σημείου πάνω στον χάρτη των Google Maps. Στη συνέχεια, κρίθηκε απαραίτητο ότι θα πρέπει ο εξυπηρετητής (Server) να βρίσκεται στο διαδίκτυο και όχι να είναι τοπικά στον υπολογιστή (localhost). Για το λόγο αυτό χρησιμοποιήθηκε η υπηρεσία DynDns. Το `dyndns` είναι μια υπηρεσία που προσφέρει domain name δωρεάν και ουσιαστικά δίνει τη δυνατότητα να «σηκωθεί» ο εξυπηρετητής στο διαδίκτυο. Τα βήματα για την απόκτηση της υπηρεσίας ήταν τα εξής: δημιουργία λογαριασμού στο <http://dyn.com/dns/>, δημιουργία του domain name το οποίο είναι <http://sofioskostas.dyndns-home.com> και έλεγχος εάν η πόρτα (port) 80 είναι ανοιχτή. Επίσης, σε περίπτωση που η πόρτα 80 δεν είναι ανοιχτή, θα πρέπει με συγκεκριμένη διαδικασία μέσω του δρομολογητή (Router) να αποκατασταθεί, καθώς και να απενεργοποιηθεί το τείχος προστασίας (Firewall) του.

Το επόμενο βήμα ήταν η υλοποίηση του AJAX Request. Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, η υλοποίηση του AJAX έγινε με τη χρήση του πλαισίου εργασίας jQuery. Ένα κομμάτι του jQuery.AJAX είναι το παρακάτω:

```
$.ajax({
    url:'http://sofioskostas.dyndns-home.com/gmaps/index.php/php_genjson?lat='+lat+'&lon='+lon+'&category='+category+'&radius='+radius,
    dataType:'json',
    success:function(data, textStatus, jqXHR)
    {
```

Παρατηρούμε ότι στην αρχή συντάσσεται ο υπερσύνδεσμος που θα ενεργοποιήσει ή καλύτερα θα αιτηθεί στον Server να επεξεργαστεί τα δεδομένα που του στέλνει και να του επιστρέψει μια απάντηση (Response). Η απάντηση θα είναι το αποτέλεσμα του χωρικού ερωτήματος που αναλύθηκε παραπάνω, σε μορφή JSON. Για το λόγο αυτό, στο πεδίο `dataType` αναγράφεται η μορφή των δεδομένων που θα επιστραφούν, δηλαδή στη συγκεκριμένη περίπτωση αναγράφεται 'json'. Το πεδίο "success" χρησιμοποιείται στη περίπτωση που το ερώτημα και η απάντηση γυρίσει με επιτυχία. Τότε γίνονται οι κατάλληλες ενέργειες. Στην αντίθετη περίπτωση χρησιμοποιώντας τη συνάρτηση `λάθους`, εμφανίζεται κατάλληλο μήνυμα.

```
error:function(jqXHR, textStatus, errorThrown){
    navigator.notification.alert('AJAX request failed');
}
```

Αφού λειτουργεί με επιτυχία το AJAX αίτημα τότε δημιουργείται η λειτουργικότητα για την επεξεργασία της απάντησης του εξυπηρετητή. Στόχος είναι να καταχωρηθούν τα δεδομένα του πίνακα μέσα στο Google Maps έτσι ώστε να οπτικοποιηθούν στο χάρτη.

Επίσης, ένα πρόβλημα που αντιμετωπίστηκε κατά την ανάπτυξη της πλευράς του πελάτη, ήταν η τοπική αποθήκευση τιμών. Δηλαδή έπρεπε να αποθηκεύεται προσωρινά η τιμή της ακτίνας σε περίπτωση που τη διαμόρφωνε ο χρήστης. Σε περίπτωση που ο χρήστης δεν επιλέξει ακτίνα θα πρέπει να αποθηκεύεται μέχρι τη έξοδο της εφαρμογή η προεπιλεγμένη τιμή της. Η λύση σε αυτό το πρόβλημα ήταν η δυνατότητα, που δίνει η HTML5, της τοπικής αποθήκευσης. Η παρακάτω γραμμή βρίσκεται στο μενού με τις κατηγορίες και αποτελεί την προεπιλεγμένη ακτίνα που αναφέραμε παραπάνω.

```
localStorage.setItem("radius", 500);
```

Σε περίπτωση που ο χρήστης επιλέξει την ακτίνα, τότε η προεπιλεγμένη τιμή αντικαθίσταται με τη νέα. Το slider είναι η τιμή που λαμβάνεται από τη μπάρα επιλογής ακτίνας.

```
slider =$("#slider-0").val();
localStorage.setItem("radius", slider);
```

Η παρακάτω γραμμή βρίσκεται στο αρχείο googlemaps.js το οποίο είναι υπεύθυνο για την αίτηση στον εξυπηρετητή και τη δημιουργία των Google Maps. Αποτελεί τη διαδικασία λήψης της μεταβλητής που αποθηκεύτηκε τοπικά και προσωρινά.

```
radius=localStorage.getItem("radius")
```

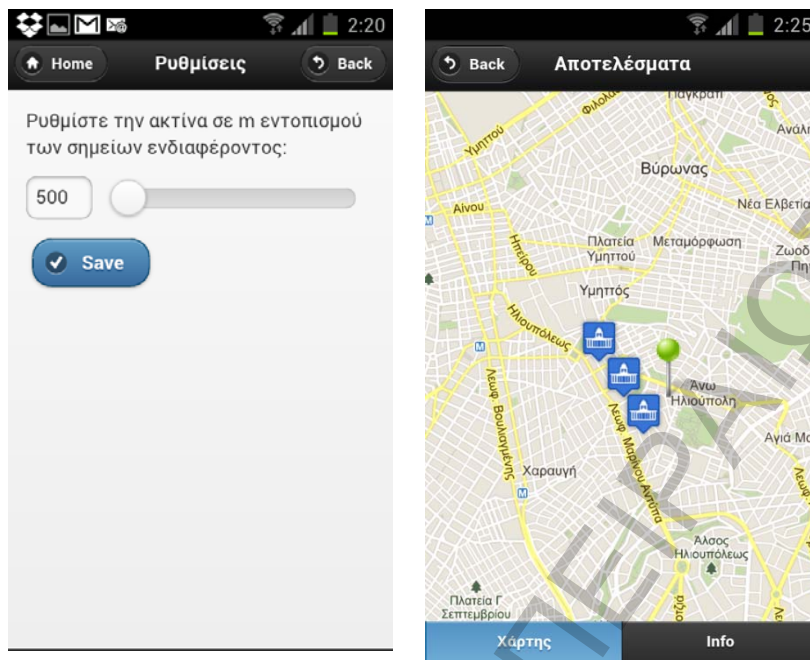
Να σημειωθεί ότι το πλήρες αρχείο javascript στο οποίο υπάρχει όλη η παραπάνω λειτουργικότητα υπάρχει στο παράρτημα α.

Επόμενο βήμα, ήταν η δημιουργία της διεπαφής της εφαρμογής. Σαν διεπαφή, εννοείται η δημιουργία των μενού, η σωστή και εύκολη πλοήγηση του χρήστη στην εφαρμογή. Στο σημείο αυτό έγινε βεβαίως η χρήση της HTML5 σε συνεργασία με το πλαίσιο εργασίας (framework) jQuery Mobile. Παρακάτω παρατίθενται εικόνες με το γραφικό περιβάλλον και την πλοήγηση της εφαρμογής.



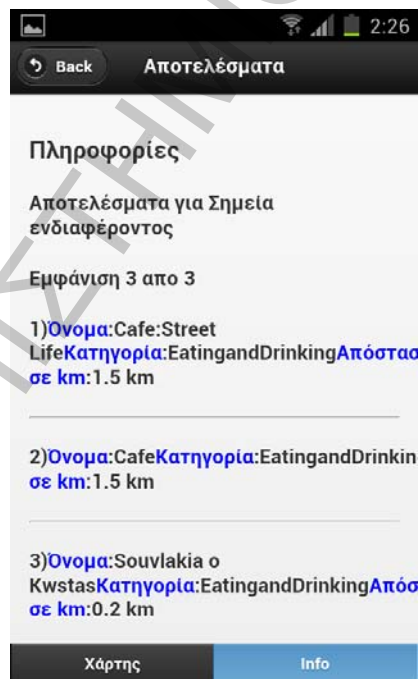
**Εικόνα 28**

Στις παραπάνω εικόνες φαίνεται το αρχικό μενού και το μενού μετά την επιλογή «κοντά μου» με τις κατηγορίες των σημείων.



Εικόνα 29

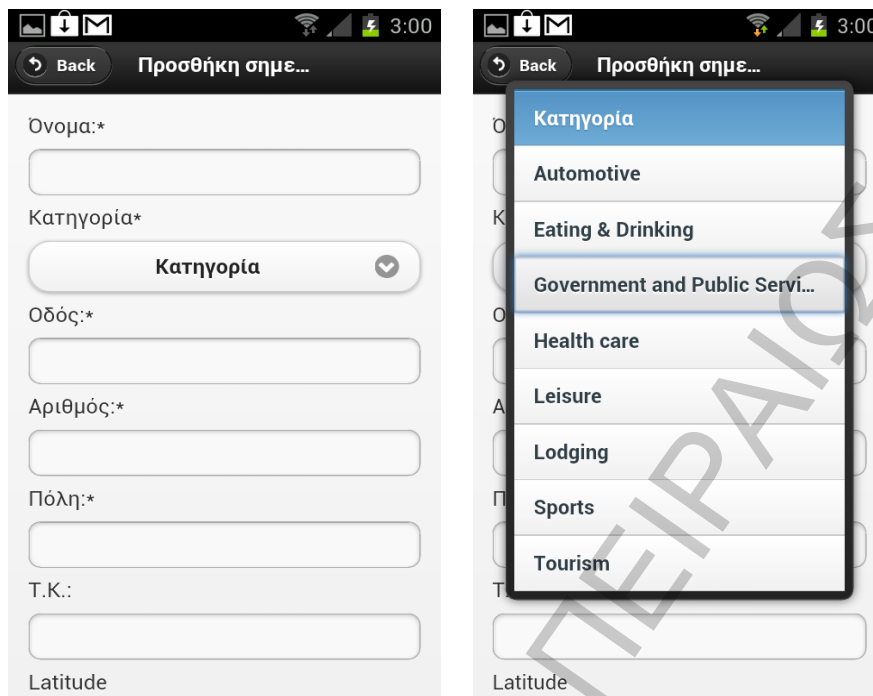
Στις δύο παραπάνω εικόνες φαίνεται η επιλογή της ακτίνας αναζήτησης και ένα παράδειγμα αναζήτησης σημείων ενδιαφέροντος γύρω από μια θέση. Το ίδιο φαίνεται και στις επόμενες εικόνες.



Εικόνα 30

Οι παρακάτω εικόνες απεικονίζουν τη φόρμα συμπλήρωσης για την προσθήκη σημείων ενδιαφέροντος.

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

**Εικόνα 31**

Στο σημείο αυτό τελειώνει το κεφάλαιο 4, στο οποίο περιγράφηκε η ανάλυση απαιτήσεων της εφαρμογής, η αρχιτεκτονική της, ο σχεδιασμός της και ο τρόπος ανάπτυξης της. Συμπεραίνεται, ότι πραγματοποιώντας μια σωστή ανάλυση απαιτήσεων και ένα σωστό σχεδιασμό της αρχιτεκτονικής αλλά και της εφαρμογής, ο τρόπος ανάπτυξης της είναι πιο ξεκάθαρος, πιο μεθοδευμένος, με μεγαλύτερη σαφήνεια στις ανάγκες του χρήστη και με μικρό ποσοστό κινδύνου για λάθος στην διαδικασία υλοποίησης. Επίσης, παρατηρήθηκε ο μεγάλος βαθμός βοήθειας που προσέφερε το πρότυπο ανάπτυξης MVC, σε συνδυασμό με το πλαίσιο εργασίας Codeigniter τόσο στην υλοποίηση, δηλαδή στον προγραμματισμό, όσο και στο σχεδιασμό της εφαρμογής. Να σημειωθεί ότι ο πλήρης κώδικας της εφαρμογής και στην πλευρά του πελάτη αλλά και στη πλευρά του εξυπηρετητή, βρίσκεται στο παράρτημα Α.

## ΚΕΦΑΛΑΙΟ 5

### ΣΥΜΠΕΡΑΣΜΑΤΑ

Στα προηγούμενα κεφάλαια παρουσιάστηκε η ανάλυση, η σχεδίαση και η υλοποίηση μιας εφαρμογής για κινητά τηλέφωνα με λειτουργικό Android, η οποία βασίζεται στη πληροφορία της θέσης του χρήστη. Η εφαρμογή αυτή, επιτρέπει στο χρήστη να ανακαλύπτει τα σημεία ενδιαφέροντος που βρίσκονται κοντά του, λαμβάνοντας επιπλέον πληροφορίες για αυτά. Επιπλέον, ο χρήστης έχει τη δυνατότητα να προσθέτει δικά του σημεία, καθώς επίσης και να ρυθμίζει την ακτίνα αναζήτησης τους σε περίπτωση που το επιθυμεί. Ακόμα, στην εργασία πραγματοποιήθηκε βιβλιογραφική έρευνα, σχετικά με άλλες εργασίες και εφαρμογές που υπάρχουν και σχετίζονται, είτε σε μεγαλύτερο είτε μικρότερο βαθμό με τη παρούσα μεταπτυχιακή διατριβή.

Όπως περιγράφηκε και στο τρίτο κεφάλαιο, η εφαρμογή στηρίζεται σε λογισμικό και τεχνολογίες ανοικτού κώδικα (open source). Ακολούθως, η χρήση τέτοιων τεχνολογιών προσέφερε αρκετά πλεονεκτήματα για την υλοποίηση της εφαρμογής. Το μηδενικό κόστος απόκτησης των λογισμικών και της χρησιμοποίησής τους ελεύθερα από το χρήστη, αποτελεί ένα από τα σημαντικότερα πλεονεκτήματα στη διαδικασία υλοποίησης των εφαρμογών εν γένει. Επιπλέον, υπήρξε ευρεία υποστήριξη προβλημάτων από σχετικά forum για την κατανόηση της φύσης των προβλημάτων που δημιουργήθηκαν κατά τη διάρκεια της ανάπτυξης, με αποτέλεσμα την επίλυση τους.

Μια από τις μεγαλύτερες προκλήσεις υλοποίησης της εφαρμογής, ήταν η εκμάθηση και η χρήση της πλατφόρμας Phonegap και αυτό γιατί αποτελεί μια καινοτόμα τεχνολογία και φιλοσοφία στην ανάπτυξη εφαρμογών για κινητά τηλέφωνα. Το μεγαλύτερο πλεονέκτημα της χρήσης της πλατφόρμας, ήταν ότι χρησιμοποιήθηκαν οι κλασσικές διαδικτυακές τεχνολογίες (HTML5, Javascript, CSS3), αναπτύσσοντας μια φορά την εφαρμογή, έχοντας τη δυνατότητα να μετατραπεί και να λειτουργήσει για κάθε γνωστό λειτουργικό σύστημα των έξυπνων κινητών τηλεφώνων, όπως το Android, Windows Phone 7, IOS, Symbian κτλ. Ωστόσο, ένα μειονέκτημα που εντοπίστηκε, είναι ότι ο χρόνος εκτέλεσης (runtime) της εφαρμογής προστίθεται στο μέγεθος της. Μεγάλο ενδιαφέρον παρουσίασε η χρήση του πλαισίου εργασίας JQuery Mobile, γιατί προσφέρει τη δυνατότητα προσθήκης γραφικών και λειτουργιών πλοήγησης μέσα στην εφαρμογή σε μικρότερο χρόνο από ότι θα χρειαζόταν σε διαφορετική περίπτωση. Η χρήση της τεχνολογίας AJAX σε συνδυασμό με το πλαίσιο εργασίας JQuery προσέφερε πολύ μεγάλη απόδοση, με συνέπεια τη μείωση χρόνου επικοινωνίας μεταξύ πελάτη και εξυπηρετητή. Σε αυτό βέβαια, βοήθησε και η χρήση της τεχνολογίας JSON με την οποία το μέγεθος δεδομένων επιστροφής από τον εξυπηρετητή ήταν πολύ μικρότερο από ότι χρησιμοποιώντας τη τεχνολογία XML. Επιπροσθέτως, πρέπει να αναφερθεί ότι η αρχιτεκτονική της εφαρμογής που βασίστηκε στο μοντέλο Πελάτη – Εξυπηρετητή, οδήγησε στην επιλογή των παραπάνω δύο τεχνολογιών.

Στη πλευρά του εξυπηρετητή, η χρήση του πλαισίου εργασίας Codeigniter προσέφερε μέγιστη απόδοση ταχύτητας επικοινωνίας μεταξύ της βάσης δεδομένων και του διαδικτυακού εξυπηρετητή, αλλά και με την πλευρά του πελάτη. Το συμπέρασμα είναι ότι το Codeigniter αποτελεί ένα απλό, εύχρηστο αλλά και πολύ δυνατό πλαίσιο εργασίας, με το οποίο μπορεί ο προγραμματιστής να δημιουργεί εφαρμογές και λειτουργίες που με την απλή χρήση της PHP θα χρειαζόταν διπλάσιο ή και τριπλάσιο χρόνο και κόπο. Επίσης, ένα μεγάλο πλεονέκτημα στη χρήση του Codeigniter ήταν ότι βασίζεται στην αρχιτεκτονική MVC η οποία έκανε ξεκάθαρο τον τρόπο που θα λειτουργεί η εφαρμογή από τη σκοπιά της αρχιτεκτονικής. Επιπλέον, ο συνδυασμός του MVC και της αρχιτεκτονικής Πελάτη - Εξυπηρετητή που επιλέχθηκε, η ανάπτυξη της εφαρμογής χωρίστηκε σε τρία μέρη. Αυτό προσέφερε πλήρη κατανόηση της λειτουργίας και του ρόλου όλων των τμημάτων της, τη μείωση του χρόνου υλοποίησης, αλλά και τη μείωση του χρόνου εντοπισμού λαθών και προβλημάτων που δημιουργήθηκαν.

Συμπεραίνεται, ότι ο συνδυασμός των τεχνολογιών που χρησιμοποιήθηκαν αλλά και της αρχιτεκτονικής που επιλέχθηκε, μείωσε το χρόνο υλοποίησης κατά πολύ μεγάλο ποσοστό με συνέπεια την δημιουργία μιας ολοκληρωμένης, αποδοτικής, γρήγορης αλλά και με πολύ καλά γραφικά εφαρμογής.

Χαρακτηριστικό της εφαρμογής, είναι οι υπηρεσίες βασισμένες στη γεωγραφική θέση του χρήστη (Location Based Services - LBS), διαπιστώνοντας ότι αποτελούν υπηρεσίες, πολλά υποσχόμενες στον κλάδο των τηλεπικοινωνιών και της πληροφορικής. Οι τεχνολογίες που υποστηρίζουν τέτοιες υπηρεσίες, ωριμάζουν συνεχώς και αυτό τις καθιστά ολοένα και πιο εμπορικά θελκτικές αλλά και οικονομικά συμφέρουσες για εκμετάλλευση. Ήδη άρχισαν να μπαίνουν συλλήβδην στη ζωή των περισσότερων ανθρώπων στις ανεπτυγμένες και βιομηχανοποιημένες χώρες. Η πλατφόρμα ανοικτού κώδικα ανάπτυξης εφαρμογής για λειτουργικό Android για κινητές συσκευές, διαπιστώθηκε ότι παρέχει όλες τις δυνατότητες για την ανάπτυξη μιας ολοκληρωμένης υπηρεσίας βασισμένης στη θέση του χρήστη. Υποστηρίζει όλες τις τεχνολογίες που διαθέτουν οι σύγχρονες συσκευές κινητής τηλεφωνίας, πρόσβαση στο διαδίκτυο μέσω Wi-Fi ή 3G, περιήγηση στον παγκόσμιο ιστό, τεχνολογίες εντοπισμού θέσης GPS κτλ.

Ωστόσο, κατά τη συγγραφή της εργασίας, αλλά και την ανάπτυξη της εφαρμογής, υπήρξαν περιορισμοί. Ο πρώτος περιορισμός αφορούσε τα δεδομένα των σημείων ενδιαφέροντος. Στόχος της εφαρμογής είναι η εύρεση σημείων ενδιαφέροντος σε όλη την Ελλάδα. Για το λόγο αυτό, η δημιουργία εγγραφών από την αρχή ήταν αδύνατο γιατί θα έπρεπε να εισαχθούν στη βάση δεδομένων χιλιάδες εγγραφές. Έτσι, τα δεδομένα που χρησιμοποιήθηκαν προήλθαν από έναν ιστότοπο, ο οποίος παρέχει γεωγραφικά δεδομένα για ακαδημαϊκούς ή προσωπικούς σκοπούς για κάθε χώρα στον κόσμο. Τα δεδομένα αυτά ήταν για όλη την Ελλάδα, ωστόσο κάποια ήταν ελλιπή, ως προς το όνομα και την κατηγορία του σημείου. Επιπλέον, τα περισσότερα ονόματα ήταν με λατινικούς χαρακτήρες. Αυτό δημιουργεί μια μελλοντική επέκταση, η οποία θα αναφερθεί παρακάτω. Ο δεύτερος περιορισμός ήταν το κόστος των δημοσιεύσεων και άρθρων, τα οποία σχετίζονται με τη παρούσα μεταπτυχιακή διατριβή. Έτσι δεν ήταν δυνατή η συγγραφή μια ολοκληρωμένης βιβλιογραφικής έρευνας στο δεύτερο κεφάλαιο.

Η εφαρμογή που αναπτύχθηκε σε αυτή την εργασία σχεδιάστηκε και οργανώθηκε έτσι ώστε να είναι πλήρως επεκτάσιμη. Είναι χωρισμένη σε διάφορα συστατικά ανάλογα με τις παρεχόμενες λειτουργίες. Έτσι είναι δυνατό μελλοντικά, προγραμματιστές να μπορέσουν να επεκτείνουν λειτουργίες της χωρίς να επηρεάζονται τα υπόλοιπα τμήματα της εφαρμογής. Μια μελλοντική επέκταση αποτελεί η περαιτέρω ανάπτυξη της αλληλεπίδρασης με τα σημεία ενδιαφέροντος. Θα ήταν εφικτό να μπορούν να προστεθούν ακόμα περισσότερες πληροφορίες για κάθε σημείο ενδιαφέροντος, όπως βίντεο, κατάλογος με πιθανούς υπερσυνδέσμους που θα μπορούσε να επισκεφθεί ένας ενδιαφερόμενος χρήστης και επιπλέον αριθμός εικόνων και στοιχείων επικοινωνίας. Επίσης, θα μπορούσε να προστεθεί περαιτέρω ανάπτυξη της αλληλεπίδρασης με το χρήστη. Στα σημεία που βρίσκονται σε οδικό δίκτυο, θα μπορούσε να προστεθεί λειτουργικότητα για τη λήψη οδηγιών με τη χρήση αυτοκινήτου χρησιμοποιώντας κάποιον αλγόριθμο οποίος θα έβρισκε τη βέλτιστη διαδρομή σε κάθε σημείο. Μια ακόμα σκέψη, είναι η δημιουργία ιστοσελίδας ώστε ο κάθε χρήστης να δημιουργεί ένα λογαριασμό και να μπορεί να προσθέτει σημεία, ενώ αντιστοίχως, κάθε καταστηματούχης ενός σημείου ενδιαφέροντος να προσθέτει προσφορές για ένα συγκεκριμένο χρονικό διάστημα που θα ορίζει εκείνος. Επιπλέον, μπορεί να δοθεί η επιλογή στο χρήστη να αναζητεί σημεία ενδιαφέροντος από μια διεύθυνση την οποία θα καθορίζει ο ίδιος, χωρίς να είναι απαραίτητα η εκάστοτε γεωγραφική του θέση. Να δοθεί η δυνατότητα στο χρήστη μέσω του ιστότοπου που αναφέρθηκε προηγουμένως η προσθήκη σχολίων και κριτικών για ένα σημείο ενδιαφέροντος, τα οποία θα μπορούν να είναι διαθέσιμα για τους υπόλοιπους χρήστες της εφαρμογής.

Τέλος, μελλοντική επέκταση θα αποτελούσε η πρόσθεση στοιχείων από τα μέσα κοινωνικής δικτύωσης όπως το Facebook ή το Twitter. Ο χρήστης θα έχει τη δυνατότητα να δει, εάν σε ένα



σημείο ενδιαφέροντος που παρουσιάζεται κοντά του, μπορεί να έχει κάνει λόγου χάριν “check in” κάποιος φίλος του.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

**ΒΙΒΛΙΟΓΡΑΦΙΑ**

Κοντόπουλος, Γ. (2010) *Ανάπτυξη διαδικτυακών εφαρμογών GIS με λογισμικό ανοιχτού κώδικα (Geoserver)*, Διπλωματική εργασία, Θεσσαλονίκη Πανεπιστήμιο Μακεδονίας, Διαθέσιμο σε: [http://dspace.lib.uom.gr/bitstream/2159/13819/1/Kontopoulos\\_Msc2010.pdf](http://dspace.lib.uom.gr/bitstream/2159/13819/1/Kontopoulos_Msc2010.pdf) [Προσπελάστηκε 12 Σεπ 2012]

Λιακέας, Γ. (2002) Η γλώσσα JavaScript, Κλειδάριθμος

Μανδουράρη, Δ. (2011) *Πληροφοριακό σύστημα για την καθοδήγηση χρηστών με χρήση GIS*, Πτυχιακή εργασία, Κρήτη Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης, Διαθέσιμο σε: <http://nefeli.lib.teicrete.gr/browse/stef/epp/2011/MandourariDimitra/attached-document-1308739738-773291-16209/Mandourari2011.pdf> [Προσπελάστηκε 13 Σεπ 2012]

Μπούρχας, Θ. (2011) *Ανάπτυξη Πλατφόρμας Δυναμικής Παροχής Υπηρεσιών Βασισμένων στη Θέση των Χρηστών*, Διπλωματική εργασία, Αθήνα Εθνικό Μετσόβιο Πολυτεχνείο, Διαθέσιμο σε: <http://artemis-new.cslab.ece.ntua.gr:8080/jspui/bitstream/123456789/5625/1/DT2011-0203.pdf> [Προσπελάστηκε 13 Σεπ 2012]

Οικονόμου, Θ. (2010) *Development of a location aware application for mobile phones*, Διπλωματική εργασία, Βόλος Πανεπιστήμιο Θεσσαλίας, Διαθέσιμο σε: [http://support.inf.uth.gr/vasi/upload/theodoros\\_oikonomou.pdf](http://support.inf.uth.gr/vasi/upload/theodoros_oikonomou.pdf) [Προσπελάστηκε 12 Σεπ 2012]

Παπαδόπουλος, Δ. (2011) *Σχεδίαση και Ανάπτυξη Πλατφόρμας Παροχής Υπηρεσιών Περιήγησης Βασισμένη στη θέση των Χρηστών*, Διπλωματική εργασία, Αθήνα Εθνικό Μετσόβιο Πολυτεχνείο, Διαθέσιμο σε: [http://artemis.cslab.ntua.gr/el\\_thesis/artemis.ntua.ece/DT2011-0144/DT2011-0144.pdf](http://artemis.cslab.ntua.gr/el_thesis/artemis.ntua.ece/DT2011-0144/DT2011-0144.pdf) [Προσπελάστηκε 14 Σεπ 2012]

Σουρέφ, Ρ. και Αργυρόπουλος, Κ. (2012) *Ανάπτυξη Εφαρμογών για κινητά τηλέφωνα με τη χρήση του λειτουργικού συστήματος Android*, Πτυχιακή εργασία, Λάρισα Ανώτατο Τεχνολογικό Ίδρυμα Λάρισας, Διαθέσιμο σε: [http://ifestos.teilar.gr/index.php?option=com\\_docman&task=docview&gid=148](http://ifestos.teilar.gr/index.php?option=com_docman&task=docview&gid=148) [Προσπελάστηκε 14 Σεπ 2012]

Φρέτζος Ηλίας (2010), *Εισαγωγή στη PostgreSQL – PostGIS*, [Online], Διαθέσιμο σε: <http://infolab.cs.unipi.gr/pre-eclass/courses/gis/lab/Lab-PostGIS-doc.pdf> [Προσπελάστηκε 28 Jun 2012]

Android Site (2009) Android SDK Reference, [Online], Διαθέσιμο σε: <http://www.android.com/> [Προσπελάστηκε 15 Jun 2012]

Apache Cordova Documentation, *API Reference and Guides*, [Online], Διαθέσιμο σε: <http://docs.phonegap.com> [Προσπελάστηκε 22 May 2012]

Boston Geographic Information Systems (2012) PostGIS 2.0 pgsq2shp Command line Cheatsheet, [Online], Διαθέσιμο σε: <http://www.bostongis.com/> [Προσπελάστηκε 20 Jun 2012]

Chen, G. and Kotz, D. (2000) *A Survey of Context-Aware Mobile Computing Research*, Dartmouth Computer Science Technical Report TR2000-381.

Codeigniter, *User Guide*, [Online], Διαθέσιμο σε: [http://codeigniter.com/user\\_guide/](http://codeigniter.com/user_guide/) [Προσπελάστηκε 17 Jul 2012]

Erlacher, C., Paulus, G., Andrae, S., Lehrer, G., Gruber, G., Moser, P., Sabitzer, K., Pfeifhofer, E., Wallner, S., Lamprecht, P., Schimanz, A., Bosnjak, J., Gutzelnig, B., Johannes, J. and Terkl, D. (2012) *OpenPOI – Location-Based Applications to Collect Free Point-of-Interest Data for Secondary School Students*

Garrett, J. James (2005) *Ajax: A New Approach to Web Applications*, [Online], Διαθέσιμο σε: <http://www.adaptivepath.com/ideas/essays/archives/000385.php> [Προσπελάστηκε 22 Jul 2012]

Google Developers, *Google Maps Javascript API v3*, [Online], Διαθέσιμο σε: <https://developers.google.com/maps/documentation/javascript/tutorial> [Προσπελάστηκε 25 May 2012]

Gregory, S. (2010) *PostgreSQL 9.0 High Performance*, Birmingham: Packt Publishing Ltd

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

Jochen, H. Schiller and Agnès, Voisard (2004) *Location Based Services*, San Francisco: Elsevier Science

JSON *Introducing JSON*, [Online], Διαθέσιμο σε: <http://json.org> [Προσπελάστηκε 13 Jul 2012]

Jquery, *Documentation*, [Online], Διαθέσιμο σε: <http://docs.jquery.com/> [Προσπελάστηκε 7 Jul 2012]

Jquery Mobile, *Documentation*, [Online], Διαθέσιμο σε: <http://jquerymobile.com/demos/1.2.0/> [Προσπελάστηκε 9 Jul 2012]

Karatzas, G. (2010) *Mobile Web Tourism Application*, Διπλωματική εργασία, University of East Anglia, Διαθέσιμο σε: [http://www.gkaratzas.com/downloads/Georgios\\_Karatzas-MscThesis.pdf](http://www.gkaratzas.com/downloads/Georgios_Karatzas-MscThesis.pdf) [Προσπελάστηκε 16 Σεπ 2012]

Loi, V. (2010) *An Investigation into Agent-based System Architecture for Mobile Information Systems*, Πτυχιακή εργασία, Bath University of Bath, Διαθέσιμο σε: <http://www.cs.bath.ac.uk/~mdv/courses/CM30082/projects.bho/2009-10/Loi-VL-dissertation-2009-10.pdf> [Προσπελάστηκε 15 Σεπ 2012]

Longley, A. P., Goodchild, F. M., Maguire, J. D., Rhind, W. D. (2011) *Geographic Information Systems and Science*, 3<sup>rd</sup> ed. John Wiley & Sons

Pollock John (2006) *Οδηγός της Javascript*, 2<sup>η</sup> έκδοση, Αθήνα: Γκιούρδας Μ.

PostGIS, *Documentation*, [Online], Διαθέσιμο σε: <http://postgis.refrains.net/documentation/> [Προσπελάστηκε 18 Ιουν 2012]

PostgreSQL, *PostgreSQL 9.1.6 Documentation*, [Online], Διαθέσιμο σε: <http://www.postgresql.org/docs/9.1/static/index.html> [Προσπελάστηκε 18 Ιουν 2012]

Philippe, R., Michel, S. and Agnès V. (2001) *Spatial Databases*, Elsevier Science Zach, E. (2011) *Berg: Smartphone shipments grew 74% in 2010*, [Online], Διαθέσιμο σε: <http://www.bgr.com> [Προσπελάστηκε 11 Jul 2012]

Quantum GIS, *QGIS Features*, [Online], Available at: <http://qgis.org/en/about-qgis/features.html> [Προσπελάστηκε 2 Jul 2012]

Rational software development team(1998) *Rational Unified Process Best Practices for Software Development Teams*, [Online], Rational Software, Διαθέσιμο σε: [http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251\\_bestpractices\\_TP026B.pdf](http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf) [Προσπελάστηκε 11 Σεπ 2012]

W3Schools, *AJAX Tutorial*, [Online], Διαθέσιμο σε: <http://www.w3schools.com/ajax/default.asp> [Προσπελάστηκε 16 Ιουλ 2012]

W3Schools, *HTML Web Storage*, [Online], Διαθέσιμο σε: [http://www.w3schools.com/html/html5\\_webstorage.asp](http://www.w3schools.com/html/html5_webstorage.asp) [Προσπελάστηκε 6 Σεπ 2012]

W3Schools, *jQuery Tutorial*, [Online], Διαθέσιμο σε: <http://www.w3schools.com/jquery/default.asp> [Προσπελάστηκε 7 Jul 2012]

Sadoun, B. and Al-Bayari, O. ,2007. Location based services using geographical information systems, Science Direct, [Online], Διαθέσιμο σε: <http://www.sciencedirect.com/science/article/pii/S0140366407002629> [Προσπελάστηκε 3 Sep 2012]

**ΠΑΡΑΡΤΗΜΑ Α****ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ****index.html**

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8?>
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<title>Pois Finder</title>
<script type="text/javascript">
function exit(){
    navigator.app.exitApp();

}
function deviceready1(){
document.addEventListener("deviceready", onDeviceReady, false);

function onDeviceReady() {
    checkConnection();
}

function checkConnection() {
    var networkState = navigator.network.connection.type;

    var states = {};
    states[Connection.UNKNOWN] = 'Unknown connection';
    states[Connection.ETHERNET] = 'Ethernet connection';
    states[Connection.WIFI] = 'WiFi connection';
    states[Connection.CELL_2G] = 'Cell 2G connection';
    states[Connection.CELL_3G] = 'Cell 3G connection';
    states[Connection.CELL_4G] = 'Cell 4G connection';
    states[Connection.NONE] = 'No network connection';

    if (networkState == Connection.NONE){
```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```

alert('Πρέπει να συνδεθείτε στο Internet ');
};
}
}

</script>
</head>
<body onLoad="deviceready1()">
<div data-role="page">
<div data-role="header">
  <a href="" data-role="button" data-transition="pop" data-icon="back" rel="external"
onClick="exit()">Exit</a>
  <h1>Pois Finder</h1>
</div><!-- /header -->
<div data-role="content" data-position="relative">
<ul data-role="listview" data-inset="true" data-filter="true">
<li data-role="list-divider" role="heading">
  Σημεία ενδιαφέροντος
</li>
<li><a href="menu.html" rel="external">Κοντά μου</a></li>
<li><a href="addpois.html" rel="external">Πρόσθεσε σημεία</a></li>
<li><a href="about.html" rel="external">Εγχειρίδιο χρήστη</a></li>

</ul>

</div><!-- /content -->

<div data-role="footer" data-position="fixed">
  <h1>University of Piraeus 2012</h1>
</div><!-- /footer -->
</div><!-- /page -->

</body>
</html>

```

#### menu.html

```

<!DOCTYPE html>
<html>
<head>
<title>Menu</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8?>

```

```

<meta name="viewport" content="width=device-width, initial-scale=1">

<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
    localStorage.setItem("radius", 500);
</script>
</head>
<body>

<div data-role="page">

<div data-role="header" data-position="fixed">
    <a href="index.html" data-role="button" data-transition="pop" data-icon="back"
    rel="external">Back</a>

    <h1>Κοντά μου</h1>
    <a href="setting.html" data-role="button" data-transition="pop" data-icon="gear"
    rel="external">Options</a>

</div><!-- /header -->

<div data-role="content" data-position="relative">
    <ul data-role="listview" data-inset="true" data-filter="true">
        <li data-role="list-divider" role="heading">
            Σημεία ενδιαφέροντος
        </li>
        <li><a href="googlemaps1.html" rel="external">Automotive</a></li>
        <li><a href="googlemaps2.html" rel="external">Eating & Drinking</a></li>
        <li><a href="googlemaps3.html" rel="external">Government and Public
        Services</a></li>
        <li><a href="googlemaps4.html" rel="external">Health care</a></li>
        <li><a href="googlemaps5.html" rel="external">Leisure</a></li>
        <li><a href="googlemaps6.html" rel="external">Lodging</a></li>
        <li><a href="googlemaps7.html" rel="external">Sports</a></li>
        <li><a href="googlemaps8.html" rel="external">Tourism</a></li>
    </ul>
</div><!-- /content -->

```

```
</div><!-- /page -->
```

```
</body>
```

```
</html>
```

### googlemaps1.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
```

```
<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
```

```
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
```

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
```

```
<link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
```

```
<script type="text/javascript"
```

```
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx  
yht1U9zcs0&sensor=false"></script>
```

```
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
```

```
<script type="text/javascript">
```

```
    category="Automotive";
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
```

```
<div data-role="page" id="page1" data-title="page1">
```

```
<div data-role="header">
```

```
    <a href="menu.html" data-role="button" data-icon="back" >Back</a>
```

```
    <h1>Αποτελέσματα</h1>
```

```
</div><!-- /header -->
```

```
<div data-role="content" id="map_canvas" style="position:absolute; width:99%;  
height:86%;" >
```

```
</div>
```

```
<div data-role="footer" data-position="fixed">
```

```
<div data-role="navbar" data-position="fixed">
```

```
    <ul>
```

```
        <li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist"  
rel="external">Χάρτης</a></li>
```

```
        <li><a href="#page2" data-role="button" rel="external">Info</a></li>
```

```
    </ul>
```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```

</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
<div data-role="page" id="page2" data-title="page2">
<div data-role="header">
  <a href="menu.html" data-role="button" data-icon="back" >Back</a>
  <h1>Αποτελέσματα</h1>
</div><!-- /header -->
<div data-role="content" id="posts">
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
  <ul>
    <li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>
    <li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Info</a></li>
  </ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### googlemaps2.html

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&amp;sensor=false"></script>
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
  category="EatingandDrinking";
</script>

```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών



```

</head>
<body>
<script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
<div data-role="page" id="page1" data-title="page1">
  <div data-role="header">
    <a href="menu.html" data-role="button" data-icon="back" >Back</a>
    <h1>Αποτελέσματα</h1>
  </div><!-- /header -->
  <div data-role="content" id="map_canvas" style="position:absolute; width:99%; height:86%;" >
  </div>
  <div data-role="footer" data-position="fixed">
    <div data-role="navbar" data-position="fixed">
      <ul>
        <li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist" rel="external">Χάρτης</a></li>
        <li><a href="#page2" data-role="button" rel="external">Info</a></li>
      </ul>
    </div><!-- /navbar -->
  </div><!-- /footer -->
</div><!-- /page -->
<div data-role="page" id="page2" data-title="page2">
  <div data-role="header">
    <a href="menu.html" data-role="button" data-icon="back" >Back</a>
    <h1>Αποτελέσματα</h1>
  </div><!-- /header -->
  <div data-role="content" id="posts">
  </div>
  <div data-role="footer" data-position="fixed">
    <div data-role="navbar" data-position="fixed">
      <ul>
        <li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>
        <li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist" rel="external">Info</a></li>
      </ul>
    </div><!-- /navbar -->
  </div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### googlemaps3.html

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```

<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
    <script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
    <link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
    <script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&amp;sensor=false"></script>
    <script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
    <script type="text/javascript">
      category="Government and Public Services";
    </script>
  </head>
  <body>
    <script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
    <div data-role="page" id="page1" data-title="page1">
      <div data-role="header">
        <a href="menu.html" data-role="button" data-icon="back" >Back</a>
        <h1>Αποτελέσματα</h1>
      </div><!-- /header -->
      <div data-role="content" id="map_canvas" style="position:absolute; width:99%;
height:86%;" >
      </div>
      <div data-role="footer" data-position="fixed">
        <div data-role="navbar" data-position="fixed">
          <ul>
            <li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Χάρτης</a></li>
            <li><a href="#page2" data-role="button" rel="external">Info</a></li>
          </ul>
        </div><!-- /navbar -->
      </div><!-- /footer -->
    </div><!-- /page -->
    <div data-role="page" id="page2" data-title="page2">
      <div data-role="header">
        <a href="menu.html" data-role="button" data-icon="back" >Back</a>
        <h1>Αποτελέσματα</h1>
      </div><!-- /header -->

```

```

<div data-role="content" id="posts">
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>
<li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Info</a></li>
</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

#### googlemaps4.html

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&amp;sensor=false"></script>
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
category="Health care";
</script>
</head>
<body>
<script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
<div data-role="page" id="page1" data-title="page1">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="back" >Back</a>
<h1>Αποτελέσματα</h1>

```

```

</div><!-- /header -->
<div data-role="content" id="map_canvas" style="position:absolute; width:99%;
height:86%;" >
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Χάρτης</a></li>
<li><a href="#page2" data-role="button" rel="external">Info</a></li>
</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
<div data-role="page" id="page2" data-title="page2">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="back" >Back</a>
<h1>Αποτελέσματα</h1>
</div><!-- /header -->
<div data-role="content" id="posts">
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>
<li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Info</a></li>
</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### googlemaps5.html

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>

```

```

<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&sensor=false"></script>
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
    category="Leisure";
</script>
</head>
<body>
<script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
<div data-role="page" id="page1" data-title="page1">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="back" >Back</a>
<h1>Αποτελέσματα</h1>
</div><!-- /header -->
<div data-role="content" id="map_canvas" style="position:absolute; width:99%;
height:86%;" >
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Χάρτης</a></li>
<li><a href="#page2" data-role="button" rel="external">Info</a></li>
</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
<div data-role="page" id="page2" data-title="page2">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="back" >Back</a>
<h1>Αποτελέσματα</h1>
</div><!-- /header -->
<div data-role="content" id="posts">
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>

```

```

    <li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Info</a></li>
  </ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### googlemaps6.html

```

<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
    <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
    <script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
    <link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
    <script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&sensor=false"></script>
    <script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
    <script type="text/javascript">
      category="Lodging";
    </script>
  </head>
  <body>
    <script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
    <div data-role="page">
      <div data-role="header">
        <a href="menu.html" data-role="button" data-icon="back" >Back</a>
        <h1>Αποτελέσματα</h1>
      </div><!-- /header -->
      <div data-role="content" id="map_canvas" style="position:absolute; width:99%;
height:86%;" >
      </div>
      <div data-role="footer" data-position="fixed">
        <div data-role="navbar" >
          <ul>
            <li><a onclick="map()" data-role="button" class="ui-btn-active">Χάρτης</a></li>
            <li><a onclick="infos()" data-role="button" >Info</a></li>

```

```

</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### googlemaps7.html

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&amp;sensor=false"></script>
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
    category="Sports";
</script>
</head>
<body>
<script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
<div data-role="page" id="page1" data-title="page1">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="back" >Back</a>
<h1>Αποτελέσματα</h1>
</div><!-- /header -->
<div data-role="content" id="map_canvas" style="position:absolute; width:99%;
height:86%;" >
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Χάρτης</a></li>
<li><a href="#page2" data-role="button" rel="external">Info</a></li>

```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```

</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
<div data-role="page" id="page2" data-title="page2">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="back" >Back</a>
<h1>Αποτελέσματα</h1>
</div><!-- /header -->
<div data-role="content" id="posts">
</div>
<div data-role="footer" data-position="fixed">
<div data-role="navbar" data-position="fixed">
<ul>
<li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>
<li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Info</a></li>
</ul>
</div><!-- /navbar -->
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### Googlemaps8.html

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script type="text/javascript" src="http://malsup.github.com/jquery.blockUI.js"></script>
<script src="jquery.mobile/jquery.mobile-1.1.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="jquery.mobile/jquery.mobile-1.1.1.min.css" />
<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDEXmxAE6wIXEOvwgVx
yht1U9zcs0&amp;sensor=false"></script>
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
category="Tourism";
</script>

```



```

</head>
<body>
<script type="text/javascript" charset="utf-8" src="googlemaps.js"></script>
<div data-role="page" id="page1" data-title="page1">
  <div data-role="header">
    <a href="menu.html" data-role="button" data-icon="back" >Back</a>
    <h1>Αποτελέσματα</h1>
  </div><!-- /header -->
  <div data-role="content" id="map_canvas" style="position:absolute; width:99%;
  height:86%;" >
  </div>
  <div data-role="footer" data-position="fixed">
  <div data-role="navbar" data-position="fixed">
    <ul>
      <li><a href="#page1" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Χάρτης</a></li>
      <li><a href="#page2" data-role="button" rel="external">Info</a></li>
    </ul>
  </div><!-- /navbar -->
  </div><!-- /footer -->
</div><!-- /page -->
<div data-role="page" id="page2" data-title="page2">
  <div data-role="header">
    <a href="menu.html" data-role="button" data-icon="back" >Back</a>
    <h1>Αποτελέσματα</h1>
  </div><!-- /header -->
  <div data-role="content" id="posts">
  </div>
  <div data-role="footer" data-position="fixed">
  <div data-role="navbar" data-position="fixed">
    <ul>
      <li><a href="#page1" data-role="button" rel="external">Χάρτης</a></li>
      <li><a href="#page2" data-role="button" class="ui-btn-active ui-state-persist"
rel="external">Info</a></li>
    </ul>
  </div><!-- /navbar -->
  </div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

### googlemaps.js

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```
$(document).ready(function){
  var marker;
  radius=localStorage.getItem("radius")
  document.addEventListener("deviceready", init, false);
  init();
  var map;
  function toggleBounce()
  {
    if (marker.getAnimation() != null) {
      marker.setAnimation(null);
    }
    else
    {
      marker.setAnimation(google.maps.Animation.BOUNCE);
    }
  }
  function init()
  {
    if(navigator.geolocation)
    {
      var options = { frequency: 300 };
      navigator.geolocation.getCurrentPosition(processPosition,
      handleError,{enableHighAccuracy: true }, options);

    }
    else
    {
      navigator.notification.alert("Geolocation not supported in this browser");
    }
  }

  function processPosition(pos)
  {
    var latlng = new google.maps.LatLng(pos.coords.latitude, pos.coords.longitude);
    lat=pos.coords.latitude;
    lon=pos.coords.longitude;

    var image0 = new google.maps.MarkerImage('http://icons.iconarchive.com/icons/icon-
    land/vista-map-markers/48/Map-Marker-Ball-Chartreuse-icon.png',
    // This marker is 20 pixels wide by 32 pixels tall.
```

```
new google.maps.Size(48, 48),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));

var myOptions = {
  center: latlng,
  zoom: 14,
  mapTypeId: google.maps.MapTypeId.ROADMAP
};
var map = new google.maps.Map(document.getElementById("map_canvas"),
myOptions);

var marker = new google.maps.Marker({
  position: latlng,
  draggable:true,
  animation: google.maps.Animation.DROP,
  icon : image0,
  map: map,
  title:"You are here! (at least within a "+pos.coords.accuracy+" meter radius)"
});

google.maps.event.addListener(marker, 'click', toggleBounce);

$.ajax({

  url:'http://sofioskostas.dyndns-
home.com/gmaps/index.php/php_genjson?lat='+lat+'&lon='+lon+'&category='+category+'&radius='+radius,
  dataType:'json',
  success:function(data, textStatus, jqXHR)
  {

    if(data.status==1) {

      html="<h3>Πληροφορίες</h3>"
      html += "<h4>Αποτελέσματα για Σημεία ενδιαφέροντος</h4>"
```

```

html += "<h4>Εμφάνιση " + data.data.length + " απο  

"+data.data.length+"</h4></p></td></tr></table>"
for (i=0; i<data.data.length; i++)
{
var str=data.data[i]['dist']
var n=str.substr(0,3);
html += "<p>"
html += "<h4>"
html += i+1
html += "<br>"
html += "<font color='blue'>Όνομα</font>:" + data.data[i]['name'] + "<br>"
html += "<font color='blue'>Κατηγορία</font>:" + data.data[i]['category'] + "<br>"
html += "<font color='blue'>Απόσταση σε km</font>:" + n + " km</h4>"
html += "<p><hr>"
html += "<span>"
$('#posts').html(html);

var latlng1 = new google.maps.LatLng(data.data[i]['st_y'], data.data[i]['st_x']);

var image1 = new google.maps.MarkerImage('icons/automotive.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));
var image2 = new google.maps.MarkerImage('icons/healthcare.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));

var image3 = new google.maps.MarkerImage('icons/gover.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.

```

```
new google.maps.Point(0, 32));

var image4 = new google.maps.MarkerImage('icons/Leisure.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));
var image5 = new google.maps.MarkerImage('icons/food.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));

var image6 = new google.maps.MarkerImage('icons/Lodging.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));

var image7 = new google.maps.MarkerImage('icons/Sports.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));

var image8 = new google.maps.MarkerImage('icons/tourism.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));
```

```

var image9 = new google.maps.MarkerImage('icons/business.png',
// This marker is 20 pixels wide by 32 pixels tall.
new google.maps.Size(32, 37),
// The origin for this image is 0,0.
new google.maps.Point(0,0),
// The anchor for this image is the base of the flagpole at 0,32.
new google.maps.Point(0, 32));

var shape = {
coord: [1, 1, 1, 20, 18, 20, 18, 1],
type: 'poly'
};

var customIcons = [];

customIcons["Automotive"] = image1;
customIcons["Health care"] = image2;
customIcons["Government and Public Services"] = image3;
customIcons["Leisure"] = image4;
customIcons["EatingandDrinking"] = image5;
customIcons["Lodging"] = image6;
customIcons["Sports"] = image7;
customIcons["Tourism"] = image8;
customIcons["Night Life and Business"] = image9;

var marker = new google.maps.Marker({
position: latlng1,
map: map,
shape: shape,
icon: customIcons[data.data[i]['category']],
title:"Here is a "+data.data[i]['name']+" and belongs on "+data.data[i]['category']+"
category!)"
});
}
}
else{

alert(data.message);

```

```

        localStorage.setItem("radius", "3000");
        location.reload();
    }
},
error: function(jqXHR, textStatus, errorThrown){
    navigator.notification.alert('AJAX request failed');
}
});
}

```

```

function handleError(err)
{
    alert('An error occurred: ' + err.code);
}
});

```

#### setting.html

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<link rel="stylesheet" href="http://code.jquery.com/mobile/1.1.1/jquery.mobile-1.1.1.min.css" />
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/mobile/1.1.1/jquery.mobile-1.1.1.min.js"></script>
<script type="text/javascript"
src="http://maps.googleapis.com/maps/api/js?key=AlzaSyAfiZGDExmxAE6wIXEOvwgVx
yht1U9zcs0&sensor=false"></script>
<script type="text/javascript" charset="utf-8" src="phonegap-8.js"></script>
<script type="text/javascript">
function radius1()
{
    slider =$("#slider-0").val();

    localStorage.setItem("radius", slider);
    $.mobile.changePage("menu.html");
}

```

```

}
</script>
</head>
<body>
<div data-role="page">
<div data-role="header">
<a href="menu.html" data-role="button" data-icon="home" >Home</a>
<h1>Ρυθμίσεις</h1>
<a href="" data-role="button" data-rel="back" data-icon="back" >Back</a>
</div><!-- /header -->
<div data-role="content" >
<label for="slider-0">Ρυθμίστε την ακτίνα σε m εντοπισμού των σημείων
ενδιαφέροντος:</label>
<input type="range" name="slider" id="slider-0" value="500" min="500" max="5000"
/><br>
<a href="" data-role="button" data-inline="true" data-icon="check" data-theme="b"
onclick="radius1()">Save</a>
</div>
<div data-role="footer" data-position="fixed">
</div><!-- /footer -->
</div><!-- /page -->
</body>
</html>

```

#### addpois.html

```

<!DOCTYPE html>
<html>
<head>
<title>Add a POI</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8?>
<link rel="stylesheet" href="http://code.jquery.com/mobile/1.1.1/jquery.mobile-
1.1.1.min.css" />
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<script src="http://code.jquery.com/mobile/1.1.1/jquery.mobile-1.1.1.min.js"></script>
<script type="text/javascript" charset="utf-8" src="cordova-2.0.0.js"></script>
<script type="text/javascript">
$(document).ready(function){
$("input[type=text]").val("");
$("select option:selected").each(function () {

```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών



```

$(this).remove(); //or whatever else
});
document.addEventListener("deviceready", init, false);
init();
function init() {
$(a[id^="submitButton"]').on("click",function(){
$("#submitButton",this).attr("disabled","disabled");
poisname=$("#poisname").val();
odos=$("#odos").val();
arithmos=$("#arithmos").val();
tk=$("#tk").val();
city=$("#city").val();
secret=$("#secret").val();
categ=$("#categ").val();
latitude=$("#latitude").val();
longitude=$("#longitude").val();
if(poisname!=" && odos!=" && arithmos!=" && city!=" && categ!=")
{
$.ajax({
url:'http://sofioskostas.dyndns-
home.com/gmaps/index.php/addpois?poisname='+poisname+'&categ='+categ+'&odos='+odos+
'&arithmos='+arithmos+'&tk='+tk+'&city='+city+'&latitude='+latitude+'&longitude='+longitude,
dataType:'text',
success:function(data, textStatus, jqXHR)
{
alert("Your form submitted sucessfully");
$.mobile.changePage("index.html");
},
error: function(jqXHR, textStatus, errorThrown){
alert("Η αποστολή απέτυχε ξαναπροσπαθήστε!");
}
});
}
else
{

```

```

    alert("Πρέπει να συμπληρώσετε τα υποχρεωτικά πεδία!");
}

return false;
});
}
});
</script>
</head>
<body>
<div data-role="page">
<div data-role="header" >
<a href="index.html" data-icon="back">Back</a>
<h2>Προσθήκη σημείων</h2></div>
<div data-role="content">
<form id="addPoisForm">
Όνομα:<input type="text" name="poisname" id="poisname"/>
<label for="select-choice-1" class="select">Κατηγορία*</label>
<select data-native-menu="false" name="categ" id="categ">
<option value="Cat">Κατηγορία</option>
<option value="Automotive">Automotive</option>
<option value="EatingandDrinking">Eating & Drinking</option>
<option value="Government and Public Services">Government and Public
Services</option>
<option value="Health care">Health care</option>
<option value="Leisure">Leisure</option>
<option value="Lodging">Lodging</option>
<option value="Sports">Sports</option>
<option value="Tourism">Tourism</option>
</select>
Οδός:<input type="text" name="odos" id="odos"/>
Αριθμός:<input type="text" name="arithmos" id="arithmos"/>
Πόλη:<input type="text" name="city" id="city"/>
Τ.Κ.:<input type="text" name="tk" id="tk"/>
Latitude<input type="text" name="latitude" id="latitude"/>
Longitude<input type="text" name="longitude" id="longitude"/>
<a href="index.html" data-role="button" data-inline="true" data-icon="back" >Back</a>
<a href="" id="submitButton" data-role="button" data-inline="true" data-theme="b" data-
icon="forward">Προσθήκη</a>
</form>
</div>

```

```
</div>
</body>
</html>
```

### gmaps\_model.php

```
<?php
class Gmaps_model extends CI_Model {

    // Fortwnei tin db library. Epsis kanei available tin db class me to $this->db antikeimeno

    public function __construct()
    {
        $this->load->database();
    }

    public function get_pois()
    {
        $lat = $_GET['lat'];
        $lon = $_GET['lon'];
        $category = $_GET['category'];
        $radius = $_GET['radius'];

        $query = $this->db->query("SELECT gid, category,name,st_x(the_geom),
st_y(the_geom),ST_Distance((ST_Transform(ST_GeomFromText(the_geom,
4326),2100)),(ST_Transform(ST_GeomFromText('POINT($lon $lat)', 4326),2100)))/1000 as
dist
FROM greece_poi
```

```
WHERE ((ST_Transform(ST_GeomFromText(the_geom,
4326),2100))@ST_Buffer(ST_Transform(ST_GeomFromText('POINT($lon $lat)',
4326),2100),$radius)))
AND category='$category' ;");
```

```
if(!$query)
{
    $this->error = $this->db->_error_message();
    $this->errno = $this->db->_error_number();
    return false;
}
//return $query->result_array();
if ($query->num_rows()>0)
{
    return $query->result_array();
}
}
}
}
```

php\_genjson.php

```
<?php
```

```
class Php_genjson extends CI_Eλεγκτή(Controller) {
```

```
public function __construct()
```

```
{
    parent::__construct();
    $this->load->model('gmaps_model');
}
```

```
public function index()
```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```

{
$data['greece_poi'] = $this->gmaps_model->get_pois();

if($data['greece_poi'] === FALSE){
$result = array();

$result["data"]=$data["greece_poi"];
$result["status"]=0;
$result["message"]="The Database query Failed and returned False!";

echo json_encode($result);
return false;

}

if(empty($data['greece_poi'])){
$result = array();

$result["data"]=$data["greece_poi"];
$result["status"]=0;
$result["message"]="Δέν βρέθηκαν αποτελέσματα κοντά σας. Θέλετε να γίνει αναζήτηση με
μεγαλύτερη ακτίνα;";

echo json_encode($result);
return false;

}

$result = array();

$result["data"]=$data["greece_poi"];
$result["status"]=1;
$result["message"]="Success!";

echo json_encode($result);
return TRUE;

echo json_encode(
    array(
        "responseStatus" =>0,
        "errorType" =>"generic",

```

```
"errors" =>"The error",
"message" =>"errormessage"

)
);
return FALSE;

}
}
```

### addpois.php

```
<?php
```

```
class Addpois extends CI_Eλεγκτή(Controller) {
```

```
public function __construct()
```

```
{
    parent::__construct();
    $this->load->model('gmaps_model2');
}
```

```
public function index()
```

```
{
```

```
$poisname=$_GET['poisname'];
$odos=$_GET['odos'];
$arithmos=$_GET['arithmos'];
$tk=$_GET['tk'];
$city=$_GET['city'];
$categ=$_GET['categ'];
$latitude=$_GET['latitude'];
$longitude=$_GET['longitude'];
$to = "sofios_kostas@hotmail.com";
$subject = 'Προσθήκη σημείων ενδιαφέροντος';
$message = "<html>
<head>
<title>Email</title>
</head>
<body>
```

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών

```
<p><h1><b>Προσθήκη σημείου</b></h1></p>
<h2><b>Στοιχεία σημείου ενδιαφέροντος:</b></h2>
<h3>Όνομα: ".$poisname." </h3><br>
<h3>Κατηγορία: ".$categ." </h3><br>
<h3>Οδός: ".$odos." </h3><br>
<h3>Αριθμός: ".$arithmos." </h3><br>
<h3>Ταχυδρομικός Κώδικας: ".$tk." </h3><br>
<h3>Πόλη: ".$city." </h3><br>
<h3>Latitude: ".$latitude." </h3><br>
<h3>Longitude: ".$longitude." </h3><br>
</body>
</html>
";
$headers = 'MIME-Version: 1.0'. "\r\n";
$headers.= 'Content-type: text/html;'. "\r\n";
$headers.= 'From: sofios_kostas@hotmail.com'. "\r\n".
'Reply-To: sofios_kostas@hotmail.com'. "\r\n".
'X-Mailer: PHP/'. phpversion();

mail($to, $subject, $message, $headers);
echo $message;
}
}
```

**ΠΑΡΑΡΤΗΜΑ Β****ΓΛΩΣΣΑΡΙ ΟΡΩΝ**

Αίτημα	Request
Ανάλυση απαιτήσεων	Requirements Analysis
Ανοικτού κώδικα	Open Source
Απάντηση	Response
Ασύγχρονη αναπαραγωγή	Asynchronous replication
Ασύρματο σύστημα θέσης	Wi-Fi position system
Γεωγραφικά πληροφοριακά συστήματα	Geographic information systems
Γεωεντοπισμός	Geolocation
Γεωκωδικοποίηση	Geocoding
Γλώσσα προγραμματισμού	Programming Languages
Γλώσσα σήμανσης	Markup Language
Γραμμή ζωής αντικειμένου	Lifeline object
Διαδικτυακός εξυπηρετητής	Web server
Διαχειριστής θέσης	Location Manager
Διαχειριστής περιεχομένου	Content Manager
Διαχειριστής τηλεφώνου	Telephony Manager
Διαχειριστής δραστηριότητας	Activity Manager
Διεπαφή	Interface
Έμφυτο	Native
Ένθετες συναλλαγές	Nested transactions
Έξυπνα κινητά τηλέφωνα	Smartphones
Εξυπηρετητής	Server
Εξομοιωτής	Emulator
Επαναληπτική ενοποιημένη διαδικασία	Rational Unified Process
Επεκτάσεις	Extensions
Εφαρμογή	Application
Λίστες ηλεκτρονικού ταχυδρομείου	Mailing lists
Παρασκήνιο	Backend
Πελάτης	Client
Πέρασμα	Parsing
Περιηγητής	Browser
Πλαίσιο εργασίας	Framework
Πόροι	Resources
Προσαρμοσμένο	Custom
Πρόσθετο	Plugin
Σενάρια δράσης	Scripts
Στοιχεία	Tags
Τοπική αποθήκευση	Local storage

Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών



Μεταπτυχιακή Διατριβή

Σοφίος Κωνσταντίνος

Υπηρεσίες βασισμένες στη θέση  
Φυλλομετρητής ιστοσελίδας  
Χωρική βάση δεδομένων

Location based services  
Web Browser  
Spatial Database

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ