



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΤΟΠΟΓΡΑΦΩΝ
ΜΗΧΑΝΙΚΩΝ

Διπλωματική εργασία :

**«Ανάπτυξη Διαδικτυακής Εφαρμογής για την Αξιολόγηση
Παραμέτρων στον Σχεδιασμό Δικτύου Ποδηλατικών
Διαδρομών »**

ΟΝΟΜΑΤΕΠΩΝΥΜΟ : ΘΕΟΔΩΡΟΠΟΥΛΟΣ ΑΘΑΝΑΣΙΟΣ - ΓΕΩΡΓΙΟΣ
ΗΜΕΡΟΜΗΝΙΑ ΠΑΡΑΔΟΣΗΣ: 8/11/2012

Περιεχόμενα

Κεφάλαιο 1	4
1.2 Οργάνωση της εργασίας	7
ΚΕΦΑΛΑΙΟ 2	8
2.1 Σημασία της χρήσης ποδηλάτου και η επίπτωση στην μορφή των σύγχρονων πόλεων	8
2.2 Μεθοδολογία σχεδιασμού δικτύου ποδηλατικών διαδρομών.....	9
Κεφάλαιο 3	12
3.1 ΣΥΣΤΗΜΑΤΑ ΓΕΩΓΡΑΦΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ	12
3.3 Αναπαράσταση (Δομές) χαρτογραφικών δεδομένων.....	15
3.3.1 Κανικοποιημένα (RASTER) Δεδομένα	16
3.3.2 Διανυσματικά (Vector) δεδομένα.....	16
3.3.3 Πλεονεκτήματα - μειονεκτήματα Raster /Vector	18
3.3.4 Μη χωρικά δεδομένα	19
3.4 Συστήματα συντεταγμένων.....	20
3.4.1 Γεωδαισία.....	20
3.4.2 Επιφάνειες Αναφοράς	21
3.4.3 Χαρτογραφικές προβολές:	24
Κεφάλαιο 4	26
Τεχνολογίες και εργαλεία	26
Ο παγκόσμιος ιστός.....	26
4.1 Η γλώσσα HTML και η δυνατότητα χαρτογράφησης.....	26
4.1.1 Html	26
4.1.2 Html και στατικοί χάρτες.....	27
4.2 Html και διαδραστικοί χάρτες.....	27
4.2.1 • Επέκταση λειτουργικότητας του πελάτη	28
4.2.2 • Επέκταση λειτουργικότητας του Εξυπηρετητή.....	28
4.2.3• Μικτές προσεγγίσεις.....	28
4.3 Java – Javascript.....	29
4.4. Υπηρεσίες χαρτογραφικού περιεχομένου στον παγκόσμιο	29
ιστό	29
4.5. Η γλώσσα kml	30
4.6. Το εργαλείο Open Layers	31
4.7 Διαδικτυακές υπηρεσίες (Web Services).....	31
4.7.1 Mash - ups	32
4.7.2 Διαδικτυακά Γεωγραφικά Πληροφοριακά Συστήματα (web - GIS)	33
4.7.3 Google Elevation API	35
Παράδειγμα υπολογισμού σημειακών υψομέτρων	39
Παράδειγμα υπολογισμού υψομέτρων διαδρομής.....	41
4.7.4 Google Chart Tools.....	44
ΚΕΦΑΛΑΙΟ 5	47
5.1 Διαδικτυακή Εφαρμογή geodata.gov.gr	47
5.2 Χάρτης ΕΜΠ.....	48
5.3 Διαδικτυακή Εφαρμογή Wikimapia.org	49
ΚΕΦΑΛΑΙΟ 6Ο.....	50
6.1 ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ.....	50
6.1.1 Αρχιτεκτονική χαρτογραφικού σκέλους εφαρμογής και επιλογή υποβάθρων.....	52
Αρχιτεκτονική Εφαρμογής	52
6.1.2 Επιλογή Χαρτογραφικού Υποβάθρου	53
6.2 Έναρξη Δημιουργίας Γεωμετρίας	56

6.3 Κλήση της υπηρεσίας Google Elevation Api.....	57
6.4 Μετασχηματισμός των αποστάσεων σε γεωδαιτικές	57
6.5 Υπολογισμοί Κλίσεων	59
6.6 Δημιουργία Διαγραμμάτων	59
6.7 Εύρεση των αποστάσεων από MMM μεταφοράς και ενημέρωση του χρήστη	59
6.8 Επιλογή των αρχείων προς αποθήκευση	60
6.9 Περιγραφή της τρόπου Λειτουργίας της Εφαρμογής.....	60
6.10 Λογισμικό που χρησιμοποιήθηκε.....	63
ΚΕΦΑΛΑΙΟ 7Ο.....	65
Σχεδίαση ποδηλατικού δικτύου στην Πολυτεχνειούπολη Ζωγράφου	65
7.1 Χρήση της εφαρμογής	65
7.2 Εισαγωγή αρχείων.....	68
7.3 Αποθήκευση ψηφιοποιημένων από τον χρήστη αρχείων	69
Κεφάλαιο 8ο	72
ΠΡΟΒΛΗΜΑΤΑ ΚΑΙ ΔΥΝΑΤΟΤΗΤΑ ΓΙΑ ΜΕΛΛΟΝΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ	72
8.1 Γεωδαιτικά Συστήματα Αναφοράς.....	72
8.2 Υπολογισμοί αποστάσεων	73
8.3 Περιορισμοί που επιβάλλονται από το Google Elevation API	74
8.4 Δειγματοληψία Υψομέτρων.....	75
8.5 Δυνατότητα Για άλλες εφαρμογές.....	76
8.5.1 Υπολογισμός θερμιδικής κατανάλωσης για συγκεκριμένες ποδηλατικές εφαρμογές	76
8.5.2 Υπολογισμός κατανάλωσης καυσίμου για συγκεκριμένη διαδρομή.....	76
8.5.3 Υπολογισμός παραγωγής καυσαερίου από κινούμενα οχήματα.....	77
8.5.4 Έκθεση διαδρομών με βάση το καυσαέριο.....	77
8.5.5 Ψηφιοποίηση και δημιουργία αρχείων	78
Συμπεράσματα	78
Βιβλιογραφία	79
Βιβλιογραφία	79
Παράρτημα	82

Κεφάλαιο 1

Εισαγωγή :

Στην παρούσα εργασία διερευνώνται οι τεχνολογίες οπτικοποίησης και διαχείρισης γεωγραφικών δεδομένων στον παγκόσμιο ιστό.

Είναι γεγονός ότι στην σημερινή εποχή, η εξέλιξη της επιστήμης των υπολογιστών, της ψηφιακής χαρτογραφίας, της γεωδαισίας και των βάσεων δεδομένων αποτέλεσε την βάση στην οποία δημιουργήθηκαν και συνεχίζουν να εξελίσσονται τα Συστήματα Γεωγραφικών Πληροφοριών.

Επίσης είναι γεγονός ότι ένα σημαντικό μέρος των αποφάσεων που λαμβάνονται έχουν κάποια γεωγραφική αναφορά. Η εφαρμογή των γεωγραφικών δεδομένων σε πολλούς τομείς οδήγησε στην συνεχή ανάγκη για πληροφορία με κάποια χωρική αναφορά. Η ανάγκη αυτή αποτέλεσε την αφορμή για την δημοσιοποίηση χαρτογραφικού περιεχομένου στον παγκόσμιο ιστό.

Σήμερα πλέον, η δημοσιοποίηση χαρτογραφικού περιεχομένου στον παγκόσμιο ιστό αποτελεί μια κοινή πρακτική. Η δημοσιοποίηση αυτή μπορεί να επιτευχθεί με διάφορους τρόπους, ανάλογα με το επιθυμητό αποτέλεσμα διαδραστικότητας και περιλαμβάνει την δυνατότητα διαχείρισης της χωρικής πληροφορίας. Επομένως η δημοσιοποίηση εμπεριέχει αφενός μεν την δυνατότητα προβολής της χαρτογραφικής πληροφορίας αλλά και την δυνατότητα επεξεργασίας της.

Αντικείμενο της παρούσας εργασίας είναι η δημιουργία μιας διαδικτυακής εφαρμογής για τον σχεδιασμό δικτύου ποδηλατόδρομων μεγάλου μήκους.

Για τον σκοπό έγινε μια διερεύνηση των εργαλείων ανοιχτού κώδικα που μπορούν να χρησιμοποιηθούν για την ανάπτυξη διαδικτυακών εφαρμογών.

Για την ανάπτυξη της διαδικτυακής εφαρμογής χρησιμοποιήθηκαν δυο εργαλεία ανοιχτού κώδικα που είναι δωρεάν διαθέσιμα στο διαδίκτυο, η Javascript βιβλιοθήκη OpenLayers, ενώ ως εξυπηρετητής δεδομένων το λογισμικό geoserver

Abstract

In the current essay the technologies of visualization and management of geographical data in the world wide web are examined.

It is a fact that in our modern era, the development of computer's science, of digital cartography , of geodesy and of data base system has contributed the base on which Geographical Informations Systems was based and continue to grow.

Furthermore, we should mention that an essential part of the decisions has a certain geographical reference. The implementation of geographical data in many fields has led to the continuous need for information with a certain spatial reference. This need has constituted the excuse for the publication of maps in the global web.

Today, publishing map data in the web constitutes a common practice. This publication can be achieved in varied ways, depending on the desirable result of interactivity and includes the possibility of managing spatial information.

Therefore, publishing includes on the one hand the possibility of projecting spatial information and on the other hand the possibility of processing it.

The creation of a web application for the designing of a net of bicycle routes of long length is the object of the current essay.

To this purpose a research of the open source tools which could be used for the development of web applications was made.

For the development of this web application two opensource tools was are freely available on the network, the javascript library openlayers, as a data server, geoserver software, were used.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω θερμά τον Επίκουρο Καθηγητή της Σχολής Αγρονόμων και Τοπογράφων Μηχανικών κύριο Βασίλειο Βεσκούκη για την πολύτιμη βοήθεια που μου προσέφερε κατά την σύνταξη της παρούσης διπλωματικής εργασίας.

1.2 Οργάνωση της εργασίας

Η παρούσα εργασία χωρίζεται σε οκτώ κεφάλαια.

Στο κεφάλαιο 1 γίνεται μια εισαγωγή στο αντικείμενο της εργασίας και παρουσιάζεται η διάρθρωσή της.

Στο κεφάλαιο 2 γίνεται μια ανάλυση της χρησιμότητας του ποδηλάτου και αναλύονται τα βήματα που εφαρμόζονται στον σχεδιασμό δικτύου ποδηλατικών διαδρομών. Γίνεται μια θεώρηση των παραμέτρων που γενικά λαμβάνονται υπόψη κατά τον σχεδιασμό ενός δικτύου ποδηλατόδρομων και αναφέρονται οι παράμετροι που λάβαμε υπόψη κατά την δημιουργία της διαδικτυακής εφαρμογής.

Στο κεφάλαιο 3 γίνεται μια παρουσίαση των κυριότερων εννοιών που αφορούν τα Συστήματα Γεωγραφικών Πληροφοριών.

Στο κεφάλαιο 4 παρουσιάζονται παρουσιάζονται θεωρητικά τα εργαλεία που χρησιμοποιούνται για την διάχυση γεωγραφικών δεδομένων στον παγκόσμιο ιστό, τα οποία αποτελούν και τις τεχνολογίες δημιουργίας μίας διαδικτυακής εφαρμογής. Ιδιαίτερη βαρύτητα δώθηκε στις τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής για τον σχεδιασμό δικτύου ποδηλατικών διαδρομών .

Στο κεφάλαιο 5 παρουσιάζονται συναφείς εφαρμογές χαρτογραφίας στον παγκόσμιο ιστό.

Στο κεφάλαιο 6 παρουσιάζεται η εφαρμογή που δημιουργήθηκε.

Στο κεφάλαιο 7 γίνεται χρησιμοποίηση της εφαρμογής σε ένα πραγματικό πρόβλημα.

Στο κεφάλαιο 8 αναφέρονται τα προβλήματα που αντιμετωπίσαμε, οι αποφάσεις που κληθήκαμε να λάβουμε καθώς και προτεινόμενες δυνατότητες για μελλοντικές εφαρμογές

ΚΕΦΑΛΑΙΟ 2

2.1 Σημασία της χρήσης ποδηλάτου και η επίπτωση στην μορφή των σύγχρονων πόλεων

Το ποδήλατο αποτελεί έναν από τους πιο φυσικούς τρόπους μετακίνησης. Επιπροσθέτως με δεδομένο το μεγάλο κυκλοφοριακό πρόβλημα των σύγχρονων πόλεων κρίνεται επιτακτική η εξεύρεση εναλλακτικών λύσεων. Το ποδήλατο προσφέρει την καλύτερη επαφή με το περιβάλλον, είναι ανεξάρτητος τρόπος μετακίνησης και δεν κοστίζει.

Όμως για πολλά χρόνια ο πολεοδομικός σχεδιασμός προσαρμόσε την πόλη στο αυτοκίνητο και στις ανάγκες του σε χώρο στερώντας τον από τους πεζούς και τα ποδήλατα. Το γεγονός αυτό είχε καταστροφικές συνέπειες στην μορφή των σύγχρονων πόλεων επιβαρύνοντας τον πολεοδομικό ιστό. Η συνεχής ζήτηση για μετακινήσεις διόγκωσε το πρόβλημα με δυσμενέστερες συνέπειες για το περιβάλλον και εν τέλει στην ζωή των πολιτών με αποτέλεσμα να αναζητούνται εναλλακτικές λύσεις στο πρόβλημα. Είναι γεγονός ότι οι περισσότερες πόλεις των ανεπτυγμένων κρατών έχουν δημιουργήσει υποδομές για το περπάτημα και το ποδήλατο διευκολύνοντας τις συνθήκες μετακίνησης με τα μέσα αυτά. Για την καθιέρωση της χρήσης του ποδηλάτου είναι σημαντικό πέρα από την ύπαρξη υποδομών να υπάρχουν συμπληρωματικές προτάσεις πολιτικής όπως λόγου χάρη η ύπαρξη θέσεων στάθμευσης, η δυνατότητα χρησιμοποίησης κοινόχρηστων ποδηλάτων κ.ο.κ.

2.2 Μεθοδολογία σχεδιασμού δικτύου ποδηλατικών διαδρομών

Για τον σχεδιασμό ενός δικτύου ποδηλατικών διαδρομών λαμβάνονται πολλοί παράμετροι υπόψη, όπως λόγου χάρη οι πολεοδομικοί και κυκλοφοριακοί παράμετροι της γεωγραφικής περιοχής για την οποία θα σχεδιαστεί.

Οι πολεοδομικοί παράμετροι σχετίζονται με την επιθυμία των πολιτών να μεταβούν από το ένα σημείο της πόλης στο άλλο. Η επιθυμία αυτή σχετίζεται άμεσα με τις χρήσεις γης της περιοχής που ένα συγκεκριμένο πλήθος πολιτών επιθυμεί να επισκεφτεί, της περιοχής πόλου-έλξης δηλαδή.

Οι κυκλοφοριακοί παράμετροι σχετίζονται με την ευκολία ή δυσκολία πρόσβασης στην περιοχή αυτή και σχετίζονται άμεσα με την γεωμετρία του οδικού δικτύου αλλά και την στάθμη εξυπηρέτησης των οδών που καταλήγουν στην περιοχή πόλο-έλξης (χρονική-χωρική ταχύτητα, συγκέντρωση, διαχωρισμός).

Από τα παραπάνω συμπεραίνουμε ότι οι χρήσεις γης παράγουν μετακινήσεις, οι οποίες πρέπει να ικανοποιούνται είτε μέσω του οδικού είτε μέσω οχημάτων σταθερής τροχιάς.

Για την χρησιμοποίηση του ποδηλάτου ως μέσου για την ικανοποίηση αναγκών μεταφοράς εύλογο είναι ότι η υποδομή που θα δημιουργηθεί (ποδηλατόδρομοι) θα πρέπει να εξυπηρετεί περιοχές που έλκουν μετακινήσεις, πόλους έλξης δηλαδή. Τέτοιες περιοχές είναι αυτές που αναπτύσσουν οικονομική δραστηριότητα (π.χ χώρους εργασίας), περιοχές που βρίσκονται πανεπιστήμια αλλά και περιοχές που συγκεντρώνουν χρήσεις γης που σχετίζονται με την αναψυχή ή και τον πολιτισμό.

Κρίσιμο στοιχείο αποτελεί το γεγονός ότι επειδή η ποδηλασία απαιτεί σωματική προσπάθεια η οποία εξαρτάται τόσο από την φυσική κατάσταση του ποδηλάτη, αλλά και από το ανάγλυφο, το οποίο εκφράζεται από τις κλίσεις του ποδηλατόδρομου. Επομένως οι κλίσεις ενός ποδηλατόδρομου θα πρέπει να είναι οι βέλτιστες. Πέρα από τις κλίσεις, ο σχεδιασμός ενός ποδηλατόδρομου πρέπει να δίνει την δυνατότητα στον ποδηλάτη να διατηρεί με όσο το δυνατόν λιγότερη προσπάθεια σταθερή την ταχύτητά του. Αυτό μεταφράζεται σε όσο το δυνατόν λιγότερες διασταυρώσεις με υπάρχοντες δρόμους, αλλά και παραχώρηση προτεραιότητας στο ποδήλατο σε αυτές.

Επιπροσθέτως επειδή όπως αναφέρθηκε οι δραστηριότητες παράγουν ανάγκη για μετακινήσεις , πρέπει ένα δίκτυο ποδηλατόδρομων να ακολουθεί κύριες οδούς που προσφέρουν αμεσότερη πρόσβαση σε δραστηριότητες.

Η μεθοδολογία χάραξης ενός δικτύου ποδηλατόδρομων πρέπει να ακολουθεί όσο το δυνατόν τις παραπάνω αρχές ώστε να μεγιστοποιείται η ωφέλεια από την χρήση του.Αναλυτικότερα η μεθοδολογία περιλαμβάνει τα ακόλουθα 4 αλληλοεξαρτώμενα βήματα σχεδιασμού :

1ο βήμα: Προσδιορισμός των βασικών ζωνών ανάπτυξης των μητροπολιτικών διαδρομών ποδηλάτου.

Κατά το βήμα αυτό γίνεται ανάλυση της περιοχής μελέτης τόσο από την κυκλοφοριακή σκοπιά όσο και από την πολεοδομική με σκοπό τον προσδιορισμό των πόλων που θα συνδέει το ποδηλατικό δίκτυο.Το θέμα που διερευνάται ουσιαστικά αναφέρεται στον αριθμό των μετακινήσεων που ξεκινούν από κάθε ζώνη και τον αριθμό των μετακινήσεων που καταλήγουν σε αυτήν (ανάλυση κατά ζώνες) ή αντίστοιχα στον αριθμό των μετακινήσεων που κάνει ένας μετακινούμενος/ή νοικοκυριό κατά την διάρκεια μίας εβδομάδας (εξατομικευμένα μοντέλα).Οι ζώνες που θα συνδέουν τους παραπάνω πόλους είναι και οι περιοχές τις οποίες θα πρέπει να διασχίζουν οι ποδηλατικές διαδρομές.

2ο βήμα: Διαμόρφωση εναλλακτικών σεναρίων ποδηλατικών διαδρομών ανά ζώνη.

Είναι προφανές ότι δυο πόλοι (περιοχές) μπορούν να συνδεθούν μεταξύ τους με πάρα πολλές εναλλακτικές διαδρομές.Εξαιτίας της φύσης του προβλήματος, το οποίο δεν είναι μονοσήμαντο δημιουργούνται εναλλακτικά σενάρια ποδηλατικών διαδρομών που να συνδέουν μεταξύ τους τις ζώνες. Κριτήρια επιλογής είναι η ασφάλεια και η άνεση της διαδρομής καθώς και ο περιβάλλον χώρος.

3ο βήμα: Αξιολόγηση των εναλλακτικών σεναρίων ποδηλατικών διαδρομών.

Το βήμα αυτό περιλαμβάνει την αξιολόγηση των εναλλακτικών ποδηλατικών διαδρομών ανά ζώνη με τα κριτήρια που προαναφέρθηκαν και έχει ως στόχο την επιλογή της βέλτιστης ποδηλατικής διαδρομής ανά ζώνη.

4ο βήμα: Συνολική αξιολόγηση του δικτύου και τελική επιλογή των μητροπολιτικών διαδρομών.

Τα αποτελέσματα του προηγούμενου βήματος δίνουν μια μορφή του δικτύου. Το δίκτυο αυτό αξιολογείται εκ νέου με κυριότερα κριτήρια κάλυψης της ανάγκης των μετακινήσεων.

Συμπερασματικά λοιπόν αναφέρουμε ότι ενδιαφέρουν τα πολεοδομικά και κυκλοφοριακά στοιχεία για τον σχεδιασμό ενός δικτύου ποδηλατόδρομων. Τα στοιχεία αυτά προφανώς διαφέρουν από περιοχή σε περιοχή. Οι παράμετροι της κυκλοφοριακής ανάλυσης που εξετάζονται όμως παραμετροποιούνται εύκολα. Αυτοί είναι η κλίση των ποδηλατόδρομων, η απόστασή τους από σημεία ενδιαφέροντος (λ.χ σταθμούς μετρό) και οι αναπτυσσόμενες ταχύτητες. Οι ταχύτητες αυτές συνδέονται αφενός μεν με την ασφάλεια των ποδηλατών αλλά και με τα επίπεδα θορύβου που δημιουργούνται.

Στην εφαρμογή που δημιουργήσαμε και η οποία παρουσιάζεται στο κεφάλαιο 6 της παρούσης υπολογίζουμε τις κλίσεις ενός σχεδιασθέντος δικτύου ποδηλατόδρομων καθώς και την απόστασή του από προκαθορισμένα σημεία ενδιαφέροντος.

Κεφάλαιο 3

3.1 ΣΥΣΤΗΜΑΤΑ ΓΕΩΓΡΑΦΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ

Τα Γεωγραφικά Συστήματα Πληροφοριών (Γ.Σ.Π. - Geographical Information Systems, GIS) είναι μία οργανωμένη συλλογή μηχανικών υπολογιστικών συστημάτων (hardware), λογισμικών συστημάτων (software), χωρικών δεδομένων και ανθρώπινου δυναμικού, με σκοπό τη συλλογή, καταχώρηση, ενημέρωση, διαχείριση, ανάλυση και απόδοση, κάθε μορφής πληροφορίας που αφορά στο γεωγραφικό περιβάλλον (Κουτσόπουλος, 2002:53). Τα Γ.Σ.Π. έχουν σαν κυρίαρχο στόχο τον χωρικό σχεδιασμό και αντιπροσωπεύουν ένα ισχυρό σύνολο εργαλείων για την συλλογή, αποθήκευση, ανάληψη ανά πάσα στιγμή, μετασχηματισμό και απεικόνιση χωρικών στοιχείων του πραγματικού κόσμου.

Ένα Γ.Σ.Π. αποτελείται από τρία βασικά συστατικά μέρη α) το υλικό, β) το λογισμικό και γ) τα δεδομένα. Το υλικό και το λογισμικό έχουν ένα καθορισμένο κύκλο ζωής, επηρεάζονται στενά από τις τεχνολογικές εξελίξεις και αντικαθίστανται συχνά από νεότερα και πιο σύγχρονα προϊόντα. Από την άλλη πλευρά, τα δεδομένα αποτελούν το πλέον δαπανηρό συστατικό των Γ.Σ.Π., καθώς η συλλογή τους απαιτεί πολύ χρόνο και προσπάθεια. Επιπλέον τα περισσότερα γεωγραφικά δεδομένα είναι δυναμικά και απαιτούν συνεχείς ενημερώσεις.

Το υλικό των Γ.Σ.Π. περιλαμβάνει τις μονάδες επεξεργασίας που αναλαμβάνουν την εκτέλεση των προγραμμάτων, τις βοηθητικές μονάδες αποθήκευσης που φιλοξενούν τα δεδομένα και τα προγράμματα και τέλος τις περιφερειακές μονάδες που χρησιμοποιούνται για την εισαγωγή των δεδομένων, τη δημιουργία χαρτών και τεχνικών εκθέσεων κ.λπ. Από την άλλη μεριά, το λογισμικό των Γ.Σ.Π. περιβάλλει το υλικό και περιλαμβάνει τα διάφορα προγράμματα εκτέλεσης λειτουργιών του υπολογιστικού συστήματος.

Τα δεδομένα αποτελούν αναπόσπαστο τμήμα κάθε πληροφοριακού συστήματος. Τα Γ. Σ. Π. έχουν την ιδιαιτερότητα της διαχείρισης και επεξεργασίας δεδομένων με χωρική και θεματική διάσταση, οι οποίες μεταβάλλονται στο χρόνο. Τα σύγχρονα πακέτα Γ.Σ.Π. διαχειρίζονται ικανοποιητικά τη χωρική και θεματική διάσταση των γεωγραφικών οντοτήτων, ενώ έμμεσα διαχειρίζονται τις χρονικές τους μεταβολές (Στεφανάκης, 2003:32).

Το μεγάλο πλήθος των πληροφοριών και η πολυπλοκότητα της δομής του αστικού χώρου, καθιστά δύσκολη και προβληματική την διαχείριση των πληροφοριών μέσω μιας συμβατικής τράπεζας πληροφοριών. Οι τεχνολογίες διαχείρισης πληροφοριών έχουν αναπτυχθεί με πολύ γρήγορους ρυθμούς τα τελευταία χρόνια, με αποτέλεσμα να προσφέρονται σήμερα δυνατότητες που ποτέ άλλοτε δεν ήταν διαθέσιμες. Γίνεται επομένως αντιληπτό ότι είναι επιτακτική η ανάγκη της εκμετάλλευσης των δυνατοτήτων της σύγχρονης τεχνολογίας όχι μόνο για καλύτερη ποιοτικά και ποσοτικά πληροφόρηση για μία πόλη, αλλά και για την καλύτερη διαχείριση των δεδομένων που χαρακτηρίζουν τις λειτουργίες της. Η τεχνολογία των Γ.Σ.Π., διαθέτει όλες τις απαραίτητες προϋποθέσεις για την οργανωμένη διαχείριση των συλλεγόμενων πληροφοριών (Διαμαντάκης, Πραστάκος, 1999).

Γενικά, όταν τίθεται το θέμα διαχείρισης πληροφοριών, ο ρόλος των Γ.Σ.Π. είναι καθοριστικός. Ένας γενικά αποδεκτός ορισμός θεωρεί ότι ένα Γ.Σ.Π. δεν είναι παρά μια πολυδύναμη βάση δεδομένων που συσχετίζεται με το γεωγραφικό χώρο. Παράλληλα όμως, είναι εφοδιασμένη με ένα ολοκληρωμένο σύστημα εργαλείων που παρέχουν την δυνατότητα οποιασδήποτε ανάλυσης και σύνθεσης πληροφορίας επιθυμεί ο χρήστης, από την παραγωγή ενός απλού πίνακα δεδομένων έως την παραγωγή οποιουδήποτε θεματικού χάρτη. Είναι δηλαδή ένα σύστημα στο οποίο, διαθέτοντας μια οποιασδήποτε μορφής οπτικοποίηση ή αναπαράσταση του χώρου σε Η/Υ, σε κάθε σημείο του μπορεί να συνδεθεί οποιοσδήποτε, απεριόριστα θεωρητικά, αριθμός κάθε είδους περιγραφικών πληροφοριών, π.χ. δημογραφικών, οικονομικών, κοινωνικών κ.λπ. Αντίστροφα έχοντας κάποιο δεδομένο, μια οποιαδήποτε πληροφορία, μπορεί να εντοπιστεί το σημείο ή τα σημεία από τα οποία προέρχεται (Υφαντής, Σαββίδης, 2004).

Η υποχρεωτική σύνδεση πληροφορίας - χώρου, με τη γεωγραφική θέση δηλαδή στην οποία η πληροφορία παράγεται ή αντιστοιχεί, διευρύνει και τη λειτουργία του συστήματος. Η παραγωγή της πληροφορίας γίνεται τοπικά, οπότε ίσως είναι πιο εφικτό να γίνεται η αποθήκευση και πρωτογενής επεξεργασία των πληροφοριών επίσης τοπικά. Αντί λοιπόν να επιδιώκεται η κεντρική συγκέντρωση της πληροφορίας, είναι καλύτερα να αποκεντρώνεται το δίκτυο εγκατεστημένων βάσεων δεδομένων. Με τον τρόπο αυτό, οι επί μέρους βάσεις δεν γιγαντώνονται, ενώ παράλληλα μπορούμε να θεωρήσουμε ότι σε πολλές περιπτώσεις η έστω πρωτογενής επεξεργασία ενδυναμώνεται από την εμπειρία και γνώση των τοπικών συνθηκών κάθε είδους (Υφαντής, Σαββίδης, 2004).

Στον ελληνικό χώρο οι προσπάθειες για ανάπτυξη Γ.Σ.Π. σε αστικές περιοχές έχουν εστιαστεί σε μεγάλο βαθμό στην δημιουργία συστημάτων που στόχος τους είναι να χρησιμοποιούνται για πολεοδομικές μελέτες καθώς και την ανάπτυξη κτηματολογίων, έκδοση πολεοδομικών αδειών, κ.λπ. (Papas, 1994). Χαρακτηριστικό των συστημάτων αυτών είναι η έμφαση στη μεγάλη γεωγραφική ακρίβεια των δεδομένων, καθώς λόγω της αξιών γης στον αστικό χώρο πιθανά λάθη έχουν σοβαρές χρηματικές επιπτώσεις, και σε μεταβλητές όπως οι χρήσεις γης, το ιδιοκτησιακό καθεστώς κλπ. Αν και τα συστήματα αυτά είναι πλήρως επαρκή για τους σκοπούς που έχουν αναπτυχθεί έχουν το πρόβλημα ότι δύσκολα μπορούν να χρησιμοποιηθούν για θέματα στρατηγικού σχεδιασμού στα οποία η απαιτούμενη πληροφορία είναι διαφορετική και οι εφαρμογές σχετίζονται με θέματα λήψης αποφάσεων (Διαμαντάκης, Πραστάκος, 1999).

3.2 Γεωγραφικά δεδομένα:

Τα Γεωγραφικά δεδομένα «αποτελούν μια ειδική κατηγορία δεδομένων, τα οποία κατανέμονται στο χώρο και μεταβάλλονται στο χρόνο». Για την περιγραφή της πραγματικότητας γίνεται η θεώρηση ότι αυτή συντίθεται από ένα σύνολο διακριτών αλλά παράλληλα αλληλοσυσχετιζόμενων μονάδων, που καλούνται οντότητες. Οντότητα ορίζεται κάθε μονάδα ή αντικείμενο με φυσική ή εννοιολογική υπόσταση» (Στεφανάκης, 2003).

Οι οντότητες συχνά κατατάσσονται σε τέσσερις κατηγορίες, εκείνες των φυσικών αντικειμένων, των διοικητικών μονάδων, των γεωγραφικών φαινομένων και των παραγόμενων πληροφοριών

Η γεωγραφική οντότητα περιέχει χωρικές πληροφορίες, τις γεωγραφικές της συντεταγμένες, και περιγραφικές πληροφορίες οι οποίες αφορούν στη φύση της οντότητας. Η αποθήκευσή τους στον ηλεκτρονικό υπολογιστή γίνεται σε διάφορες μορφές. Στις παρακάτω υποενότητες γίνεται αναφορά στην επιστήμη της χαρτογραφίας, τα προβολικά συστήματα τα συστήματα συντεταγμένων και στους τύπους αποθήκευσης των γεωγραφικών δεδομένων στον ηλεκτρονικό υπολογιστή.

3.3 Αναπαράσταση (Δομές) χαρτογραφικών δεδομένων

Τα αντικείμενα του πραγματικού κόσμου αναπαρίστανται σε ένα σύστημα γεωγραφικών δεδομένων με δυο μορφές, αυτές είναι η κανικοποιημένη μορφή (Raster) και η διανυσματική (Vector)

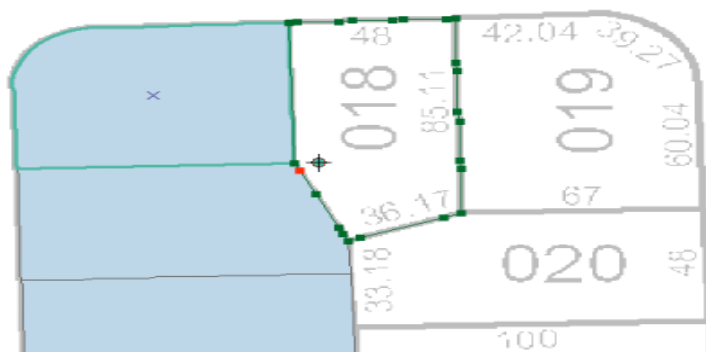
Οι δύο αυτές δομές διαχειρίζονται γεωμετρική πληροφορία, η οποία αναφέρεται στις μετρητικές (metrics) και στις τοπολογικές (topology) ιδιότητες των στοιχείων

3.3.1 Κανικοποιημένα (RASTER) Δεδομένα



Η κανικοποιημένη δομή δεδομένων (Raster) χρησιμοποιεί ένα πλέγμα Φατνίων (pixels) για να περιγράψει το αντικείμενο. Το κάθε φατνίο ορίζεται από την γραμμή και την στήλη στην οποία βρίσκεται (i,j) ως προς ένα σύστημα αναφοράς. Ο συνδυασμός των φατνίων δημιουργεί την εικόνα. Κάθε φατνίο έχει μια τιμή και μέγεθος το οποίο καθορίζει την ακρίβεια και την λεπτομέρεια του απεικονιζόμενου αντικειμένου. Τα κανικοποιημένα δεδομένα αποθηκεύονται συνήθως σε μορφή JPEG, TIF.

3.3.2 Διανυσματικά (Vector) δεδομένα



Η δομή αυτή δεδομένων χρησιμοποιεί διανύσματα για την αποθήκευση των γεωγραφικών/χαρτογραφικών δεδομένων. Κάθε διάνυσμα αποτελείται από ένα ζεύγος συντεταγμένων. Τα δεδομένα σε αυτή την μορφή αναπαρίστανται ως σημεία-γραμμές-πολύγωνα.

- Σημεία

Τα σημεία, χρησιμοποιούνται για γεωγραφικά στοιχεία που μπορούν να αναφερθούν μόνο με την θέση τους. Λ.χ., σημεία ενδιαφέροντος, γεωτρήσεις κ.λ.π.

Ανάλογα με την κλίμακα μπορούν να χρησιμοποιηθούν για να αναπαραστήσουν και επιφάνειες. Λόγου χάρη μια πόλη σε μια μεγάλη κλίμακα αναπαρίσταται ως πολύγωνο αλλά σε ένα χάρτη μικρότερης κλίμακας (λ.χ παγκόσμιος χάρτης) αναπαρίσταται ως σημείο στον χάρτη αυτόν. Τα σημεία δεν έχουν εμβαδόν.

- Γραμμές

Οι γραμμές χρησιμοποιούνται για τη περιγραφή γραμμικών στοιχείων όπως δρόμοι, λίμνες κ.λ.π.. Στις γραμμές μπορούμε να μετρήσουμε μήκος.

- Πολύγωνα

Τα δισδιάστατα πολύγωνα χρησιμοποιούνται για την απεικόνιση γεωγραφικών στοιχείων που καταλαμβάνουν κάποια έκταση, όπως λίμνες, πόλεις, χρήσεις γης κ.λ.π.. Στα πολύγωνα μπορούμε να μετρήσουμε περίμετρο και εμβαδόν.

Κάθε ένα vector στοιχείο μπορεί να σχετίζεται με μία εγγραφή σε μια βάση δεδομένων, όπου εκτός από τη γεωμετρία του μπορεί να περιγράφονται τα χαρακτηριστικά του. Λ.χ ένα διανυσματικό αρχείο χρήσεων γης μπορεί να περιέχει πληροφορία για την πυκνότητα του πληθυσμού, πολεοδομικά στοιχεία κ.ο.κ. Οι περιγραφικές αυτές πληροφορίες που περιέχονται στο αρχείο μπορούν να χρησιμοποιηθούν για την χρωματική διαβάθμιση του θεματικού επιπέδου λ.χ

πορτοκαλί όπου οι χρήσεις γης είναι αμιγείς, κόκκινο όπου δεν επιτρέπεται η δόμηση κ.ο.κ.

3.3.3 Πλεονεκτήματα - μειονεκτήματα *Raster /Vector*

VECTOR:

Πλεονεκτήματα

- Οι οντότητες αναπαρίστανται στην αρχική ανάλυση
- Επιτρέπεται η δημιουργία τοπολογίας
- Έχουν μεγαλύτερες δυνατότητες ανάλυσης
- Είναι πιο ευέλικτα στον χειρισμό (εστίαση, αποκοπή κ.λ.π)

Μειονεκτήματα

- Η θέση του κάθε διανύσματος αποθηκεύεται ξεχωριστά

RASTER:

Πλεονεκτήματα

- Εύκολη παραγωγή, η θέση του κάθε φατνίου δίνει αυτόματα και τη θέση της οντότητας, εύκολη ποσοτική ανάλυση,
- Εύκολη η λειτουργία της επικάλυψης
- Ιδανικά για συνεχή δεδομένα

Μειονεκτήματα

- Μεγαλύτερος όγκος από τα vector
- Εξαρτώνται από το μέγεθος του pixel επηρεάζοντας έτσι τις διαδικασίες ανάλυσης
- Ακατάλληλα για γραμμικές οντότητες

3.3.4 Μη χωρικά δεδομένα

Τα μη χωρικά δεδομένα που συνδέονται με κάποιο γεωμετρικό στοιχείο ή κάποιο raster κελί, μπορούν να αποθηκεύονται σε πίνακες Βάσεων Δεδομένων. Για τα μεν vector κάθε εγγραφή μπορεί να περιέχει τα χωρικά και τα μη χωρικά δεδομένα στα απαραίτητα πεδία. Στα raster δεδομένα, η μη χωρική πληροφορία μπορεί να αποθηκεύεται σαν τιμή του κελιού, ή όταν αυτό δεν επαρκεί, η τιμή του κελιού μπορεί να χρησιμοποιηθεί σαν δείκτης που το συσχετίζει με μία εγγραφή κάποιου πίνακα σε μία Βάση Δεδομένων.

3.4 Συστήματα συντεταγμένων

3.4.1 Γεωδαισία:

Γεωδαισία είναι η γεωεπιστήμη που ασχολείται με την μέτρηση και την αναπαράσταση ολόκληρης της γήινης επιφάνειας ή μεγάλου μέρους της (ανώτερη γεωδαισία) ή σχετικά μικρών τμημάτων της, τα οποία θεωρούνται επίπεδα (τοπογραφία). Οι μετρήσεις αυτές πρέπει να αναφέρονται ως προς ένα σύστημα αναφοράς. Το σύστημα αυτό αναφοράς καλείται σύστημα συντεταγμένων.

Σύστημα συντεταγμένων είναι ένα πλαίσιο παραμέτρων μέσω του οποίου μπορούμε να προσδιορίσουμε την θέση ενός σχήματος. Υπάρχουν πολλά και μπορούν να επινοηθούν άπειρα συστήματα. Κάθε εικόνα λόγου χάρη έχει ένα σύστημα συντεταγμένων που ορίζεται από τις γραμμές και τις στήλες της. (μια εικόνα είναι ένας πίνακας)

Επιπροσθέτως και το γεωδαιτικό σύστημα αναφοράς μιας χώρας έχει ένα τουλάχιστον σύστημα συντεταγμένων (ελλειψοειδές, καρτεσιανό).

Τα συστήματα συντεταγμένων που χρησιμοποιούνται στην γεωδαισία είναι :

- Το **γεωγραφικό** σύστημα συντεταγμένων (ϕ, λ, h), το οποίο θεωρεί ότι η επιφάνεια της γης είναι σφαίρα. Βασίζεται στις γωνίες σε σχέση με τον ισημερινό και τον μεσημβρινό. Τα ύψη καθορίζονται σε σχέση με το μέσο επίπεδο της θάλασσας.
- Το **γεωκεντρικό** σύστημα συντεταγμένων (X, Y, Z), το οποίο είναι ένα ορθοκανονικό σύστημα συντεταγμένων αρχή του οποίου θεωρείται το κέντρο της γης. Οι συντεταγμένες του δεν έχουν φυσική σημασία.
- Το **Καρτεσιανό** σύστημα συντεταγμένων το οποίο χρησιμοποιεί ως επιφάνεια αναφοράς το επίπεδο.

3.4.2 Επιφάνειες Αναφοράς:

Σύμφωνα με τον Λιβιεράτο, χαρτογραφική απεικόνιση ή προβολή είναι «η συστηματική διαδικασία μεταφοράς του πλέγματος των μεσημβρινών και παραλλήλων, δηλαδή του γεωγραφικού καννάβου, σε ένα επίπεδο έτσι ώστε τα σημεία του καννάβου στη σφαίρα να βρίσκονται σε σχέση αμφοιμονοσήμαντης αντιστοιχίας με εκείνα του επιπέδου. Υπάρχει απειρία δυνατοτήτων απεικόνισης του καννάβου, των μεσημβρινών και παραλλήλων της Γης σε ένα επίπεδο. Η επιλογή του κατάλληλου επιπέδου εξαρτάται από το τι αναπτυσσόμενη επιφάνεια θα ελέγχουμε για την προβολή, από την θέση στο χώρο που θα έχει η αναπτυσσόμενη επιφάνεια σε σχέση με τον άξονα περιστροφής της Γης και από το ποιες γεωμετρικές ιδιότητες θέλουμε να έχει ο χάρτης» (Λιβιεράτος Ε, 1988).

Σφαιρική Γη:

Η αντίληψη ότι η γη είναι σφαιρική ξεκινά χιλιάδες χρόνια πριν. Ουσιαστικά πρόκειται για ένα απλοποιημένο μαθηματικό μοντέλο που προσεγγίζει με μικρότερη ακρίβεια την επιφάνεια της γης (από ότι ένα Ελλειψοειδές Εκ Περιστροφής). Η σφαίρα για μικρές σχετικά αποστάσεις μπορεί να προσομοιώσει το σχήμα της γης με ικανοποιητική ακρίβεια, ενώ συνηθίζεται να αποτελεί την επιφάνεια αναφοράς σε εφαρμογές ναυσιπλοΐας αλλά και χαρτογραφίας. Η μαθηματική έκφραση της θέσης των σημείων της επιφάνειας της σφαίρας ακτίνας R (ακτίνα της σφαιρικής γης $R = 6371 \text{ km}$), όταν το κέντρο του συστήματος αναφοράς είναι το κέντρο της σφαίρας είναι :

$$X^2 + Y^2 + Z^2 = R^2$$

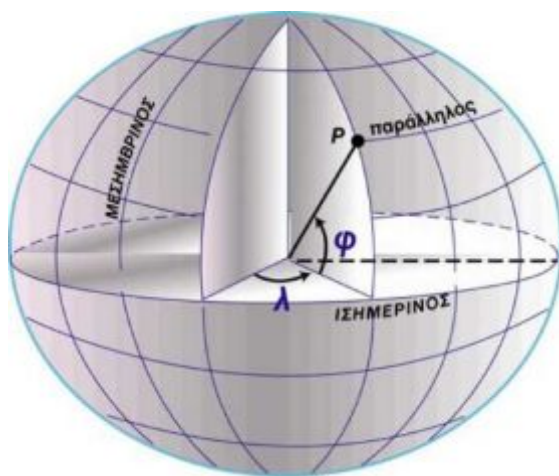
Οι υπολογισμοί στην επιφάνειά της γίνονται με τη βοήθεια των τύπων της σφαιρικής τριγωνομετρίας, που είναι σχετικά απλοί.

Αν η επιφάνεια αναφοράς ήταν το επίπεδο, τότε οι συντεταγμένες ενός σημείου θα ήταν η ευθεία που ένωνε το σημείο με την αρχή των αξόνων.

Όταν όμως η επιφάνεια αναφοράς είναι σφαίρα ή ΕΕΠ οι ευθείες γραμμές των αξόνων x και y γίνονται καμπύλες, ώστε να προσαρμοσθούν στην επιφάνεια αναφοράς. Οι καμπύλες αυτές γραμμές αποτελούν τους νέους άξονες

συντεταγμένων και είναι κάθετες μεταξύ τους σχηματίζονταν ένα πλέγμα μεσημβρινών και παραλλήλων.

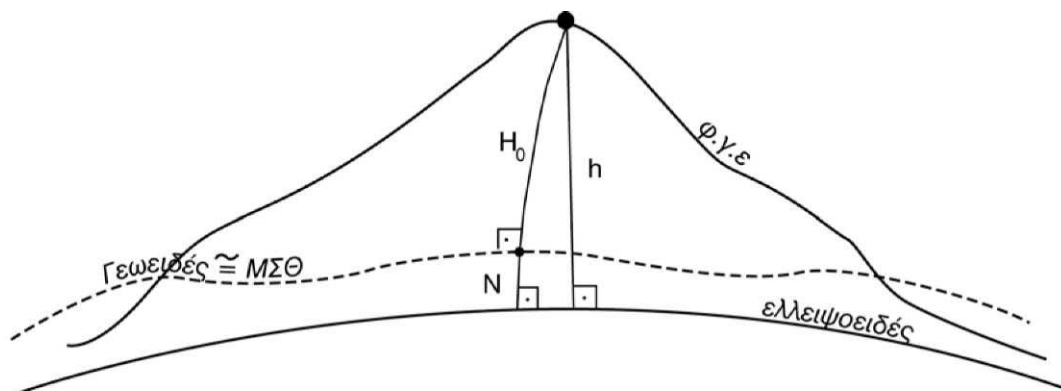
Μεσημβρινό καλούμε κάθε μέγιστο κύκλο στην επιφάνεια της σφαίρας που προκύπτει από την τομή της με ένα επίπεδο που περιέχει τον άξονα περιστροφής της. **Παράλληλο, ονομάζουμε** τον κύκλο που προέρχεται από την τομή της σφαίρας με ένα επίπεδο κάθετο στον άξονα περιστροφής της. Αν το επίπεδο αυτό περνά και από το κέντρο της σφαίρας (**ισημερινό επίπεδο**) ο παράλληλος αυτός που είναι μέγιστος κύκλος λέγεται **ισημερινός**.



Σχήμα 3.1. Δίκτυο μεσημβρινών και παραλλήλων και γεωγραφικές συντεταγμένες (ϕ, λ) στη σφαίρα

Γεωειδής Γη

Η πραγματική επιφάνεια της γης είναι το γεωειδές, το οποίο ορίζεται ως εκείνη η ισοδυναμική επιφάνεια του πραγματικού γήινου πεδίου βαρύτητας, που προσαρμόζεται όσο το δυνατόν καλύτερα στη μέση στάθμη των θαλασσών στο σύνολο της γης. Με προσέγγιση μέτρου το γεωειδές ταυτίζεται με τη μέση στάθμη της θάλασσας, στους ωκεανούς, η οποία θεωρείται ότι «επεκτείνεται» και κάτω από τις ηπείρους με φανταστικές σήραγγες. Επειδή οι υπολογισμοί συντεταγμένων καθίστανται εξαιρετικά περίπλοκοι έχοντας ως επιφάνεια αναφοράς το γεωειδές, συνήθως αυτό αντικαθίσταται από μια άλλη επιφάνεια, το Ελλειψοειδές Εκ Περιστροφής. Το Ελλειψοειδές Εκ Περιστροφής μπορεί να χρησιμεύσει ως μια απλούστερη επιφάνεια αναφοράς, στην οποία μπορούν να εκφραστούν γεωδαιτικές, χαρτογραφικές εργασίες, καθώς και το ίδιο το γεωειδές. Για το λόγο αυτό το ελλειψοειδές αυτό ονομάζεται και **ελλειψοειδές αναφοράς ή γήινο ελλειψοειδές**. Από την επιφάνεια του ελλειψοειδούς αυτού μετρώνται επίσης και τα γεωμετρικά υψόμετρα.



Σχήμα 3.2. Σχετικές θέσεις της μέσης στάθμης της θάλασσας του γεωειδούς και του ελλειψοειδούς αναφοράς

3.4.3 Χαρτογραφικές προβολές:

Όταν αναφερόμαστε στην χαρτογραφική προβολή εννοούμε την απεικόνιση της επιφάνεια αναφοράς (η οποία μπορεί να είναι είτε σφαίρα είτε ελλειψοειδές εκ περιστροφής) στο επίπεδο του χάρτη. Η απεικόνιση αυτή συνοδεύεται από παραμορφώσεις. Κάθε προβολή διατηρεί ορισμένες μόνο γεωμετρικές ιδιότητες των αντικειμένων που παρουσιάζονται στον χάρτη αναλλοίωτες, αλλά δεν υπάρχει χαρτογραφική προβολή που να διατηρεί όλες τις γεωμετρικές ιδιότητες αναλλοίωτες.

Ανάλογα με τις παραμορφώσεις που επιφέρουν σε γεωμετρικά μεγέθη, οι απεικονίσεις διακρίνονται σε: σύμμορφες, ισοδύναμες και αζιμουθιακές.

Σύμμορφες Προβολές:

Σύμμορφες ονομάζονται οι απεικονίσεις οι οποίες διατηρούν αναλλοίωτη την μορφή στοιχειωδών σχημάτων, διατηρούν δηλαδή το σχήμα των χωρικών αντικειμένων. Οι σύμμορφες απεικονίσεις χρησιμοποιούνται ευρέως για δημιουργία χαρτών πλοήγησης. Όπως προαναφέραμε καμία χαρτογραφική προβολή δεν διατηρεί αναλλοίωτες τις γεωμετρικές ιδιότητες των αντικειμένων,

Έτσι ενώ οι Σύμμορφες απεικονίσεις διατηρούν αναλλοίωτο, έχουν μεταβαλλόμενη κλίμακα, με αποτέλεσμα να εισάγονται σφάλματα στην μέτρηση εμβαδών.

Χαρακτηριστικές σύμμορφες απεικονίσεις είναι η μερκατορική προβολή αλλά προβολή Lambert. Τα κυριότερα χαρακτηριστικά της σύμφωνα με τον Robinson είναι:

- Οι μεσημβρινοί και ο ισημερινός, μαζί με τους παράλληλους κύκλους ή κύκλους πλάτους, όπως αλλιώς λέγονται, να αναπαρίστανται ως ένα γραμμικό δίκτυο παραλλήλων και καθέτων μεταξύ τους γραμμών.
- Τυχούσα λοξοδρομική καμπύλη στην επιφάνεια της γης να αναπαρίσταται ως ευθεία γραμμή στο χάρτη. Υπενθυμίζεται ότι λοξοδρομική καμπύλη είναι μία ευθεία πάνω στην επιφάνεια της γης η οποία ενώνει δύο σημεία της επιφάνειάς της

- Διατηρείται η ομοιότητα των στοιχειωδών σχημάτων της επιφάνειας της γης - δηλαδή η απεικόνιση είναι σύμμορφη.

Ισοδύναμες προβολές

Ισοδύναμες ονομάζονται οι απεικονίσεις που διατηρούν αναλλοίωτα τα εμβαδά. Εξαιτίας της ιδιότητάς τους αυτής χρησιμοποιούνται για δημιουργία παγκόσμιων χαρτών, αλλά και για την δημιουργία χαρτών γενικής χρήσης.

Χαρακτηριστικά παραδείγματα ισοδύναμων προβολών είναι η προβολή Bonne, Sanson, Mollweide.

Αζιμουθιακή προβολή (επίπεδη)

Χαρακτηριστικό αυτής της μεθόδου είναι ότι τα σημεία της σφαιρικής επιφάνειας της Γης προβάλλονται σε ένα επίπεδο (χάρτη) εφαπτόμενο σε ένα



Σχήμα 3.3 Αζιμουθιακή Προβολή

σημείο της επιφάνειας που θέλουμε να προβάλουμε. Σε αυτές τις προβολές οι μεσημβρινοί προβάλλονται ως ευθείες γραμμές.

Κεφάλαιο 4

Τεχνολογίες και εργαλεία

Ο παγκόσμιος ιστός

Παγκόσμιος ιστός είναι το δίκτυο των συνδεδεμένων υπολογιστών και δικτύων σε παγκόσμια κλίμακα. Οι υπολογιστές αυτοί για να μεταφέρουν πληροφορίες, χρησιμοποιούν μια συγκεκριμένη «γλώσσα», η οποία ονομάζεται πρωτόκολλο επικοινωνίας. Η γλώσσα αυτή για τον παγκόσμιο ιστό ονομάζεται http. Το πρωτόκολλο αυτό ορίζει τον τρόπο και τους κανόνες επικοινωνίας μεταξύ των υπολογιστών.

Για να επικοινωνήσουν δυο υπολογιστές μεταξύ τους ο ένας υπολογιστής πρέπει να υποβάλλει (αυτός ονομάζεται πελάτης/client) ένα αίτημα με την εντολή GET του πρωτοκόλλου http. Ο άλλος υπολογιστής (που ονομάζεται εξυπηρετητής/server) επεξεργάζεται το αίτημα του πρώτου και απαντάει μέσω της εντολής PUT του πρωτοκόλλου http.

Ο τρόπος αυτός επικοινωνίας αποτελεί την αρχιτεκτονική επικοινωνίας μεταξύ του πελάτη και του εξυπηρετητή. Απαραίτητο είναι ο πελάτης (client) να διαθέτει ένα λογισμικό που καλείται φυλλομετρητής (web browser). Πιο διαδεδομένοι φυλλομετρητές είναι ο MS internet explorer, ο Mozilla Firefox, ο Google Chrome κ.α.

Η γλώσσα HTML είναι μια ειδική γλώσσα που αναγνωρίζεται από όλους τους φυλλομετρητές και χρησιμοποιείται από τους φυλλομετρητές για να ερμηνεύσει το περιεχόμενο μιας ιστοσελίδας.

4.1 Η γλώσσα HTML και η δυνατότητα χαρτογράφησης

4.1.1 Html

Η γλώσσα HTML (**H**yper**T**ext **M**arkup **L**anguage) είναι ένας δηλωτικός τρόπος να πούμε στον φυλλομετρητή τι θέλουμε να εμφανίσει, είτε αυτό πρόκειται για κείμενο, εικόνες ή πολυμέσα. Επομένως οι ιστοσελίδες είναι γραμμένες σε αυτή την γλώσσα. Δεν είναι ολοκληρωμένη γλώσσα προγραμματισμού αλλά

γλώσσα περιγραφής ιδιοτήτων των στοιχείων που αποτελούν μία ιστοσελίδα μέσα από ένα σύνολο από ετικέτες (tags) που φωλιάζουν και μορφοποιούν το περιεχόμενο.

Τα κείμενα HTML δεν απαιτούν ειδικό λογισμικό για να συνταχθούν, αλλά μπορούν να δημιουργηθούν χρησιμοποιώντας έναν απλό κειμενογράφο.

Ένα κείμενο HTML πρέπει να έχει επέκταση .html ή .htm.

Όταν ένας χρήστης επισκέπτεται μια ιστοσελίδα τότε ο φυλλομετρητής που χρησιμοποιεί διαβάζει το κείμενο HTML και ανάλογα με τις ετικέτες που αυτό περιέχει μορφοποιεί την σελίδα και την εμφανίζει στην οθόνη του χρήστη.

4.1.2 Html και στατικοί χάρτες

Η δημοσιοποίηση χαρτογραφικού περιεχομένου μπορεί να γίνει με πολλούς τρόπους και με διαφορετικά επίπεδα διαδραστικότητας.Επομένως υπάρχουν στατικοί χάρτες, οι οποίοι μπορεί να είναι μια απλή εικόνα αλλά και πιο διαδραστικές χαρτογραφικές εφαρμογές.

Η HTML δίνει την δυνατότητα δημιουργίας στατικών χαρτών μέσω της εισαγωγής εικόνας σε μια ιστοσελίδα.

Η HTML δίνει την δυνατότητα για δημιουργία πιο διαδραστικών χαρτογραφικών εφαρμογών μέσω του ορισμού περιοχών στις εικόνες και την ανάθεση σε αυτών υπερσυνδέσμων σε άλλες σελίδες ή αρχεία.Οι περιοχές αυτές καλούνται ενεργές περιοχές και αντιστοιχούν σε απλούς γεωμετρικούς τύπους (παραλληλόγραμμο,κύκλος,πολύγωνο) και οι οποίοι εμφανίζονται στην οθόνη όταν ο χρήστης κάνει κλικ στην αντίστοιχη ενεργή περιοχή.

4.2 Html και διαδραστικοί χάρτες

Η δημοσιοποίηση διαδραστικών χαρτών απαιτεί την εγκατάσταση ειδικών εφαρμογών τόσο από τον πελάτη, όσο και από την εξυπηρετητή.Επομένως υπάρχουν τρεις αντιμετώπισεις, η επέκταση της λειτουργικότητας του πελάτη, η επέκταση της λειτουργικότητας του εξυπηρετητή καθώς και οι μικτές προσεγγίσεις.

4.2.1 • Επέκταση λειτουργικότητας του πελάτη

Ένας τρόπος επέκτασης της λειτουργικότητας του πελάτη (client) είναι η εγκατάσταση στον υπολογιστή του, ενός λογισμικού ,που ονομάζεται plugin και εμπλουτίζει τη λειτουργικότητα του φυλλομετρητή (web browser). Με αυτόν τον τρόπο ο φυλλομετρητής είναι σε θέση να αναγνωρίσει και άλλες μορφές αρχείων , λόγου χάρη αρχεία SVG, GML, KML κ.α ,τα οποία και προσθέτουν επιπρόσθετες δυνατότητες στον φυλλομετρητή ως προς την αξιοποίηση χαρτογραφικού περιεχομένου.Πέρα από τα επιπρόσθετα λογισμικά (plugins) έχουν δημιουργηθεί γλώσσες προγραμματισμού για το διαδίκτυο (π.χ Javascript, Java Applets) που δίνουν την δυνατότητα για δημιουργία πιο διαδραστικών χαρτογραφικών εφαρμογών.

4.2.2 • Επέκταση λειτουργικότητας του Εξυπηρετητή

Η διαδραστικότητα των χαρτών στον παγκόσμιο ιστό μπορεί να γίνεται και από την πλευρά του εξυπηρετητή.Στην περίπτωση αυτή η επεξεργασία γίνεται από τον εξυπηρετητή κατόπιν αιτήματος του πελάτη. Ο εξυπηρετητής επεξεργάζεται τα αιτήματα και αποστέλλει το αποτέλεσμα στον πελάτη.Το αποτέλεσμα αυτό αναλαμβάνει να οπτικοποιήσει στην οθόνη του χρήστη ο φυλλομετρητής που χρησιμοποιεί.Αυτή η διαδικασία συνήθως γίνεται μέσω ενός πρωτοκόλλου cgi (Common Gateway Interface)

4.2.3• Μικτές προσεγγίσεις

Μια άλλη προσέγγιση είναι η μικτή.Σε αυτή την περίπτωση γίνεται τόσο επέκταση της λειτουργικότητας του εξυπηρετητή όσο και του πελάτη, συνδιάζοντας τις τεχνολογίες που αναφέραμε πριν (CGI, Javascript).

Στην παρούσα εργασία έγινε επέκταση της λειτουργικότητας του πελάτη, χρησιμοποιώντας την γλώσσα Javascript όπως θα αναφερθεί και πιο αναλυτικά στο κεφάλαιο 6 της παρούσης.

4.3 Javascript

Η Javascript, είναι γλώσσα στο επίπεδο του χρήστη.

Ένα πρόγραμμα γραμμένο σε Javascript, περιλαμβάνεται ως τμήμα ενός εγγράφου HTML και δεν χρειάζεται μετάφραση για να διαβαστεί από τον φυλλομετρητή. Εξάλλου τα προγράμματα της Javascript δεν χρειάζονται ειδικό λογισμικό για να γραφτούν. Η Javascript χρησιμοποιεί αντικείμενα, αλλά δεν είναι καθαρά αντικειμενοστραφής γλώσσα προγραμματισμού, είναι μία γλώσσα βασισμένη σε αντικείμενα (object based), αλλά τα αντικείμενα στην Javascript χρησιμοποιούνται ως οντότητες που δεν έχουν τη δυναμική των πραγματικών αντικειμένων μιας αντικειμενοστραφούς γλώσσας.

Τα αντικείμενα μπορούν να θεωρηθούν ως σύνολα ιδιοτήτων .

Η Javascript χρησιμοποιεί την εξής σύνταξη για να περιγράψει κάποια ιδιότητα :

α <όνομα αντικειμένου>.<ιδιότητα>

Οι εντολές της περικλείονται μεταξύ των ετικετών:

<SCRIPT> </SCRIPT>

4.4. Υπηρεσίες χαρτογραφικού περιεχομένου στον παγκόσμιο

ιστό

Οι υπηρεσίες χαρτογραφικού περιεχομένου στον παγκόσμιο ιστό (web services for mapping) αφορούν την επικοινωνία και διαλειτουργικότητα μεταξύ των χαρτογραφικών εφαρμογών και την ανταλλαγή χαρτογραφικών δεδομένων μέσω του διαδικτύου. Οι υπηρεσίες αυτές είναι σχεδιασμένες έτσι ώστε να μπορούν να χρησιμοποιηθούν εύκολα από το χρήστη, χωρίς να χρειάζονται εξεδικευμένες γνώσεις..

Αναλυτικότερα, ο πελάτης (client) κάνει το αίτημα στον εξυπηρετητή (server) ο οποίος με τη σειρά του επιστρέφει το αποτέλεσμα. Για την διαδικασία αυτή, για την ανάκτηση των χαρτογραφικών δεδομένων , απαιτείται η γνώση της διεύθυνσης του εξυπηρετητή και μερικών βασικών παραμέτρων των δεδομένων (ονόματα θεματικών επιπέδων, formats κλπ.). Η αναζήτηση αυτών των χαρτογραφικών δεδομένων που έχει ζητήσει ο πελάτης, μπορεί πολλές

φορές να εμπλέκει και άλλους εξυπηρετητές για την διεκπεραίωση του αιτήματος.

Τα χαρτογραφικά αιτήματα του πελάτη, διατυπώνονται από αυτόν μέσα στην γραμμή διεύθυνσης URL και μέσα τους περιλαμβάνουν την διεύθυνση του εξυπηρετητή, την υπηρεσία καθώς και διάφορες παραμέτρους που έχει θέσει. Τα τυπικά αιτήματα που γίνονται από τον πελάτη προς τον εξυπηρετητή είναι:

- **Get Capabilities Διαθέσιμα δεδομένα**
- **GetMap Ένα στιγμιότυπο των δεδομένων**
- **GetFeature Δεδομένα που ικανοποιούν τις παραμέτρους που έχουν τεθεί**
- **GetFeatureInfo Τιμές γνωρισμάτων ενός αντικειμένου**

Ο OGC (Open Geospatial Consortium), που είναι μία διεθνής συμφωνία που έχει γίνει από οργανισμούς θέσπισης προτύπων, έχει προδιαγράψει τους βασικούς τρόπους διάθεσης γεωγραφικών και χαρτογραφικών δεδομένων στον Παγκόσμιο Ιστό με τυποποιημένες υπηρεσίες και πρότυπα όπως τα WMS, WFS, WCS, WMC, SLD, GML.

Στην παρούσα εργασία χρησιμοποιήθηκε η υπηρεσία WFS για την αποστολή διανυσματικών δεδομένων από τον εξυπηρετητή στον πελάτη.

4.5. Η γλώσσα *kml*

Η KML είναι γλώσσα περιγραφής χωρικών δεδομένων τριών διαστάσεων και η οποία βασίζεται στην γλώσσα XML. Τα αρχεία KML μπορούν να οπτικοποιηθούν στους χάρτες των φυλλομετρητών γης, όπως π.χ Google Earth και Google Maps. Τα αρχεία KML συχνά δημοσιοποιούνται στην δυαδική του μορφή KMZ για λόγους συμπίεσης.

Τα αρχεία KML περιγράφουν στην ουσία σε μορφή XML σημεία, γραμμές, πολύγωνα, τρισδιάστατα αντικείμενα και ετικέτες ονοματολογίας. Καθώς η KML είναι βασισμένη στην XML εφαρμόζει την δομή των φωλιασμένων στοιχείων με τις παραμέτρους του. Για κάθε στοιχείο σε ένα KML αρχείο περιέχονται οι συντεταγμένες του (στο WGS'84, το ύψος παρατήρησης, το σύμβολο που θα χρησιμοποιηθεί κ.α)

Ένα αρχείο KML δημιουργείται είτε χειρονακτικά σε έναν επεξεργαστή κειμένου (Notepad), είτε σε έναν επεξεργαστή κειμένων XML. Άλλος τρόπος δημιουργίας αρχείου KML, είναι μέσω της εφαρμογής Google Earth όπου σχεδιάζονται τα αντικείμενα με τα εργαλεία της διεπαφής και αποθηκεύονται σε μορφή KML ή KMZ. Τα αρχεία KML έχουν τύχει μεγάλης αποδοχής και πρόσφατα έχουν αποτελέσει τυποποίηση του OGC, ενώ υπάρχουν πάρα πολλά δωρεάν διαθέσιμα προγράμματα στο διαδίκτυο για την μετατροπή του σε άλλες μορφές αρχείων ή αντίστροφα.

4.6. Το εργαλείο Open Layers

Τα Open Layers είναι μία βιβλιοθήκη JavaScript για την οπτικοποίηση γεωγραφικών δεδομένων στον Παγκόσμιο Ιστό. Η βιβλιοθήκη των Open Layers είναι συμβατή με τους σύγχρονους φυλλομετρητές (web browsers) και είναι ανεξάρτητη από τον εξυπηρετητή. Τα openlayers παρέχουν ένα javascript api για την δημιουργία πλούσιων γεωγραφικών δεδομένων για το διαδίκτυο, αντίστοιχα με αυτά των Google Maps και MSN Virtual Earth, με την σημαντική διαφορά ότι τα Open Layers είναι ελεύθερο λογισμικό που αναπτύχθηκε για και από την κοινότητα ανοιχτού λογισμικού. Επιπλέον, είναι συμβατά με τις τυποποιήσεις του Open Geospatial Consortium, δηλαδή τα πρωτόκολλα Web Mapping Service (WMS) και Web Feature Service (WFS).

4.7 Διαδικτυακές υπηρεσίες (Web Services)

Πολλοί είναι οι ορισμοί που έχουν διατυπωθεί για τις υπηρεσίες διαδικτύου (web – services). Σύμφωνα με την IBM τα web services είναι «μια νέα γενιά εφαρμογών web. Είναι αυτοπεριγραφικές, ανεξάρτητες αρθρωτές εφαρμογές που μπορούν να δημοσιευτούν, να εντοπισθούν και να κληθούν από το web. Εκτελούν συναρτήσεις, που μπορεί να είναι οτιδήποτε από ένα απλό αίτημα έως μια περίπλοκη επιχειρησιακή διαδικασία. Μόλις μια υπηρεσία διαδικτύου αναπτυχθεί, άλλες εφαρμογές (και άλλες υπηρεσίες διαδικτύου) μπορούν να την αναζητήσουν και να την καλέσουν» (wikipedia.org).

Ουσιαστικά επιτρέπουν ένα πρόγραμμα σε έναν υπολογιστή να καλέσει μια λειτουργία σε έναν άλλον υπολογιστή χωρίς να παίζει ρόλο το λειτουργικό

σύστημα, η γλώσσα προγραμματισμού ή ο κατασκευαστής. Καλούνται μέσω του τυπικού HTTP πρωτοκόλλου μόνο που αντί να ζητούν σελίδα, ζητούν την κλήση μιας λειτουργίας χρησιμοποιώντας ένα πρωτόκολλο επικοινωνίας όπως τα “get ” ή “post”.

Υπηρεσίες διάχυσης χαρτογραφικού περιεχομένου

Οι δημοφιλέστερες υπηρεσίες διάχυσης χαρτογραφικού περιεχομένου αποτελούν οι τυποποιήσεις του Open Geospatial Consortium. Στην επίσημη ιστοσελίδα του, ο OGC αναφέρεται ως ένας διεθνής μη κερδοσκοπικός οργανισμός που ηγείται στην ανάπτυξη τυποποιήσεων για διαδικτυακές υπηρεσίες χωρικού περιεχομένου. Μέσω των προγραμμάτων του OGC έχουν αναπτυχθεί μία σειρά από εργαλεία για τη διάχυση γεωγραφικών δεδομένων στον Παγκόσμιο Ιστό.

Παρακάτω αναλύονται δύο σημαντικές διαδικτυακές υπηρεσίες αυτού του οργανισμού

WMS: Η υπηρεσία WMS (Web Map Service) αποτελεί μία μέθοδο διάχυσης χαρτών σε μορφή εικόνας (GIF, PNG κλπ) στον Παγκόσμιο Ιστό. Με το αίτημα GetMap ο πελάτης ζητάει έναν χάρτη από ένα εξυπηρετητή ο οποίος διαθέτει την συγκεκριμένη υπηρεσία.

WFS: Η υπηρεσία WFS (Web Feature Service) αποτελεί μία μέθοδο διάχυσης γεωγραφικών οντοτήτων στον Παγκόσμιο Ιστό. Σε αντίθεση με την υπηρεσία WMS, η οποία επιστρέφει χάρτες σε μορφή εικόνας, η υπηρεσία WFS επιστρέφει διανυσματικά δεδομένα σε μορφή GML.

4.7.1 Mash – ups

Ένα χαρτογραφικό Mash-up είναι μια ιστοσελίδα ή μια διαδικτυακή εφαρμογή που χρησιμοποιεί και συνδιάζει δεδομένα από διαφορετικές πηγές για την δημιουργία μιας νέα υπηρεσίας, με στόχο να τους προσδώσει μεγαλύτερη λειτουργικότητα.

Με την ραγδαία πρόοδο του παγκόσμιου ιστού πολλές είναι οι υπηρεσίες που συνδυάζουν δεδομένα από άλλες πηγές, ενώ διάφορες εταιρίες ή οργανισμοί

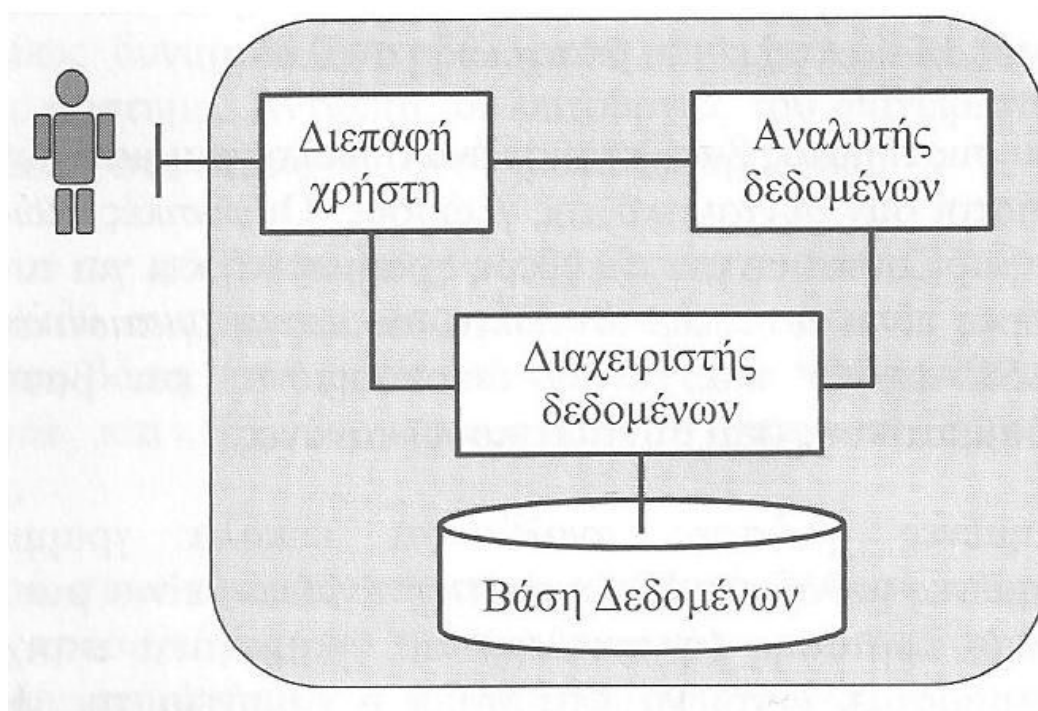
έχουν προχωρήσει στην ανάπτυξη api's κάνοντας πολύ εύκολη την ενσωμάτωση δεδομένων από άλλες πηγές (λ.χ google maps api).

4.7.2 Διαδικτυακά Γεωγραφικά Συστήματα Πληροφοριών

Η ραγδαία αύξηση της χρήσης του διαδικτύου επηρέασε τα Γεωγραφικά Συστήματα Πληροφοριών με αποτέλεσμα να δημιουργούνται συνεχώς εφαρμογές Γεωγραφικών Πληροφοριών που είναι προσβάσιμες από το διαδίκτυο. Οι εφαρμογές αυτές συνήθως συνδιάζουν δεδομένα από διάφορες πηγές, (λ.χ Google, Yahoo κ.α) δημιουργώντας χαρτογραφικά Mash – ups όπως περιγράφηκε προηγουμένως, αυξάνοντας τις λειτουργικότητά τους.

Αρχιτεκτονική εφαρμογών οπτικοποίησης Γεωγραφικών Δεδομένων

Ένα τυπικό πληροφοριακό σύστημα αποτελείται από 4 βασικά τμήματα, όπως προκύπτει και από το παρακάτω σχήμα.



Σχήμα 1: Αρχιτεκτονική τυπικού πληροφοριακού συστήματος
(Πηγή: Στεφανάκης Ε, 2005)

Σύμφωνα με το σχήμα, στο χαμηλότερο επίπεδο βρίσκεται η βάση δεδομένων, στην οποία είναι αποθηκευμένα όλα τα απαραίτητα δεδομένα. Η βάση δεδομένων επικοινωνεί με το διαχειριστή δεδομένων ο οποίος διαχειρίζεται τα δεδομένα. Ο αναλυτής δεδομένων υποδεικνύει στο διαχειριστή αναλυτικές λειτουργίες διαχείρισης των δεδομένων της βάσης με στόχο την εξαγωγή πληροφοριών. Τέλος η διεπαφή αποτελεί συνήθως ένα το μέσο επικοινωνίας του χρήστη με το σύστημα, μεταφέροντας στην εφαρμογή τις εντολές του χρήστη και τα αποτελέσματα της επεξεργασίας των εντολών του πάλι στο χρήστη.

Αρχιτεκτονική διαδικτυακού γεωγραφικού πληροφοριακού συστήματος

Η αρχιτεκτονική ενός διαδικτυακού συστήματος γεωγραφικών πληροφοριών φέρεται στο παρακάτω σχήμα.



Σχήμα 2: αρχιτεκτονική ενός διαδικτυακού συστήματος γεωγραφικών πληροφοριών (Στεφανάκης,2009)

Όπως φαίνεται και στο σχήμα κύριο χαρακτηριστικό των διαδικτυακών εφαρμογών αποτελεί ο συνδιασμός δεδομένων (στο σχήμα φαίνεται μόνο οι χάρτες της Google, αλλά συνδιαστικά θα μπορούσε να είναι από διάφορες πηγές). Ο πελάτης στέλνει ερωτήματα GET και παίρνει απάντηση PUT μέσω του πρωτοκόλλου HTTP. Ο εξυπηρετητής μπορεί να λαμβάνει δεδομένα που είναι αποθηκευμένα τοπικά ή σε άλλον απομακρυσμένο εξυπηρετητή.

Χαρακτηριστικό παράδειγμα τέτοιων περιπτώσεων απομακρυσμένου εξυπηρετητή είναι οι εξυπηρετητές των τριών δημοφιλών χαρτογραφικών υποβάθρων Google maps, Bing aps, Yahoo maps όπου μπορούν να χρησιμοποιηθούν από εκατομμύρια χρήστες ταυτόχρονα .

4.7.3 Google Elevation API

Εισαγωγή

Το Google Elevation API είναι μια υπηρεσία που προσφέρει την δυνατότητα αναζήτησης υψομέτρων για κάθε σημείο στην επιφάνεια της γης. Επιπροσθέτως ο χρήστης μπορεί να αναζητήσει υψομέτρα κατά μήκος διαδρομής, δίνοντας την δυνατότητα υπολογισμού υψομετρικών διαφορών κατά μήκος διαδρομών.

Τι μπορούμε να κάνουμε με το Google Elevation API

Με την χρήση του Google Elevation API μπορούμε να πάρουμε υψομετρικά δεδομένα για την επιφάνεια της γης, συμπεριλαμβανομένου και του βάθους των ωκεανών (τα οποία επιστρέφονται ως αρνητικές τιμές). Στις περιπτώσεις

που η Google δεν έχει στην κατοχή της ακριβή υψομετρικά δεδομένα στην συγκεκριμένη περιοχή που ζητήθηκε, η υπηρεσία (Google Elevation API) θα εφαρμόσει παρεμβολή και θα επιστρέψει την μέση τιμή του υψομέτρου χρησιμοποιώντας τα υψόμετρα από τις τέσσερις κοντινότερες περιοχές.

Με την χρήση του Google Elevation API ένας χρήστης μπορεί να αναπτύξει εφαρμογές για πεζοπορία και ορειβασία, εφαρμογές για κινητά τηλέφωνα που χρειάζονται υψομετρική πληροφορία ή χαμηλής ακρίβειας τοπογραφικές εργασίες.

Η πρόσβαση στις υπηρεσίες του Google Elevation API επιτυγχάνεται μέσω μίας σύνδεσης HTTP.

Περιορισμοί χρήσης

Η χρήση του Google Elevation API υπόκειται σε περιορισμό 2,500 υψομετρικών αιτημάτων ανά ημέρα. Η εμπορική έκδοση της υπηρεσίας υπόκειται σε περιορισμό 100,000 υψομετρικών αιτημάτων ανά ημέρα. Σε κάθε υψομετρικό αίτημα ο χρήστης μπορεί να ζητήσει υψομετρική πληροφορία μέχρι και 512 περιοχών, αλλά δεν πρέπει να ξεπεράσει τις 25000 περιοχές ανά ημέρα. Το όριο αυτό υπάρχει για να μην υπάρχει κατάχρηση της υπηρεσίας. Αν ξεπεραστεί το συγκεκριμένο όριο ανά ημέρα η υπηρεσία απλώς σταματά να δουλεύει και ενεργοποιείται την επόμενη ημέρα. Οι διευθύνσεις URLs του Elevation API περιορίζονται στους 2048 χαρακτήρες, πριν την κωδικοποίηση του URL.

Αιτήματα για υψόμετρο

Το Google Elevation API επιστρέφει υψομετρικά δεδομένα για τοποθεσίες επάνω στην γη. Μπορούμε να ορίσουμε τις τοποθεσίες για τις οποίες ζητάμε υψόμετρα με έναν από τους παρακάτω δυο τρόπους :

- Σαν ένα σύνολο από ένα ή περισσότερα σημεία
- Σαν ένα σύνολο συνδεδεμένων σημείων κατά μήκος μιας διαδρομής.

Κάθε ένας από αυτούς τους δυο τρόπους χρησιμοποιεί το γεωγραφικό μήκος/πλάτος, τις γεωγραφικές συντεταγμένες δηλαδή για την ανίχνευση των περιοχών.

Μια διεύθυνση URL του Google Elevation API πρέπει να έχει την ακόλουθη μορφή:

<http://Maps.googleapis.com/Maps/API/Elevation/outputFormat?Parameters>

Μορφή Αποτελεσμάτων

Το Elevation API προς το παρόν υποστηρίζει τις ακόλουθες μορφές εξόδου:

json επιστρέφει αποτελέσματα σε JavaScript Object Notation (JSON).

xml επιστρέφει αποτελέσματα σε XML.

Χρήση παραμέτρων

Τα αιτήματα προς το Elevation API χρησιμοποιούν διαφορετικές παραμέτρους βασισμένες στο αν το αίτημα είναι για διακριτές περιοχές ή για ταξινομημένη διαδρομή. Για τις διακριτές περιοχές, τα αιτήματα για υψόμετρα επιστρέφουν πληροφορίες στις συγκεκριμένες περιοχές που εισήχθησαν στο αίτημα. Για τις διαδρομές, τα αιτήματα υψομέτρων λαμβάνονται δειγματοληπτικά κατά μήκος της δοσμένης διαδρομής.

Όπως συμβαίνει σε όλες τις URL διευθύνσεις, οι παράμετροι χωρίζονται με τη χρήση του συμβόλου (&) . Η λίστα των παραμέτρων και οι πιθανές τους τιμές δηλώνονται παρακάτω:

Αιτήματα Περιοχής (Positional Requests):

- `locations` (απαιτείται) καθορίζει την περιοχή(ές) στη γη από τις οποίες θα επιστραφούν οι υψομετρικές πληροφορίες. Αυτή η παράμετρος δέχεται είτε μία τοποθεσία ως ένα ζευγάρι γεωγραφικές συντεταγμένες είτε πολλαπλά ζεύγη γεωγραφικών συντεταγμένων που έχουν εισαχθεί ως πίνακας ή ως κωδικοποιημένη πολλαπλή γραμμή (polyline).

Δειγματοληπτικά Αιτήματα Διαδρομής (Sampled Path Requests):

- `path` (απαιτείται) καθορίζει μία διαδρομή επάνω στη γη για την οποία θα επιστραφούν υψομετρικές πληροφορίες. Αυτή η παράμετρος καθορίζει ένα σύνολο δύο ή περισσότερων διατεταγμένων ζευγαριών (γεωγραφικό μήκος, γεωγραφικό πλάτος) ορίζοντας μία διαδρομή κατά μήκος της επιφάνειας της γης.
- `samples` (απαιτείται) προσδιορίζει τον αριθμό των δειγματοληπτικών σημείων κατά μήκος μιας διαδρομής για τα οποία επιστρέφεται η υψομετρική πληροφορία.

Παράμετροι Αναφοράς (Reporting Parameters):

- `sensor` (απαιτείται) προσδιορίζει εάν τα δεδομένα της εφαρμογής χρησιμοποιούν αισθητήρα για τον καθορισμό της θέσης του χρήστη. Οι εφαρμογές που καθορίζουν τη θέση του χρήστη μέσω αισθητήρα πρέπει να εισάγουν `sensor=true` στην κλήση της URL διεύθυνσης του Elevation API. Εάν η εφαρμογή δε χρησιμοποιεί αισθητήρα ο χρήστης εισάγει `sensor=false`.

Elevation Responses

Για κάθε έγκυρο αίτημα, η υπηρεσία Υψομέτρων (Elevation Service) θα επιστρέφει μία απάντηση στη μορφή που δίνεται στη διεύθυνση URL του αιτήματος. Κάθε απάντηση θα περιέχει τα ακόλουθα στοιχεία:

- έναν κώδικα `status`, το οποίο μπορεί να είναι ένα από τα παρακάτω:
 1. OK δείχνοντας ότι η κλήση API ήταν επιτυχής
 2. INVALID_REQUEST δείχνοντας ότι η κλήση API ήταν αλλοιωμένη
 3. OVER_QUERY_LIMIT δείχνοντας ότι το πρόγραμμα αιτημάτων έχει υπερβεί το όριο χρήσης.
 4. REQUEST_DENIED δείχνοντας ότι το API δεν ολοκλήρωσε τη διαδικασία κλήσης.

5. UNKNOWN_ERROR δείχνοντας άγνωστης μορφής σφάλμα.

- έναν πίνακα αποτελεσμάτων results που περιλαμβάνει τα ακόλουθα στοιχεία:

1. Ένα στοιχείο location (που περιέχει lat και lng στοιχεία) της θέσης για την οποία υπολογίζονται οι υψομετρικές πληροφορίες.

2. Ένα στοιχείο elevation το οποίο δείχνει το υψόμετρο της περιοχής σε μέτρα.

3. Ένα στοιχείο resolution που δείχνει την μέγιστη απόσταση μεταξύ σημείων που υλοποιήθηκε παρεμβολή για τον υπολογισμό των υψομέτρων, σε μέτρα.

Παραδείγματα κλήσης της υπηρεσίας Elevation API

Παράδειγμα υπολογισμού σημειακών υψομέτρων

Το ακόλουθο παράδειγμα ζητάει το υψόμετρο σε μορφή JSON.

```
var elevator;
var map;
var infowindow = new google.maps.InfoWindow();
var denali = new google.maps.LatLng(63.3333333, -150.5);

function initialize() {
    var mapOptions = {
        zoom: 8,
        center: denali,
        mapTypeId: 'terrain'
    }
    map = new google.maps.Map(document.getElementById("map_canvas"),
    mapOptions);
```

```

// Create an ElevationService
elevator = new google.maps.ElevationService();

// Add a listener for the click event and call getElevation on that
location
google.maps.event.addListener(map, 'click', getElevation);
}

function getElevation(event) {

    var locations = [];

    // Retrieve the clicked location and push it on the array
    var clickedLocation = event.latLng;
    locations.push(clickedLocation);

    // Create a LocationElevationRequest object using the array's one
value
    var positionalRequest = {
        'locations': locations
    }

    // Initiate the location request
    elevator.getElevationForLocations(positionalRequest,
function(results, status) {
    if (status == google.maps.ElevationStatus.OK) {

        // Retrieve the first result
        if (results[0]) {

            // Open an info window indicating the elevation at the
clicked position
            infowindow.setContent("The elevation at this point
is " + results[0].elevation + " meters.");
            infowindow.setPosition(clickedLocation);
            infowindow.open(map);
        } else {
            alert("No results found");
        }
    } else {
        alert("Elevation service failed due to: " + status);
    }
    });
}
}

```

Η ακόλουθη εικόνα δείχνει πως απεικονίζονται οι παραπάνω πληροφορίες σε χάρτη.



Παράδειγμα υπολογισμού υψομέτρων διαδρομής

Το παράδειγμα που ακολουθεί ζητάει υψομετρική πληροφορία κατά μήκος μίας ευθείας γραμμικής διαδρομής από το Mt. Whitney της Καλιφόρνια στο Badwater της Καλιφόρνια, το χαμηλότερο και υψηλότερο σημείο στις ηπειρωτικές ΗΠΑ. Ζητούνται τρία δείγματα ώστε να περιληφθούν τα δύο ακραία σημεία και ένα ενδιάμεσο.

```
var elevator;
var map;
var chart;
var infowindow = new google.maps.InfoWindow();
var polyline;

// The following path marks a general path from Mt.
// Whitney, the highest point in the continental United
// States to Badwater, Death Vallet, the lowest point.
var whitney = new google.maps.LatLng(36.578581, -118.291994);
var lonepine = new google.maps.LatLng(36.606111, -118.062778);
var owenslake = new google.maps.LatLng(36.433269, -117.950916);
var beattyjunction = new google.maps.LatLng(36.588056, -116.943056);
var panamintsprings = new google.maps.LatLng(36.339722, -117.467778);
var badwater = new google.maps.LatLng(36.23998, -116.83171);

// Load the Visualization API and the columnchart package.
google.load("visualization", "1", {packages: ["columnchart"]});

function initialize() {
  var mapOptions = {
    zoom: 8,
    center: lonepine,
```

```

        mapTypeId: 'terrain'
    }
    map = new google.maps.Map(document.getElementById("map_canvas"),
    mapOptions);

    // Create an ElevationService.
    elevator = new google.maps.ElevationService();

    // Draw the path, using the Visualization API and the Elevation
    service.
    drawPath();
}

function drawPath() {

    // Create a new chart in the elevation_chart DIV.
    chart = new
    google.visualization.ColumnChart(document.getElementById('elevation_c
    hart'));

    var path = [ whitney, lonepine, owenslake, panamintsprings,
    beattyjunction, badwater];

    // Create a PathElevationRequest object using this array.
    // Ask for 256 samples along that path.
    var pathRequest = {
        'path': path,
        'samples': 256
    }

    // Initiate the path request.
    elevator.getElevationAlongPath(pathRequest, plotElevation);
}

// Takes an array of ElevationResult objects, draws the path on the
map
// and plots the elevation profile on a Visualization API
ColumnChart.
function plotElevation(results, status) {
    if (status == google.maps.ElevationStatus.OK) {
        elevations = results;

        // Extract the elevation samples from the returned results
        // and store them in an array of LatLngs.
        var elevationPath = [];
        for (var i = 0; i < results.length; i++) {
            elevationPath.push(elevations[i].location);
        }

        // Display a polyline of the elevation path.
        var pathOptions = {
            path: elevationPath,
            strokeColor: '#0000CC',
            opacity: 0.4,
            map: map
        }
        polyline = new google.maps.Polyline(pathOptions);

        // Extract the data from which to populate the chart.
        // Because the samples are equidistant, the 'Sample'
        // column here does double duty as distance along the

```

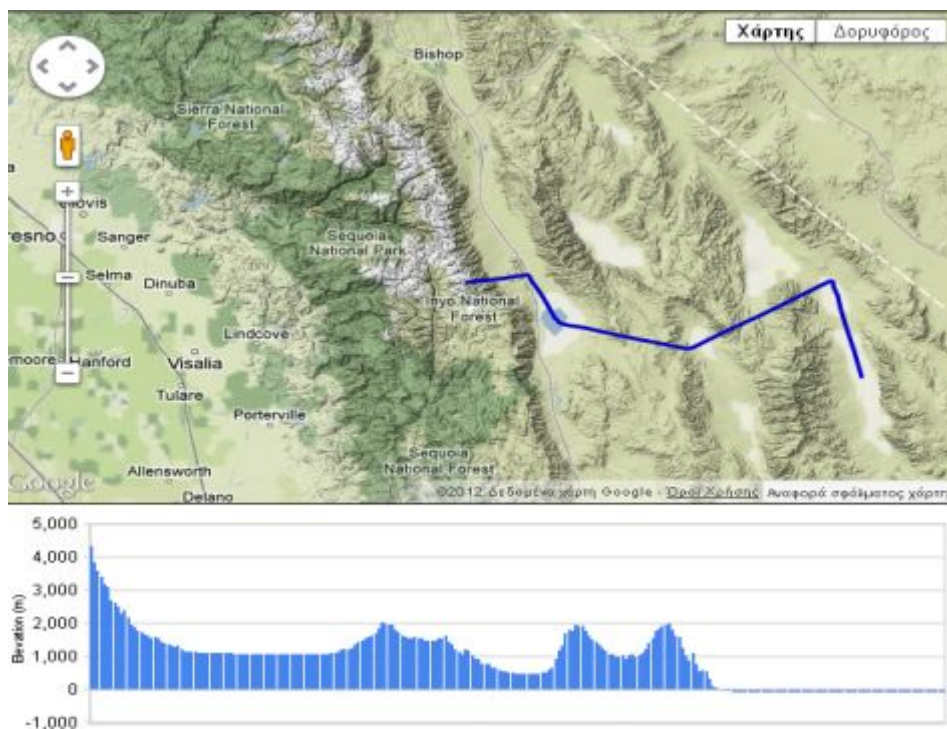
```

// X axis.
var data = new google.visualization.DataTable();
data.addColumn('string', 'Sample');
data.addColumn('number', 'Elevation');
for (var i = 0; i < results.length; i++) {
    data.addRow(['', elevations[i].elevation]);
}

// Draw the chart using the data within its DIV.
document.getElementById('elevation_chart').style.display =
'block';
chart.draw(data, {
    width: 640,
    height: 200,
    legend: 'none',
    titleY: 'Elevation (m)'
});
}
}

```

Η ακόλουθη εικόνα δείχνει πως απεικονίζονται οι παραπάνω πληροφορίες σε χάρτη.



4.7.4 Google Chart Tools

Τα Google Chart Tools παρέχουν έναν πολύ εύχρηστο τρόπο οπτικοποίησης δεδομένων σε ιστοσελίδα. Από απλά γραμμικά διαγράμματα μέχρι ιεραρχικά δένδροειδή διαγράμματα, η βιβλιοθήκη γραφημάτων παρέχει ένα μεγάλο αριθμό τύπων διαγραμμάτων.

Βιβλιοθήκη διαγραμμάτων

Τα διαγράμματα είναι ουσιαστικά κώδικας JavaScript και υπάρχουν πολλές επιλογές διαγραμμάτων προς επιλογή. Πέρα από την προκαθορισμένη εμφάνιση υπάρχει η δυνατότητα παραμετροποίησης διαγραμμάτων. Τα διαγράμματα είναι αλληλεπιδραστικά με άλλα στοιχεία της σελίδας. Τα διαγράμματα δημιουργούνται με την τεχνολογία HTML5/SVG ώστε να είναι συμβατά με διάφορους περιηγητές ιστού αλλά και σε διάφορες συσκευές.

Παραδείγματα κλήσης της υπηρεσίας Google Chart Tools

Google Chart Tools

```
<html>
  <head>
    <title>
      Cinematics
    </title>
    <style type="text/css">
      .header {
        color: purple;
        background-color: #abc;
        font-size: 40px;
        text-align: center;
      }
    </style>
    <script type="text/javascript"
src="http://www.google.com/jsapi"></script>
    <!-- Note that you need to replace the key ABCDEF with your Maps
API key. -->
    <!-- Sign up for a Maps API key here:
http://code.google.com/apis/maps/signup.html -->
    <script
src="http://maps.google.com/maps?file=api&v=2&api=key=ABCDEF"
type="text/javascript"></script>
    <script type="text/javascript">
      google.load('visualization', '1',
        {'packages': ['table', 'map', 'corechart']});
      google.setOnLoadCallback(initialize);

      function initialize() {
        // The URL here is the URL of the spreadsheet.
        // This is where the data is.
        var query = new google.visualization.Query(
```

```

        'https://spreadsheets.google.com/pub?key=pCQbetd-
CptF0r8qmCOLZGg');
        query.send(draw);
    }

    function draw(response) {
        if (response.isError()) {
            alert('Error in query');
        }

        var ticketsData = response.getDataTable();
        var chart = new google.visualization.ColumnChart(
            document.getElementById('chart_div'));
        chart.draw(ticketsData, {'isStacked': true, 'legend':
'bottom',
        'vAxis': {'title': 'Number of tickets'}});

        var geoData = new google.visualization.DataTable();
        geoData.addColumn('string', 'City');
        geoData.addColumn('string', 'Name');
        geoData.addColumn('boolean', 'Food');
        geoData.addRows(3);
        geoData.setCell(0, 0, 'London');
        geoData.setCell(1, 0, 'Paris');
        geoData.setCell(2, 0, 'Moscow');
        geoData.setCell(0, 1, 'Cinematics London');
        geoData.setCell(1, 1, 'Cinematics Paris');
        geoData.setCell(2, 1, 'Cinematics Moscow');
        geoData.setCell(0, 2, true);
        geoData.setCell(1, 2, true);
        geoData.setCell(2, 2, false);

        var geoView = new google.visualization.DataView(geoData);
        geoView.setColumns([0, 1]);

        var table =
            new
google.visualization.Table(document.getElementById('table_div'));
        table.draw(geoData, {showRowNumber: false});

        var map =
            new
google.visualization.Map(document.getElementById('map_div'));
        map.draw(geoView, {showTip: true});

        // Set a 'select' event listener for the table.
        // When the table is selected,
        // we set the selection on the map.
        google.visualization.events.addListener(table, 'select',
            function() {
                map.setSelection(table.getSelection());
            });

        // Set a 'select' event listener for the map.
        // When the map is selected,
        // we set the selection on the table.
        google.visualization.events.addListener(map, 'select',
            function() {
                table.setSelection(map.getSelection());
            });
    }
}

```

```

</script>
</head>

<body>
  <div class="header">Cinematics</div>

  <table align="center">
    <tr valign="top">
      <td style="width: 50%;">
        <div id="map_div" style="width: 400px; height: 300;"></div>
      </td>
      <td style="width: 50%;">
        <div id="table_div"></div>
      </td>
    </tr>
    <tr>
      <td colspan=2>
        <div id="chart_div" style="align: center; width: 700px; height: 300px;"></div>
      </td>
    </tr>
  </table>

</body>
</html>

```

Η ακόλουθη εικόνα δείχνει πως απεικονίζονται οι παραπάνω πληροφορίες σε χάρτη

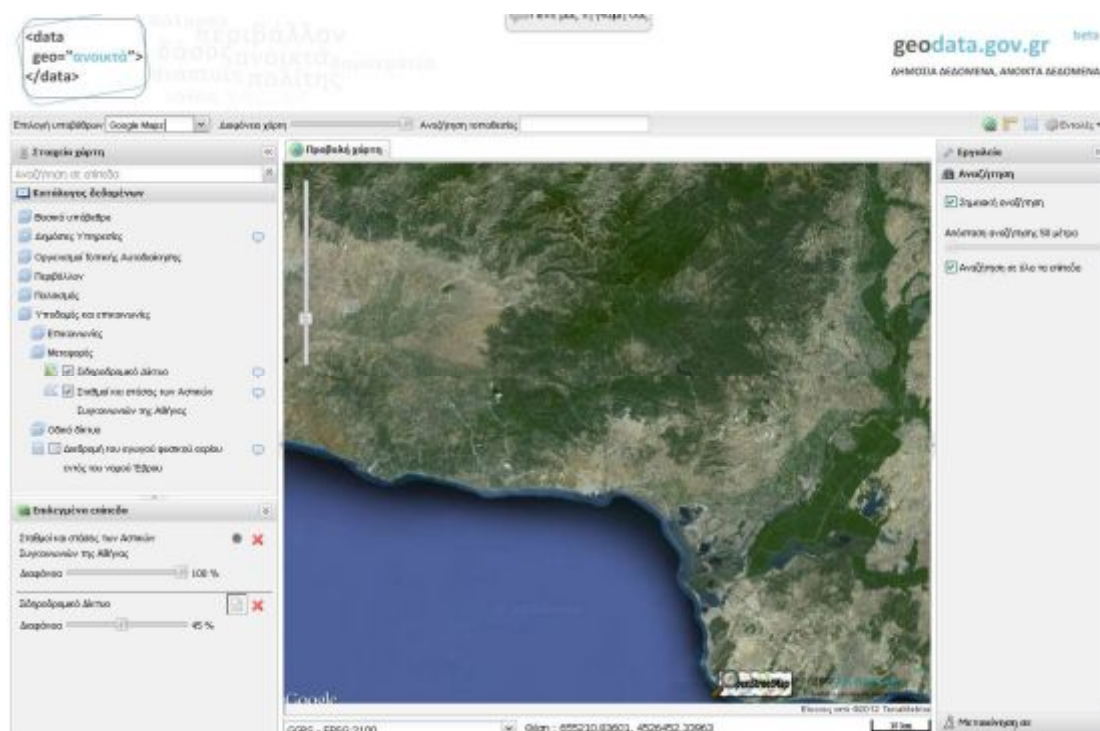


ΚΕΦΑΛΑΙΟ 5

Σήμερα είναι αρκετές οι εφαρμογές χαρτογραφικού περιχομένου στον παγκόσμιο ιστό. Από αυτές άλλες χαρακτηρίζονται από αυξημένη διαδραστικότητα και άλλες από περιορισμένη. Παρακάτω παρατίθενται κάποιες χαρακτηριστικές εφαρμογές που δίνουν την δυνατότητα αλληλεπίδρασης μεταξύ χρήστη και χάρτη

5.1 Διαδικτυακή Εφαρμογή *geodata.gov.gr*

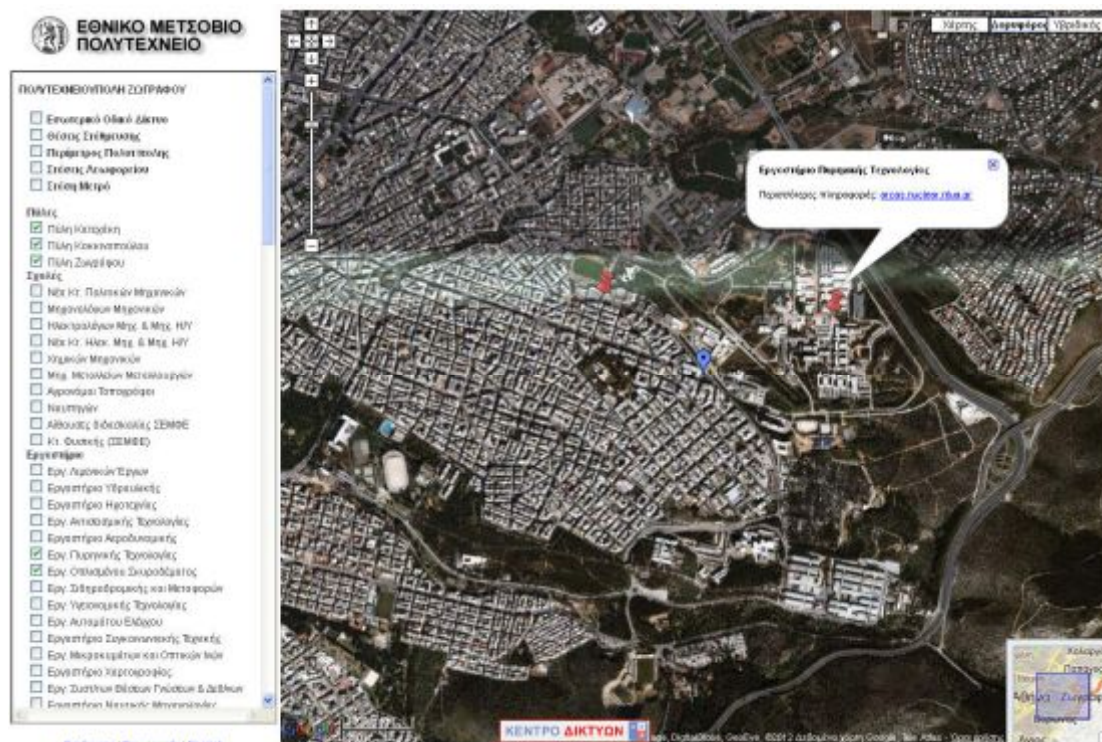
Η ιστοσελίδα *geodata.gov.gr* αποτελεί μια πύλη διάθεσης χωρικών δεδομένων από δημόσιες υπηρεσίες σε κάθε ενδιαφερόμενο. Η συγκεκριμένη ιστοσελίδα χρησιμοποιεί ως υπόβαθρα του χάρτες της Κτηματολόγιο Α.Ε, τους χάρτες της Google, τους χάρτες της Microsoft καθώς και τους χάρτες του Open Street Maps. Για την δημιουργία της εφαρμογής έχει χρησιμοποιηθεί το δωρεάν λογισμικό MapServer για την παραγωγή των χαρτών και Geonetwork για την οργάνωση των μεταδεδομένων. Επισημαίνεται ότι ο χρήστης έχει τη δυνατότητα να κατεβάσει τα δεδομένα που τον ενδιαφέρουν μέσω υπηρεσιών WMS, WFS σε διάφορες μορφές αρχείων, όπως ενδεικτικά shp, kml κ.α.



Εικόνα 1. Διαδικτυακή Εφαρμογή geodata.gov.gr

5.2 Χάρτης ΕΜΠ

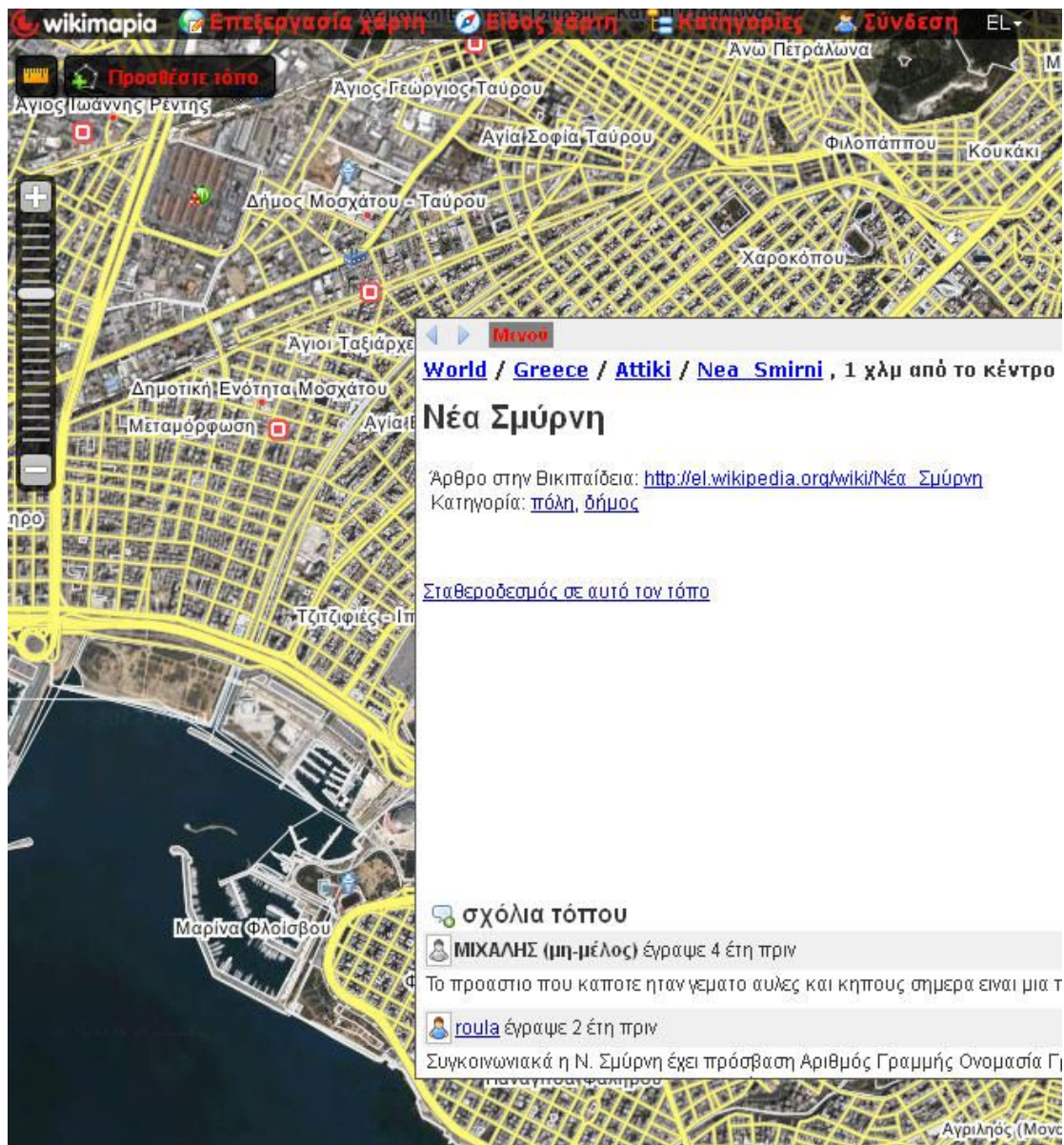
Ο ψηφιακός χάρτης του Εθνικού Μετσοβίου Πολυτεχνείου είναι μια διαδικτυακή εφαρμογή που περιέχει πληροφορίες για το Πολυτεχνείο, όπως ενδεικτικά οι θέσεις κάθε τμήματος, συγκοινωνιακές πληροφορίες, οδικό δίκτυο εντός της Πολυτεχνειούπολης Ζωγράφου κ.α. Η υλοποίηση του παρόντος χάρτη έγινε με χρήση του Google Maps Api.



Εικόνα 2. Διαδικτυακή Εφαρμογή Χάρτη ΕΜΠ

5.3 Διαδικτυακή Εφαρμογή Wikimapia.org

Η διαδικτυακή εφαρμογή που βρίσκεται στην ηλεκτρονική ιστοσελίδα Wikimapia.org είναι μια πρωτότυπη προσπάθεια συνδυασμού των πληροφοριών που βρίσκονται στην ιστοσελίδα της Wikipedia έχοντας ως υπόβαθρο τους χάρτες της Google. Υπάρχουν διάφορα θεματικά επίπεδα στον χάρτη, τα οποία συνδέονται (υπάρχει παραπομπή) στην ιστοσελίδα Wikipedia.org, στα οποία ο χρήστης μπορεί να απευθυνθεί για περισσότερες πληροφορίες. Η υλοποίηση έγινε με χρήση του Google Maps Api και πλεονέκτημα (για πολλούς θα μπορούσε να χαρακτηριστεί ως μειονέκτημα) της ιστοσελίδας αποτελεί ότι ο χρήστης μπορεί να προσθέσει τις δικές του πληροφορίες.



Εικόνα 3. Διαδικτυακή Εφαρμογή Χάρτη Wikimapia.org

ΚΕΦΑΛΑΙΟ 6Ο

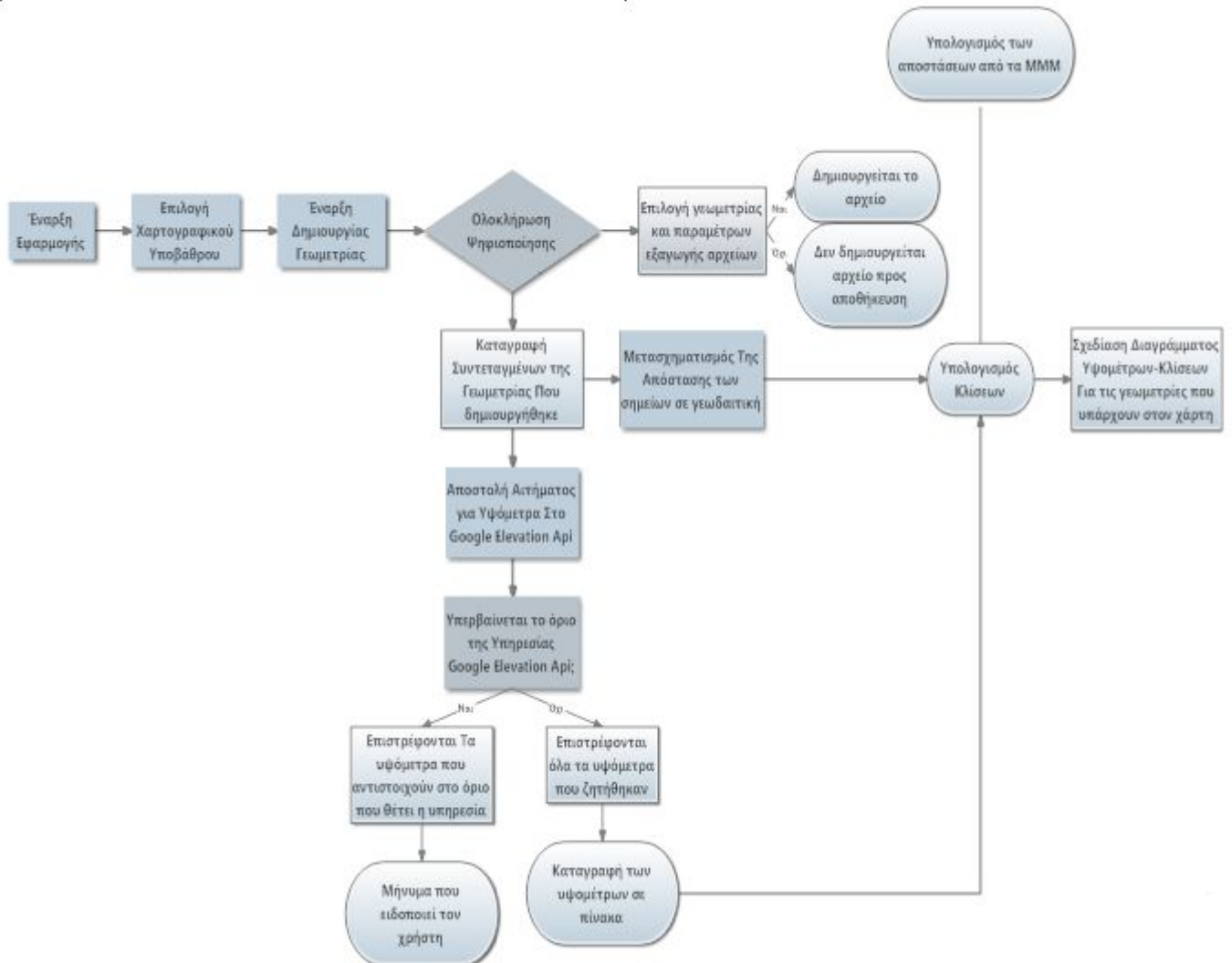
Γενικά

Πρίν την υλοποίηση οποιασδήποτε εφαρμογής είναι απαραίτητος ο σχεδιασμός της εφαρμογής. Στο στάδιο αυτό θα επιλέξουμε τι θέλουμε να πραγματοποιεί η εφαρμογή μας και ποια εργαλεία θα χρησιμοποιήσουμε για τον σκοπό αυτό

6.1 ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Κομμάτι της λειτουργίας της εφαρμογής είναι δημιουργία διαδικτυακής υπηρεσίας, η οποία χρησιμοποιεί χαρτογραφικό υπόβαθρο που διανέμεται δωρεάν στο διαδίκτυο (Google, Open Street Maps, Ktimatologio A.E, Bing Maps) και η απεικόνιση συγκεκριμένων γεωγραφικών δεδομένων στο υπόβαθρο αυτό (συγκεκριμένα οι θέσεις των σταθμών του μετρό της Αθήνας, του ηλεκτρικού και του προαστιακού σιδηρόδρομου της Αθήνας). Η εφαρμογή θα διαθέτει μια εργαλειοθήκη ψηφιοποίησης, δίνοντας την δυνατότητα στον χρήστη να δημιουργεί τα δικά του γεωμετρικά στοιχεία, χρησιμοποιώντας ως χαρτογραφικό υπόβαθρο τους χάρτες που χρησιμοποιεί η εφαρμογή. Πέρα από την δυνατότητα ψηφιοποίησης, θα παρέχεται η δυνατότητα στον χρήστη να εισάγει τα δικά του αρχεία, τα οποία και θα μπορούν να προληθούν στο χαρτογραφικό υπόβαθρο. Επιπροσθέτως θα παρέχεται στους χρήστες η δυνατότητα μεγέθυνσης, σμίκρυνσης, μετακίνησης και επιλογής στο διαθέσιμο χαρτογραφικό υπόβαθρο. Κατά την ψηφιοποίηση οι συντεταγμένες των γεωμετρικών στοιχείων θα καταγράφονται και θα στέλνονται στην υπηρεσία elevation api της google η οποία και θα επιστρέφει τα υψόμετρα των κορυφών που ψηφιοποιήθηκαν (και των οποίων οι συντεταγμένες καταγράφηκαν). Με χρήση των συνεταγμένων και των αντίστοιχων υψομέτρων θα υπολογίζεται η απόσταση των κορυφών μεταξύ και η αντίστοιχη κλίση των πλευρών του δικτύου (κλίση= υψομετρική διαφορά προς απόσταση). Επιπροσθέτως θα υπολογίζεται η απόσταση του ποδηλατικού δικτύου από συγκεκριμένα σημεία ενδιαφέροντος. Τέλος θα

παρέχεται η δυνατότητα αποθήκευσης των αρχείων που δημιουργήσε ο χρήστης. Σχηματικά η πορεία εργασιών φαίνεται στο παρακάτω διάγραμμα :

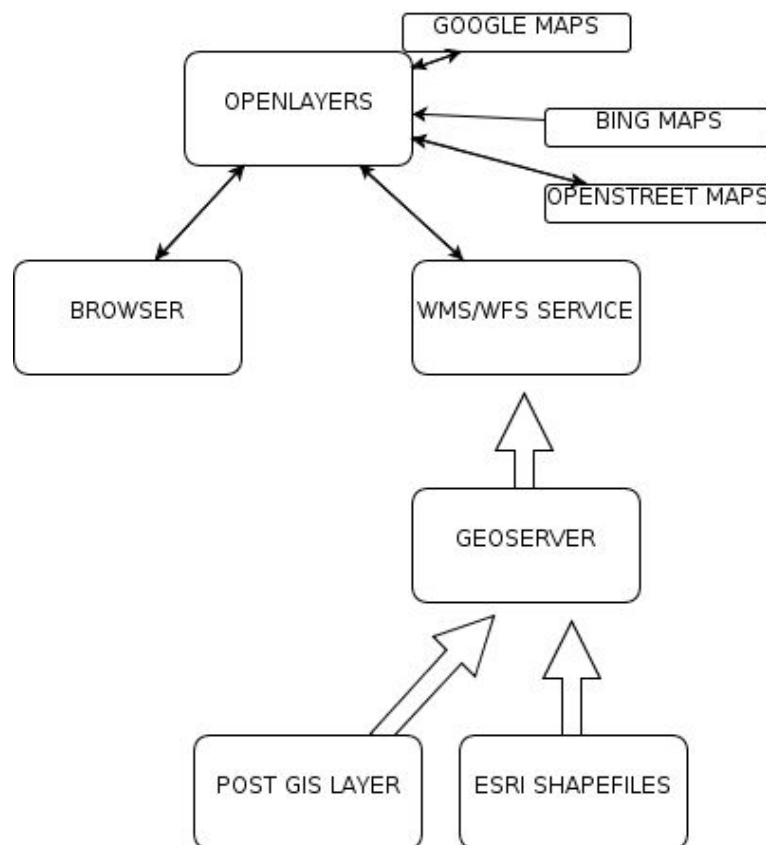


Διάγραμμα 1 : Πορεία Εργασιών που θα υλοποιηθούν

6.1.1 Αρχιτεκτονική χαρτογραφικού σκέλους εφαρμογής και επιλογή υποβάθρων

Αρχιτεκτονική Εφαρμογής:

Η αρχιτεκτονική της open source εφαρμογής συντίθεται σε τρία τμήματα. Παρακάτω περιγράφονται τα τμήματα αυτά καθώς και η μεταξύ τους επικοινωνία.



Σχήμα 3: Αρχιτεκτονική διαδικτυακής εφαρμογής

Το πρώτο τμήμα απαρτίζεται από τα δεδομένα, τη Χωρική Βάση Δεδομένων που περιέχει τις πληροφορίες για τις Γεωγραφικές Θέσεις και τα δεδομένα που αποθηκεύονται σε συστήματα αρχείων.

Το δεύτερο τμήμα απαρτίζεται από εξυπηρετητές (servers) οι οποίοι διαχειρίζονται τα γεωγραφικά δεδομένα ανάλογα με τα αιτήματα του πελάτη (client). Οι εξυπηρετητές που χρησιμοποιούνται είναι ο GeoServer ο οποίος είναι υπεύθυνος για την ανάλυση και διαχείριση των δεδομένων που προσφέρονται στο πελάτη μέσω της WMS υπηρεσίας του OGC. Συγκεκριμένα διαχειρίζεται τη Χωρική Βάση Δεδομένων, το διανυσματικό υπόβαθρο. Επίσης χρησιμοποιούνται οι Servers της Google για τα Google Maps και τα Open Street Maps, καθώς και οι χάρτες της Κτηματολόγιο Α.Ε.

Τα γεωγραφικά δεδομένα αποστέλλονται στον πελάτη μέσω διαδικτυακών υπηρεσιών.


Τέλος στο τρίτο επίπεδο βρίσκονται οι διαδικτυακές διεπαφές του χρήστη με την οποία αποστέλλονται αιτήματα και απεικονίζονται τα αποτελέσματα, και του διαχειριστή όπου υπάρχουν δυνατότητες επεξεργασίας των δεδομένων. Οι διεπαφές έχουν πρόσβαση στα ίδια δεδομένα μέσω των ίδιων εξυπηρετητών η διαφορά τους έγκειται στη χρήση διαφορετικών υπηρεσιών. Για τις ανάγκες της εφαρμογής χρησιμοποιήθηκε η διαδικτυακή διεπαφή OPENLAYERS έκδοσης 2.11.

6.1.2 Επιλογή Χαρτογραφικού Υποβάθρου:

Στις μέρες μας υπάρχει η δυνατότητα επιλογής του κατάλληλου χαρτογραφικού υποβάθρου από μία σειρά υπηρεσιών που προσφέρουν δεδομένα (κυρίως δορυφορικές εικόνες) δωρεάν σε κάθε ενδιαφερόμενο.

Χαρτογραφικά Υπόβαθρα που χρησιμοποιήσαμε:


Τα χαρτογραφικά υπόβαθρα που χρησιμοποιήσαμε είναι τα εξής:

 **Οι χάρτες της Κτηματολόγιο Α.Ε:** Οι χάρτες είναι διαθέσιμοι προς κάθε ενδιαφερόμενο από την ιστοσελίδα:

<http://gis.ktimanet.gr/wms/ktbasemap/default.aspx>.

Πρόκειται για τους πρόσφατους ορθοφωτοχάρτες που παρήχθησαν για την δημιουργία του Εθνικού Κτηματολογίου και καλύπτουν το σύνολο της

Ελληνικής Επικράτειας. Η κλίμακά τους είναι 1:5000 για το σύνολο της χώρας, ενώ για τα μεγάλα αστικά κέντρα είναι 1:1000. Το σύστημα συντεταγμένων που χρησιμοποιήθηκε για την παραγωγή των χαρτών αυτών είναι το Προβολικό σύστημα που χρησιμοποιεί το Εθνικό Γεωδαιτικό Σύστημα Αναφοράς 1987 ('ΕΓΣΑ' 87). Το υπόβαθρο αυτό είναι ακριβέστερο από τα υπόλοιπα υπόβαθρα καθώς πρόκειται για γεωμετρικά διορθωμένες αεροφωτογραφίες και όχι απλές δορυφορικές εικόνες. Το πρόβλημα στην χρησιμοποίησή τους είναι το γεωδαιτικό σύστημα στο οποίο αυτές αναφέρονται ('ΕΓΣΑ ' 87) και πιο συγκεκριμένα στο ότι αυτό είναι διαφορετικό από το Προβολικό Σύστημα που χρησιμοποιεί η Google καθώς και άλλοι πάροχοι (π.χ Yahoo, Bing maps) για την παροχή των δικών τους δεδομένων. Επομένως είναι απαραίτητος ο μετασχηματισμός των συντεταγμένων των χαρτών της Κτηματολόγιο Α.Ε προκειμένου να τους χρησιμοποιήσουμε σε συνδιασμό με χάρτες άλλων παρόχων.

 **Οι χάρτες της Google:** Οι χάρτες της Google είναι το πλέον διαδεδομένο και ευρύτερα χρησιμοποιούμενο στο διαδίκτυο χαρτογραφικό υπόβαθρο. Πρόκειται για δορυφορικές εικόνες υψηλής και πολύ υψηλής ανάλυσης (ανάλογα με την απεικονιζόμενη περιοχή) που καλύπτουν όλο τον κόσμο. Οι δορυφορικές εικόνες αυτές είναι διαθέσιμες μέσω μιάς υπηρεσίας χαρτών σε όποιον υπολογιστή διαθέτει πρόσβαση στο διαδίκτυο. Ενδιαφέρον παρουσιάζει η δυνατότητα που δίνει η Google σε κάθε χρήστη να εισάγει εξατομικευμένους χάρτες στην ιστοσελίδα του, κάνοντας χρήση του Google Maps Api, αλλά και στην δημιουργία χαρτογραφικών mash-ups.

Η Google πέρα από το επίπεδο των δορυφορικών εικόνων προσφέρει στους χρήστες το Google Streets, που αποτελεί οδικό χάρτη, το Google Hybrid που αποτελεί υβριδικό χαρτογραφικό προϊόν, προβάλλοντας το οδικό δίκτυο σε δορυφορικές εικόνες. Τέλος προσφέρεται ένα ακόμα θεματικό επίπεδο, το Google Physical, που αποτελεί μια προσπάθεια φυσικής αναπαράστασης της γήινης επιφάνειας. Επισημαίνεται ότι όλες οι εικόνες της Google είναι στο σύστημα WGS'84.



Ο χάρτης του Open Street Maps: Ο χάρτης του Open Street Maps αποτελεί μια προσπάθεια δημιουργίας ενός παγκόσμιου χάρτη από απλούς χρήστες. Ο χάρτης είναι προϊόν συνδυασμού δεδομένων από διάφορες πηγές. Επισημαίνεται ότι δίνεται η δυνατότητα σε εγγεγραμμένους χρήστες να εισάγουν δεδομένα GPS, καθώς και να τροποποιούν τα διανυσματικά δεδομένα. Το σύστημα συντεταγμένων που χρησιμοποιεί το Open Street Maps είναι το WGS'84.

Συλλογή Δεδομένων και Εισαγωγή τους σε βάση Γεωγραφικών / Χαρτογραφικών Δεδομένων.

Στόχος όπως προαναφέρθηκε ήταν η δημιουργία διαδικτυακής εφαρμογής που χρησιμοποιεί δωρεάν διαθέσιμα χαρτογραφικά υπόβαθρα (Google, Open Street Maps, Κτηματολόγιο Α.Ε) και πάνω στα οποία θα γίνει η προβολή των δικών μας δεδομένων (σημεία ενδιαφέροντος) αλλά και αριθμητικοί υπολογισμοί με βάση τα δεδομένα που εισάγει ο χρήστης. Τα δεδομένα που θα απεικονίζει η εφαρμογή ήταν προϊόν δικής μας συλλογής.

Η συλλογή των Γεωγραφικών/Χαρτογραφικών δεδομένων τα οποία θα απεικονίζει η εφαρμογή μας έγινε στο δωρεάν λογισμικό Quantum GIS. Αρχικά έγινε η επιλογή της περιοχής μελέτης (Αθήνα). Έπειτα βρήκαμε τον χάρτη χρήσεων Γης της συγκεκριμένης περιοχής, ο οποίος λειτούργησε ως υπόβαθρο για την ψηφιοποίηση των δεδομένων μας. Επιλέξαμε να ψηφιοποιήσουμε τους σταθμούς μετρό, ηλεκτρικού και προαστιακού σιδηρόδρομου. Στο σημείο αυτό πρέπει να επισημάνουμε ότι πρακτικά θα μπορούσαμε να δημιουργήσουμε οποιοδήποτε θεματικό επίπεδο και μετά να το προβάλλουμε στην διαδικτυακή μας υπηρεσία (ενδεικτικά και όχι περιοριστικά διαφορετικές χρήσεις γης, συγκοινωνιακό δίκτυο, σημεία ενδιαφέροντος κ.ο.κ).

Κατόπιν τα σημεία που δημιουργήθηκαν στο προηγούμενο στάδιο φορτώθηκαν σε χωρική βάση δεδομένων ώστε να συνδεθούν με τον εξυπηρετητή δεδομένων (Geoserver). Ιδιαίτερη προσοχή πρέπει να δοθεί στο γεωδαιτικό σύστημα στο οποίο βρίσκονται τα δεδομένα μας (ΕΓΣΑ'87), το οποίο διαφέρει από το σύστημα που είναι οι χάρτες της Google και του Open

Street Maps.Επομένως απαιτείται μετασχηματισμός των δεδομένων μας, εφόσον επιθυμούμε να τα προβάλλουμε στους προαναφερθέντες χάρτες.

Επικοινωνία εξυπηρετητή (Geoserver) με την χωρική βάση δεδομένων.

Ο εξυπηρετητής επικοινωνεί με τον πελάτη, διαβάζει τα ερωτήματά του, συνδέεται με την χωρική βάση δεδομένων και ανακτά τα δεδομένα, τα οποία τα επιστρέφει στον πελάτη με κάποια υπηρεσία WMS/WFS. Στην συγκεκριμένη εργασία χρησιμοποιήσαμε την υπηρεσία WFS, για την οπτικοποίηση των δεδομένων στον πελάτη.

Ρύθμιση Open Layers

Επόμενο βήμα είναι η εγκατάσταση και ρύθμιση της διεπαφής με τον πελάτη, έτσι ώστε να υπάρχει επικοινωνία μεταξύ των διάφορων εξυπηρετητών δωρεάν παροχής χαρτογραφικού περιεχομένου (Google, Open Street Maps, Κτηματολόγιο Α.Ε) και του εξυπηρετητή (Geoserver) που παρέχει τα διανυσματικά δεδομένα (αρχείο σταθμών μετρό, ηλεκτρικού και ησαπ).

Ουσιαστικά διαβάζουμε τους δωρεάν εξυπηρετητές χαρτών. Στην περίπτωση των χαρτών της Κτηματολόγιο Α.Ε μετασχηματίζουμε το σύστημα συντεταγμένων, ώστε αυτό να είναι συμβατό με τον υπολοίπων εξυπηρετητών και τέλος δηλώνουμε την υπηρεσία WFS ώστε η εφαρμογή να διαβάζει τα δεδομένα μας από το δικό μας εξυπηρετητή. Στο παράρτημα 1 υπάρχουν εικόνες της εφαρμογής. Ο κώδικας που συντάχθηκε για την υλοποίησή της παρατίθεται στο παράρτημα 2.

6.2 Έναρξη Δημιουργίας Γεωμετρίας

Ο χρήστης θα μπορεί να δημιουργήσει πολλαπλές διαδρομές χρησιμοποιώντας τα εργαλεία ψηφιοποίησης που διαθέτει η εφαρμογή ή να προσθέτει στον χάρτη έτοιμα αρχεία (που έχουν δημιουργηθεί από άλλες εφαρμογές). Στην περίπτωση που ο χρήστης επιλέξει να φορτώσει αρχεία στην εφαρμογή και όχι να τα ψηφιοποιήσει εξ αρχής προβλέπεται έλεγχος συμβατότητας με την μορφή που έχει επιλέξει. Και στις δυο αυτές περιπτώσεις θα γίνεται καταγραφή των συντεταγμένων της γεωμετρίας (σημείο, γραμμή, πολύγωνο). Στην περίπτωση των γραμμών και των

πολυγώνων θα καταγράφεται η αλληλουχία των συντεταγμένων των κορυφών όπως αυτές εισήχθησαν στην εφαρμογή (ψηφιοποίηση ή εισαγωγή).

6.3 Κλήση της υπηρεσίας Google Elevation Api

Μετά την καταγραφή των συντεταγμένων από το προηγούμενο βήμα στέλνεται αίτημα στην υπηρεσία Google Elevation Api για την παροχή των υψομέτρων (βλέπε κεφάλαιο 4). Όπως αναφέρθηκε όμως η χρήση της υπηρεσίας Elevation Api υπόκειται σε περιορισμούς (2500 αιτήματα ανά ημέρα). Αν τα όρια αυτά ξεπεραστούν ο χρήστης ενημερώνεται. Στην περίπτωση που τα αιτήματα για υψόμετρο είναι μέσα στα όρια που θέτει η υπηρεσία, τότε τα υψόμετρα επιστρέφονται. Στο σημείο αυτό πρέπει να τονίσουμε ότι το αίτημα για υψόμετρα δεν γίνεται μόνο για τις κορυφές που ψηφιοποιεί ο χρήστης (στην περίπτωση γραμμών) αλλά και δειγματοληπτικά κατά μήκος των κορυφών. Ο λόγος που γίνεται αυτό είναι για να μην χαθεί υψομετρική πληροφορία ανάμεσα στις κορυφές η οποία οφείλεται σε αδυναμία αναγνώρισης του βαθους από τον χρήστη ελλείψη εμπειρίας. Όπως και να έχει ο σωστός σχεδιασμός μίας εφαρμογής προϋποθέτει την προσπάθεια για λιγότερη εμπλοκή του χρήστη στην διαδικασία.

6.4 Μετασχηματισμός των αποστάσεων σε γεωδαιτικές

Το πρόβλημα έγκυται στην εύρεση της απόστασης δυο σημείων που βρίσκονται στην επιφάνεια της γης δοθέντως των γεωγραφικών τους συντεταγμένων.

Οι συντεταγμένες που έχουμε αναφέρονται σε βαθμούς και όχι σε μέτρα. Είναι δεδομένο ότι για να γίνονται οι αποστάσεις κατανοητές από τον χρήστη αλλά και να έχουν φυσική σημασία πρέπει αυτές να αναφέρονται σε μονάδα που να είναι κατανοητή από όλους, αυτή η μονάδα είναι τα μέτρα.

Αν η επιφάνεια αναφοράς που χρησιμοποιούσαμε ήταν το επίπεδο η αντιμετώπιση του προβλήματος θα ήταν πολύ εύκολη (2ο θεμελιώδες πρόβλημα της τοπογραφίας) και η απόσταση δυο σημείων θα ήταν η ευθεία γραμμή που ενώνει τα σημεία. Όμως η επιφάνεια αναφοράς που χρησιμοποιούμε είναι η σφαίρα. Επομένως αναζητούμε την απόσταση δυο σημείων πάνω στην σφαίρα (η επιφάνεια της γης θεωρείται ως σφαίρα). Η

απόσταση αυτή όμως δεν είναι ευθεία αλλή είναι μια καμπύλη γραμμή (γεωδαιτική γραμμή) και συγκεκριμένα η καμπύλη μικρότερου μήκους που ενώνει τα σημεία (ορθοδρομική απόσταση).Επειδή η σφαιρική γεωμετρία είναι διαφορετική από την ευκλείδεια γεωμετρία η εξίσωση της απόστασης παίρνει διαφορετική μορφή.

Η λύση του προβλήματος δίνεται από την φόρμουλα haversine, η οποία χρησιμοποιείται για την εύρεση απόστασης μεταξύ δυο σημείων με γεωγραφικές συντεταγμένες (και επιφάνεια αναφοράς την σφαίρα) και η οποία χρησιμοποιείται ευρέως στην ναυσιπλοΐα αλλά και σε κάθε πρόβλημα εύρεσης γωνιών και αποστάσεων με επιφάνεια αναφοράς την σφαίρα.Η εξίσωση αυτή γράφεται ως εξής

$$d = 2r \arcsin \left(\sqrt{\text{haversin}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{haversin}(\lambda_2 - \lambda_1)} \right)$$

$$= 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Όπου

r : είναι η ακτίνα της σφαίρας, της γης εν προκειμένω και η οποία θεωρείται ίση με 6371 χιλιόμετρα

ϕ_1, ϕ_2 : Είναι το γεωγραφικό πλάτος κάθε σημείου

λ_1, λ_2 : Είναι το γεωγραφικό μήκος κάθε σημείου

Στο σημείο αυτό πρέπει να επισημάνουμε ότι η εφαρμογή της σχέσης αυτής στην περίπτωση της γης δεν είναι απόλυτα σωστή, γεγονός που οφείλεται στο ότι η γη δεν είναι μια τέλεια σφαίρα αλλά η ακτίνα της μεταβάλλεται από 6356.78 km στους πόλους σε 6378.14 km στον ισημερινό .Επιπροσθέτως η ακτίνα καμπυλότητας δεν είναι σταθερή.Μια πιο ασφαλής αντιμετώπιση θα ήταν αν θεωρούσαμε την γη ως ελλειψοειδές και να χρησιμοποιούσαμε την φόρμουλα Vincenty για την λύση του προβλήματος.Η λύση αυτή θα παρείχε μεγαλύτερη ακρίβεια, η οποία όμως στην περίπτωση μας δεν είναι απαραίτητα, καθώς το μήκος που σχεδιάζει ο χρήστης δεν είναι τόσο μεγάλο οπότε η αντιμετώπιση της σφαιρικής γης δίνει ικανοποιητικά αποτελέσματα.

6.5 Υπολογισμοί Κλίσεων

Μετά την επιστροφή των υψομέτρων από την υπηρεσία Google Elevation Api και τον μετασχηματισμό των αποστάσεων σε γεωδαιτικές υπολογίζονται οι κλίσεις. Υπενθυμίζεται ότι οι κλίσεις είναι το αποτέλεσμα της διαίρεσης της υψομετρικής διαφοράς δυο σημείων προς την απόσταση που τα χωρίζει. Ιδιαίτερη προσοχή πρέπει να δοθεί στο πλήθος των κλίσεων που θα υπολογίσουμε και το οποίο εξαρτάται από το πλήθος των σημείων που θα ψηφιοποιήσει ο χρήστης μείον 1. Αυτό σημαίνει ότι δυο σημεία έχουν μία κλίση, πράγμα που είναι λογικό καθώς δυο σημεία δημιουργούν μια πλευρά. Για τον υπολογισμό των κλίσεων επομένως βρίσκουμε τις υψομετρικές διαφορές των σημείων και την διαιρούμε με την απόσταση που τα χωρίζει.

6.6 Δημιουργία Διαγραμμάτων

Από την στιγμή που έχουν υπολογιστεί οι κλίσεις μπορούν να σχεδιαστούν σε διάγραμμα. Για την σχεδιάσή του χρησιμοποιείται μια διαδικτυακή εφαρμογή, το Google Chart Tools (βλέπε κεφάλαιο 4). Οι παράμετροι που δίνονται στην υπηρεσία αυτή για την δημιουργία διαγράμματος είναι τα υψόμετρα που επιστράφησαν από την υπηρεσία Google Elevation Api, οι κλίσεις που υπολογίσαμε στο προηγούμενο βήμα αλλά και οι αποστάσεις μεταξύ των κορυφών, ανεξάρτητα αν αυτές ψηφιοποιήθηκαν από τον χρήστη ή εισήχθησαν με την μορφή αρχείου.

6.7 Εύρεση των αποστάσεων από MMM μεταφοράς και ενημέρωση του χρήστη

Ουσιαστικά πρόκειται για ένα γεωμετρικό πρόβλημα, την εύρεση απόστασης σημείου από ευθεία. Η ευθεία ουσιαστικά αναπαριστά τον ποδηλατόδρομο ενώ το σημείο την θέση του μέσου μαζικής μεταφοράς, το οποίο μπορεί να είναι σταθμός μετρό, ηλεκτρικού ή προαστιακού σιδηρόδρομου ή απλώς μια στάση λεωφορείου. Υπολογίζονται όλες αποστάσεις, δηλαδή προς όλα τα


MMM και επιλέγονται προς προβολή η ελάχιστη απόσταση, αλλά και οι αποστάσεις στα όρια που θέτει ο χρήστης.


6.8 Επιλογή των αρχείων προς αποθήκευση

Ο χρήστης αφού έχει ψηφιοποιήσει μια οποιαδήποτε γεωμετρία (σημείο, γραμμή, πολύγωνο) έχει την δυνατότητα να αποθηκεύσει την εργασία του. Πρέπει πρώτα να καθορίσει την μορφή του αρχείου που επιθυμεί να αποθηκεύσει επιλέγοντας μια από τις μορφές αρχείων που υποστηρίζονται (GeoJSON, Atom, KML, GeoRss, GML, WKT, GPX) ή να κρατήσει την προκαθορισμένη GeoJSON. Επίσης αν επιθυμεί μπορεί να σώσει τα αρχεία σε WGS ' 84 (EPSG :4326) ή σε εγκάρσια μερκατορική προβολή (Spherical Mercator) που χρησιμοποιεί η Google για την απεικόνιση των δεδομένων της. Μετά με μια απλή αντιγραφή και επικόλληση στο notepad των windows και σώζοντας το αρχείο με την αντίστοιχη κατάληξη έχει αποθηκεύσει την γεωμετρία.







6.9 Περιγραφή της τρόπου Λειτουργίας της Εφαρμογής


Για την υλοποίηση της εφαρμογής κατασκευάστηκαν τμήματα κώδικα στην γλώσσα προγραμματισμού Javascript. Τα οποία ενώθηκαν σε ένα ενιαίο αρχείο. Οι συναρτήσεις που χρησιμοποιήθηκαν είναι οι εξής :


 **Init()** : Είναι η βασική συνάρτηση που χρησιμοποιείται. Κατά την εκτέλεση του κώδικα είναι η πρώτη συνάρτηση που τρέχει, καλώντας τις υπόλοιπες συναρτήσεις. Μέσα σε αυτή την συνάρτηση υπάρχουν και οι μεταβλητές που καλούνται να παραλάβουν το χαρτογραφικό υπόβαθρο που χρησιμοποιεί η εφαρμογή.

 **lineadded ()** : Είναι η συνάρτηση που καλείται όταν ο χρήστης ξεκινά την δημιουργία γεωμετρίας. Όταν ολοκληρωθεί η δημιουργία

γεωμετρίας καταγράφει τις συντεταγμένες της γεωμετρίας που δημιουργήθηκε και τις δίνει ως όρισμα στην συνάρτηση transform.

-  **transform()** : Είναι η συνάρτηση που αναλαμβάνει να κάνει τους μετασχηματισμούς μεταξύ συστημάτων αναφοράς. Λαμβάνει τις συντεταγμένες που καταγράφονται από την lineadded () και τις επιστρέφει στο σύστημα αναφοράς wgs'84.
-  **enhmeroshformat()**: Η συνάρτηση αυτή αναλαμβάνει να παραλάβει τις επιλογές του χρήστη ως προς την φόρτωση αρχείων στην εφαρμογή και την αποθήκευσή τους και καλεί τις συναρτήσεις domhsh και apodomhsh.
-  **domhsh()**: Η συνάρτηση αυτή παίρνει τις επιλογές του χρήστη από την συνάρτηση enhmeroshformat και ανάλογα με αυτές δημιουργεί το αντίστοιχο αρχείο για αποθήκευση (το οποίο και στέλνει σε συγκεκριμένο σημείο στην ιστοσελίδα).
-  **apodomhsh()** : Η συνάρτηση αυτή διαβάζει τις επιλογές του χρήστη από την συνάρτηση enhmeroshformat και την μορφή του αρχείου και αναλαμβάνει να διαβάσει το αρχείο από το σημείο που είναι καταγεγραμμένο και να το εισάγει στο χαρτογραφικό υπόβαθρο.
-  **calculateSlopes()** : Η συνάρτηση αυτή παίρνει ως όρισμα ένα δισδιάστατο πίνακα συντεταγμένων καθώς και έναν μονοδιάστατο πίνακα με υψόμετρα και υπολογίζει τις κλίσεις μεταξύ τους.
-  **plotElevation()**: Η συνάρτηση αυτή καλεί την συνάρτηση calculateSlopes() δίνοντας της ως όρισμα τον πίνακα συντεταγμένων που καταγράφηκαν από την lineadded καθώς και τα υψόμετρα που επιστράφηκαν από το google elevation api. Παίρνει τα αποτελέσματα από την συνάρτηση calculateSlopes() τα οποία και στέλνει στην υπηρεσία google charts για την δημιουργία του διαγράμματος. Επιπροσθέτως καλεί την συνάρτηση calculateAverage και προβάλλει στο διάγραμμα την μέση τιμή που αυτή της επιστρέφει.

 **calculateAverage()** : παίρνει τις υπολογισμένες κλίσεις από την συνάρτηση calculateSlopes() και βγάζει τον μέσο όρο τους, τον οποίο και επιστρέφει στην συνάρτηση plotElevation μέσω της οποίας καλείται.

 **Kml input()** : Αναλαμβάνει την σύνδεση με τον εξυπηρετητή δεδομένων και την φόρτωση αρχείων στην εφαρμογή. Διαβάζει ουσιαστικά το αρχείο μορφής kml στο οποίο είναι αποθηκευμένες οι θέσεις των σταθμών του μετρό, του ηλεκτρικού, του προαστιακού αλλά και οι θέσεις των στάσεων λεωφορείου. Η ύπαρξη του αρχείου αυτού είναι απαραίτητη για να τρέξει η συνάρτηση distanceTo.

distanceTo() : Διαβάζει αρχικά τα όρια των αποστάσεων που θέτει ο χρήστης. Τα όρια αυτά αφορούν την απόσταση από την γεωμετρία που υπάρχει στον χάρτη (είτε αυτή ψηφιοποιήθηκε είτε εάν ήταν έτοιμη και απλώς εισήχθη στην εφαρμογή) με την θέση των σταθμών του μετρό, του ηλεκτρικού, του προαστιακού και των στάσεων λεωφορείων. Υπολογίζει την απόσταση της γραμμής αυτής με τα σημεία (οι θέσεις των MMM είναι σημεία) και την συγκρίνει με αυτή που εισήγαγε ο χρήστης. Εάν είναι μικρότερη προβάλλει στην σελίδα την ονομασία των MMM που πληρούν αυτό το κριτήριο της απόστασης. Εάν είναι μεγαλύτερη την απορρίπτει.

6.10 Λογισμικό που χρησιμοποιήθηκε:

Παρακάτω παρουσιάζεται το λογισμικό που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής. Όλα τα προγράμματα είναι ανοικτού κώδικα (open source).

Quantum GIS

Το Quantum GIS είναι ένα εύχρηστο και δωρεάν λογισμικό Γεωγραφικών Συστημάτων Πληροφοριών με διάφορες εκδόσεις για τα πιο γνωστά λειτουργικά συστήματα. Υποστηρίζει πολλές λειτουργίες, ενώ η λειτουργικότητά του μπορεί να επεκταθεί με εφαρμογές γραμμένες από χρήστες/προγραμματιστές. Για την ανάπτυξη της εφαρμογής το Quantum χρησιμοποιήθηκε για την δημιουργία των ψηφιακών αρχείων των σταθμών των ΜΜΜ και για τον μετασχηματισμό του συστήματος συντεταγμένων τους.

Google Earth

Το Google Earth πρόκειται για έναν εξαιρετικά δημοφιλή φυλλομετρητή Γης μέσα από το οποίο προσφέρονται υψηλής ποιότητας τρισδιάστατες οπτικοποίησης της Γης αλλά και το διάστημα. Τα δεδομένα προέρχονται από την εξυπηρετητή (server) της Google. Ο φυλλομετρητής παρέχει δυνατότητες σχεδιασμού γραφικών μέσω της γλώσσας KML.

Το εργαλείο Open Layers

Τα Open Layers είναι μία βιβλιοθήκη JavaScript για την οπτικοποίηση γεωγραφικών δεδομένων στον Παγκόσμιο Ιστό. Η βιβλιοθήκη των Open Layers είναι συμβατή με τους σύγχρονους φυλλομετρητές (web browsers) και είναι ανεξάρτητη από τον εξυπηρετητή. Τα openlayers παρέχουν ένα javascript api για την δημιουργία πλούσιων γεωγραφικών δεδομένων για το διαδίκτυο, αντίστοιχα με αυτά των Google Maps και MSN Virtual Earth, με την σημαντική διαφορά ότι τα Open Layers είναι ελεύθερο λογισμικό που αναπτύχθηκε για και από την κοινότητα ανοιχτού λογισμικού. Επιπλέον, είναι

συμβατά με τις τυποποιήσεις του Open Geospatial Consortium, δηλαδή τα πρωτόκολλα Web Mapping Service (WMS) και Web Feature Service (WFS).

Geoserver

Αποτελεί μια εφαρμογή ανοιχτού κώδικα, η οποία μπορεί να αναγνωρίσει τις κυριότερες μορφές αρχείων διανυσματικών και ψηφιακών δεδομένων, όπως ενδεικτικά (.shp, Arcinfo coverages, Post/gis tables). Ο Geoserver μπορεί να χρησιμοποιηθεί για την απεικόνιση δεδομένων όλων των δημοφιλών εφαρμογών χαρτογραφίας, όπως ενδεικτικά Google Maps, Google Earth, Yahoo.

ΚΕΦΑΛΑΙΟ 7Ο

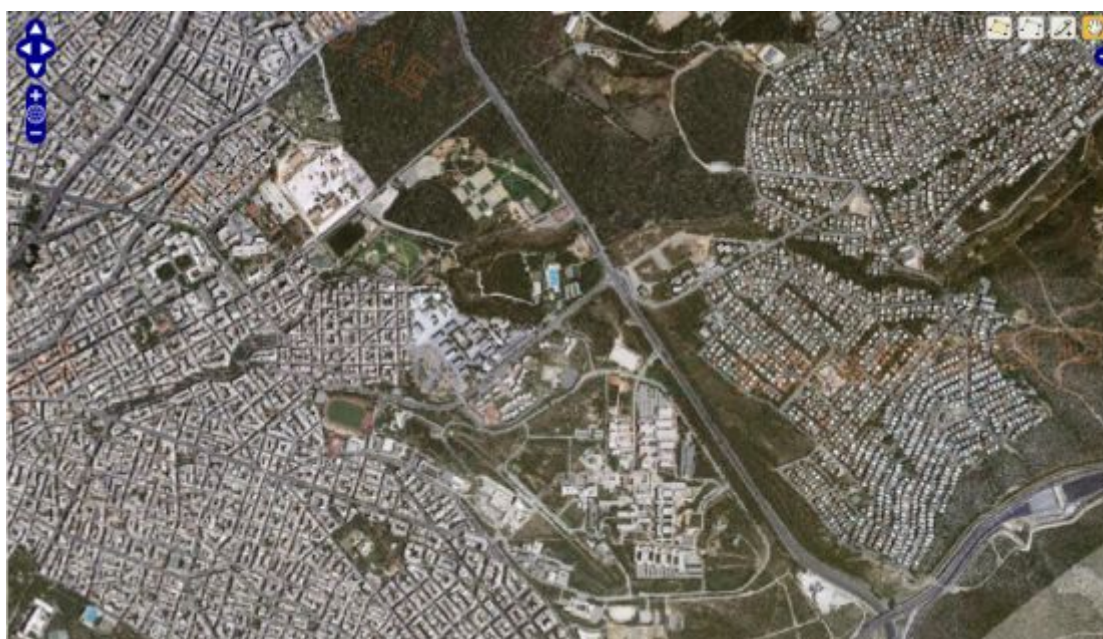
Σχεδίαση ποδηλατικού δικτύου στην Πολυτεχνειούπολη Ζωγράφου

Περιοχή Υπολογισμών

Η περιοχή που επιλέχτηκε για δοκιμή της εφαρμογής είναι η Πολυτεχνειούπολη Ζωγράφου. Πιο συγκεκριμένα θα σχεδιάσουμε ένα ποδηλατικό δίκτυο που θα συνδέει τις δυο πύλες εξόδου Κοκκινοπούλου και Κατεχάκη.

7.1 Χρήση της εφαρμογής

Με το άνοιγμα της εφαρμογής βλέπουμε την παρακάτω εικόνα



Στην οποία διακρίνουμε τα παρακάτω σύμβολα:



Επιτρέπει την μετακίνηση στον χάρτη κάνοντας κλίκ στο βέλος



Επιτρέπει την μεγέθυνση και σμίκρυνση του χάρτη κάνοντας κλίκ στο σύμβολο + ή _ αντίστοιχα. Λειτουργεί συμπληρωματικά με την ροδέλα του

ποντικιού (κάνουν την ίδια λειτουργία).



Επιτρέπει την δημιουργία πολυγώνων



Επιτρέπει την δημιουργία γραμμών



Επιτρέπει την δημιουργία σημείων



Επιτρέπει την μετακίνηση (pan) στον χάρτη



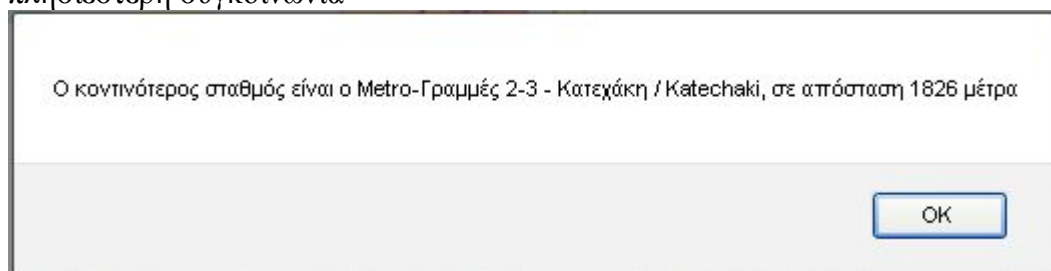
Ανοίγει ένα αναδυόμενο μενού, στο οποίο μπορούμε να επιλέξουμε χαρτογραφικό υπόβαθρο και να ενεργοποιήσουμε/απενεργοποιήσουμε layers, όπως φαίνεται και στην παρακάτω εικόνα.



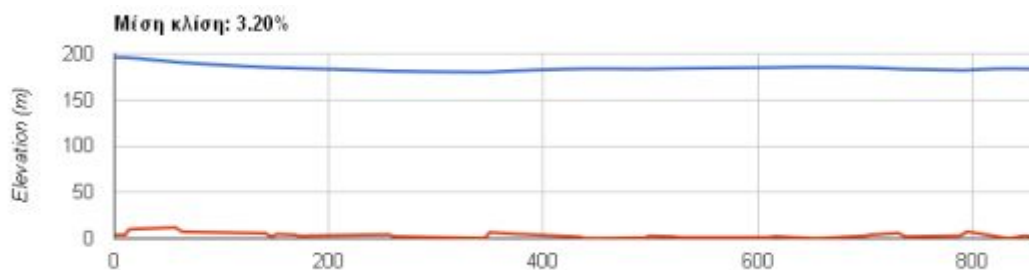
Με την επιλογή  ξεκινάμε να ψηφιοποιούμε κάνοντας διαδοχικά κλικ στην περιοχή που μας ενδιαφέρει.



Όταν ολοκληρώσουμε την ψηφιοποίηση η εφαρμογή μας ενημερώνει για την πλησιέστερη συγκοινωνία



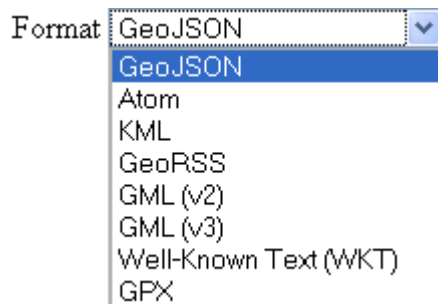
Ενώ παράλληλα δημιουργείται ένα γράφημα που περιέχει τα υψόμετρα, τις κλίσεις κατά μήκος της διαδρομής καθώς και την απόσταση από την αρχή. Ο χρήστης ενημερώνεται άμεσα για την μέση κλίση κατά μήκος της διαδρομής που έχει ψηφιοποιήσει.



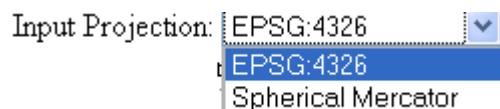
7.2 Εισαγωγή αρχείων

Στο σημείο αυτό πρέπει να επισημάνουμε ότι ο χρήστης έχει την δυνατότητα να εισάγει αρχεία (κάνοντάς τα επικόλληση) από άλλες εφαρμογές τα οποία προβάλλονται στον χάρτη. Υποστηρίζονται οι πιο διαδεδομένοι τύποι αρχείων, όπως :

GeoJSON, Atom, KML, GeoRss, GML, WKT, GPX.



Για την σωστή προβολή των αρχείων και αφού ο χρήστης επιλέξει το είδος του αρχείου, κατόπιν πρέπει να επιλέξει και το σύστημα συντεταγμένων στο οποίο είναι τα δεδομένα του. Προς το παρόν υποστηρίζεται το σύστημα WGS'84 (EPSG :4326) καθώς και εγκάρσια μερκατορική προβολή (Spherical Mercator) που χρησιμοποιεί η Google για την απεικόνιση των δεδομένων της.




Αφού ο χρήστης έχει συμπληρώσει αυτές τις επιλογές μπορεί να επικολλήσει το αρχείο του

Εδώ μπορείτε να επικολλήσετε κείμενο

Προβολή στον χάρτη

Και κάνοντας κλικ στην επιλογή προβολή στον χάρτη το αρχείο προβάλλεται στον χάρτη.

7.3 Αποθήκευση ψηφιοποιημένων από τον χρήστη αρχείων

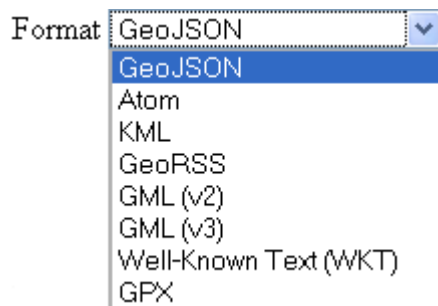
Με την επιλογή  ξεκινάμε να ψηφιοποιούμε κάνοντας διαδοχικά κλικ στην περιοχή που μας ενδιαφέρει.



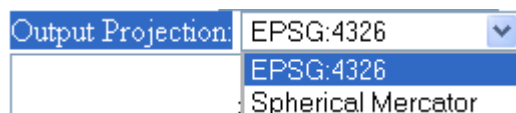
Μετά την ολοκλήρωση της ψηφιοποίησης η γραμμή γίνεται κόκκινη



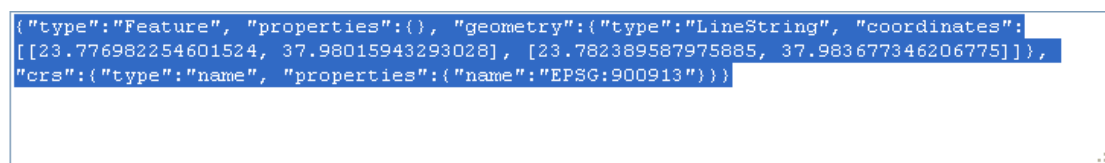
Επόμενο βήμα είναι ο ορισμός της μορφής του αρχείου που θέλουμε να εξαχθεί, αυτό γίνεται από το αναδυόμενο μενού επιλογών Format. όπως φαίνεται και στην παρακάτω εικόνα υποστηρίζονται οι ακόλουθοι τύποι αρχείων : GeoJSON, Atom, KML, GeoRss, GML, WKT, GPX



Καθώς και την προβολή των δεδομένων που θα εξαχθούν. Η επιλογή αυτή γίνεται στο αναδυόμενο μενού Output Projection και τα συστήματα που προς το παρόν υποστηρίζονται είναι το σύστημα WGS'84 (EPSG :4326) καθώς και εγκάρσια μερκατορική προβολή (Spherical Mercator) που χρησιμοποιεί η Google για την απεικόνιση των δεδομένων της



Τέλος και αφού έχουν γίνει οι επιλογές αυτές, ο χρήστης αρκεί να περάσει το ποντίκι του πάνω από την γεωμετρία που έχει ψηφιοποιήσει για να «γεμίσει» το πεδίο εξαγωγής συντεταγμένων, όπως φαίνεται στην παρακάτω εικόνα



Σημείωση: Στο σημείο αυτό θα πρέπει να τονίσουμε ότι καμία ενέργεια δεν απαιτείται από τον χρήστη αν κρατήσει τις προκαθορισμένες επιλογές

(δηλαδή μορφή αρχείων προς εξαγωγή Json format και σύστημα συντεταγμένων το WGS' 84).

Κεφάλαιο 8ο

ΠΡΟΒΛΗΜΑΤΑ ΚΑΙ ΔΥΝΑΤΟΤΗΤΑ ΓΙΑ ΜΕΛΛΟΝΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ

Κατά τη διάρκεια υλοποίησης της εφαρμογής προέκυψαν διάφορα προβλήματα τα οποία παρατίθενται παρακάτω καθώς και οι ενέργειες που κάναμε για να τα αντιμετωπίσουμε.

8.1 Γεωδαιτικά Συστήματα Αναφοράς

Ο όρος σφαιρική μερκατορική προβολή χρησιμοποιείται συχνά για να περιγράψει την προβολή που χρησιμοποιείται από διάφορους παρόχους χαρτών όπως Google Maps, Microsoft Virtual Earth, Yahoo Maps καθώς και άλλους παρόχους χαρτών.

Ο όρος σφαιρική μερκατορική προβολή χρησιμοποιείται επιπροσθέτως για να υποδηλώσει το γεγονός ότι η συγκεκριμένη προβολή που χρησιμοποιείται (από τους προαναφερθέντες παρόχους) θεωρεί την γη ως σφαίρα αντί για μια προβολή που θεωρεί την γη ως ελλειψοειδές. Το γεγονός αυτό επηρεάζει τους υπολογισμούς που γίνονται θεωρώντας την γη ως επίπεδο και για τον λόγο αυτό πρέπει να δίνουμε μεγάλη προσοχή όταν χρησιμοποιούμε χάρτες από αυτούς τους παρόχους.

Για την σωστή προβολή αρχείων χρησιμοποιώντας ως υπόβαθρο τους χάρτες που παρέχονται από αυτούς τους παρόχους πρέπει να μετασχηματίσουμε τα δεδομένα στο σύστημα που αυτοί χρησιμοποιούν. Αυτό ισχύει τόσο για τα διανυσματικά δεδομένα όσο και για τις εικόνες. Στην παρούσα διπλωματική συναντήσαμε αυτό το πρόβλημα όταν θελήσαμε να χρησιμοποιήσουμε τους χάρτες που παρέχονται από την Κτηματολόγιο Α.Ε οι οποίοι βρίσκονται στην προβολή που χρησιμοποιεί το Εθνικό Γεωδαιτικό Σύστημα Αναφοράς.

Υπήρξαν δυο τρόποι εργασίας, να μετατρέψουμε τους χάρτες που παρέχονται από την κτηματολόγιο Α.Ε στο σύστημα που χρησιμοποιούν οι άλλοι πάροχοι ή να μετατρέψουμε τους χάρτες των άλλων παρόχων στην προβολή που χρησιμοποιεί το Εθνικό Γεωδαιτικό Σύστημα Αναφοράς. Ουσιαστικά πρόκειται για έναν μετασχηματισμό εικόνων, κανονικοποιημένων δηλαδή δεδομένων. Για

λόγους ευκολίας αλλά και ταχύτητας επιλέξαμε να μετασχηματίσουμε τους χάρτες της Κτηματολόγιο Α.Ε, ώστε να είναι στο ίδιο σύστημα με τους υπολοίπους χάρτες που χρησιμοποιήσαμε. Η ίδια λογική ακολουθήθηκε και για τον μετασχηματισμό των διανυσματικών δεδομένων, συγκεκριμένα τις θέσεις των σταθμών Μετρό, Προαστιακού, ΗΣΑΠ αλλά και τις στάσεις λεωφορείων που χρησιμοποιήσαμε για την σωστή προβολή τους στο χαρτογραφικό υπόβαθρο. Τα διανυσματικά αυτά δεδομένα μετασχηματίστηκαν στην σφαιρική μερκατορική προβολή από το σύστημα συντεταγμένων που αρχικά ήταν (Εθνικό Γεωδαιτικό Σύστημα Αναφοράς).

8.2 Υπολογισμοί αποστάσεων

Το πρόβλημα έγκυται στην εύρεση της απόστασης δυο σημείων που βρίσκονται στην επιφάνεια της γης δοθέντως των γεωγραφικών τους συντεταγμένων.

Οι συντεταγμένες που έχουμε αναφέρονται σε βαθμούς και όχι σε μέτρα. Είναι δεδομένο ότι για να γίνονται οι αποστάσεις κατανοητές από τον χρήστη αλλά και να έχουν φυσική σημασία πρέπει αυτές να αναφέρονται σε μονάδα που να είναι κατανοητή από όλους, αυτή η μονάδα είναι τα μέτρα.

Αν η επιφάνεια αναφοράς που χρησιμοποιούσαμε ήταν το επίπεδο η αντιμετώπιση του προβλήματος θα ήταν πολύ εύκολη (2ο θεμελιώδες πρόβλημα της τοπογραφίας) και η απόσταση δυο σημείων θα ήταν η ευθεία γραμμή που ενώνει τα σημεία. Όμως η επιφάνεια αναφοράς που χρησιμοποιούμε είναι η σφαίρα. Επομένως αναζητούμε την απόσταση δυο σημείων πάνω στην σφαίρα (η επιφάνεια της γης θεωρείται ως σφαίρα). Η απόσταση αυτή όμως δεν είναι ευθεία αλλά είναι μια καμπύλη γραμμή (γεωδαιτική γραμμή) και συγκεκριμένα η καμπύλη μικρότερου μήκους που ενώνει τα σημεία (ορθοδρομική απόσταση). Επειδή η σφαιρική γεωμετρία είναι διαφορετική από την ευκλείδεια γεωμετρία η εξίσωση της απόστασης παίρνει διαφορετική μορφή.

Η λύση του προβλήματος δίνεται από την φόρμουλα haversine, η οποία χρησιμοποιείται για την εύρεση απόστασης μεταξύ δυο σημείων με γεωγραφικές συντεταγμένες (και επιφάνεια αναφοράς την σφαίρα) και η οποία χρησιμοποιείται ευρέως στην ναυσιπλοΐα αλλά και σε κάθε πρόβλημα

εύρεσης γωνιών και αποστάσεων με επιφάνεια αναφοράς την σφαίρα. Η εξίσωση αυτή γράφεται ως εξής

$$d = 2r \arcsin \left(\sqrt{\text{haversin}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{haversin}(\lambda_2 - \lambda_1)} \right)$$

$$= 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Όπου

r : είναι η ακτίνα της σφαίρας, της γης εν προκειμένω και η οποία θεωρείται ίση με 6371 χιλιόμετρα

Φ_1, Φ_2 : Είναι το γεωγραφικό πλάτος κάθε σημείου

λ_1, λ_2 : Είναι το γεωγραφικό μήκος κάθε σημείου

Στο σημείο αυτό πρέπει να επισημάνουμε ότι η εφαρμογή της σχέσης αυτής στην περίπτωση της γης δεν είναι απόλυτα σωστή, γεγονός που οφείλεται στο ότι η γη δεν είναι μια τέλεια σφαίρα αλλά η ακτίνα της μεταβάλλεται από 6356.78 km στους πόλους σε 6378.14 km στον ισημερινό. Επιπροσθέτως η ακτίνα καμπυλότητας δεν είναι σταθερή. Μια πιο ασφαλής αντιμετώπιση θα ήταν αν θεωρούσαμε την γη ως ελλειψοειδές και να χρησιμοποιούσαμε την φόρμουλα Vincenty για την λύση του προβλήματος. Η λύση αυτή θα παρείχε μεγαλύτερη ακρίβεια, η οποία όμως στην περίπτωση μας δεν είναι απαραίτητα, καθώς το μήκος που σχεδιάζει ο χρήστης δεν είναι τόσο μεγάλο οπότε η αντιμετώπιση της σφαιρικής γης δίνει ικανοποιητικά αποτελέσματα.

8.3 Περιορισμοί που επιβάλλονται από το Google Elevation API

Η χρήση της υπηρεσίας Google Elevation API (την οποία χρησιμοποιήσαμε για την εύρεση υψομέτρων κατά μήκος της διαδρομής που ψηφιοποιεί ο χρήστης) υπόκειται σε περιορισμούς. Πιο συγκεκριμένα υπάρχει ένα όριο 2.500 αιτημάτων για υψόμετρα ανά ημέρα. Σε κάθε χρησιμοποίηση της

υπηρεσίας ο χρήστης μπορεί να ζητήσει το υψόμετρο μέχρι 512 περιοχών, αλλά δεν μπορεί να ξεπεράσει τις 25000 περιοχές. Αν ξεπεραστεί αυτό το όριο η υπηρεσία σταματά να δουλεύει μέχρι την επόμενη ημέρα, όπου τα όρια αυτά ανανεώνονται. Κατά το στάδιο προγραμματισμού θεωρήσαμε ότι το όριο αυτό είναι αρκετά μεγάλο ώστε να προβλέψουμε κάποια εναλλακτική λύση. Ενδεχόμενη λύση θα ήταν να αποθηκεύσουμε το ψηφιακό μοντέλο αναγλύφου SRTM σε μια βάση δεδομένων και από την οποία θα ανακτούσαμε τα υψόμετρα που απαιτούνται. Η λύση αυτή δεν επιλέχτηκε αφενός μεν γιατί οι απαιτήσεις σε αποθηκευτικούς χώρους και υπολογιστική ισχύ θα ήταν μεγαλύτερες, αλλά και η ακρίβεια των υψομέτρων του SRTM είναι χαμηλότερη από την ακρίβεια των υψομέτρων που δίνει η υπηρεσία Google Elevation API (και η οποία εξαρτάται από την γεωγραφική περιοχή).

8.4 Δειγματοληψία Υψομέτρων

Μία από τις αποφάσεις που κληθήκαμε να λάβουμε κατά την εκπόνηση αυτής της διπλωματικής ήταν ανα πόσα μέτρα θα λαμβάνουμε υψομετρική τιμή.

Μια πρώτη θεώρηση θα ήταν να λαμβάνουμε υψόμετρο (μέσω της υπηρεσίας Google Elevation API) στα σημεία που ψηφιοποιεί ο χρήστης. Εξαιτίας όμως του γεγονότος ότι σε πολλούς από τους χάρτες που χρησιμοποιήσαμε ως υπόβαθρα (π.χ Google maps) το βάθος δεν μπορεί να διακριθεί, αλλά και σε αυτά τα οποία μπορεί να διακριθεί (π.χ οι χάρτες της Κτηματολόγιο Α.Ε) επειδή είναι ορθοφωτοχάρτες η ανίχνευση των απότομων υψομετρικών αλλαγών είναι δύσκολη και απαιτεί μεγάλη εμπειρία από τον χρήστη. Λόγου χάρη ο χρήστης μπορεί να ψηφιοποιήσει τις δυο πλευρές ενός ρέματος με μεγάλο βάθος. Αν επιλέξουμε να λάβουμε υψόμετρο για τις δυο αυτές πλευρές θα χαθεί αυτή η υψομετρική διαφορά που υπάρχει ανάμεσά τους, επομένως και οι κλίσεις που θα υπολογιστούν στην συνέχεια θα είναι λάθος.

Για τον λόγο αυτό επιλέξαμε να κάνουμε χρήση της υπηρεσίας Google Elevation API όχι μόνο για τα σημεία που ψηφιοποιεί ο χρήστης αλλά και για κάποια ενδιάμεσα.

8.5 Δυνατότητα Για άλλες εφαρμογές

Η εφαρμογή της διπλωματικής εργασίας μπορεί να χρησιμοποιηθεί σε ένα πλήθος άλλων εφαρμογών που απαιτούν κάποια χωρική αναφορά.

Ενδεικτικά αναφέρονται κάποιες επιπρόσθετες εφαρμογές :

8.5.1 Υπολογισμός θερμιδικής κατανάλωσης για συγκεκριμένες ποδηλατικές εφαρμογές.

Το ανάγλυφο και πιο συγκεκριμένα η κλίση ενός ποδηλατόδρομου επηρεάζει την θερμιδική κατανάλωση ενός ποδηλάτη. Άλλα στοιχεία που επηρεάζουν είναι το βάρος και η φυσική κατάσταση του ποδηλάτη, η αντίσταση του αέρα αλλά και στοιχεία της χάραξης του ποδηλατόδρομου, π.χ σε πόσες διασταυρώσεις αναγκάζεται να σταματήσει (απαιτείται μεγαλύτερη προσπάθεια, άρα και μεγαλύτερη κατανάλωση θερμίδων για να ανακτήσει την ταχύτητα που είχε ξεκινώντας από την αδράνεια). Η εφαρμογή που δημιουργήθηκε θα μπορούσε με κάποιες παραδοχές (π.χ την αντίσταση του αέρα) να υπολογίζει την θερμιδική κατανάλωση. Ενδιαφέρον θα είχε αν ο χρήστης μπορούσε να υπολογίσει με βάση τα ατομικά του στοιχεία (λόγου χάρη βάρος) την θερμιδική κατανάλωση για συγκεκριμένες ποδηλατικές διαδρομές.

8.5.2 Υπολογισμός κατανάλωσης καυσίμου για συγκεκριμένη διαδρομή

Η κλίση ενός δρόμου και πιο συγκεκριμένα το πρόσημο της κλίσης (εάν είναι ανωφέρεια ή κατωφέρεια) επηρεάζει κατά ένα μεγάλο βαθμό την κατανάλωση καυσίμου. Στην κατωφέρεια η κατανάλωση καυσίμου είναι ασφαλώς μικρότερη από όταν το όχημα κινείται σε ανωφέρεια. Ενδιαφέρον θα είχε η δυνατότητα να εισάγει ο χρήστης το όχημά του από μια λίστα οχημάτων (στα οποία τα χαρακτηριστικά του αυτοκινήτου θα ήταν γνωστά (π.χ η κατανάλωση καυσίμου) ή ο χρήστης να μπορεί να εισάγει ο ίδιος τις παραμέτρους του οχήματός του και να υπολογίζεται η κατανάλωση καυσίμου. Βεβαίως αυτή θα ήταν μια απλοποιημένη προσέγγιση καθώς η κατανάλωση καυσίμου

εξαρτάται και από άλλες παραμέτρους όπως η συντήρηση του κινητήρα κ.α, αλλά κυρίως και από την κίνηση που επικρατεί στο οδόστρωμα. Στην περίπτωση αυτή θα απαιτούνταν στοιχεία για την κίνηση που επικρατεί σε ένα οδικό τμήμα κατά την διάρκεια της ημέρας (ένας δρόμος λόγου χάρη μπορεί να μην εμφανίζει αυξημένο φόρτο σε όλη την διάρκεια της ημέρας, αλλά μόνο κάποιες συγκεκριμένες ώρες) ή ακόμα καλύτερα να υπήρχαν πραγματικά δεδομένα (REAL TIME) που να τροφοδοτούσαν το σύστημα με δεδομένα εκείνης της στιγμής ώστε ο χρήστης να ήξερε την κατανάλωση καυσίμου για κάθε διαδρομή και να μπορούσε να επιλέξει την διαδρομή που θα ακολουθήσει.

8.5.3 Υπολογισμός παραγωγής καυσαερίου από κινούμενα οχήματα

Η κλίση ενός δρόμου και πιο συγκεκριμένα το πρόσημο της κλίσης (εάν είναι ανωφέρεια ή κατωφέρεια) επηρεάζει κατά ένα μεγάλο βαθμό εκτός από την κατανάλωση καυσίμου και την εκπομπή ρύπων. Πολύ μεγάλη επίδραση εκτός από την κλίση έχει και η ταχύτητα που αναπτύσσεται σε αυτό το οδικό τμήμα. Ο συνδυασμός της εφαρμογής που δημιουργήθηκε με ένα μοντέλο υπολογισμού παραγωγής καυσαερίων κατά μήκος διαδρομών θα μπορούσε να χρησιμοποιηθεί σε μελέτες περιβαλλοντικών επιπτώσεων.

8.5.4 Έκθεση διαδρομών με βάση το καυσαέριο

Με δεδομένο ότι η έκθεση σε καυσαέριο δεν είναι ίδια σε όλα τα σημεία της πόλης (αντιθέτως αυτή μεγιστοποιείται σε συγκεκριμένες περιοχές, λόγου χάρη δίπλα σε δρόμους υψηλών ταχυτήτων), θα μπορούσε να χωρίζεται ένα αστικό κομμάτι σε περιοχές ανάλογα με το καυσαέριο που εμφανίζεται σε αυτές και να υπολογίζεται η έκθεση κάποιου σε αυτό, ανάλογα αν διασχίζει το κομμάτι αυτό πεζός, με ποδήλατο ή αυτοκίνητο αλλά και τον χρόνο που εκτίθεται στο καυσαέριο (ανάλογα με την ταχύτητα διάσχισης του συγκεκριμένου κομματιού).

8.5.5 Ψηφιοποίηση και δημιουργία αρχείων

Με προσθήκη κάποιων επιπρόσθετων βοηθητικών εργαλείων ψηφιοποίησης, ο χρήστης θα μπορούσε να έχει στα χέρια του ένα ολοκληρωμένο σχεδιαστικό εργαλείο στο οποίο θα μπορούσε να ψηφιοποιήσει τα δεδομένα που τον ενδιαφέρουν, έχοντας μάλιστα την δυνατότητα επιλογής υποβάθρου (π.χ τους ορθοφωτοχάρτες της Κτηματολόγιο Α.Ε) χωρίς να χρειάζεται να έχει στον τοπικό του υπολογιστή τίποτα (ούτε υπόβαθρο, αλλά ούτε και κάποιο ακριβό σχεδιαστικό πρόγραμμα).

Συμπεράσματα

Τα τελευταία χρόνια το διαδίκτυο έχει μπει στην καθημερινή ζωή της πλειονότητας των πολιτών. Η εξέλιξη των ιστοσελίδων από απλά κείμενα σε διαδραστικές εφαρμογές καθώς και η ανάπτυξη γλωσσών προγραμματισμού που έρχονται να καλύψουν αυτή την ανάγκη, δεν θα μπορούσαν να αφήσουν ανεπηρέαστα τα συστήματα γεωγραφικών εφαρμογών. Η ανάπτυξη εφαρμογών και η δωρεάν διάθεση χαρτογραφικών υποβάθρων από διάφορους παρόχους (λόγου χάρη Google, Yahoo, Microsoft) έδωσαν την δυνατότητα σε απλούς χρήστες να δημιουργήσουν τις δικές τους εφαρμογές. Από την απλή εισαγωγή ενός χάρτη σε μια ιστοσελίδα μέχρι και την δημιουργία σύνθετων εφαρμογών με δυναμική αναζήτηση σε βάσεις δεδομένων, τα Συστήματα Γεωγραφικών Πληροφοριών σταδιακά μετεξελίσσονται από τις καθιερωμένες Desktop εφαρμογές σε διαδραστικές εφαρμογές προσβάσιμες σε όλους από το διαδίκτυο.

Μεγάλη σημασία έχει το γεγονός ότι η πλειονότητα των εργαλείων που έχουν δημιουργηθεί είναι δωρεάν και τα οποία επιτρέπουν σε χρήστες σχεδόν χωρίς καμία ή με ελάχιστες γνώσης προγραμματισμού να αναπτύξουν τις δικές τους εφαρμογές χωρίς κανένα απολύτως κόστος,

Βιβλιογραφία

Στεφανάκης Εμμανουήλ : Τεχνολογίες Δημοσιοποίηση Χαρτογραφικού Περιεχομένου στον Παγκόσμιο Ιστό (Εκδόσεις Νέων Τεχνολογιών, 2009)

Ερευνητικό Πρόγραμμα ΕΜΠ :

Θεματική συμπλήρωση και εξειδίκευση, ως προς το σχεδιασμό δικτύου ποδηλατικών διαδρομών μεγάλου μήκους στην Αθήνα, της έρευνας μέτρων εφαρμογής ενιαίου πολεοδομικού και κυκλοφοριακού σχεδιασμού (ΜΑΡΤΙΟΣ 2011)

ΕΠΙΣΤ. Υπεύθυνος: **Θ. Βλαστός, Καθηγητής ΕΜΠ**

David Flanagan : Javascript, The definitive Guide, 4th Edition
(Εκδόσεις O' REILLY)

Erik Hazzard : OpenLayers 2.10 Beginner's Guide (Εκδόσεις PACKT)

Antonio Santiago Perez : OpenLayers CookBook (Εκδόσεις PACKT)

Josie Wernecke : The KML Handbook: Geographic Visualization for the Web
(Εκδόσεις Kindle)

Brian Youngblood : Geoserver Beginner's Guide (Εκδόσεις PACKT)

Tyler Mitchell: Web Mapping Illustrated Using Open Source GIS Toolkits
(Εκδόσεις O'Reilly Media)

Λιβιεράτος Ε: Γενική Χαρτογραφία (Εκδόσεις Ζήτη)

Κουτσόπουλος Κ: Γεωγραφικά Συστήματα Πληροφοριών και ανάλυση Χώρου" (Εκδόσεις Παπασωτηρίου)

Robinson A, Morrison J, Muehrcke P, Kimerling A and Guptill S:
Στοιχεία Χαρτογραφίας (Εκδόσεις ΕΜΠ)

Κοτζίνος, Δ Πραστάκος, Π. Παπαγεωργίου, Μ. (1999),

Χρήση Δικτυακού ΓΣΠ (internet GIS) στη Σχεδίαση και Υλοποίηση on-line Συστήματος Πληροφοριών για Μεταφορές, Πρακτικά 1 ου Πανελλήνιου Συνεδρίου της Ελληνικής Εταιρείας Γεωγραφικών Συστημάτων Πληροφοριών, Αθήνα, 9-10 Δεκεμβρίου 1999.

Υφαντής Ι. Σαββαΐδης Π.

Γεωγραφικά Συστήματα Πληροφοριών: Ολοκληρωμένη Διαχείριση Εφαρμογών, Α.Π.Θ

Αραποστάθης Ευστάθιος : Εφαρμογές οπτικοποίησης γεωγραφικών δεδομένων στον Παγκόσμιο Ιστό: η περίπτωση του Εθνικού Αρχαιολογικού Μουσείου Αθηνών, Μεταπτυχιακή Εργασία Χαροκόπειο Πανεπιστήμιο

Κουρουτζίδου Φωτεινή : Εφαρμογή Τεχνολογιών δημοσιοποίησης χαρτογραφικού περιεχομένου στον παγκόσμιο ιστό: η περίπτωση του βιβλίου γεωγραφίας της Ε' Δημοτικού. Μεταπτυχιακή Εργασία Χαροκόπειο Πανεπιστήμιο

Φάκα Αντιγόνη: Γεωγραφικά Συστήματα Πληροφοριών στην Τοπική Αυτοδιοίκηση: Η Κοινότητα Πικερμίου. Μεταπτυχιακή Εργασία ΕΜΠ

Ιστοσελίδες

Διαδικτυακή εφαρμογή Geodata.gov.gr : <http://geodata.gov.gr/geodata/>

Διαδικτυακή εφαρμογή Ψηφιακού Χάρτη ΕΜΠ:
<http://map.ntua.gr/google.htm>

Διαδικτυακή εφαρμογή Wikimapia.org :
<http://wikimapia.org/#lat=37.9833&lon=23.7333&z=10&l=14&m=b>

Geoserver User Manual : <http://docs.geoserver.org/stable/en/user/>

Google elevation api:

<https://developers.google.com/maps/documentation/javascript/elevation>

Παράρτημα

Παρακάτω παρατίθενται ο κώδικας των συναρτήσεων που χρησιμοποιήσαμε για την υλοποίηση της εφαρμογής

```
var map;  
var WGS84_google_mercator,formats,vector_layer;  
var WGS84;  
var elevator, chart;  
  
function enhmeroshformat() {  
    var in_options = {  
        'internalProjection': map.baseLayer.projection,  
        'externalProjection': new  
OpenLayers.Projection(OpenLayers.Util.getElement("inproj").value)  
    };  
    var out_options = {  
        'internalProjection': map.baseLayer.projection,  
        'externalProjection': new  
OpenLayers.Projection(OpenLayers.Util.getElement("outproj").value)  
    };  
    var gmlOptions = {  
        featureType: "feature",  
        featureNS: "http://example.com/feature"  
    };  
    var gmlOptionsIn = OpenLayers.Util.extend(  
        OpenLayers.Util.extend({}, gmlOptions),  
        in_options  
    );  
    var gmlOptionsOut = OpenLayers.Util.extend(  
        OpenLayers.Util.extend({}, gmlOptions),  
        out_options  
    );  
    var kmlOptionsIn = OpenLayers.Util.extend(  
        {extractStyles: true}, in_options);  
    formats = {  
        'in': {  
            wkt: new OpenLayers.Format.WKT(in_options),  
            geojson: new OpenLayers.Format.GeoJSON(in_options),  
            georss: new OpenLayers.Format.GeoRSS(in_options),  
            gml2: new OpenLayers.Format.GML.v2(gmlOptionsIn),  
            gml3: new OpenLayers.Format.GML.v3(gmlOptionsIn),  
            kml: new OpenLayers.Format.KML(kmlOptionsIn),  
            atom: new OpenLayers.Format.Atom(in_options),  
            gpx: new OpenLayers.Format.GPX(in_options)  
        },  
        'out': {
```

```

        wkt: new OpenLayers.Format.WKT(out_options),
        geojson: new OpenLayers.Format.GeoJSON(out_options),
        georss: new OpenLayers.Format.GeoRSS(out_options),
        gml2: new OpenLayers.Format.GML.v2(gmlOptionsOut),
        gml3: new OpenLayers.Format.GML.v3(gmlOptionsOut),
        kml: new OpenLayers.Format.KML(out_options),
        atom: new OpenLayers.Format.Atom(out_options),
        gpx: new OpenLayers.Format.GPX(out_options)
    }
};
}

```

```

function init() {
    // ορισμός προβολών
    // προβολή World Geodetic System 1984 (lon/lat)
    WGS84 = new OpenLayers.Projection("EPSG:4326");

    // προβολή WGS84 Google Mercator (meters)
    WGS84_google_mercator = new OpenLayers.Projection("EPSG:900913");
    //Initialize the map

    //δημιουργία χάρτη openlayers map στο
    // <div> html element με id="map"
    map = new OpenLayers.Map ("map", {
        controls: [
            // pan/zoom
            new OpenLayers.Control.Navigation(),
            //προβολή εργαλείων the pan/zoom
            new OpenLayers.Control.PanZoom(),
            //προβολή layer switcher
            new OpenLayers.Control.LayerSwitcher(),
            //προβολή συντεταγμένων της θέσης του ποντικιού
            //σε ένα <div> html element με id coordinates
            new OpenLayers.Control.MousePosition({
                div: document.getElementById("coordinates")
            })
        ],
        projection: WGS84_google_mercator,
        displayProjection: WGS84
    });

    var openstreetmap = new OpenLayers.Layer.OSM();
    var google_maps = new OpenLayers.Layer.Google("Google Maps", {
        numZoomLevels: 20
    });
}

```

```

});

var google_satellite = new OpenLayers.Layer.Google("Google Satellite", {
    type: google.maps.MapTypeId.SATELLITE,
    numZoomLevels: 20
});

KT = new OpenLayers.Layer.WMS("ktimatologio.",
    "http://gis.ktimanet.gr/wms/wmsopen/WmsServer.aspx",
    { layers: "basic", reaspect: "false", transparent: 'false' }, { 'buffer': 0 });
KT.isBaseLayer = true;
KT.setTileSize(new OpenLayers.Size(500, 500));

////////////////////////////////////
// // fortosh arxeioy kml pou periexei //////////////////////////////////
// toys sta8moys metro hsap kai proastiakou////////////////////////////////
// auto to arxeio 8a xrhsimopoih8ei gia na vre8oun
// oi apostaseis apo thn sxediasmenh grammh
    kmlLayer = new OpenLayers.Layer.Vector("point of interest", {
        protocol: new OpenLayers.Protocol.HTTP({
            url: "point of interest.kml",
            format: new OpenLayers.Format.KML({
                extractStyles: true,
                extractAttributes: true,
                internalProjection: WGS84_google_mercator
            })
        }),
        strategies: [new OpenLayers.Strategy.Fixed()]
    });

////////////////////////////////////
vector_layer = new OpenLayers.Layer.Vector('Editable Vectors');
map.addControl(new OpenLayers.Control.EditingToolbar(vector_layer));

    map.addControls(new OpenLayers.Control.ModifyFeature(vector_layer,
{displayClass: 'olControlModifyFeature'}));
var modifyControl = new OpenLayers.Control.ModifyFeature(vector_layer);
map.addControl(modifyControl);

```

```

    map.addLayers([openstreetmap, google_maps,
google_satellite,vector_layer,kmlLayer,KT]);
    var mapextent = new OpenLayers.Bounds(23.78068, 37.97492, 23.78102,
37.98899).transform(WGS84, map.getProjectionObject());
    map.zoomToExtent(mapextent);

    //////////////////////////////////////

    //δημιουργία style
    var vector_style = new OpenLayers.Style({
        'fillColor': '#669933',
        'fillOpacity': .8,
        'strokeColor': 'red',
        'strokeWidth': 3,
        'pointRadius': 8
    });

    //δημιουργία style map object

    var vector_style_map = new OpenLayers.StyleMap({
        'default': vector_style
    });

    //Προσθήκη του style map στα διανύσματα
    vector_layer.styleMap = vector_style_map;

    if(!map.getCenter()){
        map.zoomToMaxExtent();
    }

    //////////////////////////////////////

    function modifyChanged(checked) {
    if(checked) {
        modifyControl.activate();
    } else {
        modifyControl.deactivate();
    }
    }

    //
    // Τροποποίηση της ιδιότητας 'mode'
    //
    function changeMode() {
        var reshape = dijit.byId("reshape").get("checked");
        var resize = dijit.byId("resize").get("checked");
        var rotate = dijit.byId("rotate").get("checked");
        var drag = dijit.byId("drag").get("checked");

        var mode = null;

```

```

    if(reshape) {
        mode |= OpenLayers.Control.ModifyFeature.RESHAPE;
    }
    if(resize) {
        mode |= OpenLayers.Control.ModifyFeature.RESIZE;
    }
    if(rotate) {
        mode |= OpenLayers.Control.ModifyFeature.ROTATE;
    }
    if(drag) {
        mode |= OpenLayers.Control.ModifyFeature.DRAG;
    }

    modifyControl.deactivate();
    modifyControl.mode = mode;
    modifyControl.activate();
}

```

```

function changeFilter(value) {

    modifyControl.deactivate();
    map.removeControl(modifyControl);
    modifyControl.destroy();

    var geometryTypes = null;
    if(value=="POINT") {
        geometryTypes = ["OpenLayers.Geometry.Point"];
    } else if(value=="PATH") {
        geometryTypes = ["OpenLayers.Geometry.LineString"];
    } else if(value=="POLYGON") {
        geometryTypes = ["OpenLayers.Geometry.Polygon"];
    }
    modifyControl = new OpenLayers.Control.ModifyFeature(vectorLayer, {
        geometryTypes: geometryTypes
    });
    map.addControl(modifyControl);
    modifyControl.activate();
}

```

```

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

```

```

vector_layer.events.register('featureadded', this, lineAdded);

```

```

var options = {
    hover: true,
    onSelect: domhsh
};

```

```

var select = new OpenLayers.Control.SelectFeature(vector_layer, options);
map.addControl(select);
        new OpenLayers.Control.LayerSwitcher(),
select.activate();
    enhmeroshformat();

// Δημιουργία υπηρεσίας Elevation
elevator = new google.maps.ElevationService();
// Δημιουργία διαγράμματος στο αντίστοιχο div.
chart = new
google.visualization.LineChart(document.getElementById('elevation_chart'));

};

var kmlLayer;
function lineAdded(e) {
    var feature = e.feature;
    // υψόμετρα
    var nodes =
feature.geometry.clone().transform(WGS84_google_mercator,WGS84).getVertices();

    var locations = [];
    for (var i=0; i<nodes.length; i++) {
        var lon = nodes[i].x;
        var lat = nodes[i].y;

        locations.push(new google.maps.LatLng(lat, lon));
    }
    // console.log("Added " + locations.length + " locations");

    // Create a PathElevationRequest object using this array.
    // Ask for 256 samples along that path.
    var pathRequest = {
        'path': locations,
        'samples': 256
    }

    // αρχικοποίηση του path request.
    elevator.getElevationAlongPath(pathRequest, plotElevation);
    // elevator.getElevationForLocations({locations: locations}, plotElevation);

    //εισαγωγή απόστασης από τον χρήστη
    var maxDistance =
parseFloat(document.getElementById('maxDistance').value);
    if (!maxDistance) {
        // αν ο χρήστης βάλει βλακείες βάζουμε μια προκαθορισμένη τιμή, π.χ. 5000:

```

```

    maxDistance = 5000;
    // και ενημερώνουμε το πεδίο, ώστε να το καταλάβει
    document.getElementById('maxDistance').value = maxDistance;
}

// αποσταση από σταθμό
var distances = [];
for(var i=0;i<kmlLayer.features.length;i++) {
    var station = kmlLayer.features[i];
    var distance = feature.geometry.distanceTo(station.geometry);

    if (distance < maxDistance) {

        station.attributes['distanceFromLine'] = distance;
        // δεν χρειάζεται για αυτό που κάνουμε

        distances.push({
            'station': station.attributes.name,
            'distance': distance
        });
    }
}

if (distances.length > 0) {
    // ορίζουμε κατευθείαν την compare, απλά η διαφορά αρκεί

    distances.sort(function(a,b) {
        // το a, b είναι τώρα αντικείμενα με πεδία: station και distance
        // μας νοιάζει να τα ταξινομήσουμε
        return a.distance - b.distance;
    });

    // χρησιμοποιούμε την ίδια θέση στο HTML και
    var table = document.getElementById("distances");

    table.innerHTML = "";

    // βάζουμε τίτλους στις στήλες
    var header = table.insertRow(-1);
    header.insertCell(-1).innerHTML = 'Σταθμός';
    header.insertCell(-1).innerHTML = 'Απόσταση';

    for (var i = 0; i < distances.length; i++) {
        var row = table.insertRow(-1);
        // το όνομα του σταθμού
        row.insertCell(-1).innerHTML = distances[i].station;
        // και η απόσταση
        row.insertCell(-1).innerHTML = distances[i].distance.toFixed(0) + ' μέτρα';
    }
}

```



```

        // αυτο δεν το θέλουμε πια (;)
        /*
        if (distances.length > 0) {
            var closestStation = distances[0].station;
            var minDistance = distances[0].distance;
            window.alert("Ο κοντινότερος σταθμός είναι ο " + closestStation + ", σε
απόσταση " + minDistance.toFixed(0) + " μέτρα");
        }
        */
    }
}

function plotElevation(results, status) {
    if (status == google.maps.ElevationStatus.OK) {
        var sampleDist =
google.maps.geometry.spherical.computeDistanceBetween(results[0].location,
results[1].location);

        var slopes = calculateSlopes(results);

        // Extract the data from which to populate the chart.
        // Because the samples are equidistant, the 'Sample'
        // column here does double duty as distance along the
        // X axis.
        var data = new google.visualization.DataTable();
        data.addColumn('number', 'Distance');
        data.addColumn('number', 'Elevation');
        data.addColumn('number', 'Slope');
        for (var i = 0; i < results.length; i++) {
            data.addRow([Math.round(i * sampleDist), results[i].elevation, slopes[i]]);
        }

        // σχεδίαση διαγράμματος με τα αντίστοιχα δεδομένα στο div.
        document.getElementById('elevation_chart').style.display = 'block';
        chart.draw(data, {
            width: 800,
            height: 200,
            legend: 'none',
            titleY: 'Elevation (m)',
            title: 'Μέση κλίση: ' + calculateAverage(slopes).toFixed(2) + '%'
        });
    }
}

// elevationData:
function calculateSlopes(elevationData) {
    //////////////////////////////////////
    //ypologismos apostaseon////////////////////////////////////

```

```

////////////////////////////////////////
var distances = [], slopes = [];
for (i=0; i < elevationData.length-1; i++) {
  distances[i] =
google.maps.geometry.spherical.computeDistanceBetween(elevationData[i].location,
elevationData[i+1].location);
  var heightDiff = Math.abs(elevationData[i+1].elevation-
elevationData[i].elevation);
  slopes[i] = heightDiff / distances[i] * 100;
}
//console.log(distances);
//console.log(slopes);

return slopes;
}

// Πίνακας από number
function calculateAverage(values) {
  if (!values.length) {
    return result;
  }
  var result = 0;
  for (var i=0; i<values.length; i++) {
    result+=values[i];
  }
  return result/values.length;
}

function domhsh(feature) {
  var type = document.getElementById("formatType").value;

  var pretty = document.getElementById("prettyPrint").checked;
  var str = formats['out'][type].write(feature, pretty);

  document.getElementById('output').value = str;
}

function apodomhsh() {
  var element = document.getElementById('text');
  var type = document.getElementById("formatType").value;
  var features = formats['in'][type].read(element.value);
  var bounds;
  if(features) {
    if(features.constructor != Array) {
      features = [features];
    }
    for(var i=0; i<features.length; ++i) {
      if (!bounds) {
        bounds = features[i].geometry.getBounds();
      } else {

```

```

        bounds.extend(features[i].geometry.getBounds());
    }

    }
    vector_layer.addFeatures(features);
    map.zoomToExtent(bounds);
    var plural = (features.length > 1) ? 's' : '';
    element.value = features.length + ' feature' + plural + ' added';
} else {
    element.value = 'Bad input ' + type;
}
}

```