



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Δυναμική διαμόρφωση κριτηρίων γεωπληροφορικής αναζήτησης
Όνοματεπώνυμο Φοιτητή	Μαρία Λούκα
Πατρώνυμο	Παναγιώτης
Αριθμός Μητρώου	ΜΠΠΛ/ 10020
Επιβλέπων	Χρήστος Δουληγέρης, Καθηγητής
Συνεπιβλέπων Ερευνητής	Δρ. Βασίλειος Μενεκλής

Ημερομηνία Παράδοσης **Μάρτιος 2013**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Χρήστος Δουληγέρης
Καθηγητής

(υπογραφή)

Δημήτρης Αποστόλου
Επίκουρος Καθηγητής

(υπογραφή)

Παναγιώτης
Κοτζανικολάου
Λέκτορας

Πίνακας Περιεχομένων

Περίληψη.....	5
Abstract.....	5
Ευχαριστίες.....	6
Κεφάλαιο 1.....	7
1.1 Εισαγωγή.....	7
1.2 Το αντικείμενο της διατριβής.....	7
1.3 Σχετικές τεχνολογίες.....	8
1.4 Δομή της μεταπτυχιακής διατριβής.....	9
Κεφάλαιο 2.....	10
Σχετική Βιβλιογραφία.....	10
2.1 «Ανάπτυξη μίας εφαρμογής επίβλεψης και ανίχνευσης ζημιών σε μια οικία με τη χρήση εργαλείων της Google».....	10
2.2 «Εργαλεία ανάπτυξης εφαρμογών για τον παγκόσμιο ιστό».....	10
2.3 «Πληροφοριακό σύστημα για την καθοδήγηση χρηστών με χρήση GIS».....	11
2.4 «Εργαλείο για βελτιστοποίηση G.I.S συστήματος web πλατφόρμας του εργαστηρίου eLAB».....	12
2.5«Ηλεκτρονικό ινστιτούτο αισθητικής με χρήση MVC PHP Framework».....	13
2.6 «Χρήση Mobile GIS, για διαχείριση εκτάκτων αναγκών».....	13
2.7 «Ανασύνθεση του πακέτου “Web G.I.S. – Αθηνών” του εργαστηρίου PeLAB με χρήση ORACLE 8.0».....	14
2.8 Ανακεφαλαίωση.....	15
Κεφάλαιο 3.....	16
Τεχνολογίες που χρησιμοποιήθηκαν.....	16
3.1 Δυναμική ιστοσελίδα.....	16
3.2 Codeigniter.....	17
3.2.1 Χαρακτηριστικά του Codeigniter.....	17
3.2.2 Μοντέλο σχεδιασμού MVC.....	17
3.3 Google maps.....	19
3.4 Google Maps API (Διεπαφή Προγραμματισμού Εφαρμογών Google Maps).....	20
3.5 Javascript.....	20
3.6 HTML.....	20
3.7 Γεωγραφικά Συστήματα Πληροφοριών (GIS).....	21
3.7.1 PostgreSQL.....	23
3.7.2 PostGIS.....	23
3.7.3 PgAdmin.....	23
3.7.4 Quantum.....	24
3.8 Ανακεφαλαίωση.....	25
Κεφάλαιο 4.....	26
Αρχιτεκτονική της εφαρμογής.....	26
4.1 Υλοποίηση.....	26
4.2 Διαγράμματα UML.....	33
4.2.1 Διάγραμμα περίπτωσης (Use case diagram).....	33

4.2.2 Διάγραμμα Τάξεων(Class Diagrams).....	35
4.2.3 Διάγραμμα Δραστηριοτήτων (Activity diagram).....	36
4.2.4 Διάγραμμα Διανομής (Deployment Diagram).....	37
4.2.5 Διάγραμμα ακολουθίας- σειράς (Διάγραμμα αλληλεπίδρασης).....	37
4.3 Πλήρης περιγραφή της εφαρμογής.....	39
4.4 Ανακεφαλαίωση.....	41
Κεφάλαιο 5.....	42
Συμπεράσματα.....	42
5.1 Συμπεράσματα για τις τεχνολογίες που χρησιμοποιήθηκαν.....	42
5.1.1 Συμπεράσματα από τη χρήση των γεωγραφικών συστημάτων (GIS)....	42
5.1.2 Συμπεράσματα από τη χρήση των τεχνολογιών διαδικτύου.....	42
5.3 Μελλοντικές βελτιώσεις της εφαρμογής.....	43
Βιβλιογραφία.....	44
Παράρτημα - Ο κώδικας της εφαρμογής	46

ΠΕΡΙΛΗΨΗ

Η παρούσα μεταπτυχιακή διατριβή πραγματεύεται την ιδέα της υλοποίησης μιας εφαρμογής όπου θα εφαρμόζεται δυναμική διαμόρφωση κριτηρίων γεωπληροφοριακής αναζήτησης. Στόχος της εργασίας είναι να δημιουργηθεί μια εφαρμογή η οποία θα δίνει τη δυνατότητα στο χρήστη να αλληλεπιδρά πάνω σε ένα χάρτη κάνοντας κλικ πάνω του και να εμφανίζονται σε νέα ιστοσελίδα σημεία ενδιαφέροντος που βρίσκονται σε ορισμένη ακτίνα γύρω από το σημείο που έγινε επιλογή πάνω στο χάρτη από το χρήστη.

Ο χάρτης των περιοχών των νησιών Πάρου και Νάξου δημιουργήθηκε με τη βοήθεια του Google maps και η δυνατότητα να μπορεί να κάνει κλικ ο χρήστης πάνω στο χάρτη με τη βοήθεια του Google maps API. Οι συντεταγμένες στέλνονται στο Codeigniter, το οποίο είναι ένα πλαίσιο (framework) της PHP, όπου εκεί γίνεται επεξεργασία αυτών. Έπειτα γίνεται σύνδεση με το διακομιστή του δικτύου και την βάση δεδομένων όπου είναι αποθηκευμένα τα σημεία ενδιαφέροντος και επιστρέφονται τα αποτελέσματα της επεξεργασίας στον χρήστη με τη βοήθεια πάλι του Codeigniter.

Ο χρήστης αρχικά εκτελεί το πρόγραμμα - φυλλομετρητή του Διαδικτύου για να μπορέσει να εκτελέσει την εφαρμογή. Η εκτέλεση της εφαρμογής είναι απλή και εύκολη, αυτό όμως δεν αναιρεί την πολυπλοκότητα της εφαρμογής σε τεχνικό επίπεδο και τη δυσκολία της δημιουργίας της.

ABSTRACT

This masters thesis discusses the idea of implementing an application that will apply dynamic configuration geoinformation search criteria. The aim of the work was to create an application that will allow users to interact on a map by clicking on it, and being able to see new website with landmarks located within a certain radius around the point was selected on the map by the user.

The map of the areas of the islands of Paros and Naxos are first created with the help of Google maps and then the ability clicks on the map using the Google maps API is materialized. The coordinates are sent to Codeigniter, framework of PHP, where are they processed moreover a connection is established with the web server and the database where the landmarks are stored. The processing results are returned to the user with the help of Codeigniter.

The user first runs the program - a web browser to be able to execute the application. The running of the application is simple and easy, but that does not negate the implementations complexity, the technical level and the difficulty of its creation.

ΕΥΧΑΡΙΣΤΙΕΣ

Οι τεχνολογίες που μελετήθηκαν αλλά και τα συμπεράσματα που εξήχθησαν από τη μελέτη των συγκεκριμένων τεχνολογιών, αποτελούν μια πολύ σημαντική παρακαταθήκη σε προσωπικό επίπεδο αναφορικά με τη μελέτη, τη χρήση και την διαχείριση των διαθέσιμων τεχνολογιών. Ήταν πολύ ευχάριστη η διαδικασία εκμάθησης και εξερεύνησης αυτών διότι τα αντικείμενα τους είναι ενδιαφέροντα και σύγχρονα. Ήταν μια προσωπική πρόκληση να αντεπεξέλθω στις απαιτήσεις της εφαρμογής.

Για την επιτυχή ολοκλήρωση της εργασίας, ήταν καταλυτική η βοήθεια του αξιότιμου καθηγητή κυρίου Χρήστου Δουληγέρη που επέβλεπε την πρόοδο της εργασίας. Επίσης εξίσου πολύτιμη ήταν και η βοήθεια του αξιότιμου διδάκτορα κυρίου Βασίλη Μενεκλή, ο οποίος επεσήμανε εύστοχα διάφορα προβλήματα που έπρεπε να μελετηθούν, έδινε τις κατευθυντήριες γραμμές προκειμένου να γίνει πλήρης και εμπειρισταωμένη μελέτη και με ενθάρρυνε στην όλη προσπάθεια να διεκπεραιώσω επιτυχώς την μεταπτυχιακή διατριβή μου.

Θέλω να ευχαριστήσω από καρδιάς και τους δύο για την καθοδήγηση και τη συμπαράσταση που μου παρείχαν καθ' όλη την εξέλιξη της υλοποίησης του προγράμματος και της συγγραφής της μεταπτυχιακής διατριβής, αλλά και όλους τους διδάσκοντες του μεταπτυχιακού προγράμματος σπουδών «Πληροφορική» του Πανεπιστημίου Πειραιώς για την πολύ καλή συνεργασία κατά τη διάρκεια του κύκλου σπουδών.

Τέλος θέλω να ευχαριστήσω την οικογένειά μου και κυρίως τα παιδιά μου που μου έλεγαν καλή επιτυχία πριν τις εξετάσεις και τους συμφοιτητές μου για την ηθική στήριξη που μου παρείχαν προκειμένου να συμμετάσχω και να ολοκληρώσω το μεταπτυχιακό πρόγραμμα σπουδών «Πληροφορική».

ΚΕΦΑΛΑΙΟ 1

1.1 Εισαγωγή

Καθώς το διαδίκτυο έχει μπει για τα καλά στη καθημερινή ζωή μας, έχουν αναπτυχθεί εφαρμογές για την καλύτερη χρήση του και την εκμετάλλευση των δυνατοτήτων που μας πρόσφερε. Έτσι λοιπόν η Google ανέπτυξε την υπηρεσία Google maps η οποία χαρτογράφησε ψηφιακά σχεδόν όλο τον πλανήτη και ταυτόχρονα μπορεί ο χρήστης να περιπλανηθεί οπουδήποτε στο χάρτη και η οποία δίνει τη δυνατότητα μεγαλύτερης ανάπτυξης των Γεωγραφικών Συστημάτων Πληροφοριών (GIS). Τα Συστήματα Γεωγραφικών Πληροφοριών (ΣΓΠ), γνωστό ευρέως και ως G.I.S. (Geographic Information Systems), είναι συστήματα διαχείρισης χωρικών δεδομένων (spatial data) και συσχετισμένων ιδιοτήτων.

Η παρούσα μεταπτυχιακή διατριβή ασχολείται με την υλοποίηση μιας εφαρμογής όπου πραγματοποιεί δυναμική διαμόρφωση κριτηρίων γεωπληροφοριακής αναζήτησης και στηρίζεται πάνω στα Συστήματα Γεωγραφικών Πληροφοριών (GIS) με συνδυασμό Τεχνολογιών Διαδικτύου.

Η ιδέα είναι να δημιουργηθεί μια εφαρμογή η οποία θα εκτελείται στο περιβάλλον του φυλλομετρητή του Διαδικτύου με εμφάνιση του χάρτη από το Google maps και συγκεκριμένα των περιοχών των νησιών Πάρος και Νάξος όπου ο χρήστης θα κάνει κλικ μέσα στην περιοχή αυτή και αν υπάρχουν καταστήματα σε ορισμένη ακτίνα από το σημείο που επιλέχθηκε από το χρήστη, θα εμφανιστούν τα καταστήματα με σχετικές πληροφορίες για αυτά όπως όνομα ιδιοκτήτη, είδος καταστήματος και το νησί στο οποίο βρίσκεται. Το χαρακτηριστικό της εφαρμογής αυτής είναι ότι γίνεται δυναμικά η αλλαγή των κριτηρίων αναζήτησης καθώς ο χρήστης είναι ελεύθερος να κάνει κλικ οπουδήποτε στο χάρτη.

Το ιδιαίτερο χαρακτηριστικό της εφαρμογής αυτής είναι η χρήση του πλαισίου PHP (framework) Codeigniter το οποίο χρησιμοποιεί το πρότυπο MVC (Model-View-Controller). Το MVC είναι μία αρχιτεκτονική προγραμματισμού που βασίζεται στο διαχωρισμό του κώδικα λογικής από τον κώδικα εμφάνισης και αλληλεπίδρασης με το χρήστη. Αυτή η αρχιτεκτονική αν υιοθετηθεί επιτρέπει την καλύτερη οργάνωση του κώδικα, καθώς και την εύκολη συντήρηση του κώδικα. Επίσης επιτρέπει σε πολλούς προγραμματιστές να δουλεύουν στο ίδιο σύστημα ταυτόχρονα, καθώς και την ανάπτυξη και αποτελμάτωση κάθε κομματιού κώδικα ξεχωριστά. Στην πράξη, αυτό επιτρέπει στις ιστοσελίδες να περιέχουν ελάχιστο κώδικα αφού η παρουσίαση γίνεται ξεχωριστά από το PHP κώδικα.

1.2 Το αντικείμενο της διατριβής

Η βασική ιδέα που πραγματεύεται η διατριβή, είναι η υλοποίηση μιας εφαρμογής, η οποία θα δίνει την δυνατότητα στο χρήστη να κάνει δυναμική αναζήτηση πάνω σε ένα χάρτη. Καθώς ο χρήστης θα επιλέγει πάνω στο χάρτη και θα δημιουργεί σημάδι (marker), θα εμφανίζονται σημεία ενδιαφέροντος που έχουν αποθηκευτεί στη βάση δεδομένων τα οποία βρίσκονται σε ορισμένη ακτίνα γύρω από το σημάδι.

Θα υπάρχει λοιπόν μία σελίδα φτιαγμένη με Javascript , HTML και Css η οποία θα φορτώνει ένα χάρτη με τη βοήθεια του google maps API. Ο χάρτης θα είναι ζουμαρισμένος στην περιοχή της Πάρου και της Νάξου και κάθε φορά που θα γίνεται κλικ μέσα στο google map θα δημιουργούνται κόκκινα σημεία δηλαδή σημάδια (markers) και μετέπειτα θα εμφανίζονται τα σημεία ενδιαφέροντος που έχουμε καταχωρήσει στη βάση δεδομένων που βρίσκονται σε ορισμένη ακτίνα γύρω από το σημείο που επιλέχθηκε. Στη βάση δεδομένων μας τα σημεία ενδιαφέροντος αντιπροσωπεύουν καταστήματα, όπως φαρμακεία, καφετέριες, εμπορικά κτλ. και έχουν καταχωρηθεί λεπτομέρειες για τα σημεία ενδιαφέροντος όπως κωδικός, όνομα ιδιοκτήτη και είδος και αυτές οι λεπτομέρειες θα εμφανίζονται στο τελικό αποτέλεσμα.

Επίσης έχει φτιαχτεί μία βάση δεδομένων, η οποία διαχειρίζεται από το pgAdmin όπου έχουν καταχωρηθεί κάποια σημεία ενδιαφέροντος και καθώς κάνουμε κλικ πάνω στο χάρτη, θα στέλνουμε αίτημα στη βάση με τη βοήθεια τεχνολογιών διαδικτύου (php framework-Codeigniter) και θα μας επιστρέφονται τα σημεία ενδιαφέροντος όπου βρίσκονται σε ορισμένη ακτίνα γύρω από το σημείο που επιλέχθηκε και τα οποία θα εμφανίζονται με τη βοήθεια πάλι των τεχνολογιών διαδικτύου (php framework-Codeigniter).

Οι παρακάτω απαιτήσεις θεωρήθηκαν βασικές για την εφαρμογή:

1. να διαθέτει κατανοητό και φιλικό προς το χρήστη περιβάλλον επικοινωνίας (user interface) έτσι ώστε να μπορεί ο χρήστης να τη χρησιμοποιεί με ευκολία,
2. να μπορεί να εμφανίζει όλα τα στοιχεία που περιγράφουν τα σημεία ενδιαφέροντος που είναι σε ορισμένη ακτίνα και θα είναι χρήσιμα στον χρήστη,
3. να εμφανίζει μήνυμα αποτυχίας εύρεσης σημείων ενδιαφέροντος αν δεν υπάρχουν τέτοια σημεία σε ορισμένη ακτίνα,
4. να μπορεί να στέλνει επιτυχώς τις συντεταγμένες του σημαδιού που επιλέγει ο χρήστης αρχικά στον ελεγκτή (controller) και μετά στο μοντέλο (model) του Codeigniter εκεί που συνδέεται με τη βάση και εκτελείται το ερώτημα,
5. να μπορεί ο χρήστης αφού έχει επιλέξει πάνω στο χάρτη και έχει δει τα αποτελέσματα ή το μήνυμα αν δεν υπάρχουν, να γυρνάει πίσω στην αρχική σελίδα με το χάρτη για νέα αναζήτηση.

1.3 Σχετικές τεχνολογίες

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκαν, αναπτύχθηκαν και συνδυάστηκαν διάφορες τεχνολογίες, για την δημιουργία του περιβάλλοντος αλληλεπίδρασης με το χρήστη που περιέχει επιπλέον χάρτη, την δημιουργία της βάσης δεδομένων για την αποθήκευση και ανάκτηση χωρικών πληροφοριών και την σύνδεση αυτών των δύο για την εύρεση των αποτελεσμάτων των κριτηρίων αναζήτησης του χρήστη.

Τα Συστήματα Γεωγραφικών Πληροφοριών τα οποία είναι ψηφιακά συστήματα, ικανά να ενσωματώσουν, αποθηκεύσουν, προσαρμόσουν, αναλύσουν και παρουσιάσουν γεωγραφικά συσχετισμένες (geographically-referenced) πληροφορίες. Στην παρούσα μεταπτυχιακή διατριβή αποτελούνται από :

- Postgis[1],
- PostgreSQL[21],
- pgAdminIII[22],
- Quantum[6]

με τα οποία αποθηκεύονται, οργανώνονται και οπτικοποιούνται οι χωρικές πληροφορίες, δηλαδή στην παρούσα μεταπτυχιακή διατριβή τα σημεία ενδιαφέροντος, τα καταστήματα με τις πληροφορίες τους και τις ιδιότητες τύπου γεωμετρίας (geometry).

Οι Τεχνολογίες Διαδικτύου, δηλαδή οι τεχνολογίες που δημιουργούν το περιβάλλον διεπαφής με το χρήστη, λαμβάνουν τα κριτήρια που βάζει ο χρήστης, τα επεξεργάζονται και τέλος εμφανίζουν τα αποτελέσματα σε επίπεδο του Διαδικτύου αποτελούνται από:

- Google maps - Google maps API με το οποίο φορτώνεται ο χάρτης πάνω στον οποίο θα γίνει αναζήτηση των πληροφοριών [9],
- Javascript, Html και Css με τα οποία δημιουργείται η σελίδα πάνω στην οποία θα βρίσκεται ο χάρτης, η σελίδα με την οποία εμφανίζονται τα αποτελέσματα του ερωτήματος και η σελίδα με την οποία εμφανίζεται το μήνυμα ότι δεν υπάρχουν αποτελέσματα,

- Codeigniter το οποίο είναι πλαίσιο (framework) ρηρ και διαχειρίζεται τις πληροφορίες από το χάρτη και από τη βάση δεδομένων για να επεξεργαστούν τα δεδομένα και να οπτικοποιηθούν τα αποτελέσματα[10].

1.4 Δομή της μεταπτυχιακής διατριβής

Παρακάτω αναφέρονται και περιγράφονται περιληπτικά τα περιεχόμενα των κεφαλαίων της μεταπτυχιακής διατριβής.

Στο δεύτερο κεφάλαιο (Σχετική βιβλιογραφία) θα παρουσιαστούν δημοσιεύσεις με αντίστοιχο θέμα με την παρούσα πτυχιακή εργασία και θα γίνει σύγκριση, δηλαδή θα εντοπιστούν τυχόν διαφορές, ομοιότητες που μπορεί να έχουν.

Στο τρίτο κεφάλαιο (Τεχνολογίες που χρησιμοποιήθηκαν) θα αναφερθούν και θα αναλυθούν οι τεχνολογίες που χρησιμοποιήθηκαν στην πτυχιακή εργασία δηλαδή το Codeigniter, Google maps, Google Maps API, Javascript , Html, Css, PostGis, PostgreSQL, pgAdmin και πώς αυτές χρησιμοποιήθηκαν στην παρούσα μεταπτυχιακή διατριβή.

Στο τέταρτο κεφάλαιο (Αρχιτεκτονική της εφαρμογής) θα αναλυθεί η αρχιτεκτονική της εφαρμογής δηλαδή θα αναλυθεί ακριβώς «τι έγινε» και πώς υλοποιήθηκαν οι τεχνολογίες για να επιτευχθεί η τελική εργασία και τέλος θα εξηγηθεί και παραστατικά η αρχιτεκτονική της εφαρμογής με τη βοήθεια UML.

Στο πέμπτο κεφάλαιο (Συμπεράσματα) θα παρουσιαστούν και θα αναλυθούν τα συμπεράσματα από τη δημιουργία και τη χρήση της εφαρμογής και θα αναφερθούν μελλοντικές βελτιώσεις της εφαρμογής μας.

Στο τέλος των κεφαλαίων υπάρχει η Βιβλιογραφία και ένα Παράρτημα. Το Παράρτημα περιέχει τον κώδικα της εφαρμογής με σχετικά σχόλια για την εύκολη ανάγνωσή του.

ΚΕΦΑΛΑΙΟ 2

ΣΧΕΤΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

Σε αυτό το κεφάλαιο θα αναφερθούν αντίστοιχες διατριβές με την παρούσα μεταπτυχιακή διατριβή και θα πραγματοποιηθεί σύγκριση μεταξύ τους, δηλαδή θα επισημανθούν οι διαφορές και οι ομοιότητες που μπορεί να έχουν.

2.1 «Ανάπτυξη μίας εφαρμογής επίβλεψης και ανίχνευσης ζημιών σε μια οικία με τη χρήση εργαλείων της Google»

Αυτή η εργασία γράφτηκε από τους σπουδαστές Παναγιώτου Παναγιώτη και Παπαδόπουλο Νίκο του Τμήματος Εφαρμογών Πληροφορικής στη διοίκηση και στην οικονομία του Παραρτήματος Λευκάδας του Τεχνολογικού Εκπαιδευτικού Ιδρύματος Ιονίων Νησιών με θέμα «Ανάπτυξη μίας εφαρμογής επίβλεψης και ανίχνευσης ζημιών σε μια οικία με τη χρήση εργαλείων της Google» με επιβλέποντα καθηγητή τον Αθανάσιο Κακαρούντα το 2011.

Σύγκριση

Στόχος της παραπάνω διατριβής είναι να δημιουργηθεί μια εφαρμογή που να παρέχει έλεγχο και ασφάλεια σε κτίρια από πιθανούς κινδύνους. Η εφαρμογή θα έχει τη δυνατότητα μέσω αισθητήρων να καταγράφει τι γίνεται στο κτίριο, να στέλνει τις πληροφορίες στη βάση δεδομένων (και εκεί να ελέγχονται τα δεδομένα) και αν διαπιστωθεί ότι υπάρχει κίνδυνος, να ενημερώνει το σύστημα ώστε να αποφευχθούν δυσάρεστες καταστάσεις (φωτιά, διαρροή φυσικού αερίου). Δευτερεύοντα στόχο αποτελεί η ανάπτυξη μιας εφαρμογής μέσα από τις δυνατότητες που δίνονται από τα εργαλεία της Google.

Στην παραπάνω εργασία αναπτύχθηκε μια εφαρμογή επίβλεψης και ανίχνευσης ζημιών σε μια οικία δηλαδή ο στόχος της εργασίας είναι διαφορετικός αλλά η μεθοδολογία μοιάζει καθώς χρησιμοποιεί εργαλεία όπως Γεωγραφικά Συστήματα Πληροφοριών (GIS), Google maps και Google Maps API τα οποία χρησιμοποιήθηκαν και στην παρούσα πτυχιακή. Επίσης χρησιμοποιήθηκαν και τεχνολογίες αισθητήρων οι οποίες δεν χρησιμοποιούνται στην παρούσα πτυχιακή. Επίσης χρησιμοποιήθηκε η PHP απλά και όχι το Codeigniter που είναι php πλαίσιο (framework) που χρησιμοποιείται στην παρούσα μεταπτυχιακή διατριβή.

Επιλέχτηκε η συγκεκριμένη εργασία για μελέτη σύγκρισης με την παρούσα μεταπτυχιακή διατριβή γιατί με την μελέτη αυτή καταλήγουμε στο συμπέρασμα ότι μπορεί κάποια συγκεκριμένα εργαλεία να εξυπηρετήσουν τελείως διαφορετικούς στόχους. Δηλαδή τα Γεωγραφικά Συστήματα Πληροφοριών (GIS), τα Google maps, τα Google Maps API και η php στην παραπάνω εργασία χρησιμοποιήθηκαν για να δημιουργηθεί μια εφαρμογή που να παρέχει έλεγχο και ασφάλεια σε κτίρια από πιθανούς κινδύνους ενώ στην παρούσα μεταπτυχιακή διατριβή για να γίνεται δυναμική διαμόρφωση κριτηρίων γεωπληροφοριακής αναζήτησης. Βέβαια η χρήση αυτών των εργαλείων στην παραπάνω εργασία βοήθησε στην εξοικείωση των εργαλείων και στην πρακτική εφαρμογή αυτών. Επίσης με την μελέτη αυτή φαίνεται η διαφορά ανάμεσα στη php όπου χρησιμοποιήθηκε στην παραπάνω εργασία με το πλαίσιο php Codeigniter όπου χρησιμοποιήθηκε στην παρούσα πτυχιακή διατριβή και τα πλεονεκτήματα χρήσης πλαισίου php και ειδικά του Codeigniter και στην επιλογή χρήσης του Codeigniter στην παρούσα εργασία.

2.2 «Εργαλεία ανάπτυξης εφαρμογών για τον παγκόσμιο ιστό»

Αυτή η εργασία γράφτηκε από το σπουδαστή Γλυκύ Νικόλα του Τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών Τηλεπικοινωνιών και Δικτύων της Πολυτεχνικής σχολής του Πανεπιστημίου Θεσσαλίας με θέμα «Εργαλεία ανάπτυξης εφαρμογών για τον παγκόσμιο ιστό» με επιβλέποντες καθηγητές τον Κατσαρό Δημήτριο και τον Μποζάνη Παναγιώτη το 2011.

Σύγκριση

Στην παραπάνω διατριβή μελετάται και υλοποιείται η ανάπτυξη εργαλείων που επιτρέπουν τη γρήγορη ανάπτυξη ιστοσελίδων και εργαλείων για τον παγκόσμιο ιστό. Παρουσιάζονται τα εργαλεία και οι τεχνολογίες που χρησιμοποιήθηκαν καθώς και οι αρχιτεκτονικές και μοτίβα (patterns) που ακολουθούνται για την υλοποίηση. Η τελική υλοποίηση γίνεται με τη μορφή ενός MVC (Μοντέλο-Προβολή-Ελεγκτής) πλαισίου (Framework) που περιέχει τα κατάλληλα εργαλεία και επιτρέπει ταχεία ανάπτυξη εφαρμογών για τον παγκόσμιο ιστό, σε σύγκριση με τις άλλες μεθόδους. Επίσης αναπτύσσεται ένα Σύστημα Διαχείρισης Περιεχομένου (CMS, Content Management System) που χρησιμοποιεί το παρόν πλαίσιο (Framework). Τέλος παρουσιάζεται η υλοποίηση της εφαρμογής xyz ως παράδειγμα χρήσης του πλαισίου (Framework)-CMS.

Οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίησή του είναι οι γλώσσες HTML, PHP και SQL για τη μορφοποίηση, τη διαμόρφωση των λειτουργιών και τη διαμόρφωση της βάσης δεδομένων αντίστοιχα. Από την άλλη πλευρά τα εργαλεία που χρησιμοποιήθηκαν περιλαμβάνουν τον Apache server (διακομιστή παγκόσμιου ιστού), το σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων MySQL (διακομιστή), το εργαλείο phpMyAdmin γραμμένο σε PHP για τη διαχείριση του MySQL και τέλος τα εργαλεία Visual Paradigm και Pacestar Uml Diagrammer για τη δημιουργία των εξής: σχήματος σχέσης, οντότητας σχέσης και διαγράμματος περιπτώσεων χρήσεως.

Στην παρούσα μεταπτυχιακή διατριβή αναπτύσσονται ιστοσελίδες και εργαλεία για τον παγκόσμιο ιστό, δηλαδή δημιουργείται ιστοσελίδα με το χάρτη για να σημειωθούν τα σημεία (markers) που θα χρησιμοποιηθούν οι συντεταγμένες αυτών και για την παρουσίαση των αποτελεσμάτων δηλαδή των σημείων ενδιαφέροντος που βρίσκονται σε απόσταση ενός χιλιομέτρου από τα σημεία που κάναμε κλικ. Επίσης χρησιμοποιείται το πρότυπο MVC (Μοντέλο-Προβολή-Ελεγκτής) μέσω του πλαισίου PHP το Codeigniter για την υλοποίηση της εφαρμογής. Για την μορφοποίηση ομοίως χρησιμοποιήθηκε η γλώσσα HTML, για τη διαμόρφωση λειτουργιών PHP και τη διαμόρφωση της βάσης δεδομένων SQL. Χρησιμοποιείται και ο Apache server (διακομιστής παγκόσμιου ιστού) και το εργαλείο phpAdminIII για τη διαχείριση του PostgreSQL.

Επιλέχθηκε η παραπάνω εργασία γιατί σε αυτή φαίνονται τα πλεονεκτήματα χρήσης πλαισίου PHP με πρότυπο MVC (Μοντέλο-Προβολή-Ελεγκτής) και το γεγονός αυτό συνέβαλε στη χρήση πλαισίου το οποίο στηρίζεται σε πρότυπο MVC (Μοντέλο-Προβολή-Ελεγκτής) στην παρούσα μεταπτυχιακή διατριβή δηλαδή το Codeigniter.

2.3 «Πληροφοριακό σύστημα για την καθοδήγηση χρηστών με χρήση GIS»

Αυτή η εργασία γράφτηκε από την σπουδάστρια Μανδουράρη Δήμητρα του Τμήματος Εφαρμοσμένης Πληροφορικής & Πολυμέσων της Σχολής Τεχνολογικών Εφαρμογών του Τεχνολογικού Εκπαιδευτικού Ιδρύματος Κρήτης με θέμα «Πληροφοριακό σύστημα για την καθοδήγηση χρηστών με χρήση GIS» με επιβλέποντα καθηγητή τον Παπαδάκη Νίκο το 2011.

Σύγκριση

Η παραπάνω διατριβή έχει ως σκοπό την ανάδειξη του σχεδιασμού και της υλοποίησης ενός πληροφοριακού συστήματος για την καθοδήγηση των χρηστών με τη χρήση GIS (Geographic Information System). Το σύστημα GIS ή αλλιώς Σύστημα Γεωγραφικών Πληροφοριών (ΣΓΠ) είναι ένα εργαλείο "έξυπνου χάρτη", το οποίο επιτρέπει στους χρήστες του να αποτυπώσουν μια περίληψη του πραγματικού κόσμου, να δημιουργήσουν διαδραστικά ερωτήσεις χωρικού ή περιγραφικού χαρακτήρα (αναζητήσεις δημιουργούμενες από τον χρήστη), να αναλύσουν τα χωρικά δεδομένα (spatial data), να τα προσαρμόσουν και να τα αποδώσουν σε αναλογικά μέσα (εκτυπώσεις χαρτών και διαγραμμάτων) ή σε ψηφιακά μέσα (αρχεία χωρικών δεδομένων, διαδραστικοί χάρτες στο Διαδίκτυο). Οι χρήστες του ιστοτόπου μπορούν να έχουν μια εικόνα για το που βρίσκονται τα σημεία που τους ενδιαφέρουν αναζητώντας τα στο χάρτη μέσω του ειδικού πεδίου αναζήτησης. Πιο συγκεκριμένα η αναζήτηση αφορά σημεία που βρίσκονται στο νομό Λασιθίου. Για την ανάπτυξη του δικτυακού τόπου χρησιμοποιήθηκαν οι γλώσσες

προγραμματισμού διαδικτύου html, php, javascript και MySQL καθώς και το σύστημα διαχείρισης βάσεων δεδομένων EasyPHP.

Ο στόχος της παραπάνω διατριβής ήταν η κατασκευή μιας δυναμικής ιστοσελίδας, ενός πληροφοριακού συστήματος, που σκοπό θα είχε την καθοδήγηση των χρηστών της σε μια ορισμένη γεωγραφική περιοχή. Για να δημιουργηθεί μια δυναμική ιστοσελίδα δεν αρκούν μόνο γλώσσες προγραμματισμού όπως η HTML και η Javascript, αλλά χρειάζονται πιο εξειδικευμένες γλώσσες όπως είναι η PHP. Με το πακέτο λογισμικού EasyPHP και τις γλώσσες PHP και MySQL μπορεί να γίνει η διαχείριση της βάσης δεδομένων. Με τη βοήθεια της βάσης δεδομένων τοποθετούνται τα σημάδια (markers) στο χάρτη και οι χρήστες έχουν τη δυνατότητα να εμφανίζουν μόνο τα σημεία για τα οποία ενδιαφέρονται (π.χ. μόνο τα πρατήρια βενζίνης ή/και τους κινηματογράφους). Επίσης υπάρχει πεδίο αναζήτησης στο οποίο ο χρήστης μπορεί να εισάγει τη διεύθυνση την οποία ψάχνει. Το σύστημα, στην περίπτωση αυτή, θα αναζητήσει το σημείο εκείνο και αν υπάρχει στη βάση θα κάνει ζουμ, δημιουργώντας έτσι μια εικόνα του πραγματικού κόσμου.

Εδώ οι στόχοι της παρούσας μεταπτυχιακής διατριβής με την παραπάνω διατριβή μοιάζουν καθώς γίνεται αναζήτηση σε μια βάση δεδομένων στηριζόμενοι σε σημεία που βρίσκονται στο χάρτη, η διαφορά είναι ότι στην παρούσα μεταπτυχιακή διατριβή ο χρήστης βάζει τα σημάδια κάνοντας κλικ και γίνεται αναζήτηση των σημείων ενδιαφέροντος με τη βοήθεια ερωτήματος. Στα εργαλεία που χρησιμοποιήθηκαν η HTML, Javascript και php χρησιμοποιούνται και στην παρούσα μεταπτυχιακή διατριβή, ενώ το πακέτο λογισμικού EasyPHP και MySQL δεν χρησιμοποιούνται καθώς έχουμε τα εργαλεία του πλαισίου της PHP Codeigniter και PostgreSQL αντίστοιχα.

2.4 «Εργαλείο για βελτιστοποίηση G.I.S συστήματος web πλατφόρμας του εργαστηρίου PeLAB»

Αυτή η εργασία γράφτηκε από τους σπουδαστές Ραφαηλίδη Ιωάννη και Φλώρο Ευάγγελο του Τμήματος Εφαρμοσμένης Πληροφορικής & Πολυμέσων της Σχολής Τεχνολογικών Εφαρμογών του Τεχνολογικού Εκπαιδευτικού Ιδρύματος Κρήτης με θέμα «Εργαλείο για βελτιστοποίηση G.I.S συστήματος web πλατφόρμας του εργαστηρίου PeLAB» με επιβλέποντα καθηγητή τον Σωτήριο Λεβέντη το 2002.

Σύγκριση

Με την παραπάνω διατριβή δημιουργήθηκε ένα εργαλείο ανάπτυξης εφαρμογών G.I.S το οποίο συνεργάζεται με εργαλεία διαχείρισης βάσης δεδομένων, όπως είναι η Microsoft Access. Δημιουργήθηκαν αρχεία δεδομένων τα οποία περιέχουν πληροφορίες σχετικά με τους δρόμους της εφαρμογής, όπως οι κόμβοι που σχηματίζουν έναν δρόμο ώστε αυτός να αναπαρασταθεί γραφικά πάνω σε ένα χάρτη οποιαδήποτε και αν είναι η μορφή του. Για την ολοκλήρωση της διατριβής χρησιμοποιήθηκαν τα προγράμματα Microsoft Excel, Microsoft Access, Microsoft Visual InterDev 6.0, Microsoft Visual Basic 6.0 καθώς επίσης και Java.

Η μόνη ομοιότητα ανάμεσα στις εργασίες είναι ότι χρησιμοποιούνται τα γεωγραφικά συστήματα GIS και ότι τα GIS συνεργάζονται με βάσεις δεδομένων. Οι διαφορές είναι στα εργαλεία διαχείρισης βάσης δεδομένων καθώς στην παραπάνω διατριβή χρησιμοποιείται η Microsoft Access.

Επιλέχθηκε η παραπάνω εργασία για μελέτη επειδή γίνεται πολύ καλή χρήση των Γεωγραφικών συστημάτων (GIS) και αυτό βοήθησε σαν παράδειγμα στην παρούσα μεταπτυχιακή διατριβή. Επίσης επειδή χρησιμοποιήθηκαν τελείως διαφορετικά εργαλεία για τη διαχείριση της βάσης δεδομένων και για τη διαδικτυακή πλατφόρμα όπως Microsoft Excel, Microsoft Access, Microsoft Visual InterDev 6.0, Microsoft Visual Basic 6.0 καθώς επίσης και Java όπου φαίνεται η διαφορά στη χρήση των εργαλείων που χρησιμοποιήθηκαν στην παρούσα εργασία όπως PostgreSQL, Codeigniter, HTML και τα πλεονεκτήματα αυτής της χρήσης.

2.5 «Ηλεκτρονικό ινστιτούτο αισθητικής με χρήση MVC PHP Framework»

Αυτή η εργασία γράφτηκε από το σπουδαστή Σταυγιανουδάκη Ιωάννη του Τμήματος Εφαρμοσμένης Πληροφορικής & Πολυμέσων της Σχολής Τεχνολογικών Εφαρμογών του Τεχνολογικού Εκπαιδευτικού Ιδρύματος Κρήτης με θέμα «Ηλεκτρονικό ινστιτούτο αισθητικής με χρήση MVC PHP Framework» με επιβλέπουσα καθηγήτρια την Λογκάκη Μαρία το 2002.

Σύγκριση

Σκοπός της παραπάνω διατριβής είναι η δημιουργία δυναμικής ιστοσελίδας στην οποία στήθηκε ένα ηλεκτρονικό ινστιτούτο αισθητικής για την παροχή στους επισκέπτες πληροφορίες και υπηρεσίες ομορφιάς. Χρησιμοποιηθεί για την υλοποίηση του ένα από τα πιο διάσημα ανοικτού κώδικα (open source) PHP πλαίσια (frameworks) το Codeigniter. Είναι βασισμένο στην αρχιτεκτονική MVC (Model-View-Controller).

Το μοντέλο (Model) είναι το κομμάτι που διαχειρίζεται λειτουργίες πάνω σε δεδομένα της βάσης, το View έχει να κάνει με το τι φαίνεται στο χρήστη, δηλαδή οι σελίδες της εφαρμογής μας και το Controller ελέγχει ενέργειες του χρήστη, τις αναγνωρίζει και τις μεταφέρει στο κατάλληλο σημείο του κώδικα όπου θα εκτελεστεί μια συγκεκριμένη λειτουργία. Επίσης δέχεται τα δεδομένα που προκύπτουν από τα Models και τα μεταφέρει στο κατάλληλο View.

Το σύστημα θα αποτελείται από μια βάση MySQL, όπου θα χρησιμοποιηθεί για να αποθηκευτούν στοιχεία των χρηστών, των προϊόντων και των υπηρεσιών που θα παρέχονται στο ηλεκτρονικό σύστημα. Όταν ο επισκέπτης εισέλθει στην αρχική σελίδα του ινστιτούτου αισθητικής θα έχει τη δυνατότητα να περιηγηθεί σε προϊόντα, όπου μπορεί να δει πληροφορίες, και να τα προσθέσει σε ένα Shopping cart έτσι ώστε να τα αγοράσει με διάφορους τρόπους πληρωμής και να τα παραλάβει είτε στο σπίτι του είτε από το φυσικό κατάστημα του ινστιτούτου.

Επίσης για να μπορέσει να έχει πρόσβαση σε όλες τις σελίδες και για παραπάνω ενέργειες θα υπάρχει ένα πλήρες σύστημα χρήστη. Κάνοντας login αν είναι μέλος, register αν δεν είναι, θα υπάρχει δυνατότητα e-appointment, δηλαδή θα μπορεί να κλείσει ραντεβού για τις υπηρεσίες που παρέχονται με άκρως διαδραστικό τρόπο, φιλικό και εύχρηστο για κάθε απαιτητικό επισκέπτη.

Θα υπάρχει σελίδα διαχείρισης του χρήστη όπου θα μπορεί να βλέπει παραγγελίες ή ραντεβού που έχουν κλείσει πελάτες. Πληροφορίες για πελάτες, όπως και θα έχει την δυνατότητα να δημοσιεύει άρθρα, νέα να προσθέτει καινούργια προϊόντα, υπηρεσίες ή να διαχειρίζεται προϊόντα και υπηρεσίες. Για την επιτυχημένη ολοκλήρωση χρησιμοποιούνται τεχνολογίες και εργαλεία όπως: MVC Framework (Codeigniter), Xhtml, CSS PHP5, MySQL, XML, AJAX (JQuery), Photoshop.

Ο στόχος της παραπάνω εργασίας είναι πολύ διαφορετικός από το στόχο της παρούσας πτυχιακής εργασίας καθώς δεν χρησιμοποιούνται γεωγραφικά συστήματα GIS και η αναζήτηση δεν γίνεται με τη βοήθεια του χάρτη. Το κοινό σημείο είναι ότι δημιουργείται δυναμική ιστοσελίδα με τη βοήθεια του Codeigniter και γίνεται αναζήτηση σε μια βάση δεδομένων. Η διαχείριση της γίνεται με το εργαλείο MySQL ενώ στην παρούσα πτυχιακή εργασία η διαχείριση της βάσης δεδομένων γίνεται με τη βοήθεια της PostgreSQL και της PhpAdmin καθώς διαχειρίζονται χωρικά δεδομένα.

2.6 «Χρήση Mobile GIS, για διαχείριση εκτάκτων αναγκών»

Αυτή η εργασία γράφτηκε από το σπουδαστή Γιώργο Τσιουτσιουλική Τμήμα Πληροφορικής του Ιόνιου Πανεπιστήμιου με θέμα «Χρήση Mobile GIS, για διαχείριση εκτάκτων αναγκών» με επιβλέποντα καθηγητή τον Σπύρος Σιούτας το 2010.

Σύγκριση

Η παραπάνω διατριβή, προσπαθεί να παρουσιάσει μια ιδιαίτερη χρήση των τεχνολογιών που καλύπτονται από την ομπρέλα του όρου Mobile GIS και την χρήση τους για συλλογή και

ανάλυση δεδομένων σεισμολογικού τύπου. Η βασική ιδέα πίσω από την ενασχόληση με τον τομέα της Σεισμολογίας, ήταν η δημιουργία μιας πλήρους γεωγραφικής βάσης δεδομένων, στην οποία θα γίνεται η συλλογή αποτελεσμάτων των σεισμολογικών πορισμάτων των κτηρίων της Κέρκυρας, η οποία πέρα από της δυνατότητες ανάλυσης στο υλικό όπου θα γίνει η συλλογή κατά τη διάρκεια της εργασίας, θα μπορέσει να διατηρηθεί και να χρησιμοποιηθεί επί μακρόν, έτσι ώστε είτε μόνη της, είτε σε συνδυασμό με άλλες ομάδες δεδομένων, να αναλυθεί περαιτέρω και να προσφέρει πλήθος πληροφορίας, η οποία σήμερα δεν μπορεί να βρεθεί κάπου. Αναλυτικότερα, η διατριβή περιλαμβάνει ένα αρκετά αξιόλογο θεωρητικό υπόβαθρο, για τους τομείς των βάσεων δεδομένων, της σεισμολογίας, της χαρτογραφίας και τέλος της τεχνολογίας GPS.

Ο τομέας της παραπάνω διατριβής είναι η Σεισμολογία ο οποίος είναι διαφορετικός από τον τομέα που ασχολείται η παρούσα πτυχιακή εργασία. Το κοινό σημείο είναι στα γεωγραφικά συστήματα GIS που χρησιμοποιούνται και στις δύο διατριβές καθώς εισάγονται, επεξεργάζονται και οπτικοποιούνται χωρικά δεδομένα. Στην προκειμένη περίπτωση τα χωρικά δεδομένα εισάγονται από το mobile GPS ενώ στην παρούσα μεταπτυχιακή διατριβή τα δεδομένα εισάγονται από το χάρτη καθώς κάνουμε κλικ πάνω σε αυτό.

Επιλέχθηκε η συγκεκριμένη εργασία για μελέτη διότι παρουσιάζεται εκτενέστατα η χρήση των γεωγραφικών συστημάτων και αυτό βοήθησε και στην παρούσα μεταπτυχιακή διατριβή καθώς χρησιμοποιούνται γεωγραφικά συστήματα. Βέβαια στην παραπάνω εργασία χρησιμοποιούνται στον τομέα της Σεισμολογίας και έτσι φαίνεται η ευρεία περιοχή εφαρμογής αυτών.

2.7 «Ανασύνθεση του πακέτου “Web G.I.S. – Αθηνών” του εργαστηρίου PeLAB με χρήση ORACLE 8.0»

Αυτή η εργασία γράφτηκε από τους σπουδαστές Ζαφειριάδου Ευστρατία και Καζάκη Μιχαήλ του Τμήματος Η/Υ Συστημάτων του Τ.Ε.Ι. Πειραιά με θέμα «Ανασύνθεση του πακέτου “Web G.I.S. – Αθηνών” του εργαστηρίου PeLAB με χρήση ORACLE 8.0» με επιβλέποντα καθηγητή τον Σωτήριο Λεβέντη το 2001.

Σύγκριση

Στην παραπάνω διατριβή χρησιμοποιήθηκαν νέα εργαλεία ανάπτυξης εφαρμογών G.I.S. τα οποία συνεργάζονται με εργαλεία διαχείρισης βάσης δεδομένων, όπως είναι η ORACLE 8.0. Δημιουργήθηκε νέα βάση διαχείρισης δεδομένων η οποία περιέχει οδούς, την περιοχή στην οποία βρίσκεται κάθε οδός, τον Ταχυδρομικό Κώδικα καθώς και τις γεωγραφικές συντεταγμένες της περιοχής που βρίσκεται η οδός. Η παρούσα εφαρμογή είναι αναπτυσσόμενη αφού παρέχει την πύλη διασύνδεσης των εφαρμογών G.I.S. με εργαλεία διαχείρισης βάσης δεδομένων. Για την ολοκλήρωση της διατριβής χρησιμοποιήθηκαν τα προγράμματα Microsoft Excel, Microsoft Access, Microsoft Visual InterDev 6.0 και Oracle 8.0.

Το μόνο κοινό σημείο με την παραπάνω διατριβή με την παρούσα μεταπτυχιακή διατριβή είναι ότι χρησιμοποιούνται τα γεωγραφικά συστήματα G.I.S και ότι συνεργάζονται με εργαλεία διαχείρισης δεδομένων αλλά τα εργαλεία αυτά δεν είναι ίδια. Δεν χρησιμοποιούνται τα προγράμματα Microsoft Excel, Microsoft Access, Microsoft Visual InterDev 6.0 και Oracle 8.0. αλλά PostgreSQL, PgAdmin και Google maps και Javascript, HTML.

Επιλέχθηκε αυτή η συγκεκριμένη εργασία για μελέτη διότι παρουσιάζεται εκτενέστατα η χρήση των γεωγραφικών συστημάτων και αυτό βοήθησε και στην παρούσα μεταπτυχιακή διατριβή καθώς χρησιμοποιούνται γεωγραφικά συστήματα. Βέβαια στην παραπάνω εργασία χρησιμοποιούνται πολύ διαφορετικά εργαλεία διαχείρισης βάσης δεδομένων και το γεγονός αυτό δείχνει τα ιδιαίτερα χαρακτηριστικά των εργαλείων που χρησιμοποιούνται στην παρούσα μεταπτυχιακή διατριβή και των σκοπών που εξυπηρετούν αυτά.

2.8 Ανακεφαλαίωση

Παραπάνω αναλύθηκαν άλλες διατριβές και παρουσιάστηκαν οι διαφορές και οι ομοιότητες αυτών με την παρούσα μεταπτυχιακή διατριβή. Επιλέχθηκαν οι συγκεκριμένες εργασίες για μελέτη διότι κάποιες είναι χρήσιμες ως παράδειγμα εφαρμογής κάποιων συγκεκριμένων εργαλείων και κάποιες άλλες που χρησιμοποιούν τελείως διαφορετικά εργαλεία βοήθησαν στην επιλογή των εργαλείων που χρησιμοποιήθηκαν τελικά. Επίσης βλέπουμε την ευρεία χρήση που μπορεί να έχουν κάποια εργαλεία και τους διαφορετικούς στόχους που μπορεί να εξυπηρετούν.

Στο επόμενο κεφάλαιο θα αναλυθούν διεξοδικά οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν στην παρούσα μεταπτυχιακή διατριβή, τα πλεονεκτήματά τους και η αξιοποίηση αυτών στη συγκεκριμένη εφαρμογή.

ΚΕΦΑΛΑΙΟ 3

ΤΕΧΝΟΛΟΓΙΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ

3.1 Δυναμική ιστοσελίδα

Δυναμική ιστοσελίδα (dynamic web page) είναι μια ιστοσελίδα η οποία δημιουργείται δυναμικά την στιγμή της πρόσβασης σε αυτή ή την στιγμή που ο χρήστης αλληλεπιδρά με τον εξυπηρετητή ιστοσελίδων. Οι δυναμικές ιστοσελίδες θεωρούνται δομικό στοιχείο της νέας γενιάς του παγκόσμιου ιστού όπου η πληροφορία διαμοιράζεται σε πολλαπλές ιστοσελίδες. Η δυναμική ιστοσελίδα μπορεί να δημιουργείται δυναμικά από ένα σενάριο εντολών, το οποίο εκτελείται τοπικά στο πελάτη, στον εξυπηρετητή ή και στον πελάτη και στον εξυπηρετητή.

Δυναμική ιστοσελίδα τοπικά στο πελάτη. Η σελίδα συνήθως περιέχει ένα σενάριο εντολών το οποίο ενσωματώνει μια διαδραστικότητα με τον χρήστη (η διαδραστικότητα γίνεται με χρήση πληκτρολογίου - ποντικιού). Το περιεχόμενο της ιστοσελίδας δημιουργείται δυναμικά στον τοπικό υπολογιστή μετά την εκτέλεση του σεναρίου εντολών το οποίο λαμβάνεται από τον απομακρυσμένο εξυπηρετητή.

Δυναμική ιστοσελίδα στον εξυπηρετητή. Μια δυναμική ιστοσελίδα συνήθως δημιουργείται από ένα λογισμικό το οποίο τρέχει στον απομακρυσμένο εξυπηρετητή ιστοσελίδων. Ο χρήστης-πελάτης επισκέπτεται τον εξυπηρετητή ιστοσελίδων ο οποίος δημιουργεί δυναμικά την ιστοσελίδα την οποία επιστρέφει σε μορφή HTML στον πελάτη. Τέτοιες τεχνολογίες για την δημιουργία δυναμικών ιστοσελίδων στον εξυπηρετητή είναι η τεχνολογίες ASP, ColdFusion, Perl, PHP, WebDNA και άλλες. Οι γλώσσες αυτές συνήθως (με εξαίρεση την ASP.NET και την JSP) χρησιμοποιούν ένα σενάριο εντολών CGI κατά την δυναμική δημιουργία των σελίδων.

Στην παρούσα μεταπτυχιακή διατριβή γίνεται ένας συνδυασμός των δύο παραπάνω, δηλαδή πραγματοποιείται ανταλλαγή περιεχομένου και δεδομένων μεταξύ του υπολογιστή του πελάτη και του απομακρυσμένου εξυπηρετητή και η σελίδα διαμορφώνεται από το σενάριο εντολών το οποίο τρέχει τοπικά στο πελάτη. Αυτό επιτυγχάνεται καθώς στην αρχική σελίδα που περιέχει το χάρτη και εμφανίζεται στο φυλλομετρητή του Διαδικτύου με τη διεύθυνση URL: localhost/ergasia7/index.php/pages/view/map και το αρχείο map.php βρίσκεται στην τοποθεσία C:\Program Files\Bitnami WAPPStack\apache2\hotdocs\ergasia7\application\views\pages, καθώς ο χρήστης κάνει κλικ, δημιουργείται σημάδι (marker) πάνω στο χάρτη και λαμβάνονται οι συντεταγμένες του σημαδιού και με ανακατεύθυνση (redirect) σε μια μέθοδο του controller. Έτσι λοιπόν επιδρά ο χρήστης πάνω στο σύστημα πατώντας πάνω στο χάρτη. Από εκεί και πέρα αναλαμβάνει το Codeigniter, πλαίσιο (framework) της php όπου γίνεται η σύνδεση με τον εξυπηρετητή, όπου στην προκειμένη περίπτωση είναι το Apache web Server. Οι συντεταγμένες του σημαδιού στέλνονται από το controller σε μια μέθοδο του model, όπου γίνεται σύνδεση με τη βάση δεδομένων και εκτελείται το ερώτημά μας και επιστρέφεται στο χρήστη ιστοσελίδα σε μορφή HTML της οποίας το περιεχόμενο εξαρτάται από το εκτελέσιμο ερώτημα.

3.2 Codeigniter

Το Codeigniter είναι ένα πλαίσιο (framework) ανοιχτού λογισμικού για την κατασκευή δυναμικών ιστοσελίδων με PHP. Είναι ένα ισχυρό πλαίσιο PHP που καταλαμβάνει ελάχιστο χώρο, χτίστηκε για προγραμματιστές της PHP που χρειάζονται ένα απλό και κομψό σύνολο εργαλείων για να δημιουργήσουν πλήρεις δυνατότητες εφαρμογών διαδικτύου (web).

3.2.1 Χαρακτηριστικά του Codeigniter

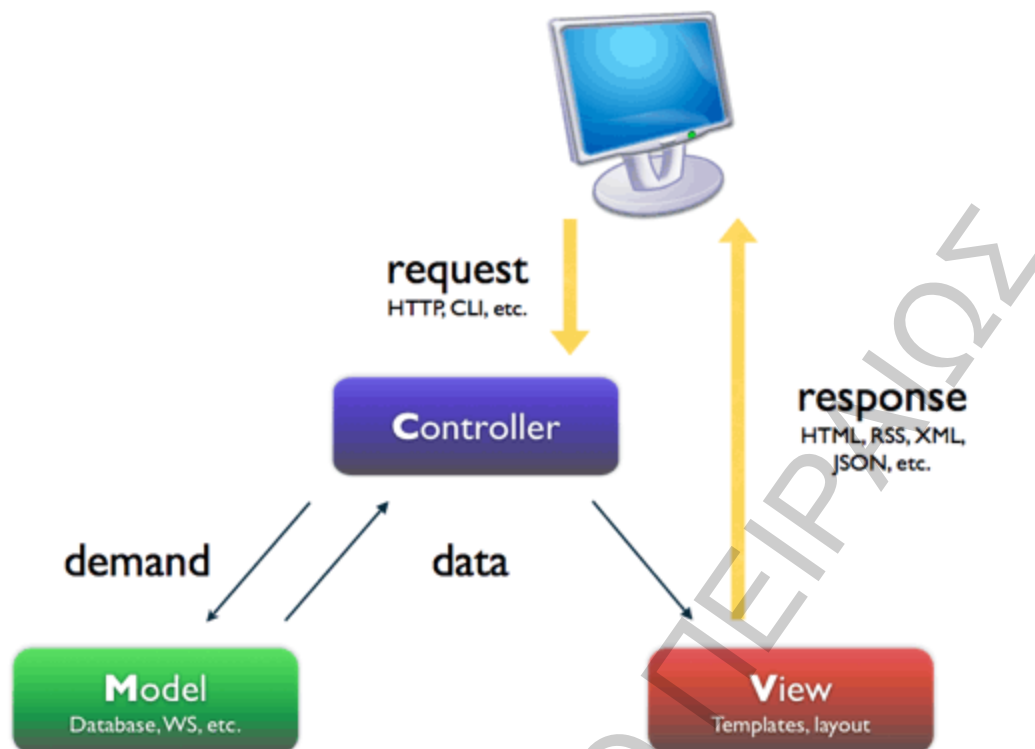
Το Codeigniter είναι πλαίσιο (framework) PHP, το οποίο παρέχεται δωρεάν, είναι αληθινά ελαφρύ, γρήγορο και μπορεί να επεκταθεί εύκολα. Το Codeigniter απευθύνεται σε αυτούς που θέλουν ένα πλαίσιο με ένα μικρό ίχνος, να έχει εξαιρετική απόδοση, ευρεία συμβατικότητα. Επίσης είναι ένα πλαίσιο που απαιτεί σχεδόν μηδαμινή διαμόρφωση, που δεν απαιτεί χρήση γραμμής εντολών, που δεν έχει περιορισμούς κωδικοποίησης και χαρακτηρίζεται για την απλοϊκότητα του.

3.2.2 Μοντέλο σχεδιασμού MVC

Χρησιμοποιώντας το μοντέλο σχεδιασμού MVC (Model-View-Controller) το πρόγραμμα χωρίζεται σε τρία βασικά μέρη :

1. Το μοντέλο (*Model*) αναπαριστά τα δεδομένα της εφαρμογής
2. Η προβολή (*View*) παράγει μια παρουσίαση των δεδομένων του μοντέλου
3. Ο ελεγκτής (*Controller*) διαχειρίζεται και κατευθύνει τις αιτήσεις που κάνει ο χρήστης

Στην Εικόνα1 εξηγείται η σύνδεση των βασικών μερών του προτύπου MVC. Δηλαδή πώς από τον υπολογιστή του χρήστη γίνεται αίτημα από τη σελίδα που είναι φορτωμένη στο φυλλομετρητή στον Ελεγκτή και από εκεί καλεί το Μοντέλο στο οποίο γίνεται η κλήση της Βάσης Δεδομένων και πραγματοποιείται το ερώτημα για να επιστραφούν τα αποτελέσματα στον Ελεγκτή. Έπειτα καλούνται τα αρχεία που βρίσκονται στην Προβολή και εμφανίζονται στον υπολογιστή του χρήστη.



Εικόνα 1. Μια βασική αίτηση στο MVC (πηγή[19])

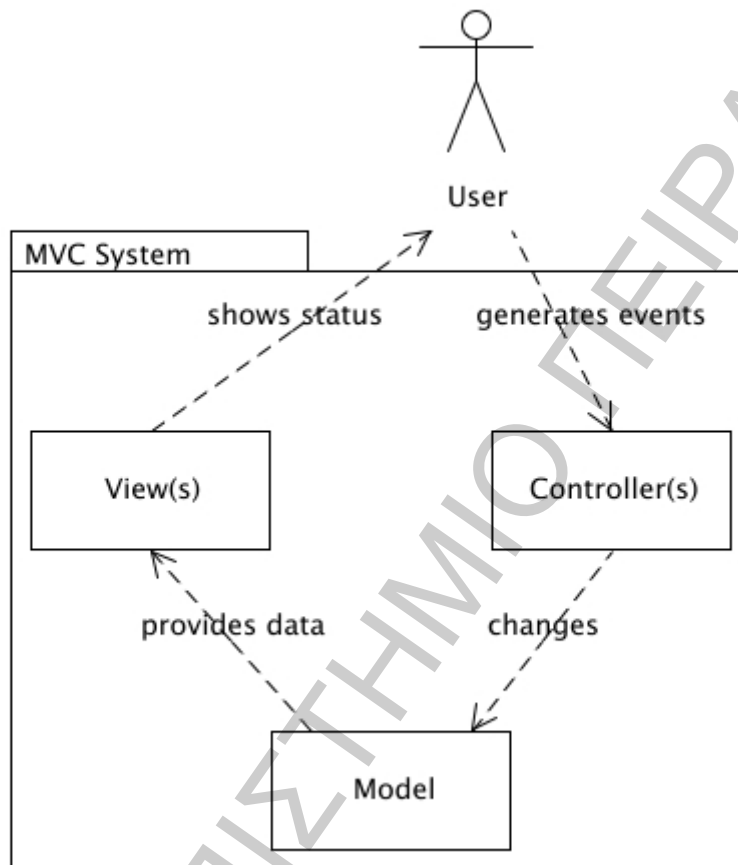
Το MVC είναι μια αρχιτεκτονική προγραμματισμού που βασίζεται στο διαχωρισμό κώδικα λογικής από τον κώδικα εμφάνισης και αλληλεπίδρασης με το χρήστη. Αυτή η αρχιτεκτονική αν υιοθετηθεί επιτρέπει την καλύτερη οργάνωση του κώδικα, καθώς και εύκολη συντήρηση του κώδικα. Επίσης επιτρέπει σε πολλούς προγραμματιστές να δουλεύουν στο ίδιο σύστημα ταυτόχρονα, καθώς και ανάπτυξη και αποτελμάτωση κάθε κομματιού κώδικα ξεχωριστά. Στην πράξη, αυτό επιτρέπει στις ιστοσελίδες να περιέχουν ελάχιστο κώδικα αφού η παρουσίαση είναι ξεχωριστά από τον κώδικα PHP.

Το Μοντέλο (Model) αναπαριστά τις δομές δεδομένων. Συνήθως οι κλάσεις του μοντέλου περιέχουν λειτουργίες που βοηθούν στην ανάκτηση, εισαγωγή και ενημέρωση των πληροφοριών στη βάση δεδομένων. Η Προβολή (View) είναι η πληροφορία που παρουσιάζεται στο χρήστη. Μία Προβολή μπορεί να είναι κανονικά μια ιστοσελίδα, αλλά σε CodeIgniter, ένα View που μπορεί επίσης να είναι ένα κομμάτι της σελίδας, όπως μια κεφαλίδα ή υποσέλιδο. Ο Ελεγκτής (Controller) λειτουργεί ως μεσάζων μεταξύ του Μοντέλου και της Προβολής, καθώς και κάθε άλλους πόρους που απαιτούνται για την επεξεργασία της αίτησης HTTP και να δημιουργήσει μια ιστοσελίδα.

Το CodeIgniter έχει μια αρκετά χαλαρή προσέγγιση για το MVC δεδομένου ότι δεν απαιτούνται τα μοντέλα. Μπορεί κατ' επιλογή του αρχιτέκτονα λογισμικού, να αγνοηθούν τα μοντέλα και να φτιαχτεί η αίτηση χρησιμοποιώντας μόνο Ελεγκτές και Προβολές. Το CodeIgniter επιτρέπει επίσης να ενσωματωθούν ήδη υπάρχοντες κώδικες, ή ακόμα και την ανάπτυξη βασικών βιβλιοθηκών του συστήματος, έτσι ώστε να λειτουργήσει με τρόπο που να έχει περισσότερο νόημα για τους χρήστες.

Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε το Codeigniter για να χρησιμοποιηθούν οι συντεταγμένες του σημαδιού (marker) αφού κάνει κλικ ο χρήστης πάνω στο χάρτη, δηλαδή περνάνε οι συντεταγμένες σε μια μέθοδο του Ελεγκτή pages και συγκεκριμένα στην μέθοδο landmarks και αυτή καλεί μια μέθοδο του μοντέλου landmarks_model που θα

περαστούν οι συντεταγμένες για να εκτελεστεί το ερώτημα που θα βρεθούν τα σημεία ενδιαφέροντος που βρίσκονται σε απόσταση 1 km από το σημάδι. Βέβαια στο Μοντέλο ήδη έχει καλεστεί η βάση δεδομένων που περιέχει τα σημεία ενδιαφέροντος. Τα αποτελέσματα θα επιστραφούν στην μέθοδο πάλι landmarks του Ελεγκτή και θα εμφανιστούν μέσω της Προβολής και συγκεκριμένα μέσω του αρχείου landmarks που είναι μέσα στο φάκελο Views. Αυτό φαίνεται και στην εικόνα, δηλαδή η σύνδεση των βασικών μερών του Codeigniter.



Εικόνα 2. Σχηματική παράσταση του MVC (πηγή[20])

3.3 Google maps

Το Google Maps είναι μια δικτυακή (web) εφαρμογή και τεχνολογία χαρτογράφησης που παρέχονται από το Google και στελεχώνουν υπηρεσίες οι οποίες είναι βασισμένες σε χάρτη, συμπεριλαμβανομένης της ιστοσελίδας Google Maps, Google Transit, (είναι μια διαδικτυακή υπηρεσία που βοηθά τους χρήστες να φτάσουν στον προορισμό τους, παρέχοντας εύκολες διαδρομές και πληροφορίες κατεύθυνσης χρησιμοποιώντας τις διαθέσιμες δημόσιες συγκοινωνίες) και χάρτες ενσωματωμένους σε τρίτες ιστοσελίδες μέσω του Google Maps API. Προσφέρει οδικούς χάρτες, σχεδιασμό διαδρομής για ταξίδι με τα πόδια, με αυτοκίνητο ή με ποδήλατο ή πρόγραμμα εντοπισμού δημόσιας και αστικής συγκοινωνίας για πολλές χώρες σε όλο τον κόσμο.

Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε το Google maps για να έχουμε διαθέσιμο χάρτη σε ιστοσελίδα ώστε να μπορεί ο χρήστης να κάνει κλικ πάνω του και να δημιουργηθεί σημάδι (marker) και να βρεθούν τα σημεία ενδιαφέροντος σε απόσταση 1km γύρω από αυτό με τη βοήθεια του Codeigniter.

3.4 Google Maps API (Διεπαφή Προγραμματισμού Εφαρμογών Google Maps)

Το Google Maps API είναι μια διεπαφή προγραμματισμού εφαρμογών (Application Programming Interface – API) της Google η οποία δίνει τη δυνατότητα ενσωμάτωσης ενός δυναμικού χάρτη σε ιστοσελίδες με τη χρήση Javascript. Το API παρέχει έναν αριθμό εργαλείων για το χειρισμό χαρτών και την προσθήκη περιεχομένου σε αυτούς μέσω διαφόρων υπηρεσιών, όπως και στην επίσημη ιστοσελίδα του Google Maps (<http://maps.google.com/>), επιτρέποντας έτσι τη δημιουργία εύρωστων εφαρμογών που εκμεταλλεύονται γεωγραφικές πληροφορίες.

Το Google Maps API είναι μια δωρεάν υπηρεσία, διαθέσιμη για οποιοδήποτε ιστότοπο που είναι ελεύθερος για το κοινό. Το μόνο που χρειάζεται είναι ένα κλειδί API (API key) που μπορεί να ζητηθεί εφόσον υπάρχει λογαριασμός στη Google και αυτομάτως συνδέεται με το λογαριασμό.

Βασικό στοιχείο οποιασδήποτε εφαρμογής Google Maps API είναι ο ίδιος ο χάρτης, ο οποίος φορτώνεται και αρχικοποιείται με συγκεκριμένες διαστάσεις (ύψος και πλάτος), κέντρο (γεωγραφικό πλάτος, γεωγραφικό μήκος), επίπεδο εστίασης και τύπο (κανονικό, δορυφορικό, υβριδικό). Στη συνέχεια είναι δυνατή η αλληλεπίδραση με το χάρτη μέσω διαφόρων μηχανισμών όπως είναι τα συμβάντα (events), οι έλεγχοι (controls), οι επιστρώσεις (overlays) αλλά και συναρτήσεις επεξεργασίας των χαρακτηριστικών του. Στην παρούσα μεταπτυχιακή εργασία χρησιμοποιήθηκε το εργαλείο Google Maps JavaScript API v3. Αυτή η έκδοση είναι σχεδιασμένη για αυτούς που γνωρίζουν τον προγραμματισμό JavaScript και αντικειμενοστραφείς έννοιες προγραμματισμού. Θα πρέπει επίσης να είναι εξοικειωμένοι με το Google Maps από την πλευρά του χρήστη του.

Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε το Google maps API στο αρχείο map καθώς χρησιμοποιήθηκε το συμβάν (event) για την λειτουργία του κλικ, δηλαδή αν κάνει κλικ ο χρήστης πάνω στο χάρτη να δημιουργηθεί σημάδι (marker).

3.5 Javascript

Η JavaScript είναι γλώσσα προγραμματισμού ηλεκτρονικών υπολογιστών, η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου και την εκτέλεση κώδικα στην πλευρά του πελάτη (client-side) σε ιστοσελίδες.

Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε η Javascript στην κατασκευή της αρχικής σελίδας με το χάρτη, δηλαδή το αρχείο map όπου θα σταλούν οι συντεταγμένες του σημείου που πατάμε με τη βοήθεια ανακατεύθυνσης (redirection).

3.6 Html

Η HTML (ακρωνύμιο του αγγλικού HyperText Markup Language – Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων.

Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες, οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες, κλπ.

Ο σκοπός ενός φυλλομετρητή του Διαδικτύου (web browser) είναι να διαβάζει τα έγγραφα HTML και τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο φυλλομετρητής δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας.

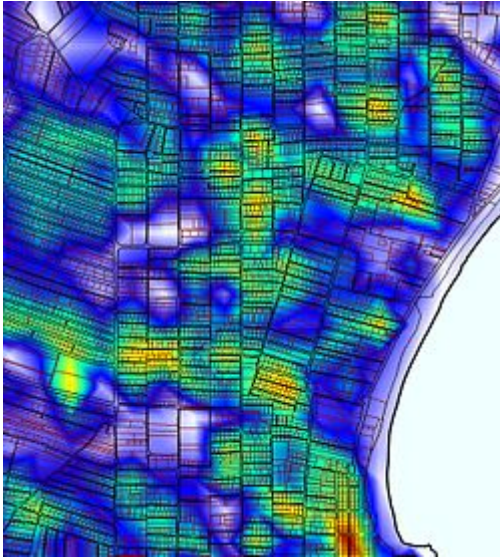
Τα στοιχεία της HTML χρησιμοποιούνται για να κτίσουν όλους του ιστότοπους. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου) καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, παραθέσεις και άλλα. Μπορεί επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

Οι φυλλομετρητές μπορούν επίσης να αναφέρονται σε στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου.

Στην παρούσα μεταπτυχιακή διατριβή η HTML χρησιμοποιήθηκε στην κατασκευή της αρχικής σελίδας με το χάρτη, δηλαδή το αρχείο map, της σελίδα με τα αποτελέσματα των σημείων ενδιαφερόντων, δηλαδή το αρχείο landmarks και της σελίδας με το μήνυμα ότι δεν υπάρχουν αποτελέσματα, δηλαδή το αρχείο noResults που βρίσκονται μέσα στο φάκελο views.

3.7 Γεωγραφικά Συστήματα Πληροφοριών (GIS)

Τα Συστήματα Γεωγραφικών Πληροφοριών (ΣΓΠ), γνωστό ευρέως και ως G.I.S. (Geographic Information Systems), είναι συστήματα διαχείρισης χωρικών δεδομένων (spatial data) και συσχετισμένων ιδιοτήτων. Στην πιο αυστηρή μορφή του είναι ένα ψηφιακό σύστημα, ικανό να ενσωματώσει, αποθηκεύσει, προσαρμόσει, αναλύσει και παρουσιάσει γεωγραφικά συσχετισμένες (geographically-referenced) πληροφορίες. Σε πιο γενική μορφή, ένα ΣΓΠ είναι ένα εργαλείο "έξυπνου χάρτη", το οποίο επιτρέπει στους χρήστες του να αποτυπώσουν μια περίληψη του πραγματικού κόσμου, να δημιουργήσουν διαδραστικά ερωτήσεις χωρικού ή περιγραφικού χαρακτήρα (αναζητήσεις δημιουργούμενες από τον χρήστη), να αναλύσουν τα χωρικά δεδομένα (spatial data), να τα προσαρμόσουν και να τα αποδώσουν σε αναλογικά μέσα (εκτυπώσεις χαρτών και διαγραμμάτων) ή σε ψηφιακά μέσα (αρχεία χωρικών δεδομένων, διαδραστικοί χάρτες στο Διαδίκτυο. Τα συστήματα GIS, όπως και τα συστήματα CAD, αποτυπώνουν χωρικά δεδομένα σε γεωγραφικό ή χαρτογραφικό ή καρτεσιανό σύστημα συντεταγμένων. Βασικό χαρακτηριστικό των ΣΓΠ είναι ότι τα χωρικά δεδομένα συνδέονται και με περιγραφικά δεδομένα, π.χ. μια ομάδα σημείων που αναπαριστούν θέσεις πόλεων συνδέεται με ένα πίνακα όπου κάθε εγγραφή εκτός από τη θέση περιέχει πληροφορίες όπως ονομασία, πληθυσμός κλπ. Στην Εικόνα 3 και στην Εικόνα 4 φαίνονται δύο χάρτες οι οποίοι χρήσιμοι και αξιοποιήσιμοι. Στην Εικόνα 3 φαίνεται ο χάρτης Οικιστικής Πυκνότητας κατάλληλος για λήψη αποφάσεων και στην Εικόνα 4 βρίσκεται ένα χάρτης Πυκνότητας πληθυσμού που μπορεί να χρησιμοποιηθεί για οικονομικούς και εμπορικούς σκοπούς.



Εικόνα 3. Χάρτης Οικιστικής Πυκνότητας, κατάλληλος για λήψη αποφάσεων, π.χ. ανάπτυξη υποδομών ΟΤΑ (πηγή [18])



Εικόνα 4. Χάρτης Πυκνότητας Πληθυσμού (πηγή[18])

Η χαρακτηριστική δυνατότητα που παρέχουν τα GIS είναι αυτή της σύνδεσης της χωρικής με την περιγραφική πληροφορία (η οποία δεν έχει από μόνη της χωρική υπόσταση). Η τεχνολογία που χρησιμοποιείται για την λειτουργία αυτή βασίζεται:

- Είτε στο σχεσιακό (relational) μοντέλο δεδομένων, όπου τα περιγραφικά δεδομένα πινακοποιούνται χωριστά και αργότερα συσχετίζονται με τα χωρικά δεδομένα μέσω κάποιων μοναδικών τιμών που είναι κοινές και στα δύο είδη δεδομένων.
- Είτε στο αντικειμενοστραφές (object-oriented) μοντέλο δεδομένων, όπου τόσο τα χωρικά όσο και τα περιγραφικά δεδομένα συγχωνεύονται σε αντικείμενα, τα οποία μπορεί να μοντελοποιούν κάποια αντικείμενα με φυσική υπόσταση (π.χ. κατηγορία = "δρόμος", όνομα = "Πανεπιστημίου", γεωμετρία = "[X1,Y1],[X2,Y2]...", πλάτος = "20μέτρα").

Το αντικειμενοστρεφές μοντέλο τείνει να χρησιμοποιείται όλο και περισσότερο σε εφαρμογές GIS εξαιτίας των αυξημένων δυνατοτήτων που παρέχει σε σχέση με το σχεσιακό μοντέλο της δυνατότητας για την εύκολη και απλοποιημένη μοντελοποίηση σύνθετων φυσικών φαινομένων και αντικειμένων με χωρική διάσταση.

Πολλές φορές η ολοκληρωμένη έννοια των GIS (integrated GIS concept) επεκτείνεται για να συμπεριλάβει τόσο τα δεδομένα (που αποτελούν ουσιαστικά τον πυρήνα τους), το λογισμικό και τον μηχανικό εξοπλισμό, όσο και τις διαδικασίες και το ανθρώπινο δυναμικό, που αποτελούν αναπόσπαστα τμήματα ενός οργανισμού, ο οποίος έχει σαν πρωταρχική του δραστηριότητα την διαχείριση πληροφορίας με την βοήθεια GIS.

Στην παρούσα μεταπτυχιακή διατριβή τα Συστήματα Γεωγραφικών Πληροφοριών χρησιμοποιήθηκαν για να αποθηκευτούν τα σημεία ενδιαφέροντος, δηλαδή τα καταστήματα με τις πληροφορίες τους όπου περιέχουν χωρικά δεδομένα.

3.7.1 PostgreSQL

Η PostgreSQL είναι μια σχεσιακή βάση δεδομένων ανοικτού κώδικα με πολλές δυνατότητες. Η ανάπτυξη της διαρκεί ήδη πάνω από δύο δεκαετίες και βασίζεται σε μια αποδεδειγμένα καλή αρχιτεκτονική η οποία έχει δημιουργήσει μια ισχυρή αντίληψη των χρηστών της γύρω από την αξιοπιστία, την ακεραιότητα δεδομένων και την ορθή λειτουργία.

Η PostgreSQL τρέχει σε όλα τα βασικά λειτουργικά συστήματα, στα οποία περιλαμβάνονται το Linux, το UNIX (AIX, BSD, HP-UX, SGI, IRIX, MAC OS X, Solaris, Tru64) και τα Windows. Επίσης υποστηρίζει αποθήκευση μεγάλων δυαδικών αντικειμένων (binary), όπως εικόνες, ήχοι ή βίντεο.

Η βάση δεδομένων σε PostgreSQL χρησιμοποιήθηκε για να αποθηκευτούν και να οργανωθούν τα σημεία ενδιαφέροντος που περιέχουν και πληροφορίες που είναι σε χωρική μορφή (geometry). Το PostGIS έχει έναν δυναμικό γεωμετρικό τύπο δεδομένων, τον geometry. Οι στήλες που περιέχουν γεωμετρικά δεδομένα ορίζονται με αυτό τον τύπο.

3.7.2 PostGIS

Το PostGIS είναι ένα λογισμικό ανοικτού κώδικα που αποτελεί επέκταση της PostgreSQL για χωρικά δεδομένα. Υποστηρίζει δεδομένα διανυσματικής μορφής και δεδομένα σε ψηφιδωτή μορφή, χωρικά ευρετήρια δένδροειδούς μορφής (R-Trees) και μετατροπή δεδομένων μεταξύ συστημάτων γεωγραφικών συντεταγμένων. Το PostGIS τρέχει σε όλα τα βασικά λειτουργικά συστήματα, στα οποία περιλαμβάνονται το Linux, Windows, Mac OS X.

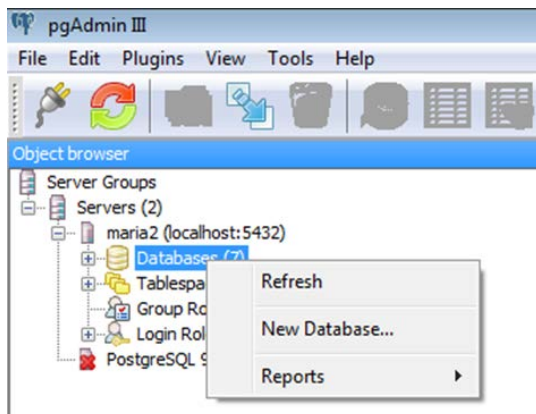
Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε το PostGIS γιατί τα δεδομένα μου είναι χωρικά καθώς εισάγονται σημεία ενδιαφέροντος με γεωγραφικές συντεταγμένες.

3.7.3 PgAdminIII

Η διαχείριση της βάσης δεδομένων πραγματοποιήθηκε από το pgAdmin. Η συγκεκριμένη υπηρεσία εγκαθίσταται ταυτόχρονα με την βάση δεδομένων και υποστηρίζεται σε πολλές πλατφόρμες υπολογιστών. Το εργαλείο pgAdmin εγκαθίσταται αυτόματα μαζί την PostgreSQL και αποτελεί το προκαθορισμένο (default) γραφικό περιβάλλον εκτέλεσης ερωτημάτων.

Έχει σχεδιαστεί για να ικανοποιήσει τις ανάγκες των χρηστών, από απλό γράψιμο ερωτημάτων σε γλώσσα SQL έως την ανάπτυξη πολύπλοκων βάσεων δεδομένων.

Στην Εικόνα 5 φαίνεται το περιβάλλον του pgAdminIII και συγκεκριμένα το υπομενού που εμφανίζεται κάνοντας επιλογή με το πάτημα του δεξιού πλήκτρου του ποντικιού πάνω στο Databases.

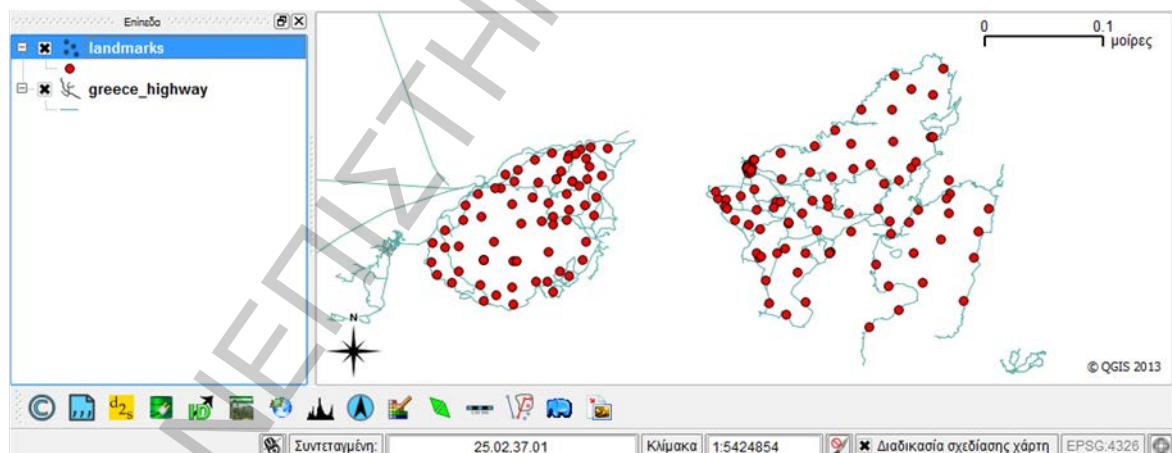


Εικόνα 5. Το περιβάλλον του pgAdminIII

3.7.4 Quantum

Το Quantum GIS (QGIS) είναι ένα ισχυρό και φιλικό προς το χρήστη, μορφής ανοικτού κώδικα (Open Source) Γεωγραφικό Σύστημα Πληροφοριών (GIS), το οποίο τρέχει σε Linux, Unix, Mac OSX, Windows και Android. Το Quantum GIS υποστηρίζει διανυσματικές, ψηφιδωτές μορφές και μορφές βάσεων δεδομένων. Το Quantum GIS είναι υπό την άδεια Άδεια Δημόσιας Χρήσης GNU.

Στην Εικόνα 6 φαίνεται το περιβάλλον του Quantum και συγκεκριμένα ο χάρτης των δύο νησιών με τα σημεία ενδιαφέροντος πάνω σε αυτό.



Εικόνα 6. Το περιβάλλον του Quantum GIS

Στην παρούσα μεταπτυχιακή διατριβή το συγκεκριμένο εργαλείο χρησιμοποιήθηκε για την οπτικοποίηση της Βάσης δεδομένων με τα σημεία ενδιαφέροντος, δηλαδή τα καταστήματα που είχαν εισαχθεί μέσω του PgAdmin. Δηλαδή φορτώνονται τα σημεία που βρίσκονται στη βάση δεδομένων μας πάνω σε ένα χάρτη της Ελλάδος και συγκεκριμένα στις περιοχές της Πάρου και της Νάξου.

3.8 Ανακεφαλαίωση

Στο κεφάλαιο αυτό αναλύθηκαν οι τεχνολογίες, τα εργαλεία και τα πλεονεκτήματά τους που χρησιμοποιούνται στην εφαρμογή της παρούσας μεταπτυχιακής διατριβής και πώς ακριβώς χρησιμοποιούνται. Στο επόμενο κεφάλαιο θα περιγραφεί όλη η διαδικασία της υλοποίησης της εφαρμογής, τα προγράμματα που χρησιμοποιήθηκαν και πώς αυτά βοήθησαν σταδιακά στην εξέλιξη της εφαρμογής. Θα αναπτυχθούν επίσης σχεδιαγράμματα UML και θα γίνει πλήρης περιγραφή της εφαρμογής.

ΚΕΦΑΛΑΙΟ 4

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Σε αυτό το κεφάλαιο θα περιγραφεί όλη η διαδικασία της υλοποίησης της εφαρμογής, τα προγράμματα που χρησιμοποιήθηκαν και πώς αυτά βοήθησαν σταδιακά στην εξέλιξη της εφαρμογής. Θα αναπτυχθούν σχεδιαγράμματα UML και θα γίνει πλήρης περιγραφή της εφαρμογής.

4.1 Υλοποίηση

Για την υλοποίηση της πτυχιακής εργασίας χρησιμοποιήθηκαν το Apache server, CodeIgniter (πλαίσιο PHP), PostgreSQL και PostGIS. Για να μπορέσουν να εγκατασταθούν τα προγράμματα αυτά χρειάστηκε να κάνουμε αρχικά εγκατάσταση του BitNami WAPPStack πηγαίνοντας στην ιστοσελίδα <http://bitnami.org/stack/wappstack>.

Το BitNami WAMP Stack απλοποιεί σε μεγάλο βαθμό την ανάπτυξη και την επέκταση των εφαρμογών PHP. Περιλαμβάνει έτοιμες προς λειτουργία εκδόσεις των Apache, MySQL, PHP και phpMyAdmin και των προαπαιτούμενων τους. Το BitNami WAMP Stack διανέμεται δωρεάν υπό την άδεια Apache 2.0. Τα συστατικά του BitNami Stack WAMP είναι αυτά που αναφέρονται παρακάτω:

- PHP scripting language
- Apache Web server
- MySQL database server
- PostgreSQL database server (optional)
- phpMyAdmin web application management tool for MySQL databases
- phpPgAdmin web application management tool for PostgreSQL databases (optional)
- SQLite database

Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε ο Apache Διακομιστής Δικτύου (Web Server) και το PostgreSQL διακομιστής βάσης δεδομένων (database server) και το phpMyAdmin διακτυακή εφαρμογή διαχείρισης βάσης δεδομένων (web application management tool for MySQL databases).

Εκτός από τη βάση και τις βιβλιοθήκες PHP, η στοίβα περιλαμβάνει δημοφιλή PHP πλαίσια που απλοποιεί το έργο της δημιουργίας εφαρμογών από το μηδέν.

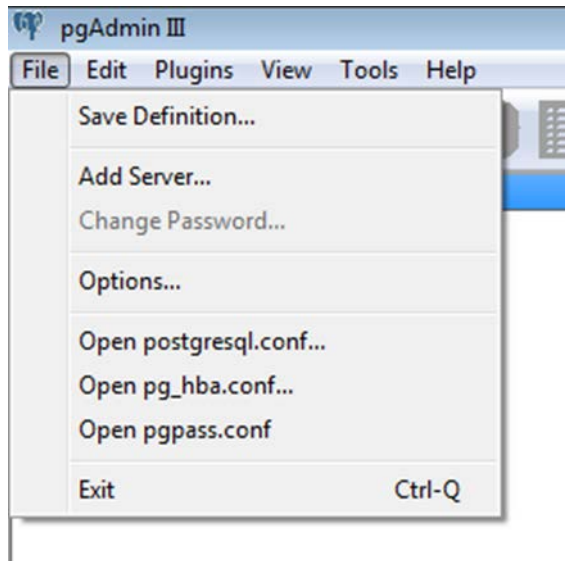
- Zend Framework
- Symfony
- CodeIgniter
- CakePHP
- Smarty
- Laravel

Στην εργασία χρησιμοποιήθηκε το CodeIgniter.

Έπειτα γίνεται εγκατάσταση του PostGIS πηγαίνοντας στην παρακάτω ιστοσελίδα:

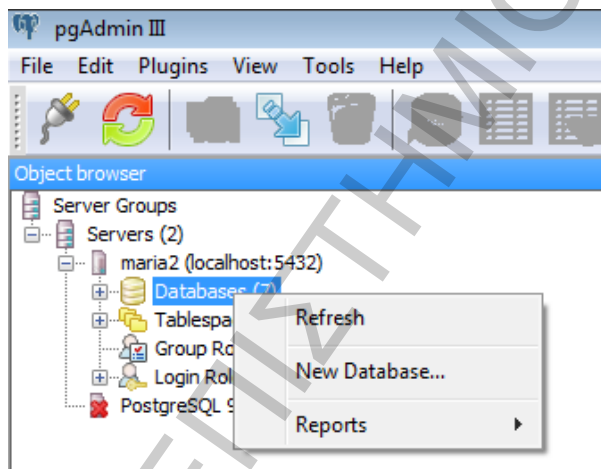
<http://postgis.refractor.net/download/windows/> . Η εκδοχή PostGIS πρέπει να είναι συμβατή με το PostgreSQL που είναι εγκατεστημένο με το WAPPStack για αυτό το λόγο κάνουμε αλλαγή της διεύθυνσης εγκατάστασης έτσι ώστε να είναι συμβατή με τη διεύθυνση της εγκατάστασης του WAPPStack PostgreSQL: C:\Program Files\BitNami WAPPStack\postgresql

Στο phpAdminIII αρχικά προσθέτουμε Εξυπηρετητή πατώντας File-> Add Server, όπως φαίνεται στην Εικόνα 7..



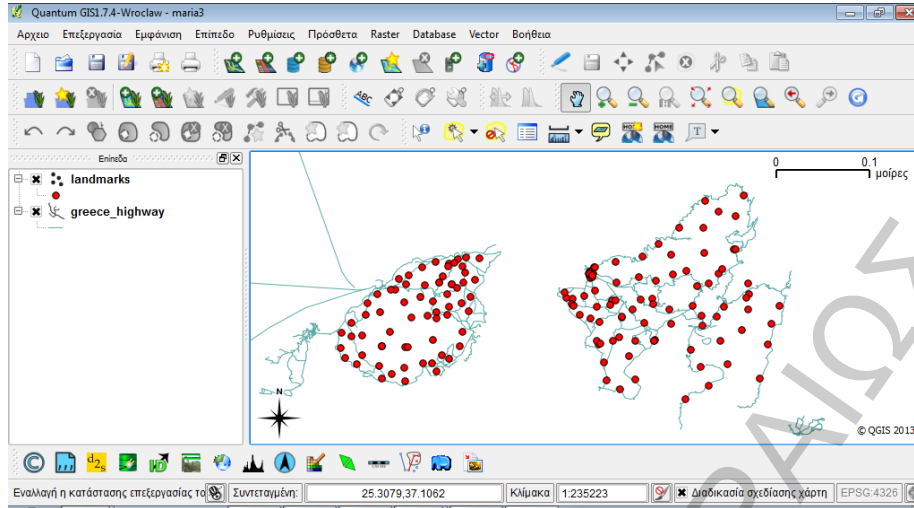
Εικόνα 7. Το μενού επιλογών

Έπειτα αφού φτιάξουμε τον Εξυπηρετητή δημιουργούμε τη βάση δεδομένων μας επιλέγοντας με το δεξί πλήκτρο του ποντικιού το Databases και μετά το New Database, όπως φαίνεται στην Εικόνα 8.



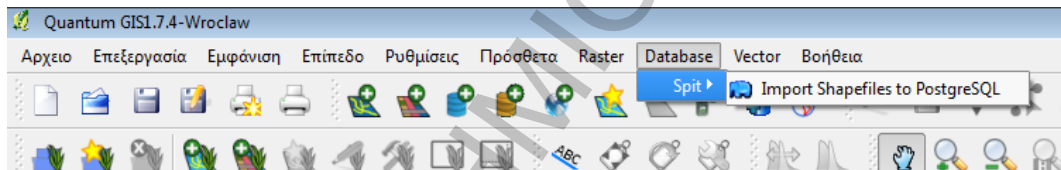
Εικόνα 8. Το υπομενού για τη δημιουργία Βάσης Δεδομένων

Κάνουμε επίσης εγκατάσταση του προγράμματος Quantum για την οπτικοποίηση των χωρικών δεδομένων πηγαίνοντας στην ιστοσελίδα <http://www.qgis.org/>. Το περιβάλλον του προγράμματος φαίνεται στην Εικόνα 9.



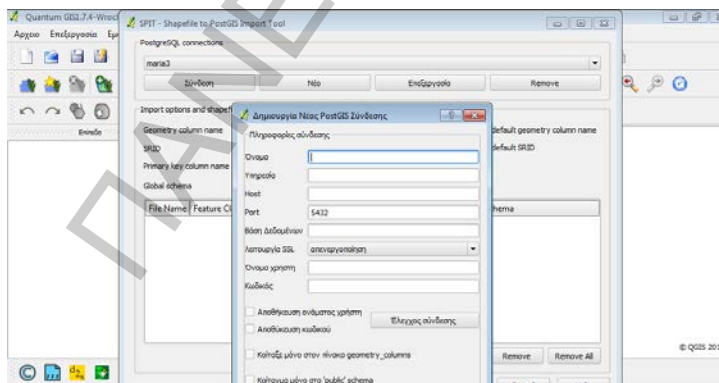
Εικόνα 9. Το περιβάλλον του Quantum QGIS

Πατάμε αρχείο -> Νέο Project. Έπειτα επιλέγουμε Database->Split-> Import Shapefiles to PostgreSQL για να εισάγουμε το χάρτη της Ελλάδος που θα χρησιμοποιήσουμε και για να εισάγουμε τα σημεία ενδιαφέροντος στα νησιά Πάρος και Νάξος, όπως φαίνεται στην Εικόνα 10.



Εικόνα 10. Το μενού Database

Θα πρέπει να κάνουμε σύνδεση με τη βάση δεδομένων που έχουμε αρχικά φτιάξει με το rhrAdminIII, για αυτό πατάμε Νέο και εμφανίζεται το παράθυρο όπως φαίνεται στην παραπάνω εικόνα. Σαν Host βάζουμε Localhost και Βάση δεδομένων τη βάση δεδομένων που έχουμε φτιάξει στο rhrAdminIII. Στο όνομα χρήστη βάζουμε αυτό της postgres και κωδικό αυτό που είχαμε βάλει, όπως φαίνεται στην Εικόνα 11.



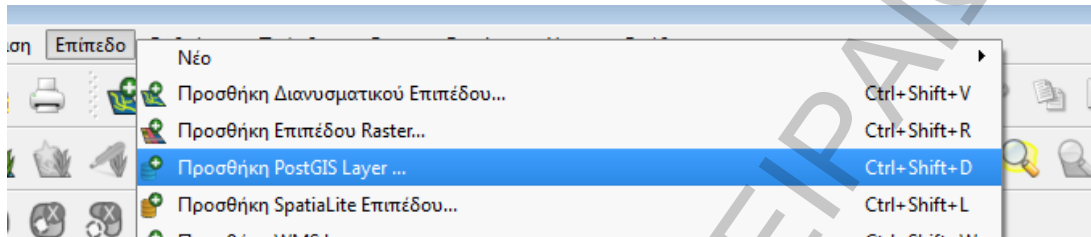
Εικόνα 11. Το παράθυρο της Νέας PostGIS Σύνδεσης

Κάνουμε σύνδεση και μετά βάζουμε στα Συστήματα χωρικής αναφοράς (Spatial Reference Systems SRID) το 4326 που έχει μονάδα μέτρησης τις μοίρες (Decimal Degrees) και ισχύουν τα παρακάτω:

-180 <= γεωγραφικό μήκος(longitude) <= 180

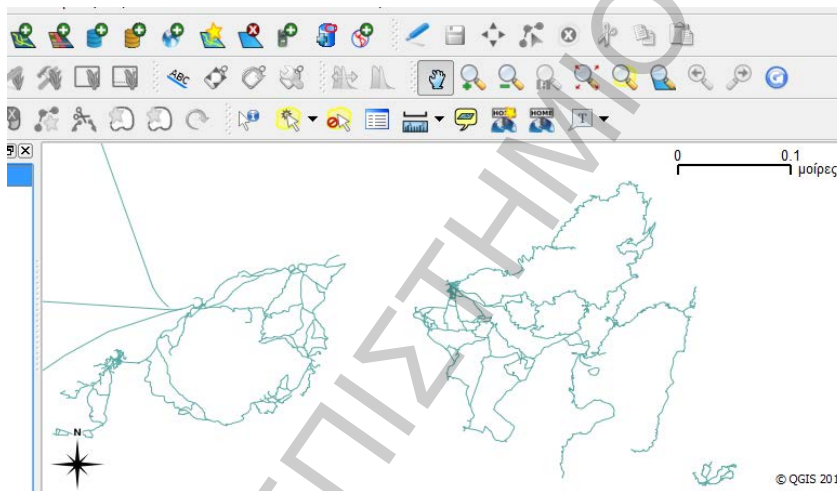
-90 <= γεωγραφικό πλάτος (latitude)<= 90

Πατάμε προσθήκη για να εισάγουμε το Χάρτη της Ελλάδος το οποίο το κατεβάσαμε από το παρακάτω υπερσύνδεσμο http://downloads.cloudmade.com/europe/southern_europe/greece/attiki#downloads_breadcr. Ύστερα επιλέγουμε Επίπεδο->Προσθήκη PostGISLayer και κάνουμε σύνδεση στη βάση δεδομένων για να δούμε την οπτικοποίηση, όπως φαίνεται στην Εικόνα 12.



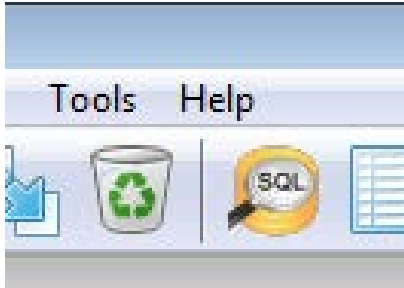
Εικόνα 12. Το μενού Επίπεδο

Έτσι λοιπόν έχουμε την Εικόνα 13.



Εικόνα 13. Ο χάρτης των νησιών Πάρου και Νάξου

Στο phpAdminIII φτιάχνουμε τον πίνακα Landmarks όπου θα κάνουμε εισαγωγή των σημείων ενδιαφέροντος. Στο περιβάλλον του Script που πηγαίνουμε πατώντας SQL φτιάχνουμε τον παρακάτω πίνακα με πρωτεύον κλειδί το landmark_id. Στην Εικόνα 14 φαίνεται το πλήκτρο SQL του προγράμματος.



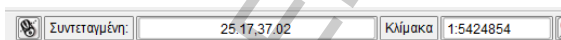
Εικόνα 14. Το πλήκτρο SQL

```
CREATE TABLE landmarks
(
  landmark_id integer NOT NULL,
  landmark_name character varying(50),
  landmark_kind character varying,
  island character varying,
  landmark_geometry geometry,
  CONSTRAINT landmarks_pkey PRIMARY KEY (landmark_id)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE landmarks
  OWNER TO postgres;
```

Επίσης φτιάχνουμε τον πίνακα να έχει ευρετήριο σύμφωνα με το πεδίο geometry για να γίνεται γρήγορα η αναζήτηση του ερωτήματος που θα φτιάξουμε .

```
CREATE INDEX landmarks_spatial_index
ON landmarks USING GIST(landmark_geometry);
```

Εισάγουμε τα σημεία ενδιαφέροντος όπως φαίνεται στην παρακάτω εικόνα καθώς μπορούμε να δούμε τις συντεταγμένες των σημείων από το χάρτη στο Quantum, όπως φαίνεται στην Εικόνα 15.



Εικόνα 15. Οι συντεταγμένες του σημείου του ποντικιού

Στην Εικόνα 16 φαίνεται το περιβάλλον της SQL όπου γίνονται η εισαγωγή των σημείων ενδιαφέροντος.

```
INSERT INTO landmarks
VALUES(13,'ΖΩΣΗ','ΦΑΡΜΑΚΕΙΟ','ΠΑΡΟΣ',ST_GeomFromText('POINT(25.155 37.025)', 4326));

INSERT INTO landmarks
VALUES(14,'ΠΛΑΚΑ','ΦΑΡΜΑΚΕΙΟ','ΠΑΡΟΣ',ST_GeomFromText('POINT(25.1872 37.0554)', 4326));

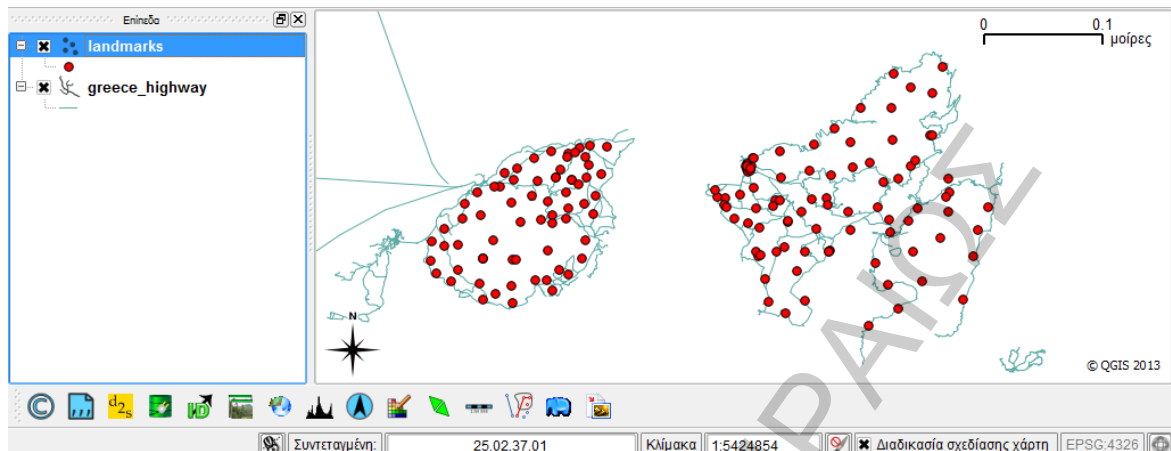
INSERT INTO landmarks
VALUES(15,'ΖΗΚΟΣ','ΦΑΡΜΑΚΕΙΟ','ΠΑΡΟΣ',ST_GeomFromText('POINT(25.2100 37.0315)', 4326));

INSERT INTO landmarks
VALUES(17,'ΛΟΥΚΟΥ','ΓΥΜΝΑΣΤΗΡΙΟ','ΠΑΡΟΣ',ST_GeomFromText('POINT(25.1341 37.0370)', 4326));

INSERT INTO landmarks
VALUES(18,'ΚΑΒΑΣΤΟΥ','ΓΥΜΝΑΣΤΗΡΙΟ','ΠΑΡΟΣ',ST_GeomFromText('POINT(25.1528 37.0046)', 4326));
```

Εικόνα 16. Εισαγωγή σημείων ενδιαφέροντος

Στην Εικόνα 17 φαίνεται το περιβάλλον του Quantum και τα σημάδια που έχουν εισαχθεί στο χάρτη.



Εικόνα 17. Τα σημεία ενδιαφέροντος στο χάρτη

Έπειτα φτιάχνουμε το codeigniter καθώς έχουμε κατεβάσει το φάκελο Codeigniter και το αποθηκεύουμε στην παρακάτω τοποθεσία με την ονομασία ergasia7:

C:\Program Files\BitNami WAPPStack\apache2\htdocs

Αρχικά στο αρχείο config που βρίσκεται στην παρακάτω τοποθεσία αλλάζουμε το εξής:

```
$config['base_url'] = 'http://localhost/ergasia7';
```

Στο αρχείο database που βρίσκεται και αυτό στην ίδια τοποθεσία αλλάζουμε τα παρακάτω:

```
$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'postgres';
$db['default']['password'] = 'maria';
$db['default']['database'] = 'maria3';
$db['default']['dbdriver'] = 'postgre';
$db['default']['dbprefix'] = "";
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = "";
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_general_ci';
$db['default']['swap_pre'] = "";
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;
```

Φτιάχνουμε το αρχείο map.php (ο κώδικας του οποίου βρίσκεται στο παράρτημα) και βρίσκεται στην τοποθεσία

C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\views\pages και φορτώνεται από το φυλλομετρητή του διαδικτύου αν βάλουμε τη διεύθυνση URL:

<http://localhost/ergasia7/index.php/pages/view/map>

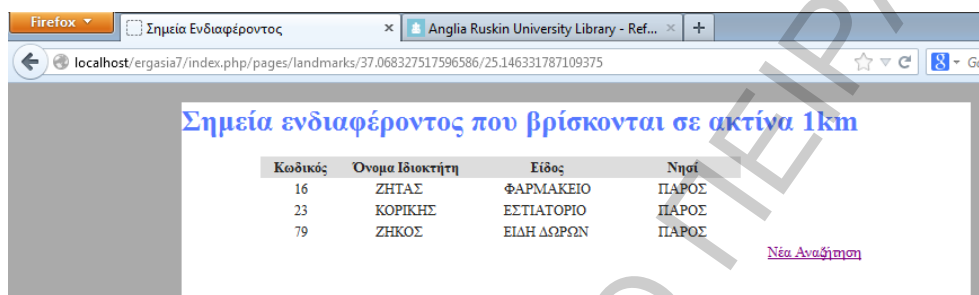
το οποίο περιέχει το χάρτη από το Google map ο οποίος είναι εστιασμένος κατάλληλα για να

φαίνονται τα δύο νησιά Πάρος και Νάξος, τις εντολές για να μπορεί ο χρήστης να δημιουργήσει σημάδι (marker) και να είναι δυνατόν να χρησιμοποιηθούν οι συντεταγμένες του και με ανακατεύθυνση να σταλεί στο controller pages στην μέθοδο public function landmarks(\$lat, \$lng) όπου οι μεταβλητές \$lat είναι το γεωγραφικό πλάτος και \$lng είναι το γεωγραφικό μήκος.

Το αρχείο pages.php που βρίσκεται στην παρακάτω τοποθεσία

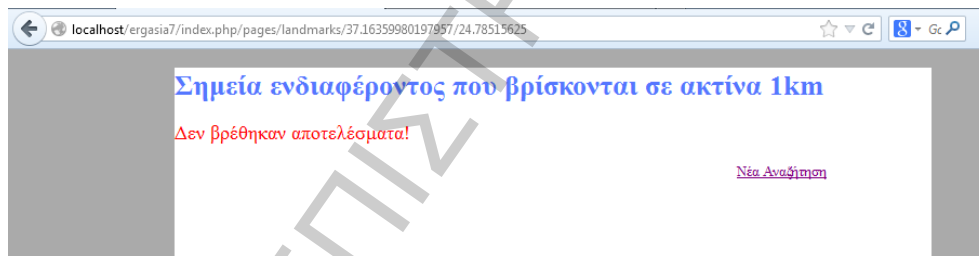
C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\controllers

(ο κώδικας του εμφανίζεται στο παράρτημα), είναι υπεύθυνο για τη φόρτωση του αρχείου map στο φυλλομετρητή με την μέθοδο view, τη σύνδεση με το model landmark_model, να πάρει τις συντεταγμένες του σημαδιού (marker) με την μέθοδο landmarks(\$lat, \$lng) και να τις στείλει στο Landmark_model στην μέθοδο get_landmarks(\$lat,\$lng) και να εμφανίσει τα αποτελέσματα, δηλαδή τα σημεία ενδιαφέροντος που βρίσκονται σε ακτίνα 1km από το σημάδι, αν υπάρχουν φορτώνοντας το αρχείο landmarks.php που είναι μέσα στο views και αν δεν υπάρχουν να εμφανίσει μήνυμα ότι δεν υπάρχουν αποτελέσματα φορτώνοντας το αρχείο noResults.php. Στην Εικόνα 18 φαίνονται τα αποτελέσματα όπως εμφανίζονται στο χρήστη.



Εικόνα 18. Εμφάνιση αποτελεσμάτων

Στην Εικόνα 19 φαίνεται το μήνυμα ότι δεν βρέθηκαν αποτελέσματα όπως εμφανίζεται στο χρήστη.



Εικόνα 19. Εμφάνιση μηνύματος ότι δεν υπάρχουν αποτελέσματα

Το αρχείο landmark_model που βρίσκεται στην παρακάτω τοποθεσία :

C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\models

Φορτώνει τη βάση δεδομένων με την μέθοδο public function __construct(), λαμβάνει τις συντεταγμένες του σημαδιού (marker) από το controller με την μέθοδο get_landmarks(\$lat, \$lng), εκτελεί το ερώτημα που θα μας δώσει τα σημεία ενδιαφέροντος τα βάσης δεδομένων που βρίσκονται σε ακτίνα 1 km και γυρνάει τα αποτελέσματα στο αρχείο του controller.

4.2 Διαγράμματα UML

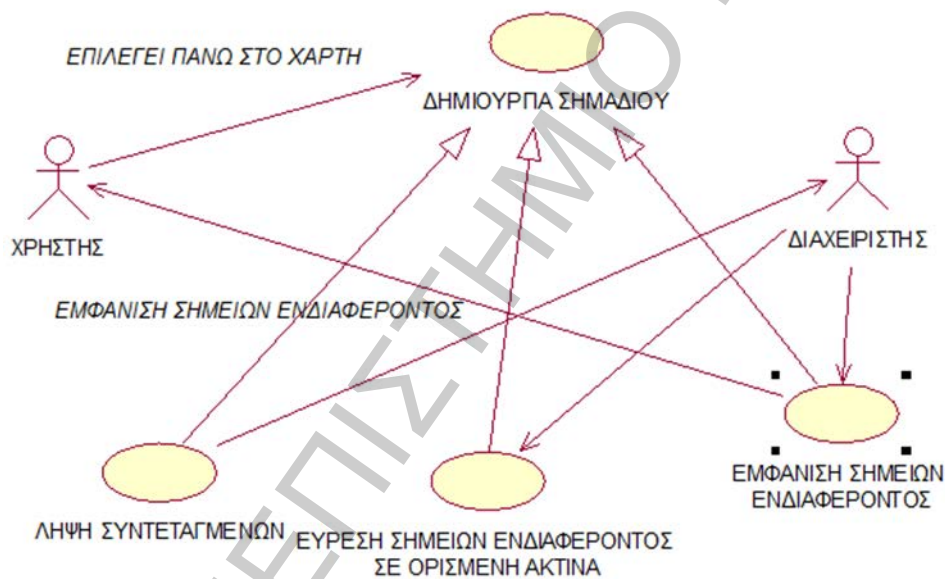
4.2.1 Διάγραμμα περίπτωσης (Use case diagram)

Στα διαγράμματα που ακολουθούν μπορούμε να δούμε αναλυτικά τις απαιτήσεις του συστήματος από την οπτική γωνία του χρήστη του συστήματος.

Το μοντέλο Use Case περιλαμβάνει:

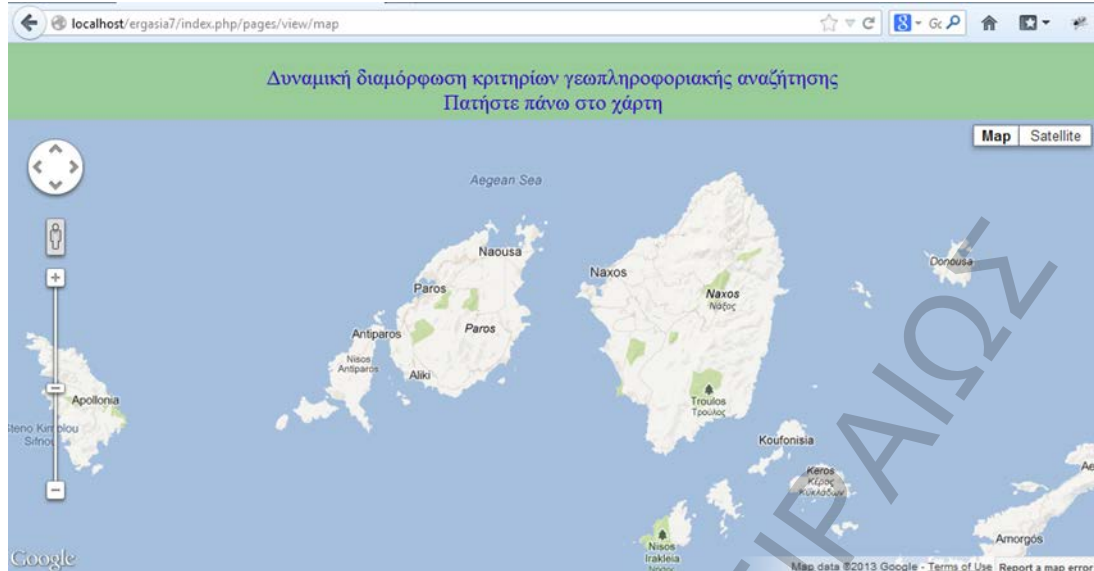
1. Τις περιπτώσεις χρήσης (Use cases)
2. Τους ενεργοποιούς (actors)

Ένας ενεργοποιός αναπαριστά το ρόλο που παίζεται από ένα άτομο ή πράγμα που αλληλεπιδρά με το σύστημα. Οι ενεργοποιοί προσδιορίζονται εύκολα αν παρατηρήσουμε τους άμεσους χρήστες του συστήματος καθώς και κάθε άλλο σύστημα που αλληλεπιδρά με αυτό που αναπτύσσεται. Επιπλέον οι περιπτώσεις χρήσης ορίζονται παρατηρώντας τις σειρές αλληλεπίδρασης για κάθε ενεργοποιό. Στην Εικόνα 20 βρίσκεται το διάγραμμα περίπτωσης όπου φαίνεται η συμβολή του χρήστη (actor) και του διαχειριστή (actor).



Εικόνα 20. Διάγραμμα περίπτωσης

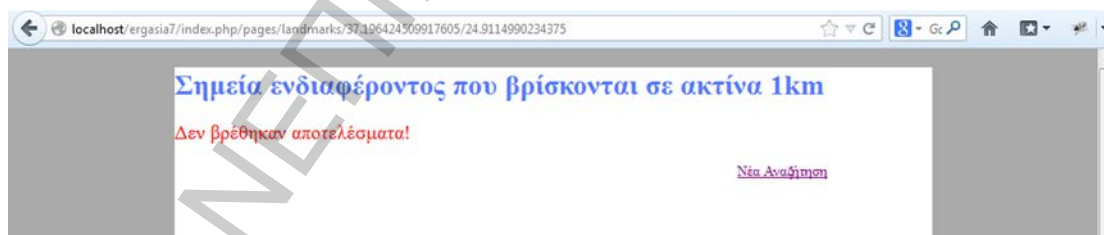
Ο χρήστης για να μπει στο χάρτη βάζει τη διεύθυνση URL: localhost/ergasia/index.php/pages/view/map και εμφανίζεται ο αρχική σελίδα με το χάρτη όπως φαίνεται και στην Εικόνα 21.



Εικόνα 21. Η αρχική σελίδα με τον χάρτη των περιοχών της Πάρου και Νάξου

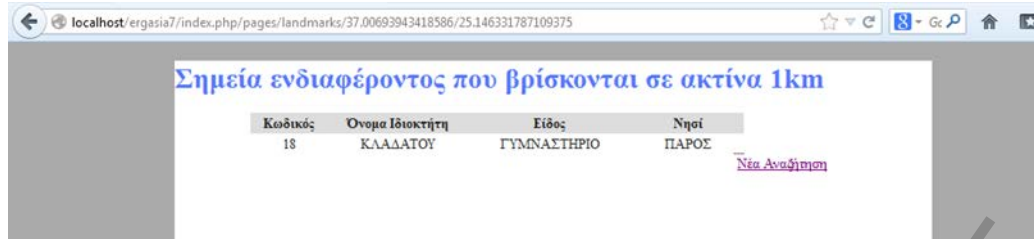
καθώς το αρχείο pages είναι το αρχείο του controllers και το αρχείο map είναι το αρχείο που φορτώνει το χάρτη και βρίσκεται μέσα στο views στο φάκελο pages και αυτή η φόρτωση γίνεται μέσα από την μέθοδο view του αρχείου pages.

Ο χρήστης κάνει κλικ πάνω στο χάρτη του Google map και δημιουργείται ένα σημείο (marker) πάνω στο χάρτη. Στο αρχείο map εντοπίζονται οι συντεταγμένες του σημείου και στέλνονται στο αρχείο pages του controllers στην μέθοδο landmarks. Έπειτα αυτές στέλνονται στο αρχείο landmarks_model του models στην μέθοδο get_landmarks και εκτελείται το ερώτημα για να βρεθούν τα σημεία ενδιαφέροντος που είναι καταχωρημένα στη βάση δεδομένων και τα οποία βρίσκονται σε ορισμένη ακτίνα από το σημείο που κάναμε κλικ. Έπειτα επιστρέφονται τα αποτελέσματα με την εντολή return \$query ->result_array() στο αρχείο pages του controllers στην μέθοδο landmarks και αν δεν υπάρχουν φορτώνεται το αρχείο noResults όπου βγάζει μήνυμα ότι δεν υπάρχουν αποτελέσματα. Στην Εικόνα 22 φαίνεται η σελίδα που εμφανίζεται όταν δεν υπάρχουν αποτελέσματα.



Εικόνα 21. Η σελίδα που εμφανίζεται όταν δεν υπάρχουν αποτελέσματα

Πατώντας στο Νέα Αναζήτηση πηγαίνουμε πίσω στο χάρτη για νέα αναζήτηση. Αν υπάρχουν αποτελέσματα τότε φορτώνεται το αρχείο landmarks το οποίο είναι μέσα στο views και εμφανίζεται η Εικόνα 23.



Εικόνα 23. Η σελίδα που εμφανίζεται όταν υπάρχουν αποτελέσματα

Πατώντας στο Νέα Αναζήτηση πηγαίνει πίσω στο χάρτη για νέα αναζήτηση.

4.2.2 Διάγραμμα Τάξεων(Class Diagrams)

Η Τάξη αντικειμένων είναι μια αφάιρηση που περιγράφει ένα σύνολο αντικειμένων με παρόμοιες ιδιότητες, κοινή συμπεριφορά, κοινές σχέσεις με άλλες τάξεις και κοινή σημασιολογία.

Οι τάξεις ή κλάσεις (classes) αποτελούν τη βάση της κατασκευής οποιουδήποτε αντικειμενοστρεφούς συστήματος. Ενσωματώνουν τα δεδομένα και τις λειτουργίες που επενεργούν στα δεδομένα αυτά. Από τις τάξεις δημιουργούνται αντικείμενα (objects) ή περιστάσεις (instances) αυτών των τάξεων, όταν λειτουργεί το σύστημα. Σκοπός του διαγράμματος τάξεων (class diagram) είναι η απεικόνιση της στατικής δομής του συστήματος. Αυτή η δομή συμπεριλαμβάνει:

Τις τάξεις (classes) που περιέχουν:

- Ιδιότητες (attributes) που είναι συνήθως τα ιδιωτικά δεδομένα μιας τάξης.
- Λειτουργίες (operations) που είναι συνήθως οι δημόσιες λειτουργίες μιας τάξης που καλούν τα άλλα αντικείμενα.

Στην Εικόνα 24 φαίνεται το διάγραμμα κλάσεων με τις μεθόδους που χρησιμοποιούνται στην εφαρμογή και τη σύνδεση αυτών.



Εικόνα 24. Διάγραμμα τάξεων

Η κλάση pages η οποία είναι ελεγκτής (controller) έχει τις εξής μεθόδους:

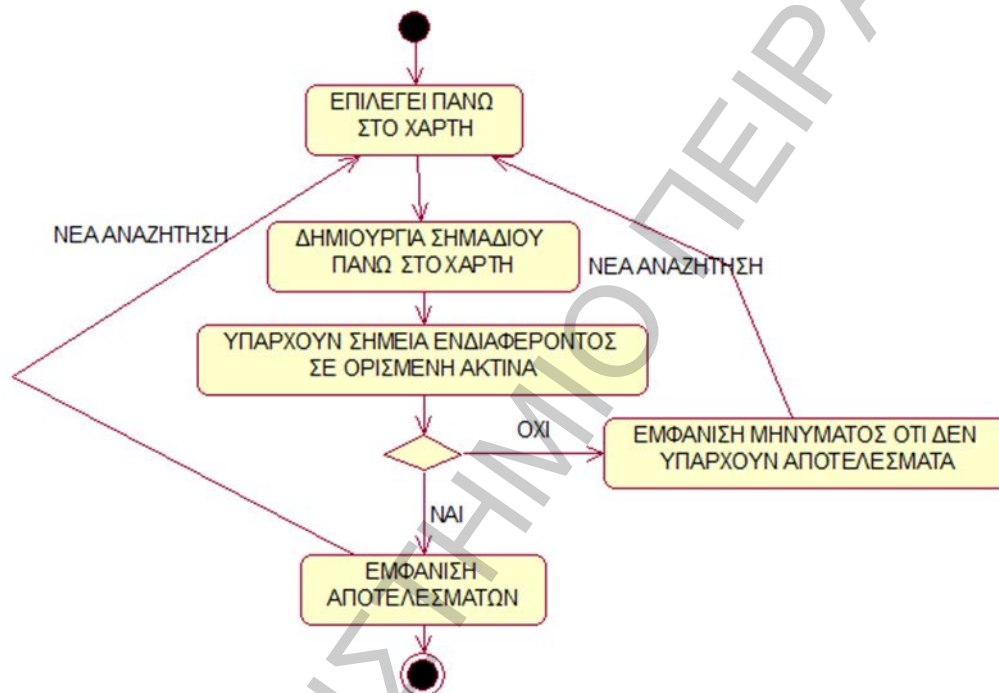
- View (\$page='map') η οποία παίρνει την μεταβλητή \$page='map' για να εμφανίζεται το αρχείο map στο browser του ιντερνέτ.
- Construct() για να φορτώνει το model landmarks_model.
- Landmarks που παίρνει τις μεταβλητές (\$lat, \$lng) από το αρχείο map, δηλαδή τις συντεταγμένες από το σημείο του κλικ που κάνουμε πάνω στο χάρτη και τις στέλνει στο αρχείο landmarks_model στην μέθοδο get_landmarks (\$lat, \$lng). Τα δεδομένα επιστρέφουν από το αρχείο landmark_model και από την μέθοδο get_landmarks.

Η κλάση landmarks_model η οποία είναι μοντέλο (model) έχει τις εξής μεθόδους:

- Construct ()για να φορτώνει τη βάση δεδομένων
- Get_landmarks(\$lat,\$lng) για να εκτελεί το ερώτημα για να βρεθούν ποια σημεία ενδιαφέροντος είναι σε απόσταση 1 km από το σημείο που έγινε κλικ πάνω στο χάρτη και επιστρέφει τα αποτελέσματα πίσω στο αρχείο pages και συγκεκριμένα στην μέθοδο landmarks (\$lat, \$lng).

4.2.3 Διάγραμμα δραστηριοτήτων (Activity diagram)

Με το διάγραμμα δραστηριοτήτων πραγματοποιείται καταγραφή των ενεργειών που εφαρμόζονται κατά την εκτέλεση μιας λειτουργίας (στιγμιότυπο της υλοποίησης της λειτουργίας), περιγράφεται η εσωτερική λειτουργία ενός αντικειμένου, περιγράφονται συσχετιζόμενες ενέργειες που εφαρμόζονται και πώς αυτές επηρεάζουν τα γύρω αντικείμενά τους. Με αυτό το διάγραμμα που φαίνεται στην Εικόνα 25 περιγράφεται η δραστηριότητα για να μπει στο σύστημα ο χρήστης.

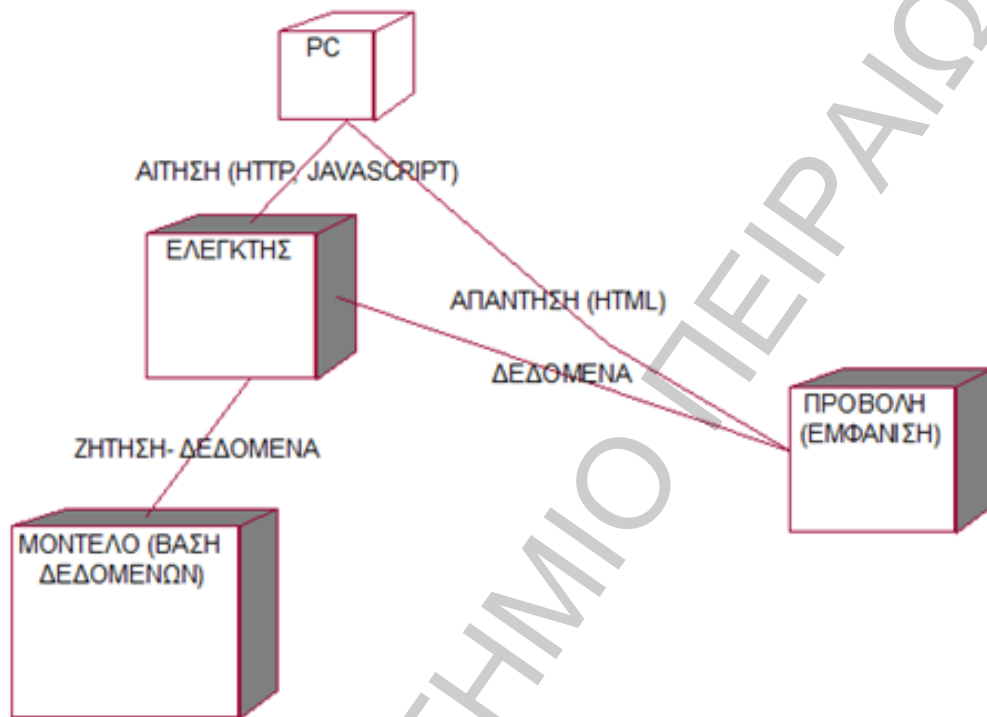


Εικόνα 25. Διάγραμμα δραστηριοτήτων του χρήστη

- Το διάγραμμα ξεκινάει με την τελεία που υποδηλώνει έναρξη της δραστηριότητας. Ο χρήστης κάνει αρχικά κλικ πάνω στο χάρτη του Google map.
- Δημιουργείται ένα σημάδι (marker) πάνω στο χάρτη.
- Καθώς λαμβάνονται οι συντεταγμένες του σημείου (marker) πάνω στο χάρτη εκτελείται το ερώτημα αν υπάρχουν σημεία ενδιαφέροντος σε ορισμένη ακτίνα από το σημείο (marker).
- Επειδή γίνεται ερώτημα υπάρχουν σημεία ενδιαφέροντος σε ορισμένη ακτίνα από το σημείο (marker) συμβολίζεται με ρόμβο γιατί είναι δύο οι απαντήσεις.
- Αν η απάντηση είναι όχι στο ερώτημα μπορεί ο χρήστης να ξαναγουρίσει πίσω στο χάρτη για να κάνει νέα αναζήτηση.
- Αν η απάντηση είναι ναι στο ερώτημα τότε εμφανίζονται τα αποτελέσματα και μπορεί ο χρήστης να ξαναγουρίσει πίσω στην αρχική σελίδα για να κάνει νέα αναζήτηση.

4.2.4 Διάγραμμα Διανομής (Deployment Diagram)

Τα διαγράμματα διανομής δείχνουν τη φυσική διάταξη των διαφόρων εξαρτημάτων (κόμβων) υλικού, που αποτελούν ένα σύστημα, όπως και τη διανομή των εκτελέσιμων προγραμμάτων σε αυτό το υλικό. Κάθε πόρος υλικού αναπαρίσταται με ένα μικρό κόμβο. Στην Εικόνα 26 φαίνεται το Διάγραμμα Διανομής και ο καταμερισμός των εκτελέσιμων ενεργειών από τα επιμέρους μέρη του Codeigniter.



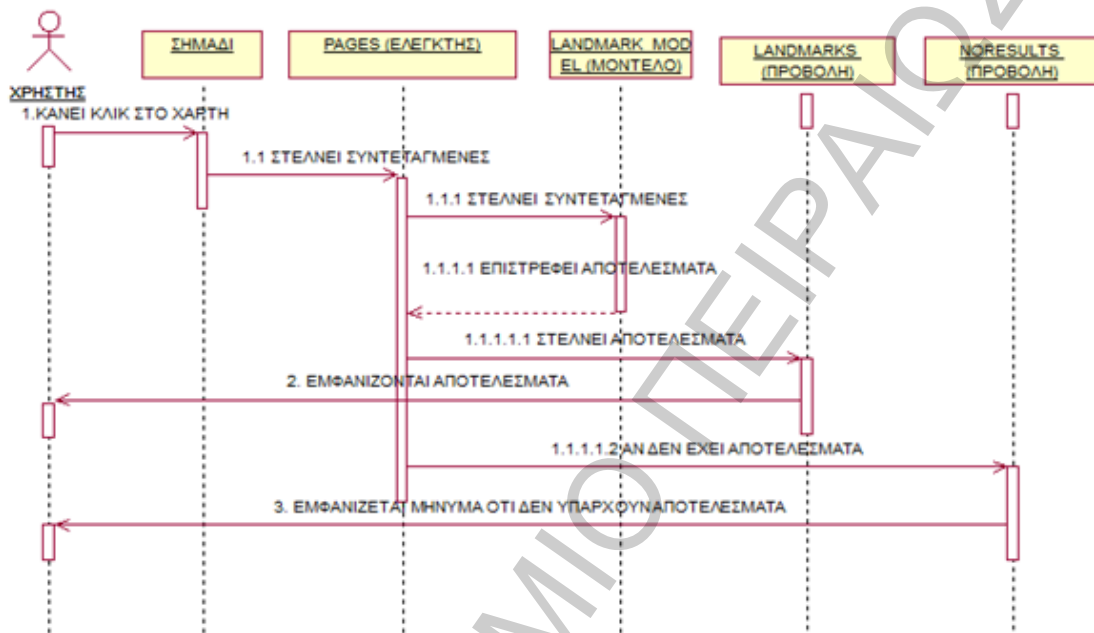
Εικόνα 26. Διάγραμμα διανομής

Ο χρήστης χειρίζεται το pc στο οποίο αρχικά βλέπει ένα αρχείο με το όνομα map τύπου .php του views του Codeigniter στο οποίο είναι ένας χάρτης από το Google map που απεικονίζει τα δύο νησιά Πάρος και Νάξος και είναι φτιαγμένο με Javascript, Html, css . Όταν πατήσει κλικ πάνω στο χάρτη, οι συντεταγμένες του σημείου (marker) με ανακατεύθυνση (redirect), λαμβάνονται από μία μέθοδο landmarks του αρχείου pages.php του controllers και αυτή η μέθοδος στέλνει τις συντεταγμένες σε ένα αρχείο του models landmarks_model.php σε μία μέθοδο get_landmarks. Σε αυτό το αρχείο του models αρχικά είχε φορτωθεί η βάση δεδομένων που περιέχει τα σημεία ενδιαφέροντος με την μέθοδο _construct. Στην μέθοδο get_landmarks εκτελείται το ερώτημα που θα αναζητήσει τα σημεία ενδιαφέροντος που είναι αποθηκευμένα στη βάση δεδομένων και βρίσκονται σε ορισμένη ακτίνα. Τα δεδομένα της αναζήτησης επιστρέφονται στο αρχείο pages.php στην μέθοδο landmarks και από εκεί στέλνονται στο αρχείο landmarks.php του views όπου εμφανίζονται τα αποτελέσματα στο pc του χρήστη. Αν δεν υπάρχουν αποτελέσματα από το ερώτημα τότε εκτελείται το αρχείο noResults.php με το οποίο εμφανίζεται ένα μήνυμα ότι δεν βρέθηκαν αποτελέσματα.

4.2.5 Διάγραμμα ακολουθίας- σειράς (Διάγραμμα αλληλεπίδρασης)

Στα συστήματα αντικειμενοστρεφούς προγραμματισμού (OOP) τα αντικείμενα δεν είναι στατικά, αλλά αλληλεπιδρούν ανταλλάσσοντας μηνύματα. Η απεικόνιση αυτής της αλληλεπίδρασης

γίνεται με τα διαγράμματα αλληλεπίδρασης. Σαν αλληλεπίδραση ορίζεται ο τρόπος με τον οποίο ανταλλάσσονται τα μηνύματα μεταξύ των αντικειμένων, ώστε να επιτευχθεί η εκτέλεση μιας εργασίας. Τα διαγράμματα σειράς από την άλλη αναπαριστούν αλληλεπιδράσεις ανάμεσα στα αντικείμενα από μια χρονική άποψη. Ο χρόνος παριστάνεται από κάτω προς τα πάνω στο διάγραμμα, που σημαίνει ότι τα μηνύματα που βρίσκονται πιο πάνω στο διάγραμμα προηγούνται χρονικά αυτών που βρίσκονται πιο χαμηλά. Στην Εικόνα 27 φαίνεται το Διάγραμμα σειράς όπου φαίνεται η ακολουθία των ενεργειών που εκτελεί το σύστημα.

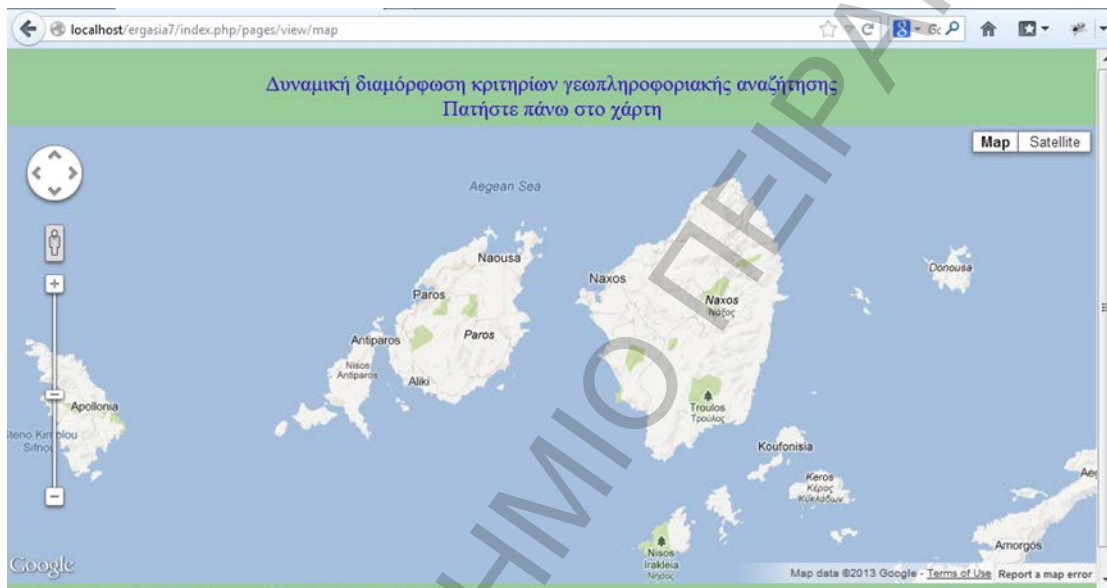


Εικόνα 27. Διάγραμμα σειράς

Ο χρήστης αρχικά κάνει κλικ πάνω στο χάρτη και δημιουργείται σημάδι (marker) καθώς έχει ήδη φορτωθεί η σελίδα: localhost/ergasia7/index.php/pages/view/map. Στέλνονται οι συντεταγμένες στο controller pages σε μία μέθοδο και αυτός αντίστοιχα τις στέλνει στο landmark_model για να εκτελεστεί εκεί το ερώτημα για να βρεθούν τα σημεία ενδιαφέροντος που είναι σε ορισμένη ακτίνα από το σημάδι. Έπειτα τα αποτελέσματα θα επιστραφούν στο controller pages για να γίνει η ερώτηση αν υπάρχουν αποτελέσματα. Αν η συνθήκη αυτή είναι να δηλαδή υπάρχουν αποτελέσματα εμφανίζονται στο χρήστη με το αρχείο landmarks που είναι μέσα στο view καθώς στέλνονται τα δεδομένα στο αρχείο αυτό. Αν δεν υπάρχουν αποτελέσματα εμφανίζεται στο χρήστη μήνυμα ότι δεν υπάρχουν αποτελέσματα καθώς φορτώνεται το αρχείο noresults που βρίσκεται μέσα στο view.

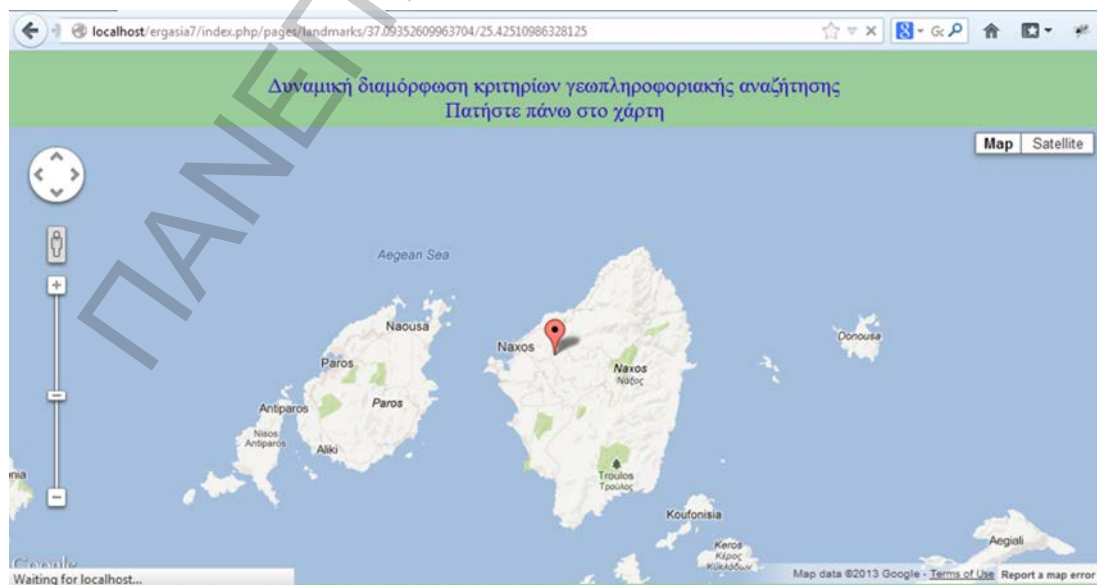
4.3 Πλήρης περιγραφή της εφαρμογής

Αρχικά ο χρήστης εκτελεί το πρόγραμμα του φυλλομετρητή του Διαδικτύου, όπως τους Internet Explorer, Opera, Mozilla Firefox κ.τ.λ. και για να μπει στο χάρτη βάζει τη διεύθυνση URL:localhost/ergasia7/index.php/pages/view/map . Στην Εικόνα 28 φαίνεται η αρχική σελίδα με το χάρτη.



Εικόνα 28. Εμφάνιση του χάρτη των περιοχών της Πάρου και Νάξου

Ο χρήστης επιλέγει πάνω στο χάρτη και δημιουργεί σημάδι (marker) και το σημάδι φαίνεται στην Εικόνα 29 με την μορφή του κόκκινου σημαδιού.

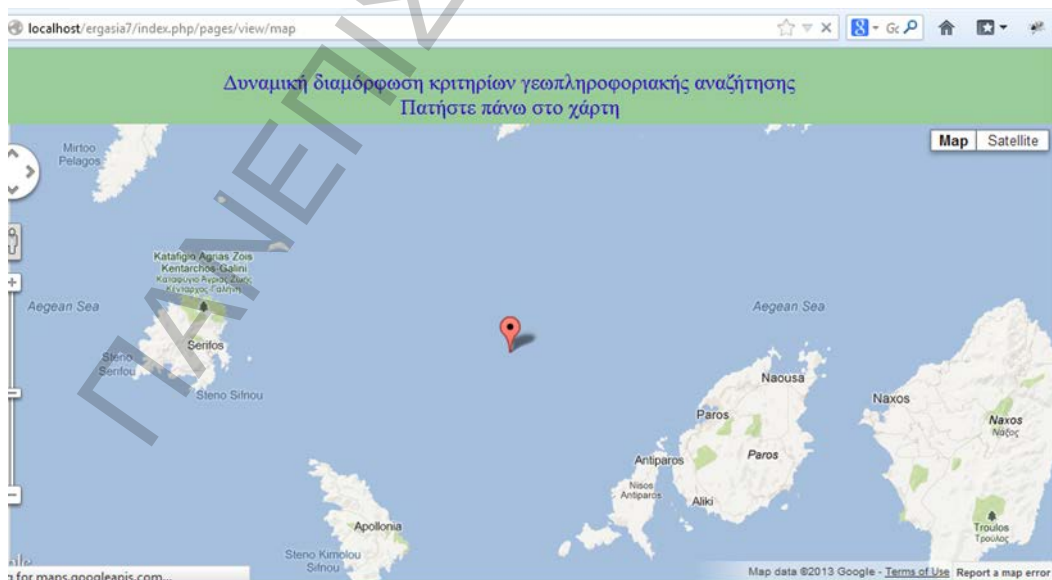


Εικόνα 29 . Ο χάρτης με το σημάδι που έκανε ο χρήστης μέσα στο νησί Νάξος

Μετά αν υπάρχουν σημεία ενδιαφέροντος, καταστήματα σε απόσταση 1 km από το σημείο που έγινε το κλικ εμφανίζεται η Εικόνα 30.

**Εικόνα 30. Τα σημεία ενδιαφέροντος και οι σχετικές πληροφορίες αυτών**

Αν πατήσει ο χρήστης σε περιοχή που δεν υπάρχουν καταστήματα σε ακτίνα 1 km, τότε εμφανίζεται το μήνυμα ότι δεν υπάρχουν αποτελέσματα, όπως φαίνονται στις Εικόνες 31 και 32.

**Εικόνα 31. Το σημάδι εκτός της περιοχής των νησιών**

Δυναμική διαμόρφωση κριτηρίων γεωπληροφοριακής αναζήτησης



Εικόνα 32. Το μήνυμα που εμφανίζεται ότι δεν υπάρχουν σημεία ενδιαφέροντος

4.4 Ανακεφαλαίωση

Σε αυτό το κεφάλαιο περιγράφηκε όλη η διαδικασία της υλοποίησης της εφαρμογής, τα προγράμματα που χρησιμοποιήθηκαν και πώς αυτά βοήθησαν σταδιακά στην εξέλιξη της εφαρμογής, αναπτύχθηκαν τα σχεδιαγράμματα UML και πραγματοποιήθηκε μια πλήρης περιγραφή της εφαρμογής. Στο επόμενο κεφάλαιο θα αναλυθούν τα συμπεράσματα κατά την πραγματοποίηση της παρούσας μεταπτυχιακής εργασίας και μελλοντικές βελτιώσεις αυτής.

ΚΕΦΑΛΑΙΟ 5

ΣΥΜΠΕΡΑΣΜΑΤΑ

Σε αυτό το κεφάλαιο θα παρουσιαστούν και θα αναλυθούν τα συμπεράσματα από τη δημιουργία και τη χρήση της εφαρμογής και θα αναφερθούν μελλοντικές βελτιώσεις της εφαρμογής μας.

5.1 Συμπεράσματα για τις τεχνολογίες που χρησιμοποιήθηκαν

5.1.1 Συμπεράσματα από τη χρήση των γεωγραφικών συστημάτων (GIS)

Η διαχείριση και ανάλυση των γεωγραφικών δεδομένων είναι ένα ερευνητικό πεδίο μείζονος σημασίας καθώς το 80% των πολιτικών και οικονομικών αποφάσεων εμπλέκουν άμεσα ή έμμεσα γεωγραφικές πληροφορίες.

Ένα γεωγραφικό σύστημα είναι υπολογιστικό σύστημα σχεδιασμένο για να υπολογίζει τη συλλογή, διαχείριση, επεξεργασία, ανάλυση, μοντελοποίηση και απεικόνιση δεδομένων που αναφέρονται στο χώρο και μεταβάλλονται στο χρόνο. Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκε γεωγραφική βάση δεδομένων για να αποθηκευτούν τα σημεία ενδιαφέροντος, δηλαδή τα καταστήματα που βρίσκονται στην περιοχή των δύο νησιών της Πάρου και της Νάξου. Το λογισμικό επεξεργασίας των γεωγραφικών πληροφοριακών συστημάτων αποτελεί την μηχανή επεξεργασίας των ΓΠΣ και οι κύριες λειτουργίες είναι η συλλογή, αποθήκευση, διαχείριση, ανάκτηση, ανάλυση και παρουσίαση χωρικών και μη χωρικών δεδομένων.

Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκαν τα εξής λογισμικά: PostgreSQL 9.1, PostGIS, PgAdminIII, Quantum GIS (1.7.4). Όλα τα εργαλεία που χρησιμοποιήθηκαν είναι ανοικτού κώδικα (open source) που σημαίνει ότι μπορούμε να τα κάνουμε εγκατάσταση δωρεάν από το Διαδίκτυο και υπάρχουν ομάδες συζητήσεων (forums) που υποστηρίζουν την ανάπτυξή τους και σημειώνουν τα παρατηρούμενα σφάλματα και βοηθούν τη βελτιστοποίησή τους.

Η PostgreSQL είναι μια σχεσιακή βάση δεδομένων, βασίζεται σε μια αποδεδειγμένα καλή αρχιτεκτονική η οποία έχει δημιουργήσει μια ισχυρή αντίληψη των χρηστών της γύρω από την αξιοπιστία, την ακεραιότητα δεδομένων και την ορθή λειτουργία.

Το PostGIS είναι ένα λογισμικό που είναι επέκταση της PostgreSQL για χωρικά δεδομένα και αυτό που είναι ξεχωριστό είναι ότι χρησιμοποιήθηκαν χωρικά ευρετήρια δένδρουειδούς μορφής (R-Trees).

Η Διαχείριση της βάσης δεδομένων στην εργασία πραγματοποιήθηκε από το pgAdmin που αποτελεί το προκαθορισμένο (default) γραφικό περιβάλλον εκτέλεσης ερωτημάτων. Η χρήση του ήταν εύκολη από απλό γράψιμο SQL ερωτημάτων για την αποθήκευση των σημείων ενδιαφέροντος, τη δημιουργία των χωρικών ευρετηρίων, την δημιουργία και την εμφάνιση των ερωτημάτων.

Το Quantum GIS (QGIS) είναι ένα ισχυρό και φιλικό εργαλείο και χρησιμοποιήθηκε για την οπτικοποίηση των χωρικών δεδομένων, δηλαδή του χάρτη των περιοχών της Πάρου και της Νάξου και τα σημεία ενδιαφέροντος πάνω σε αυτές.

5.1.2 Συμπεράσματα από τη χρήση των τεχνολογιών διαδικτύου

Το Διαδίκτυο εξελίσσεται συνεχώς γι' αυτό οι τεχνολογίες διαδικτύου προσαρμόζονται στις αλλαγές αυτές. Στην παρούσα μεταπτυχιακή διατριβή χρησιμοποιήθηκαν οι εξής τεχνολογίες: Codeigniter που είναι πλαίσιο (framework) της php, Google maps, Google Maps API (Διεπαφή Προγραμματισμού Εφαρμογών Google Maps), Javascript, HTML, CSS. Το Codeigniter το κατεβάσαμε από το Διαδίκτυο, όπως και τα εργαλεία του Google maps και του Google Maps API για να φτιάξουμε την ιστοσελίδα με τη βοήθεια του κώδικα της Javascript, HTML και CSS.

Το Codeigniter είναι ένα ισχυρό πλαίσιο (framework) για την κατασκευή δυναμικών ιστοσελίδων με PHP που καταλαμβάνει ελάχιστο χώρο. Καθώς χρησιμοποιήθηκε το μοντέλο Μοντέλου - Προβολής - Ελεγκτή (MVC) που είναι μια αρχιτεκτονική προγραμματισμού και

βασίζεται στο διαχωρισμό κώδικα λογικής από τον κώδικα εμφάνισης και αλληλεπίδρασης με το χρήστη, έγινε καλύτερη οργάνωση του κώδικα και εγγυάται για την εύκολη συντήρηση του κώδικα. Οι ιστοσελίδες περιέχουν ελάχιστο κώδικα αφού η παρουσίαση είναι ξεχωριστά από το PHP κώδικα. Αξιοποιήθηκαν οι βιβλιοθήκες του έτσι ώστε να μπορεί να πραγματοποιηθεί ο στόχος μας.

Με το Google Maps που είναι μια δικτυακή (web) εφαρμογή χαρτογράφησης χρησιμοποιήθηκε ο χάρτης της Ελλάδος ζουμαρισμένος στις περιοχές των νησιών Πάρου και Νάξου. Με το Google maps API ο χρήστης μπορεί να έχει αλληλεπίδραση με την ιστοσελίδα του χάρτη καθώς κάνει κλικ πάνω του και συνδέονται άλλες ιστοσελίδες.

Τα εργαλεία της Javascript, HTML, CSS χρησιμοποιήθηκαν για την λειτουργία και την καλύτερη εμφάνιση των ιστοσελίδων που χρησιμοποιεί ο χρήστης. Συγκεκριμένα με τη Javascript και την εντολή της ανακατεύθυνσης (redirect) στέλνονται οι συντεταγμένες στο Codeigniter για να αξιοποιηθούν.

5.3 Μελλοντικές βελτιώσεις της εφαρμογής

Θα μπορούσε να βελτιωθεί η εφαρμογή μας αν χρησιμοποιηθούν jQuery- Ajax αντί της ανακατεύθυνσης (redirect), δηλαδή για να σταλούν οι συντεταγμένες του σημαδιού (marker) στο Codeigniter και να εμφανιστούν τα αποτελέσματα της επεξεργασίας αυτών αντίστοιχα και το JSON έτσι ώστε το μέγεθος δεδομένων επιστροφής από τον εξυπηρετητή να είναι πολύ μικρότερο.

Το AJAX (Asynchronous JavaScript και XML) είναι η μέθοδος της ανταλλαγής δεδομένων με έναν διακομιστή (server), και κάποια μέρη της ιστοσελίδας ενημερώνονται χωρίς να γίνει επαναφόρτωση ολόκληρης της σελίδας. Παραδείγματα των εφαρμογών που χρησιμοποιούν AJAX: Gmail, Google Maps, Youtube, Facebook και καρτέλες.

Το jQuery παρέχει διάφορες μεθόδους για τη λειτουργικότητα AJAX. Με τις μεθόδους AJAX jQuery, μπορείτε να ζητήσετε κείμενο, HTML, XML, JSON ή από ένα απομακρυσμένο διακομιστή χρησιμοποιώντας τόσο HTTP GET και HTTP POST.

Το JSON (JavaScript Object Notation) είναι ένα κείμενο με βάση το ανοιχτό πρότυπο σχεδιασμένο για ανθρώπινη αναγνώσιμη ανταλλαγή δεδομένων. Προέρχεται από τη γλώσσα JavaScript scripting για την αναπαράσταση απλών δομών δεδομένων και συνειρμική πίνακες, που ονομάζονται αντικείμενα. Παρά τη σχέση του με το JavaScript, είναι ανεξάρτητο από τη γλώσσα Javascript με προγράμματα ανάλυσης διαθέσιμα για πολλές γλώσσες.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Regina O. Obe, Leo S. Hsu, 2011. PostGIS IN Action. Greenwich: Manning, postgis-1.5.3, 2011. Οδηγός (Manual) [pdf] Διαθέσιμο στο: < <http://www.postgis.org/download/postgis-1.5.3.pdf>> [Προσπελάσιμο στις 20 Ιουλίου 2012].
- [2] Κολιός, 2009. Χωρική Βάση Δεδομένων PostgreSQL/PostGIS και Σύστημα Γεωγραφικών Πληροφοριών QuantumGIS, Οδηγός χρήσης- ΕΛ/ΛΑΚ [pdf]] Διαθέσιμο στο : <http://old.ellak.gr/index.php?option=com_openwiki&Itemid=103&id=ellak:postgresql_postgis_quantumgis>[Προσπελάσιμο στις 22 Ιουλίου 2012].
- [3] Ηλίας Φρέντζος , 2010. Εισαγωγή στη PostgreSQL – PostGIS (Πανεπιστημιακές Σημειώσεις) [pdf]] Διαθέσιμο στο: <<http://infolab.cs.unipi.gr/pre-eclass/courses/gis/lab/Lab-PostGIS-doc.pdf>>[Προσπελάσιμο στις Ιουλίου 2012].
- [4] Γιάννης Θεοδωρίδης, Νίκος Πελέκης, 2012. Γεωγραφικά Πληροφοριακά Συστήματα (GIS) & Βάσεις Γεωγραφικών Δεδομένων (Πανεπιστημιακές Σημειώσεις) [pdf]] Διαθέσιμο στο : <<http://infolab.cs.unipi.gr/>> [Προσπελάσιμο στις 24 Ιουνίου 2012].
- [5] Βίρβου, Μ., 2008. Τεχνολογία Λογισμικού (Πανεπιστημιακές σημειώσεις). Πειραιάς PostgreSQL, 1996. PostgreSQL 9.1.7 Documentation [Ηλεκτρονικό]] Διαθέσιμο στο: <<http://www.postgresql.org/docs/9.1/interactive/index.html>>[Προσπελάσιμο στις 25 Ιουνίου 2012].
- [6] qgis.org. Quantum GIS Documentantion [Ηλεκτρονικό] Διαθέσιμο στο: <<http://www.qgis.org/en/documentation.html>>[Προσπελάσιμο στις 12 Ιουνίου 2012].
- [7] w3schools, 1999. PHP Tutorial [Ηλεκτρονικό]] Διαθέσιμο στο: <<http://www.w3schools.com/php/default.asp>>[Προσπελάσιμο στις 7 Σεπτεμβρίου 2012].
- [8] w3schools, 1999. JavaScript Tutorial [Ηλεκτρονικό]] Διαθέσιμο στο: <<http://www.w3schools.com/js/default.asp>>[Προσπελάσιμο στις 8 Σεπτεμβρίου 2012].
- [9] developers.google. Google Maps JavaScript API v3 [Ηλεκτρονικό]] Διαθέσιμο στο : <<https://developers.google.com/maps/documentation>>[Προσπελάσιμο στις 9 Σεπτεμβρίου 2012].
- [10] Ellislab, 2006. CodeIgniter User Guide Version 2.1.3 [Ηλεκτρονικό] Διαθέσιμο στο: <<http://ellislab.com/codeigniter/user-guide/>>[Προσπελάσιμο στις 1 Ιουλίου 2012].
- [11] php.net, 1997. manual [Ηλεκτρονικό]] Διαθέσιμο στο : <<http://www.php.net/manual>>[Προσπελάσιμο στις 13 Ιουλίου 2012].
- [12] jQuery Foundation, 2011. jQuery. [Ηλεκτρονικό]] Διαθέσιμο στο: < <http://www.jquery.com/>>[Προσπελάσιμο στις 3 Φεβρουαρίου 2012].
- [13] Παναγιώτου Παναγιώτης και Παπαδόπουλος Νίκος, επιβλέποντας καθηγητής Αθανάσιος Κακαρούνας, 2011 «Ανάπτυξη μίας εφαρμογής επίβλεψης και ανίχνευσης ζημιών σε μια οικία με τη χρήση εργαλείων της Google» (Πτυχιακή Εργασία) Τμήμα Εφαρμογών Πληροφορικής στη διοίκηση και στην οικονομία Παράρτημα Λευκάδας Τεχνολογικό Εκπαιδευτικό ίδρυμα Ιόνιων Νησιών [pdf] Διαθέσιμο στο: <http://www.kolivias.de/lefkadaat/asddh.pdf> [Προσπελάσιμο στις 10 Σεπτεμβρίου 2012]
- [14] Μανδουράρη Δήμητρα, επιβλέποντας καθηγητής Παπαδάκης Νίκος, 2011 «Πληροφοριακό σύστημα για την καθοδήγηση χρηστών με χρήση GIS» (Πτυχιακή Εργασία) Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων Τεχνολογικό Σχολή Τεχνολογικών Εφαρμογών Εκπαιδευτικό Ίδρυμα Κρήτης [pdf] Διαθέσιμο στο: <http://nefeli.lib.teicrete.gr/browse/stef/epp/2011/MandourariDimitra/attached-document-1308739738-773291-16209/Mandourari2011.pdf> [Προσπελάσιμο στις 10 Σεπτεμβρίου 2012]

- [15] Σταυγιανουδάκης Ιωάννης επιβλέπουσα καθηγήτρια Λογκάκη Μαρία, 2002. «Ηλεκτρονικό ινστιτούτο αισθητικής με χρήση MVC PHP Framework» (Πτυχιακή Εργασία) Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων Σχολή Τεχνολογικών Εφαρμογών Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης [pdf] Διαθέσιμο στο: <http://ebookbrowse.com/staugianoudakis-pdf-d378334402> [Προσπελάσιμο στις 11 Σεπτεμβρίου 2012]
- [16] Ραφαηλίδης Ιωάννης, Φλώρος Ευάγγελος, επιβλέποντας καθηγητής Σωτήριος Λεβέντης, 2002. «Εργαλείο για βελτιστοποίηση G.I.S συστήματος web πλατφόρμας του εργαστηρίου PeLAB» (Πτυχιακή Εργασία) Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων Σχολή Τεχνολογικών Εφαρμογών Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης [pdf] Διαθέσιμο στο: <http://pernet.teipir.gr/netlab/sl/PTYX/PDF/RAFLO-2002-S.pdf> [Προσπελάσιμο στις 11 Σεπτεμβρίου 2012]
- [17] Γιώργος Τσιουτσιουλικλής, επιβλέποντας καθηγητής Σπύρος Σιούτας, 2010. «Χρήση Mobile GIS, για διαχείριση εκτάκτων αναγκών» Τμήμα Πληροφορικής Ιόνιο Πανεπιστήμιο [pdf] Διαθέσιμο στο: http://di.ionio.gr/dbislab/images/diploma-theses/tsioutsiouliklis_2010_thesis.pdf [Προσπελάσιμο στις 12 Σεπτεμβρίου 2012]
- [18] Σύστημα Γεωγραφικών Πληροφοριών (Βικιπαίδεια, 2013) Διαθέσιμο στο: <http://el.wikipedia.org>
- [19] MVC (The World Wide Wendlick, 2012) Διαθέσιμο στο: <http://www.endlick.com>
- [20] Model View Controller Pattern (Best-Practice Software Engineering, 2011): <http://best-practice-software-engineering.ifs.tuwien.ac.at/patterns/mvc.html>
- [21] WAPP Stack (Bitnami.com, 2013) [Ηλεκτρονικό] Διαθέσιμο στο: <http://bitnami.com/stack/wapp>
- [22] pgAdmin PostgreSQL tools (pgadmin.org,2013)) [Ηλεκτρονικό] Διαθέσιμο στο: (<http://www.pgadmin.org/download/>)

ΠΑΡΑΡΤΗΜΑ

Ο ΚΩΔΙΚΑΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Παρακάτω δίνεται ο κώδικας της εφαρμογής.

Το αρχείο map.php

(C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\views\pages)

```

<!DOCTYPE html>
<html>
<!--η πρώτη σελίδα εμφάνισης του χάρτη-->
<head>
  <title>Δυναμική διαμόρφωση κριτηρίων γεωπληροφοριακής αναζήτησης</title>
  <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <style type="text/css">
<!--το στυλ της εμφάνισης της σελίδας-->
  body {
    font-size: 1.25em;
    color: #222;
    background-color:#99CC99;
  }
  html { height: 100% }
  body { height: 100%; margin: 0; padding: 0 }
  #map_canvas { height: 85% }
  div#content { height: 10%;
    display: block;
    text-align: center;
    color: #3300CC;}
</style>
<!--εισαγωγή κλειδιού για να χρησιμοποιήσουμε το χάρτη του google map-->
<script
src="https://maps.googleapis.com/maps/api/js?key=AlzaSyC_WEGLC7MaFY57_aLiUn4w
Bhw152vDTuw&sensor=false"></script>
<script>
<!--ρυθμίσεις για το χάρτη-->
  function initialize() {
    var mapOptions = {
      zoom: 10,
      center: new google.maps.LatLng(37.0261, 25.3051),
      mapTypeId: google.maps.MapTypeId.ROADMAP
    };
    var map = new google.maps.Map(document.getElementById('map_canvas'),
      mapOptions);
    <!--συμβάν για το πάτημα του ποντικιού πάνω στο χάρτη-->
    google.maps.event.addListener(map, 'click', function(e) {

```

```

    placeMarker(e.latLng, map);
    <!--στέλνονται οι συντεταγμένες του σημαδιού στηνπαρακάτω διεύθυνση-->

uri="http://localhost/ergasia7/index.php/pages/landmarks/"+e.latLng.lat()+" "+e.latLng.lng();
    window.location=uri;
    });
}
<!--τοποθέτηση του σημαδιού-->
function placeMarker(position, map) {
    var marker = new google.maps.Marker({
        position: position,
        map: map
    })
}

map.panTo(position);

}
<!--φόρτωση του παραθύρου με το σημάδι που έχει τοποθετηθείμε το κλικ του ποντικιού-->
google.maps.event.addDomListener(window, 'load', initialize);
</script>

</head>
<body>
<!--χωρισμός του παραθύρου -->
<div id="content"><p>Δυναμική διαμόρφωση κριτηρίων γεωπληροφοριακής
αναζήτησης<br/> Πατήστε πάνω στο χάρτη</p></div>
<div id="map_canvas"></div>
</html>

```

Το αρχείο Pages.php

(C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\controllers)

```
<?php
```

```

class Pages extends CI_Controller {
    /*η συνάρτηση για να εμφανίζεται το αρχείο με το χάρτη στο φυλλομετρητή του Διαδικτύου*/
    public function view($page = 'map')
    {
        $this->load->view('pages/'.$page);
    }
    /*η συνάρτηση για να φορτωθεί το μοντέλο landmarks_model*/
    public function __construct()
    {
        parent::__construct();
    }
}

```

```

    $this->load->model('landmarks_model');
}

/*η συνάρτηση που λαμβάνει τις συντεταγμένες του σημαδιού */
public function landmarks($lat, $lng)
{
    /*παιρνει τα δεδομένα από τη συνάρτηση get_landmarks($lat, $lng)*/
    $data['landmarks'] = $this->landmarks_model->get_landmarks($lat,$lng);
    /*αν τα αποτελέσματα δεν υπάρχουν φορτώνεται το αρχείο noResults*/
    if (empty($data['landmarks'])) {
        $this->load->view('noResults',$data);
    }
    else
        /*αν υπάρχουν αποτελέσματα τότε φορτώνεται το αρχείο landmarks*/
        {
            $this->load->view('landmarks',$data);
        }
    }
}

```

Το αρχείο Landmarks.model.php

(C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\models)

```

<?php
class Landmarks_model extends CI_Model {
    /*η συνάρτηση όπου φορτώνεται η βάση δεδομένων*/
    public function __construct()
    {
        $this->load->database();
    }
    /*η συνάρτηση που παίρνει ως μεταβλητές τις συντεταγμένες και τις επεξεργάζεται*/
    public function get_landmarks($lat, $lng)

    {
        /*το ερώτημα που εκτελείται για να βρεθούν τα σημεία ενδιαφέροντος της βάσης δεδομένων
σε απόσταση 1 km από το κλικ*/
        $query = $this->db->query("SELECT landmark_id, landmark_name, landmark_kind, island,
landmark_geometry
FROM landmarks
WHERE ST_buffer(ST_GeomFromText('point(".$lng." ".$lat."'),4326), 0.01) &&
landmark_geometry");

        /*επιστροφή των αποτελεσμάτων*/

return $query->result_array();
}
}

```



```
}

```

```
}

```

Το αρχείο Landmarks.php

(C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\views)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<!--η σελίδα όπου εμφανίζονται τα σημεία ενδιαφέροντος που βρίσκονται σε απόσταση 1 km-->
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="el" dir="ltr">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>Σημεία Ενδιαφέροντος</title>
  <!--το στυλ εμφάνισης της σελίδας-->
  <style media="screen" type="text/css">
    body {
      font-size: .8em;
      color: #222;
      background-color:#aaa;
    }

    #content h1 {
      color: #57f;
    }

    div#content {
      background-color:#fff;
      margin: 0 auto;
      width: 700px;
      min-height:600px;
    }

    div#results {
      width:80%;
      margin:0 auto;
    }

    table {
      border-collapse: collapse;

```

```

}

table td {
  padding-left: 30px;
  padding-right: 30px;
}

table thead {
  font-weight: bold;
  background-color: #ddd;
}

table tr {
  border-bottom: 1px dotted #000;
}

#newsearchlink {
  margin-left:520px;
  margin-top:50px;
}
</style>
</head>
<body>
<div id="content">
<!--επικεφαλίδα στη σελίδα-->
<h1>Σημεία ενδιαφέροντος που βρίσκονται σε ακτίνα 1km</h1>
<div id="results">
<table>
  <tbody>
  <thead>
    <th>Κωδικός</th><th>Όνομα Ιδιοκτήτη</th><th>Είδος</th><th>Νησί</th>
  </thead>
  <!--εμφάνιση των αποτελεσμάτων-->
  <?php foreach ($landmarks as $landmarks_item): ?>

    <tr>
      <td><?php echo $landmarks_item['landmark_id'] ?></td>
      <td><?php echo $landmarks_item['landmark_name'] ?></td>
      <td><?php echo $landmarks_item['landmark_kind'] ?></td>
      <td><?php echo $landmarks_item['island'] ?></td>
    </tr>
  <?php endforeach; ?>
  </tbody>
</table>
</div>

```

```

<!--υπερσύνδεσμος για να γυρίσει πίσω ο χρήστης στην αρχική σελίδα-->
<a href=" ../view/map" id="newsearchlink">Νέα Αναζήτηση</a>
</div>
</body>
</html>

```

Το αρχείο noResults.php

(C:\Program Files\BitNami WAPPStack\apache2\htdocs\ergasia7\application\views)

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<!--η σελίδα όπου εμφανίζεται το μήνυμα οτι δεν υπάρχουν αποτελέσματα-->
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="el" dir="ltr">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>Σημεία Ενδιαφέροντος</title>
  <!--το στυλ εμφάνισης της σελίδας-->
  <style media="screen" type="text/css">
    body {
      font-size: .8em;
      color: #222;
      background-color:#aaa;
    }

    #content h1 {
      color: #57f;
    }

    div#content {
      background-color:#fff;
      margin: 0 auto;
      width: 700px;
      min-height:600px;
    }

    div#results {
      width:80%;
      margin:0 auto;
    }

    table {
      border-collapse: collapse;
    }

```

```
table td {
  padding-left: 30px;
  padding-right: 30px;
}

table thead {
  font-weight: bold;
  background-color: #ddd;
}

table tr {
  border-bottom: 1px dotted #000;
}

#newsearchlink {
  margin-left:520px;
  margin-top:50px;
}

p.error {
  color: #f00;
  font-size:1.4em;
}
</style>
</head>
<body>
<div id="content">
<!--επικεφαλίδα της σελίδας-->
<h1>Σημεία ενδιαφέροντος που βρίσκονται σε ακτίνα 1km</h1>
<!--το μήνυμα που εμφανίζεται-->
<p class="error">Δεν βρέθηκαν αποτελέσματα!</p>
<!--δημιουργία υπερσυνδέσμου για να γυρνάει ο χρήστης πίσω στο χάρτη για νέα αναζήτηση-->
<a href="../../view/map" id="newsearchlink">Νέα Αναζήτηση</a>
```