



ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΤΜΗΜΑ ΓΕΩΓΡΑΦΙΑΣ



**Ανάπτυξη διαδικτυακής εφαρμογής για τη οπτικοποίηση
γεωγραφικών δεδομένων**

Ντούφας Βασίλειος

Επιβλέπων: Βαΐτης Μιχαήλ

Αναπληρωτής Καθηγητής

Μυτιλήνη, Ιούνιος 2020



ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΤΜΗΜΑ ΓΕΩΓΡΑΦΙΑΣ

**Ανάπτυξη διαδικτυακής εφαρμογής για τη οπτικοποίηση
γεωγραφικών δεδομένων**

Ντούφας Βασίλειος

Επιβλέπων: Βαΐτης Μιχαήλ

Αναπληρωτής Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

.....
Βαΐτης Μιχαήλ

Αναπληρωτής Καθηγητής

.....
Σουλακέλλης Νικόλαος

Καθηγητής

.....
Καβρουδάκης Δημήτριος

Επίκουρος Καθηγητής

Μυτιλήνη, Μάιος 2020

.....

Ντούφας Βασίλειος

M.Sc. Geography and Applied Geo-informatics

geom17019@geo.aegean.gr

Copyright © Ντούφας Βασίλειος

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Αιγαίου.

Περίληψη

Η ανάπτυξη του διαδικτύου και των δυνατοτήτων των ηλεκτρονικών υπολογιστών άσκησε μεγάλη επιρροή στον τρόπο παραγωγής, επεξεργασίας και διαθεσιμότητας των γεωχωρικών δεδομένων. Τα γεωγραφικά συστήματα πληροφοριών που λειτουργούν μέσω διαδικτύου αλλά και η διαδικτυακή χαρτογράφηση, χρησιμοποιείται σε καθημερινή βάση από έναν όλο και αυξανόμενο αριθμό χρηστών ενώ παράλληλα αποκτούν νέες δυνατότητες με ταχύς ρυθμούς. Στα υφιστάμενα λογισμικά δεν υπάρχει ευχρηστία για τους μη έμπειρους χρήστες, χρειάζονται ειδικές άδειες για την πρόσβαση στις πλατφόρμες που κάποιες φορές μπορεί να είναι επί πληρωμή. Στην παρούσα διπλωματική εργασία παρουσιάζεται η μέθοδος και ο τρόπος υλοποίησης μιας διαδικτυακής πλατφόρμας γεωγραφικού συστήματος πληροφοριών ως προς την καταγραφή, επεξεργασία και παρουσίαση γεωγραφικών δεδομένων σε χάρτη. Η πλατφόρμα αναπτύχθηκε χρησιμοποιώντας αυστηρά τεχνολογίες και βιβλιοθήκες ελεύθερου λογισμικού και ανοιχτού κώδικα. Η πλατφόρμα που δημιουργήθηκε είναι σχεδιασμένη ώστε να είναι άμεσα προβάσιμη και λειτουργική μέσω ενός φυλλομετρητή διαδικτύου, σε μορφή ιστοσελίδας, και προσφέρει εργαλεία επεξεργασίας και χωρικής ανάλυσης με εύκολο και κατανοητό τρόπο λειτουργίας στο χρήστη της. Βασικές βιβλιοθήκες που χρησιμοποιήθηκαν για την ανάπτυξη της ιστοσελίδας είναι η Javascript, OpenLayers, JQuery, leaflet και shapefile.js.

Abstract

The growth of the internet and the possibilities offered by computers have significantly influenced the way in which geospatial data is generated, processed and made available. Online geographic information systems and online mapping are used daily by a growing number of people and are gaining new potential at a steady pace. In the existing software there is no usability for inexperienced users, they need special permissions to access the platforms that can sometimes be paid. This dissertation presents the method and manner of implementing an online geographic information platform for the recording, processing and presentation of geographic data on a map. The platform was developed using strictly free software and open source technologies and libraries. The platform created is designed to be an instantly accessible and functional means of an internet browser, in the form of a website, and offers tools for processing and spatial analysis in an easy and understandable way for its user to operate. Key libraries used to develop the website are Javascript, OpenLayers, JQuery, leaflet and shapefile.js.

Περιεχόμενα

Περίληψη	5
Abstract	6
Περιεχόμενα.....	7
Ευρετήριο εικόνων.....	8
Ευρετήριο Πινάκων	8
Κεφάλαιο 1. Εισαγωγή	9
1.1 Σκοπός και αντικείμενο εργασίας	10
Κεφάλαιο 2. Θεωρητικό Υπόβαθρο	11
2.1 Ίντερνετ και GIS	11
2.2 Η τεχνολογία των webGIS	12
2.3 Τεχνολογικά εργαλεία διαδικτύου	16
2.4 Ελεύθερο Λογισμικό / Λογισμικό Ανοιχτού Κώδικα	22
2.5 Πλεονεκτήματα και μειονεκτήματα χρήσης Ελεύθερου Λογισμικού/ Λογισμικού ανοιχτού κώδικα	24
2.6 Λογισμικό Ανοιχτού κώδικα για WebGIS	25
Κεφάλαιο 3. Μεθοδολογία.....	27
3.1 Ανάλυση κώδικα	27
3.2 Πακέτα για την δημιουργία του WebGIS site.....	28
3.1.1 Leaflet	28
3.1.2 Shapefile.js.....	29
3.1.3 jqmarty	29
Κεφάλαιο 4. Αποτελέσματα	31
4.1 Η Εφαρμογή webGIS	31
4.2 Σύγκριση λογισμικών WebGIS.....	37
4.2.1 Esri ArcGIS Server	37
4.2.2 Autodesk MapGuide	41
4.3.3 Συζήτηση.....	45
Κεφάλαιο 5. Συμπεράσματα	46
Βιβλιογραφία	48
Παράρτημα - Κώδικας.....	49

Ευρετήριο εικόνων

Εικόνα 1 Συστατικά μέρη WebGis	13
Εικόνα 2 Τεχνολογικές εξελίξεις του Web Mapping	16
Εικόνα 3 Λογισμικά κλειστού και ανοιχτού λογισμικού.....	24
Εικόνα 4 Αρχιτεκτονική Κώδικα.....	27
Εικόνα 5 Το Ui της εφαρμογής.....	31
Εικόνα 6 Τα εργαλεία της εφαρμογής	32
Εικόνα 7 Εισαγωγή τίτλου στην εφαρμογή	32
Εικόνα 8 Εισαγωγή αρχείου Shapefile	33
Εικόνα 9 Επιλογή χρώματος και ονόματος στο αρχείο εισαγωγής	33
Εικόνα 10 Παρουσίαση αρχείου εισαγωγής στην εφαρμογή	34
Εικόνα 11 Δημιουργία χωρικής ταξινόμησης.....	34
Εικόνα 12 Δημιουργία χωρικής ταξινόμησης.....	35
Εικόνα 13 Παρουσίαση χωρικής κατανομής.....	35
Εικόνα 14 Εμφάνιση πληροφοριών από τα αρχεία shapfile	36
Εικόνα 15 Εισαγωγή πινέζας και κειμένου σε οποιοδήποτε σημείο.	36

Ευρετήριο Πινάκων

Πίνακας 1 Διαφορές Java – JavaScript.....	21
Πίνακας 2 Διαφορές Ανοιχτού και κλειστού κώδικα	23
Πίνακας 3 Σύγκριση Λογισμικών	45

Κεφάλαιο 1. Εισαγωγή

Η ανάπτυξη του διαδικτύου και η αύξηση των ταχυτήτων βοηθάει στην πρόσβαση των χρηστών στο Γεωγραφικών Συστημάτων Πληροφοριών -ΓΣΠ, οι οποίοι εργάζονται από τα γραφεία τους ή από το σπίτι τους να συντελέσουν στην ανάπτυξη εφαρμογών WebGIS. Το διαδίκτυο ως μέσο επικοινωνίας δίνει την δυνατότητα επέκτασης των γεωχωρικών δεδομένων και των προγραμμάτων διαχείρισής τους, καθιστώντας πιο εύκολη την πρόσβαση και την απόκτησή τους σε μια ευρύτερη γκάμα χρηστών. Οι WebGIS εφαρμογές θα παρέχουν εύκολη και αποτελεσματική πρόσβαση στα γεωχωρικά δεδομένα, ώστε να ληφθούν με μεγαλύτερη ευκολία αποφάσεις λειτουργικού, τακτικού και στρατιωτικού χαρακτήρα. . Το διαδίκτυο, το πρωτόκολλο μεταφοράς αρχείων, το πρωτόκολλο μεταφοράς υπερκειμένου, η γλώσσα σήμανσης υπερκειμένου και ο ενιαίος εντοπιστής πόρων, κάνουν πιο εύκολη την πρόσβαση και την μεταφορά αρχείων..

Οι γεωγραφικές επιστήμες όπως η χαρτογραφία και η γεωγραφική ανάλυση του χώρου, είναι λογικό πως εξελίχθηκαν και αυτές και απέκτησαν νέες δυνατότητες που βασίζονται στην χρήση υπολογιστών και διαδικτύου. Βασικά χαρακτηριστικά των δυναμικών εφαρμογών που αναπτύσσονται είναι η μεγάλη ευκολία που παρέχουν για την πρόσβαση και επεξεργασία μεγάλων όγκο πληροφορίας και δεδομένων.

Η υπάρχουσα ποικιλία εφαρμογών Web-GIS είναι μεγάλη και κυμαίνεται από οδηγούς πόλεων, τουριστικές εφαρμογές, ανάλυση κυκλοφοριακών δικτύων, ανάλυση εγκληματικότητας και πολλά άλλα. Θα μπορούσε να ειπωθεί ότι υπάρχουν τόσοι διαδικτυακοί χάρτες διαθέσιμοι στο διαδίκτυο, όσα και τα ενδιαφέροντα των χρηστών που τους δημιουργούν και οι ανάγκες που στοχεύουν να καλύψουν.(Συλαίου, 2012)

Εκτός από την απεικόνιση και χαρτογραφική πλοήγηση, οι χρήστες έχουν πολλές φορές ανάγκη από περισσότερο εξειδικευμένες εργασίες, όπως η δημιουργία θεματικών χαρτών. Οι υπάρχουσες εφαρμογές για το σκοπό αυτό δεν είναι ιδιαίτερα φιλικές προς τον χρήστη και απαιτούν εξειδικευμένες γνώσεις. Στην συγκεκριμένη διπλωματική θα δημιουργήσουμε μια διαδικτυακή εφαρμογή όπου θα αποτελεί ένα μέσον προσβάσιμο και εύκολο στην χρήση για κάθε πολίτη , που θα του δίνει τη δυνατότητα παρουσίασης και επεξεργασίας των γεωχωρικών δεδομένων που επιθυμεί.

Αποδέκτης χωρικών δεδομένων και χρήστης γεωγραφικών συστημάτων πληροφοριών πλέον, είναι δυναμικά οποιοσδήποτε έχει πρόσβαση στο διαδίκτυο, από ένα εξειδικευμένο ερευνητή μέχρι έναν πολίτη χωρίς προηγούμενες γνώσης περί χωρικής ανάλυσης, όπου ο καθένας όμως ανεξάρτητα του επιπέδου γνώσεών του, έχει την δυνατότητα να λάβει χρήσιμα για τον ίδιο αλλά και για τον σκοπό του αποτελέσματα.

1.1 Σκοπός και αντικείμενο εργασίας

Αντικείμενο της συγκεκριμένης εργασίας, είναι τα στάδια σχεδιασμού, η μεθοδολογία και ο τρόπος υλοποίησης ενός διαδικτυακού γεωγραφικού συστήματος πληροφοριών (Web-GIS) με χρήση εξολοκλήρου τεχνολογιών Ελεύθερου Λογισμικού/Ανοιχτού Κώδικα (ΕΛ/ΑΚ).

Μέσω της συγκεκριμένης εφαρμογής δημιουργείται μία πλατφόρμα προβάσιμη και διαθέσιμη δωρεάν για τον καθένα μέσω διαδικτύου, που προσφέρει στον χρήστη:

- τη δυνατότητα εισαγωγής γεωχωρικών δεδομένων
- τη δυνατότητα απεικόνισης των δεδομένων αυτών στον Ελλαδικό χώρο μέσω της δημιουργίας δυναμικών χαρτών
- τη δυνατότητα καθορισμού κλάσεων και ορίων ταξινόμησης σε αριθμητικά πεδία των γεωχωρικών δεδομένων κατά τη δημιουργία θεματικών χαρτών.

Επιπλέον θα πραγματοποιηθεί σύγκριση με άλλες διαδικτυακές εφαρμογές ανοιχτού ή κλειστού κώδικα σε σχέση με τον τρόπο επεξεργασίας, απεικόνισης και εξαγωγής των αποτελεσμάτων.

Σκοπός της διαδικτυακής εφαρμογής είναι να αποτελεί ένα μέσον προσβάσιμο και εύκολο στην χρήση για κάθε πολίτη που παρέχει επεξεργασία και παρουσίαση των γεωχωρικών δεδομένων χωρίς να χρειαστεί να έχει προ εγκατεστημένη κάποια εφαρμογή επί πληρωμή.

Κεφάλαιο 2. Θεωρητικό Υπόβαθρο

2.1 Ίντερνετ και GIS

Από το 1995 και μετά παρατηρήθηκε ότι το internet αποτελεί μια κυρίαρχη δύναμη στις παγκόσμιες επικοινωνίες (Smith, 1986). Παράλληλα, ο συνδυασμός του με τα γεωγραφικά συστήματα πληροφοριών αύξησε τη δυνατότητα διανομής γεωχωρικής πληροφορίας των GIS συστημάτων στους χρήστες, εκτινάσσοντας την επάρκεια και την αποτελεσματικότητά τους. Η δυναμική βοήθησε τα Γεωγραφικά Συστήματα Πληροφοριών να διεισδύσουν σε μια εντελώς νέα γκάμα πεδίων και εφαρμογών. Διαμορφώθηκε έτσι μια νέα τεχνολογία γνωστή ως Web-GIS. Η πλατφόρμα των Web-GIS συστημάτων παρέχει εργαλεία για ευέλικτη διαχείριση των γεωγραφικών δεδομένων, δημιουργώντας ένα οπτικό αποτέλεσμα με δυναμικά χαρακτηριστικά που παρουσιάζεται στην οθόνη του ηλεκτρονικού υπολογιστή του κάθε χρήστη. Κύρια χαρακτηριστικά της τεχνολογίας αυτής θεωρούνται:

1. Η ενσωμάτωση μιας πληθώρας μέσων διακίνησης πληροφοριών, δεδομένων και τεχνολογίας.
2. Η πρόσβαση σε βάσεις δεδομένων, όπου κατανομούνται χωρικά, και σε μια ποικιλία δομών με γεωγραφικά δεδομένα.
3. Παροχή δυνατοτήτων χωρικής ανάλυσης.

Η εξέλιξη στον τομέα των ηλεκτρονικών υπολογιστών, με την αύξηση της υπολογιστικής τους ισχύς, επιτρέπουν πλέον στον χρήστη να έχει πρόσβαση στα δεδομένα αυτά, δίχως πλέον να χρειάζεται εξειδικευμένο λογισμικό, αλλά χρησιμοποιώντας έναν κοινό φυλλομετρητή (Web-browser). Λόγω των μεγάλων πλεονεκτημάτων που έχει επιφέρει η υπηρεσία GIS στο διαδίκτυο, η διάδοση των υπηρεσιών αυτών παγκοσμίως είναι ευρεία. Γι' αυτό και ο σκοπός των εταιρειών που καταπιάνονται με την υπηρεσία του Web-GIS, είναι να δημιουργήσουν απλούστερα και πιο εύχρηστα συστήματα που θα δύναται να τα χρησιμοποιήσει ακόμα και ένας αρχάριος χρήστης.

Τα Συστήματα Γεωγραφικών Πληροφοριών μέσω διαδικτύου ή αλλιώς τα συστήματα Internet GIS κατέχουν ένα μεγάλο τμήμα στο κομμάτι της έρευνας και των εφαρμογών στον τομέα των ΓΣΠ (GIS), ενώ παράλληλα κάνουν ένα σημαντικό και μεγάλο βήμα για την εξέλιξη των ήδη υπάρχοντων συστημάτων ΓΣΠ σταθερού τύπου (desktop GIS). Κυβερνητικοί και εκπαιδευτικοί οργανισμοί αλλά και παραγωγοί ή απλοί χρήστες γεωχωρικών δεδομένων, έχουν επωφεληθεί από αυτά τα συστήματα, μιας και αυτά έχουν αναπτυχθεί να έχουν μια μεγάλη γκάμα δυνατοτήτων, από την απλή χαρτογράφηση μέχρι τη διαδραστική λειτουργία χάρτη με εξατομικευμένα προφίλ χρηστών με δυνατότητες μεγάλης εξειδίκευσης. (Gis, 2008)

Μία επιτυχημένη εφαρμογή GIS πρέπει να απαρτίζεται από τη σωστή πλατφόρμα λειτουργίας, τη σωστή αρχιτεκτονική και την υποστήριξη των κατάλληλων λογισμικών. Οι εξυπηρετητές (server), που διαχειρίζονται την πληροφορία πρέπει να είναι ανεξάρτητοι, αξιόπιστοι, ευέλικτοι και ασφαλείς. Για την επίτευξη ενός επιτυχημένου GIS συστήματος απαιτείται ένα καθολικό σχέδιο του όλου συστήματος, ώστε να αποφευχθούν πρόχειρες λύσεις, αλλά και επιλογή τεχνολογιών, που ενέχουν τον κίνδυνο να έχουν ξεπεραστεί μέσα σε δύο ή τρία χρόνια. Όλες οι προσπάθειες που πραγματοποιούνται τα τελευταία χρόνια στα πλαίσια των παραπάνω κατευθυντήριων

γραμμών επικεντρώνονται σε όσο το δυνατόν καλύτερες και πλήρεις απαντήσεις στα παραπάνω ερωτήματα.

Το Internet είναι ένα μοντέρνο πληροφοριακό σύστημα που δίνει την δυνατότητα να συνδεθούν εκατοντάδες χιλιάδες τηλεπικοινωνιακά δίκτυα. Τα Γεωγραφικά Συστήματα Πληροφοριών που αναπτύσσονται στο διαδίκτυο, αποτελούν μία περιοχή έρευνας και εφαρμογής που χρησιμοποιεί το Internet καθώς και άλλα υπερδίκτυακά συστήματα (συμπεριλαμβανομένων και των ασύρματων συστημάτων καθώς και των τοπικών δικτύων intranet) για να εξυπηρετηθεί η πρόσβαση, η επεξεργασία και η διάδοση των γεωγραφικών πληροφοριών καθώς και η γνώση της χωρικής ανάλυσης. Η όλη ανάπτυξη του διαδικτύου, επηρεάζει τα Γεωγραφικά Συστήματα Πληροφοριών σε τρεις διαφορετικές περιοχές: Στην πρόσβαση των δεδομένων GIS Στη διάδοση των χωρικών πληροφοριών και Στη μοντελοποίηση – επεξεργασία των Γεωγραφικών Συστημάτων Πληροφοριών. Η πιο σημαντική προφορά της συγκεκριμένης τεχνολογίας είναι η δυνατότητα επεξεργασίας, αναζήτησης και ανάλυσης χωρικών δεδομένων και αντικειμένων χωρίς να είναι απαραίτητη η αγορά κάποιου εμπορικού πακέτου GIS. Ταυτόχρονα ενισχύεται η πρόσβαση και η επαναληπτική χρήση εργαλείων ανάλυσης «κατεβάζοντας» ή «ανεβάζοντας» στο δίκτυο τα κατάλληλα εργαλεία και συστατικά μέρη ενός GIS. Στο μέλλον προβλέπεται ότι οι χρήστες θα έχουν τη δυνατότητα να εργάζονται δυναμικά με τα γεωγραφικά δεδομένα χωρίς να είναι απαραίτητη η εγκατάσταση οποιουδήποτε λογισμικού, με μοναδικό προαπαιτούμενο έναν απλό web browser.

Ένας browser (φυλλομετρητής - περιηγητής Ιστού) αποτελεί το πρόγραμμα με το οποίο κάθε χρήστης έχει τη δυνατότητα να επισκεφτεί διαδικτυακούς τόπους (ιστοσελίδες). Στην ουσία, το λογισμικό αυτό επιτρέπει στον επισκέπτη να αλληλοεπιδρά με τα χαρακτηριστικά αυτού του τόπου (απλές εικόνες, βίντεο, κείμενα απλής μορφής, μουσική, εφαρμογές flash, κείμενα/εικόνες που περιέχουν υπερσυνδέσμους προς άλλους ιστότοπους, κλπ) που είναι αναρτημένα σε ένα τοπικό δίκτυο ή στον παγκόσμιο ιστό. Ένας browser χρησιμοποιεί γλώσσα HTML για την προβολή των ιστοσελίδων. Για τον λόγο αυτό μια σελίδα ενδέχεται να εμφανίζεται με διαφορετικούς τρόπους ανάλογα με την αρχιτεκτονική του browser που χρησιμοποιείται. (Putz, 1994)

Ενδιαφέρον μάλιστα παρουσιάζει και η ανάπτυξη Open Source εφαρμογών, στις οποίες εξειδικευμένοι χρήστες μπορούν να επέμβουν στον πηγαίο κώδικά του λογισμικού και να τον διαμορφώσουν ανάλογα με τις επιθυμίες/ανάγκες τους. Η ανάπτυξη εφαρμογών Open Source καθώς και λογισμικών που επιτρέπουν την δυναμική επεξεργασία χωρικών και μη δεδομένων, αποτελεί σημαντικό αντικείμενο έρευνας με αποτέλεσμα όλες οι προσπάθειες και όλο το ενδιαφέρον των εταιρειών παραγωγής λογισμικών να περιστρέφεται ,τα τελευταία χρόνια, γύρω από αυτόν τον τομέα.

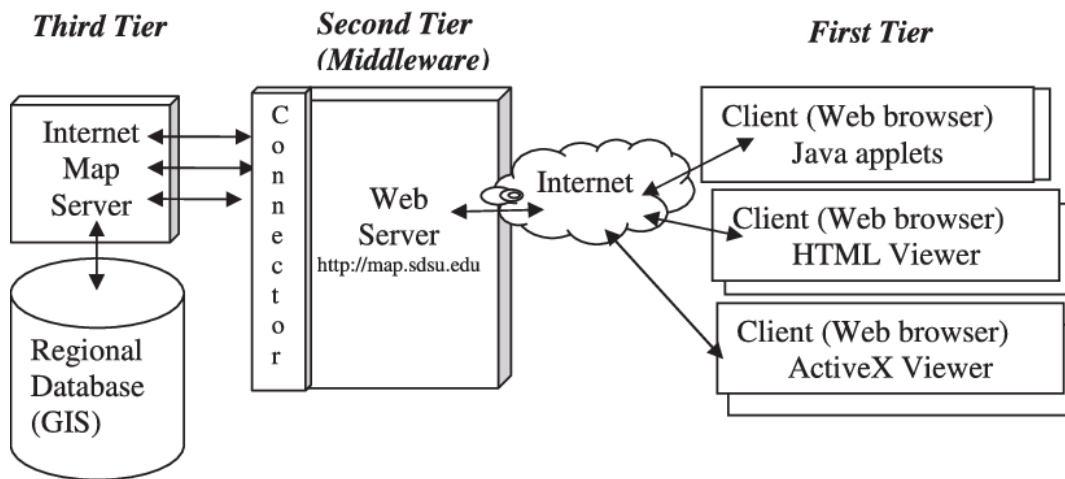
2.2 Η τεχνολογία των webGIS

Το περιβάλλον εργασίας των συστημάτων Web-GIS μπορεί πολύ εύκολα να γίνει κατανοητό. Ο χρήστης θέτει ένα ερώτημα το οποίο αποστέλλεται στον εξυπηρετητή του συστήματος. Ο web server με τη σειρά του κατευθύνει το ερώτημα στον application server, στον οποίο είναι

αποθηκευμένη όλη πληροφορία. Στην συνέχεια πραγματοποιείται επεξεργασία του ερωτήματος και συλλέγεται η απαιτούμενη πληροφορία η οποία μεταφέρεται στον Map Server και αυτός με τη σειρά του δημιουργεί έναν χάρτη σε περιβάλλον HTML, ο οποίος παρουσιάζεται στον χρήστη (client) ως απάντηση στο ερώτημα που έθεσε. Συνήθως αυτά τα αποτελέσματα παρέχονται με την μορφή εικόνων (image services), που αποτελεί για τον χρήστη μια απλή και φιλική μορφή για τη λειτουργία του συστήματος. Αντίστοιχα έχουν δημιουργηθεί και πιο σύνθετες εφαρμογές με δυναμικούς χάρτες στους οποίους δίνεται η δυνατότητα στον χρήστη να επεξεργαστεί την πρωτογενή πληροφορία όπως αυτή είναι αποθηκευμένη στο κεντρικό σύστημα (feature services). Αυτού του είδους οι εφαρμογές απευθύνονται σε πιο απαιτητικούς χρήστες, οι οποίοι διαθέτουν ισχυρά υπολογιστικά συστήματα με κατάλληλα λογισμικά (π.χ. εγκατεστημένη την γλώσσα προγραμματισμού java). (Fu, 2015)

Τα τέσσερα βασικά μέρη των Internet GIS συστημάτων, φαίνονται στην παρακάτω εικόνα (εικόνα 1) και αποτελούνται από:

- Τον πελάτη (client)
- Τον εξυπηρετητή διαδικτύου (web server) και τον εξυπηρετητή εφαρμογής (application server)
- Τον εξυπηρετητή των χαρτών (map server) και τον εξυπηρετητή της βάσης δεδομένων (data server)



Εικόνα 1 Συστατικά μέρη WebGis

Με τον όρο «client» εννοούμε το περιβάλλον εργασίας, όπου δίνεται στο χρήστη η δυνατότητα αλληλεπίδρασης με το λογισμικό των Internet GIS. Στα συστήματα αυτά, η αλληλεπίδραση με το λογισμικό βασίζεται στις λειτουργίες του διαδικτύου αλλά και στα διάφορα πρόσθετα υποπροϊόντα (add-ons), σε αντίθεση με τις περισσότερες παραδοσιακές εφαρμογές desktop-GIS που βασίζονται σε ένα γραφικό περιβάλλον για την παρουσίαση των λειτουργιών τους. Τα add-ons προστίθενται από τον εκάστοτε χρήστη (plug-ins), προσαρμόζοντας έτσι το λογισμικό στις ανάγκες του, με απώτερο σκοπό πάντα την βέλτιστη αλληλεπίδραση του με το σύστημα αλλά και με τα χωρικά δεδομένα τα οποία αυτό διαχειρίζεται. Τα plug-ins που αναπτύσσονται για τα Γεωγραφικά Συστήματα Πληροφοριών διαδικτύου, έχουν σαν στόχο να αυξήσουν τη λειτουργικότητα και την ευελιξία των συστημάτων, ώστε ο κάθε χρήστης να μπορεί να συντάξει

τα χωρικά ερωτήματα, που αυτός επιθυμεί, επιλέγοντας χαρακτηριστικά πάνω στον χάρτη που θα απεικονίζεται.

Ο Web Server, δέχεται τα αιτήματα των χρηστών, διανέμει τις ιστοσελίδες και θέτει σε λειτουργία τους εξυπηρετητές εφαρμογών. Ο Web Server αποκαλείται πολλές φορές και HTTP Server. Τα ερωτήματα των χρηστών απαντώνται με διάφορους τρόπους, οι οποίοι είναι:

- Στέλνει στον χρήστη HTML αρχεία ή εικόνες χαρτών (έτοιμες).
- Στέλνει στον χρήστη Java applets ή ActiveX Controls.
- Μεταβιβάζει ερωτήματα σε άλλα προγράμματα, θέτοντας τα σε λειτουργία για να επεξεργαστούν αυτά το ερώτημα. (πχ. Ένα CGI)
- Όταν ο web server πραγματοποιεί μεταφορά αιτημάτων σε άλλα προγράμματα, παράλληλα ενεργοποιεί και εφαρμογές από τους εξυπηρετητές εφαρμογής (application server).

Ο Application Server ή όπως λέγεται στα ελληνικά, ο εξυπηρετητής εφαρμογής, αναλαμβάνει τη διαχείριση των συναλλαγών του web server και την ασφάλεια του, ρυθμίζοντας παράλληλα την ισορροπία του συστήματος. Ένας Application Server δύναται να είναι ένα σταθερό πρόγραμμα ή ένα μέσο ώστε να συνδέεται ο Web Server με “πλαϊνές” εφαρμογές του Server (Server-side applications) όπως ένας Server χαρτών. Κύριες λειτουργίες του application server είναι η πραγματοποίηση, η διατήρηση και ο τερματισμός των συνδέσεων μεταξύ web server και map server, την λήψη, ερμηνεία και διανομή των αιτημάτων και τη διαχείριση παράλληλων αιτημάτων από τους χρήστες στον map server, και τη διαχείριση των αρχείων που ανταλλάσσονται μεταξύ του map server του data server.

Ο Map Server, δηλαδή ο εξυπηρετητής χαρτών, λαμβάνει τα αιτήματα των χρηστών από τον application server, τα επεξεργάζεται και παράγει τους χάρτες που ο χρήστης επιθυμεί. Έχει τη δυνατότητα να εκπληρώσει τα χωρικά ερωτήματα που γίνονται από τον χρήστη, να πραγματοποιήσει χωρική ανάλυση, να παράγει τους χάρτες με τα αιτήματα που έχουν θέσει οι χρήστες και να τους διανεμίει. Ο map server αναφέρεται συχνά και ως spatial server σε διάφορα λογισμικά πακέτα και η ονομασία όπως είναι κατανοητό οφείλεται κατά κόρον στις λειτουργίες που επιτελεί. Αυτές οι υπηρεσίες θα μπορούσαν να βρίσκονται σε διαφορετικούς εξυπηρετητές ως ξεχωριστά συστατικά μέρη.

Τα εξαγόμενα προϊόντα ενός map server μπορεί να έχουν μια από τις δύο μορφές που περιγράφονται στη συνέχεια:

- Δεδομένα, τα οποία έχουν υποστεί φιλτράρισμα, στέλνονται στο πρόγραμμα του χρήστη για περαιτέρω επεξεργασία.
- Απλή απεικόνιση του χάρτη σε γραφική μορφή, δηλαδή μια εικόνα που μπορεί να είναι της μορφής .gif (Graphics Interchange Format) ή .jpeg (Joint Photographic Experts Group) ή από έναν γραφικό χάρτη που περιλαμβάνει στοιχεία που ανταποκρίνονται στις ανάγκες του κάθε χρήστη που θα παρουσιάζονται με συγκεκριμένα χρώματα, υπομνήματα, στυλ, κλπ.

Τέλος, ο data server, δηλαδή ο εξυπηρετητής δεδομένων, φροντίζει τη διανομή των χωρικών/μη χωρικών δεδομένων, από μια βάση δεδομένων. Για την πρόσβαση και τη διαχείριση αυτών των δεδομένων πραγματοποιείται μέσω διαφόρων προγραμματιστικών γλωσσών. Η πιο διαδεδομένη για αυτή τη χρήση είναι η γλώσσα SQL (Structured Query Language) που πρόκειται για προγραμματιστική γλώσσα. Υπάρχουν τρεις βασικοί τρόποι πρόσβαση σε βάσεις δεδομένων:

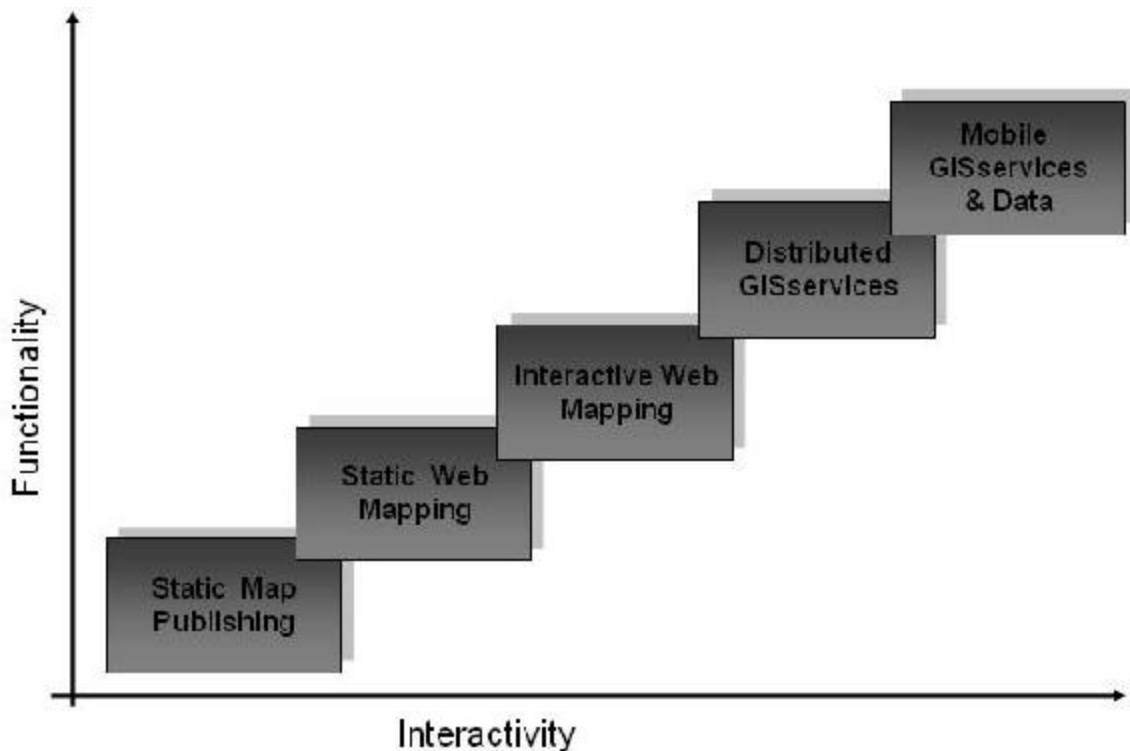
1. Μέσω ODBC
2. Μέσω Java Database Connectivity (JDBC)
3. Μέσω Object Linking and Embedding Database (OLE DB) ActiveX Data Object (ADO)

Μέσω της SQL, του ODBC ή του JDBC η εφαρμογή του χρήστη δύναται να συντάξει ερωτήματα, να ανακτήσει και να τροποποιήσει τις εγγραφές της βάσης δεδομένων στον Data Server.

Η ανάπτυξη των Γεωγραφικών Συστημάτων Πληροφοριών στο διαδίκτυο, ακολουθεί την πρόοδο της τεχνολογίας των υπολογιστών και των τηλεπικοινωνιακών δικτύων. Όπως αναφέρθηκε και σε προηγούμενες παραγράφους, τα συστήματα αυτά απαιτούσαν σημαντική υπολογιστική ισχύ, που κανένας προσωπικός ηλεκτρονικός υπολογιστής δεν διέθετε τις προηγούμενες δεκαετίες. Η εξέλιξη των ίδιων των συστημάτων, με τα χρόνια, και η αύξηση της μέσης επεξεργαστικής ισχύς στους προσωπικούς ηλεκτρονικούς υπολογιστές, οδήγησαν σε καταναεμημένες υπηρεσίες GIS που ενημερώνονται δυναμικά και προσπελούνται από πολλούς χρήστες ταυτόχρονα. (Veenendaal, 2011)

Η τεχνολογική εξέλιξη των Web GIS όπως φαίνεται στην εικόνα 2 ξεκίνησε με δημοσίευση στατικών χαρτών και εξελίχθηκε σε στατική χαρτογράφηση μέσω διαδικτύου, μέσω αλληλεπιδραστικών συστημάτων Web GIS και τελικά σε καταναεμημένες υπηρεσίες GIS. Στα αρχικά στάδια ανάπτυξης της τεχνολογίας διαδικτύου γινόταν χρήση έτοιμων χαρτών, στους οποίους η πρόσβαση γινόταν μέσω υπερσύνδεσης (hyperlink). Οι χάρτες είναι συνήθως τμήμα ενός HTML κειμένου για να εμπλουτίσουν τα περιεχόμενα αυτού. Από την άλλη οι χρήστες δεν μπορούσαν να αλληλεπιδράσουν με τους χάρτες ή να αλλάξουν την απεικόνισή τους με τον οποιονδήποτε τρόπο.

Το δεύτερο στάδιο είναι η στατική χαρτογράφηση μέσω διαδικτύου. Υλοποιείται μέσω HTML και CGI για την σύνδεση του πελάτη (μέσω web browser) με το GIS ή με άλλα προγράμματα χαρτογράφησης στους εξυπηρετητές. Οι πελάτες συντάσσουν αιτήματα, τα οποία αποστέλλονται από τον web browser. Στη συνέχεια το σύστημα GIS ή οι «μηχανές» χαρτογράφησης με τη σειρά τους δημιουργούν το χάρτη ανάλογα με το αίτημα του πελάτη και παράγουν μία εικόνα χάρτη σε πραγματικό χρόνο. Η νέα εικόνα αποστέλλεται πίσω στο χρήστη μέσω HTTP. Εντούτοις, το μειονέκτημα των τεχνολογιών χαρτογράφησης μέσω διαδικτύου έγκειται στο ότι η απόδοση του HTTP με το CGI είναι αργή, δυσκίνητη και ασταθής. Η τεχνολογία HTTP βασίζεται σε κείμενο και επιτρέπει περιορισμένη εισαγωγή και επεξεργασία από εξωτερικούς παράγοντες. Για παράδειγμα οι χρήστες δεν μπορούν να ορίσουν ή να ζωγραφίσουν έναν κύκλο ή ένα τετράγωνο πάνω στις εικόνες χάρτη.



Εικόνα 2 Τεχνολογικές εξελίξεις του Web Mapping (Zhong-Ren Peng and J. G. Speight. 2003)

Το τρίτο στάδιο εξέλιξης αφορά την αλληλεπίδραση του χρήστη της υπηρεσίας και της χαρτογράφησης μέσω διαδικτύου, όπου η αλληλεπίδραση αυτή υλοποιείται μέσω scripts όπως δυναμικά HTML και εφαρμογές όπως plug-ins, ActiveX controls και Java applets. Κάποια από τα αιτήματα που δημιουργούνται μπορεί να τα επεξεργαστεί ο πελάτης-χρήστης χωρίς την αποστολή τους σε κάποιον εξυπηρετητή. Αλλά και αυτή η προσέγγιση απαιτεί δικτυακή σύνδεση με τους εξυπηρετητές διαδικτύου (web servers) να διαμεσολαβούν για την αποθήκευση της πληροφορίας στους εξυπηρετητές.

Το τέταρτο και τελευταίο στάδιο είναι οι web υπηρεσίες, όπου τα συστατικά μέρη του GIS , επιτρέπουν στον πελάτη να μπορεί να επικοινωνεί απευθείας με τα συστατικά μέρη του εξυπηρετητή, χωρίς να είναι απαραίτητη η ύπαρξη ενός εξυπηρετητή HTTP ή ενός διαμεσολαβητή CGI. Οι κατακευματισμένες υπηρεσίες GIS βασίζονται στην επικοινωνία μεταξύ χρήστη (CORBA/Java ORB ή Microsoft SOAP) και εξυπηρετητή (CORBA/IIOP και Java ή .NET/COM).

2.3 Τεχνολογικά εργαλεία διαδικτύου

Οι λέξεις Hyper Text Markup Language (γλώσσα σήμανσης κειμένου) δημιουργούν εδώ, το ακρωνύμιο HTML, το οποίο είναι το όνομα της βασικής γλώσσας προγραμματισμού. Δημιουργήθηκε στη δεκαετία του 1980 από τον φυσικό Τιμ Μπερνερς Λι., ο οποίος εργαζόταν στο CERN. Επινόησε και κατασκεύασε το σύστημα χρήσης και διαμοιρασμού εγγράφων το λεγόμενο ENQUIRE. Στη συνέχεια δημιούργησε την προδιαγραφή της HTML, σύνθεσε τον

browser και το λογισμικό του εξυπηρετητή. Έτσι, το 1991 δημοσιεύτηκε η πρώτη μορφή της γλώσσας και ήταν ένα έγγραφο το οποίο ονομαζόταν ετικέτες HTML καθώς και μια περιγραφή για τα 20 στοιχεία που απάρτιζαν το σχεδιαστικό της μέρος. Οι περισσότερες ετικέτες της HTML ήταν σε μεγάλο βαθμό επηρεασμένες από μια άλλη μορφή τεκμηρίωσης φτιαγμένη και αυτή στο CERN, την SGMLguid, που είναι βασισμένη με τη σειρά της στην SGML. Η μόνη, ίσως, πρωτότυπη ετικέτα που δημιουργήθηκε τότε ήταν αυτή του υπερσυνδέσμου. 13 από αυτές, υπάρχουν και στην τελευταία έκδοση της HTML, ακόμα και σήμερα. Το SGML έχει εμπνευστεί και ενσωματώνει μερικές από τις τεχνικές των τυπογράφων, διαχωρίζει τη δομή από το περιεχόμενο και πάνω σε αυτήν στηρίζονται η HTML και η CSS. Οι ετικέτες της HTML, οι οποίες αποτελούν τη γλώσσα, περιλαμβάνουν μέσα στο περιεχόμενο της σελίδας, τα σύμβολα “<” και “>”, δηλαδή το «μεγαλύτερο από» και το «μικρότερο από» αντίστοιχα. Η συνήθη λειτουργία των ετικετών της HTML είναι σε ζεύγη, με την πρώτη να παίρνει την ονομασία *ετικέτα έναρξης* και η δεύτερη την ονομασία *ετικέτα λήξης*. Παράδειγμα αποτελεί το: <h1> και </h1>. Μεταξύ αυτών, οι προγραμματιστές ή οι σχεδιαστές που θα δημιουργήσουν μια σελίδα με την HTML, θα μπορούν να τοποθετήσουν κείμενο, πίνακες και εικόνες. Ο web browser είναι φτιαγμένος ώστε να διαβάζει την HTML (ερμηνεία των HTML ετικετών) και να συνθέτει ένα οπτικοακουστικό αποτέλεσμα για τον χρήστη. Ανάλογα τη διαφορετική υλοποίηση του κάθε browser, παρατηρούνται διαφορές στον τρόπο ερμηνείας των ετικετών. Με την HTML ενσωματώνονται εικόνες, αρχεία ήχου αλλά και άλλα αντικείμενα μέσα στη σελίδα που εξυπηρετούν στη διαδραστική φύση της σελίδας. Οι μέθοδοι για τη δημιουργία εγγράφου, από τους οποίους επίσης αποτελείται η HTML, περιέχουν το πρωταρχικό κείμενο αλλά και τον κώδικα που θα το μορφοποιήσει. Ο κώδικας της μορφοποίησης περιέχει τα στοιχεία του κειμένου όπως επικεφαλίδες, λίστες ,παραγράφους, συνδέσμους κλπ.(Avenue, 1810; Brooks, 2007)

Στη συγκεκριμένη πτυχιακή εργασία χρησιμοποιήθηκε η πιο πρόσφατη έκδοση της γλώσσας προγραμματισμού HTML, η HTML 5. Το σημαντικό, ίσως, πλεονέκτημα αυτής της έκδοσης είναι πως μας επιτρέπει την δημιουργία mobile web sites” τα οποία ενσωματώνουν πληθώρα δυνατοτήτων και ιδιοτήτων από τις εφαρμογές που συναντάμε στις διάφορες πλατφόρμες κινητών τηλεφώνων με αποτέλεσμα το mobile web site να αποκτά μια πιο ιδανική μορφή για χειρισμό του από ένα κινητό τηλέφωνο. Δημιουργήθηκε έτσι η ανάγκη των standards και της πλήρη συμβατότητας. Οι ανάγκες αυτές, ώθησαν το World Wide Web Consortium (W3C) και το Web Hypertext Application Technology Working Group (WHATWG) να συνεργαστούν και να δημιουργήσουν την εν λόγω αναβαθμισμένη έκδοση της HTML. νέα αυτή γλώσσα.

Οι εφαρμογές των κινητών τηλεφώνων εγκαθίστανται στο λειτουργικό σύστημά τους, με αποτέλεσμα να χαρίζουν μια εμπειρία χρήσης που θα αξιοποιεί όλες τις τεχνικές δυνατότητές του στο έπακρον. Ένα τυπικό mobile web site, σε αντίθεση με μια εφαρμογή, δεν έχει τέτοιες δυνατότητες. Ένα από τα σημαντικά χαρακτηριστικά του HTML 5 είναι πως μπορεί να προσφέρει GPS, ένα ιδιαίτερος χρήσιμο χαρακτηριστικό για τη διαφήμιση και το ηλεκτρονικό εμπόριο. Επίσης, το HTML 5 δύναται να αναπαράγει δεδομένα οπτικοακουστικού χαρακτήρα, χωρίς τη χρήση flash.

Κάποιοι βασικοί κανόνες που έχουν οριστεί για την HTML5 είναι:

- HTML, CSS, DOM, και η JavaScript (βάσεις νέων χαρακτηριστικών)
- Ανάγκη για λιγότερα plugins (όπως το Flash)
- Ακριβέστερος εντοπισμός
- Markup για να αντικατασταθεί το scripting
- Συμβατότητα σε όλες τις συσκευές

Μία γλώσσα που θα μπορούσε να φανεί χρήσιμη στο στήσιμο της συγκεκριμένης πτυχιακής είναι η Java. Η Java, αρχικά αναπτύχθηκε από τη Sun και τον James Gosling συγκεκριμένα, ο οποίοςς πειραματιζόταν πάνω στην γλώσσα C++ . Ο Gosling, πριν να οριστικοποιήσει την γλώσσα Java, είχε δημιουργήσει την πρόγονό της, μία γλώσσα με το όνομα Oak. Η Oak έμοιαζε αρκετά με την γλώσσα C++ αλλά ήταν πολύ πιο απλή. Η Oak μετονομάστηκε σε Java (ποικιλία καφέ, αγαπημένη στους δημιουργούς της) διότι το πρώτο όνομα ήταν κατοχυρωμένο ήδη, και παρουσιάστηκε επίσημα στα μέσα του 1995.(Ken Arnold, James Gosling, 2005)



Ένα μεγάλο πλεονέκτημα της γλώσσας Java συγκριτικά με τις άλλες γλώσσες, είναι πως το λειτουργικό της σύστημα και η πλατφόρμας είναι ανεξάρτητα. Δηλαδή, προγράμματα γραμμένα σε Java εκτελούνται με τον ίδιο τρόπο σε Linux, Unix, Macintosh και Windows, χωρίς να απαιτείται εκ νέου μεταγλώττιση (δηλ. compiling) ή να πραγματοποιηθούν αλλαγές στον πηγαίο κώδικα ανάλογα το λειτουργικό σύστημα που πρόκειται να εκτελεστεί το πρόγραμμα.

Τα χαρακτηριστικά της γλώσσας προγραμματισμού Java χωρίζονται σε 8 κατηγορίες:

1. Απλή

Όπως προαναφέρθηκε, η βάση της γλώσσας Java είναι η C++, αλλά με λιγότερα στοιχεία, όπως η αφαίρεση της έννοιας των δεικτών και τα προβλήματα τα οποία έπρεπε να επιλυθούν εξαιτίας τους. Άρα, η Java είναι σίγουρα πιο απλή και ευκολονόητη διότι, οι προγραμματιστές πρώτα μαθαίνουν την C++ και έτσι στη συνέχεια τους είναι πολύ εύκολο να μάθουν την Java. Επίσης, περιλαμβάνει βασικούς τύπους δεδομένων τριών τύπων: α) αριθμούς, β) λογικές μεταβλητές (Boolean) και γ)τους πίνακες. Βασικό χαρακτηριστικό της Java είναι η διαχείριση μνήμης, γνωστό ως "μονάδα συλλογής σκουπιδιών» (garbage collector) και γίνεται αυτόματα. Σε επίπεδο προγραμματισμού, δεν χρειάζεται πλέον ο δημιουργός να φροντίζει για την απελευθέρωση της

μνήμης που δεσμεύει και διαχειρίζεται , γιατί η εικονική μηχανή της Java (Virtual Machine Java - VMJ) το φροντίζει αυτό μόνη της. Τέλος, υποστηρίζεται η δημιουργία των πολυνηματικών προγραμμάτων.

2. Αντικειμενοστραφής

Στον κόσμο της πληροφορικής, ο όρος αντικειμενοστραφής προγραμματισμός - ΑΠ (object-oriented programming), αναφέρεται σε ένα υπόδειγμα που εμφανίστηκε στα τέλη της δεκαετίας του '60 ενώ καθιερώθηκε κατά τη δεκαετία του '90 αντικαθιστώντας σε μεγάλο βαθμό τον δομημένο προγραμματισμό που ήταν ένα παραδοσιακό υπόδειγμα τότε. Έχει να κάνει με μια προγραμματιστική μέθοδο, που υποστηρίζεται από διαφορετικές κατάλληλες γλώσσες προγραμματισμού, όπως η Java στην περίπτωση μας, όπου ο χειρισμός των δεδομένων και των διαδικασιών γίνεται από κοινού μέσα από μια αυτόνομη δομή δεδομένων που έχει τη δική της ταυτότητα (χαρακτηριστικά). Η δομή αυτή των δεδομένων ονομάζεται «αντικείμενο» και δημιουργεί ένα πραγματικό στιγμιότυπο στη μνήμη ενός τύπου δεδομένων που ορίζεται από τον χρήστη και ονομάζεται κλάση. Στην Java, οι κλάσεις αποτελούνται από α) μεθόδους, β) μεταβλητές. Μέθοδοι ονομάζονται οι συναρτήσεις που μπορεί να απευθύνεται το αντικείμενο μιας κλάσης. Μεταβλητές ονομάζονται τα δεδομένα που ορίζουν την κατάσταση των αντικειμένων.

3. Υψηλής απόδοσης

Η γλώσσα Java μπορεί να μην έχει την αποδοτικότητα της C++ αλλά όμως οι περισσότερες εφαρμογές που περιέχουν γραφικά αποδίδονται εξαιρετικά από την Java.

4. Εύρωστη

Οι προγραμματιστές της Java, δημιούργησαν μία γλώσσα η οποία θα ήταν απλή αλλά δυνατή. Όπως είπαμε και προηγουμένως, οι δείκτες δεν χρησιμοποιούνται με αποτέλεσμα ένα πρόγραμμα Java να μην έχει δυνατότητα να καταστρέψει ή να αναγράψει δεδομένα. Επιπροσθέτως, η Java συμβαδίζει τις γλώσσες Lisp & SmallTalk, οι οποίες απελευθερώνουν αυτόματα την καταμερισμένη μνήμη που πλέον δεν χρησιμοποιείται πια.

5. Πολυνηματική

Πολυνηματικό καλείται ένα πρόγραμμα που έχει σχεδιαστεί να εκτελεί ταυτόχρονα παραπάνω από μια διαδικασίες. Έχει δηλαδή την ικανότητα να εκτελεί ένα ανεξάρτητο νήμα (όπως ονομάζεται μια ροή εκτέλεσης μέσα σε μια εφαρμογή) ή πολλά ανεξάρτητα νήματα ταυτόχρονα. Ουδέτερη Αρχιτεκτονική

Στην Java, ένας κώδικας εντολών γραμμένος σε bytes, μεταγλωττίζεται από τον ειδικό μεταγλωττιστή της γλώσσας (compiler). Έτσι, η ίδια η γλώσσα Java, αποκτά μια αρχιτεκτονική που είναι ουδέτερη. Η ουδετερότητα της αρχιτεκτονικής δίνει τη δυνατότητα σε κάθε λειτουργικό σύστημα να διαβάσει και να εκτελέσει τα προγράμματα που είναι γραμμένα στη συγκεκριμένη γλώσσα.

Ασφάλεια

Η Java εκτός από απλή και δυνατή είναι και μία γλώσσα πολύ ασφαλής. Εξασφαλίζεται από τον ίδιο τον compiler ότι ο κώδικας του προγράμματος δεν παραβιάζει τους κανόνες ασφαλείας. Όταν οι κλάσεις φορτώνονται στο σύστημα, η μονάδα που φορτώνει τις κλάσεις φροντίζει ώστε αυτές να μην παραβιάζουν τους περιορισμούς του κάθε συστήματος, διασφαλίζοντας έτσι το σύστημα. Ακόμα, η ασφάλεια προς το API εμποδίζει τις μικροεφαρμογές να δημιουργούν προβλήματα στο σύστημα. Επίσης, ο κώδικας που εκτελείται μέσω του περιβάλλοντος χρόνου εκτέλεσης, σε μορφή byte code, ελέγχεται ώστε να υπακούει και αυτός στους κανόνες ασφαλείας.

6. Εύκολη στη μεταφορά

Στην Java, το μέγεθος όλων των τύπων δεδομένων είναι προκαθορισμένο ώστε να μπορεί να λειτουργεί σε κάθε λειτουργικό σύστημα ή μηχανήμα.

Στην παρούσα διπλωματική εργασία, χρησιμοποιήθηκε η γλώσσα Javascript. Η JavaScript δημιουργήθηκε από τον Brendan Eich ο οποίος εργαζόταν για λογαριασμό της Netscape (εταιρία λογισμικού). Εκείνη την περίοδο, η javascript αναπτυσσόταν υπό την ονομασία Mocha, όπου αργότερα ονομάστηκε LieScript για να φτάσει στην τελική της ονομασία JavaScript, επηρεασμένη από την γλώσσα Java, όπου έτσι παραμένει γνωστή μέχρι και σήμερα. Στα τέλη του 1996 η εταιρεία Netscape, υπέβαλε τη γλώσσα προγραμματισμού Javascript στην Ecma International. Η Ecma International είναι μια οργάνωση που τυποποιεί τις γλώσσες προγραμματισμού και τις εξετάζει ως βιομηχανικό πρότυπο. Η τυποποιημένη μορφή τελικά ονομάστηκε ECMAScript.

Αρχικά η Javascript ήταν βασισμένη στη γλώσσα προγραμματισμού C και την χρησιμοποιούσαν για γράψιμο κώδικα από την πλευρά του browser του χρήστη, ο οποίος ήταν ο client (πελάτης). Δηλαδή, το τελικό περιεχόμενο HTML δεν διαμορφώνεται στον διακομιστή, αλλά στον browser που είναι το πρόγραμμα περιήγησης του επισκέπτη και όχι στο χώρο του διακομιστή. Η συγκεκριμένη γλώσσα προγραμματισμού, δεν χρειάζεται ιδιαίτερες δυνατότητες από τον sever για να εκτελεστεί αλλά βασίζεται στον browser των επισκεπτών. Επιπροσθέτως, δύναται να ενσωματωθεί σε HTML σελίδες που είναι στατικές. Η JavaScript μπορεί να χρησιμοποιηθεί και σε εφαρμογές που βρίσκονται εκτός ιστοσελίδων, για παράδειγμα σε έγγραφα μορφής .pdf σε προγράμματα περιήγησης που είναι site-specific και σε widgets περιβάλλοντος εργασίας είτε σε ένα λογισμικό κινητού τηλεφώνου είτε σε λογισμικό σταθερού υπολογιστή. Μοιάζει με την C++ αλλά σαν στόχο έχει την αξιοπιστία στις εφαρμογές του διαδικτύου. Παρόλο που πολλά ονόματα αλλά και συμβάσεις έχουν εμπνευστεί/αντιγραφεί από τη γλώσσα Java, υπάρχει μεγάλη διαφορά μεταξύ των δύο γλωσσών αφού έχουν σημαντικές διαφορές στη σημασιολογία. Στη JavaScript, οι βασικές αρχές που πλαισιώνουν τον σχεδιασμό της, παίρνονται από τις γλώσσες προγραμματισμού Scheme και Self. Ο κώδικας Javascript μίας σελίδας αποτελείται από τις ετικέτες. Συνοπτικά, η γλώσσα Javascript είναι μία γλώσσα scripting και σχεδιάστηκε για να προσθέσει αλληλεπίδραση στις ιστοσελίδες. Μπορούμε να φτιάξουμε σενάρια που θα έχουν τη δυνατότητα να εκτελούν τις εργασίες αυτοματοποιημένα μία σελίδα Web ή ένα παράθυρο να ανοίγει ή να κλείνει).

Παρακάτω βλέπουμε ένα συνοπτικό πίνακα 1 με τις διαφορές ανάμεσα σε Java και Javascript.

Java	JavaScript
Περίπλοκη στη χρήση.	Η Javascript είναι σχετικά εύκολη στην χρήση.
Οι εφαρμογές αργούν ελαφρώς στην εκτέλεση.	Η εκτέλεση των εφαρμογών JavaScript πραγματοποιούνται ταχύτατα.
Απαιτείται το Java Developer's Kit για τη δημιουργία της εφαρμογής.	Η Javascript απαιτεί μόνο ένας φυλλομετρητής συμβατός με Javascript.
Τα προγράμματα της Java ενσωματώνονται σε σελίδες HTML μέσω ετικέτας αφού έχουν μεταγλωττιστεί και έχουν γίνει εκτελέσιμα σε ανεξάρτητα αρχεία.	Τα προγράμματα της JavaScript ενσωματώνονται με τη μορφή σεναρίου στις σελίδες HTML χωρίς να χρειάζεται μεταγλώττιση ή διερμήνευση σε ανεξάρτητο εκτελέσιμο αρχείο.
Η Java είναι πλήρης γλώσσα προγραμματισμού με πολλές δυνατότητες.	Η javascript είναι κατάλληλη για πιο απλές χρήσεις.
Η Java είναι προσανατολισμένη στο αντικείμενο και είναι καθαρά αντικειμενοστρεφής προγραμματιστική γλώσσα .	Η Javascript βασίζεται στο αντικείμενο, αλλά δεν είναι αντικειμενοστρεφής γλώσσα προγραμματισμού.
Η Java χρησιμοποιεί τις μεταβλητές της με αυστηρούς κανόνες.	Η Javascript χαρακτηρίζεται από χαλαρότητα ως προς τις μεταβλητές.

Πίνακας 1 Διαφορές Java – JavaScript

2.4 Ελεύθερο Λογισμικό / Λογισμικό Ανοιχτού Κώδικα

Με τον όρο Ελεύθερο Λογισμικό ή Λογισμικό ανοιχτού κώδικα, αναφερόμαστε σε ένα λογισμικό το οποίο διατίθεται με ειδικές άδειες που επιτρέπουν τη χρήση, αντιγραφή, μελέτη, τροποποίηση και διανομή του λογισμικού χωρίς περιορισμούς. Ένα ελεύθερο λογισμικό δεν αποσκοπεί στο κέρδος, σε αντίθεση με τα λογισμικά κλειστού κώδικά όπου ο πηγαίος κώδικας είναι κρυφός, το λογισμικό είναι κλειστό, απαγορεύεται η αντιγραφή ή/και η διανομή σε τρίτου κ.α. Το Ελεύθερο Λογισμικό, δεν σημαίνει πως είναι και λογισμικό που διανέμεται δωρεάν, καθώς επιτρέπεται στον κάθε διανομέα να ορίσει μια ένα αντίτιμο, εάν αυτός το επιθυμεί, κατά την χρήση του λογισμικού από τον χρήστη. Παρόλα αυτά, τα ελεύθερα λογισμικά στην συντριπτική τους πλειοψηφία διανέμονται δωρεάν. (Harper, 2006)

Από τα πιο ευρέως διαδεδομένα και γνωστά ελεύθερα λογισμικά που υπάρχουν, είναι ο φυλλομετρητής Mozilla Firefox, η σουίτα LibreOffice και ο πυρήνας του λειτουργικού συστήματος Linux.

- Ένα Ελεύθερο λογισμικό προσφέρει τη δυνατότητα στους χρήστες, που έχουν γνώσεις προγραμματισμού, να εκτελέσουν, διανείμουν, μελετήσουν, αντιγράψουν, τροποποιήσουν το λογισμικό με όποιο τρόπο αυτοί επιθυμούν, με απόλυτη ελευθερία. Οι τέσσερις κύριες ελευθερίες που προσφέρει ένα ελεύθερο λογισμικό είναι: Ελευθερία εκτέλεσης του προγράμματος για οποιονδήποτε σκοπό. (Ελευθερία 0)
- Ελευθερία μελέτης του τρόπου λειτουργίας του και πρόσβαση στον πηγαίο κώδικα του προγράμματος δίνοντας τη δυνατότητα προσαρμογής στις ανάγκες του εκάστοτε χρήστη. (Ελευθερία 1)
- Ελευθερία αντιγραφής και διανομής του προγράμματος με σκοπό την εξυπηρέτηση του κοινωνικού συνόλου. (Ελευθερία 2)
- Ελευθερία στους χρήστες να έχουν πρόσβαση στον πηγαίο κώδικα για να βελτιώσουν το πρόγραμμα και να δημοσιεύσουν αυτές τις βελτιώσεις στο ευρύ κοινό, με απώτερο σκοπό πάλι την ωφέλεια όλου του κοινωνικού συνόλου. (Ελευθερία 3)

Οι προγραμματιστές που παράγουν ελεύθερο λογισμικό/λογισμικό ανοιχτού κώδικα είναι χρήστες που δουλεύουν μεμονομένα, είτε επαγγελματίες στον τομέα της πληροφορικής, η φοιτητές. Αρκετές είναι και οι εταιρείες που παράγουν και υποστηρίζουν ελεύθερο λογισμικό/λογισμικό ανοιχτού κώδικά. Στις πρώτες μέρες της ύπαρξης του ελεύθερου λογισμικού, οι χρήστες βασίζονταν στην υποστήριξη μέσω ηλεκτρονικής επικοινωνίας με άλλους χρήστες π.χ. μέσω διάφορων forums που επικεντρώνονταν σε αυτό το σκοπό ή σε άμεση επικοινωνία με τον ίδιο τον δημιουργό. Σύντομα όμως δημιουργήθηκαν εταιρείες με αποκλειστικό σκοπό την παροχή βοήθειας σε χρήστες ελεύθερου λογισμικού. Σήμερα το μεγαλύτερο ποσοστό των χρηστών ελεύθερου λογισμικού/λογισμικού ανοιχτού κώδικα, απαιτεί την παροχή υπηρεσιών τεχνικής υποστήριξης για την σωστότερη και αποδοτικότερη χρήση και λειτουργία του λογισμικού.

Ελευθερία της χρήσης ενός τέτοιου λογισμικού σημαίνει πως κάθε άτομο ή κάθε επιχείρηση που το επιθυμεί, έχει τη δυνατότητα να κάνει χρήση του λογισμικού σε οποιοδήποτε σύστημα και για οποιοδήποτε είδος εργασίας (οικιακή χρήση, χρήση από υπαλλήλους εταιρίας, δημιουργία server ιστοσελίδων κ.α.), χωρίς να απαιτείται εκ των προτέρων αδειοδότηση από τον δημιουργό. Ο

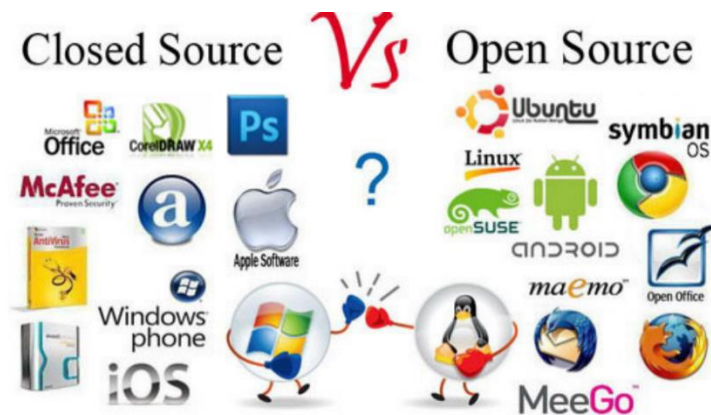
χρήστης είναι ο μόνος υπεύθυνος που θα καθορίσει τη χρήση ενός ελεύθερου λογισμικού και όταν το διανείμει σε έναν άλλο, δεν θα έχει δικαίωμα καθορισμού του είδους χρήσης του από τον άλλο χρήστη.

Η απαρχή ενός ελεύθερου λογισμικού/λογισμικού ανοιχτού κώδικα, στις περισσότερες περιπτώσεις ξεκινάει επειδή ο δημιουργός του έχει την ανάγκη να δημιουργήσει ένα εργαλείο ή μια εφαρμογή. Για τον λόγο αυτό το προσαρμόζει πλήρως στις δικές του ανάγκες και προτιμήσεις. Τα περισσότερα λογισμικά αυτού του είδους βασίζονται σε τεχνικά κριτήρια για αυτή την επιλογή. Βασικά κριτήρια αποτελούν: α) Οι δυνατότητες του λογισμικού, β) οι υπολογιστικοί πόροι που απαιτούνται και γ) τα οικονομικά κριτήρια. Σε ένα περιβάλλον επιχείρησης, τα κριτήρια για την επιλογή ενός τέτοιου λογισμικού συνδέονται άμεσα με την επιχείρηση και τους σκοπούς της. Σε μία τέτοια περίπτωση η τεχνική υποστήριξη, η επιχειρησιακή συνέχεια και οι πιθανές δεσμεύσεις, είναι σημαντικά κριτήρια για την επιλογή του λογισμικού. .

Είναι πολλά που πρέπει να λάβουμε υπόψη για να αποφασίσουμε τι λογισμικό ανταποκρίνεται στις ανάγκες μας με τον καλύτερο τρόπο. Στον παρακάτω πίνακα (Πίνακας 2) θα παρουσιαστούν οι πιο βασικές διαφορές ανάμεσα στα λογισμικά ανοιχτού κώδικα και στα λογισμικά κλειστού κώδικα.

ΚΛΕΙΣΤΟΣ ΚΩΔΙΚΑΣ	ΑΝΟΙΧΤΟΣ ΚΩΔΙΚΑΣ
Σημαντικά επίπεδα ασφαλείας ακόμα και αν γίνονται συχνά στόχοι επιθέσεων.	Παρέχουν υψηλή ασφάλεια.
Κόστος.	Το κόστος είναι μηδενικό.
Δημιουργούνται από κλειστές και συγκεκριμένες ομάδες προγραμματιστών.	Υποστηρίζεται από πολύ μεγαλύτερη ομάδα προγραμματιστών.
Απαγορεύεται η οποιαδήποτε αλλαγή στο λογισμικό.	Το λογισμικό δύναται να προσαρμοστεί στις ανάγκες του χρήστη.
Οι ενημερώσεις γίνονται τακτικά. (Χωρίς να προβάλλονται οι αλλαγές που γίνονται)	Οι ενημερώσεις γίνονται τακτικά.
Χορηγείται εγχειρίδιο χρήσης κατά την αγορά.	Παρέχεται βοήθεια μέσα από τις κοινότητες των προγραμματιστών και άλλους χρήστες.
Παρέχεται εγγύηση για την ποιότητα του προϊόντος.	Παρέχουν καλή λειτουργία και αξιοπιστία.

Πίνακας 2 Διαφορές Ανοιχτού και κλειστού κώδικα



Εικόνα 3 Λογισμικά κλειστού και ανοιχτού λογισμικού

2.5 Πλεονεκτήματα και μειονεκτήματα χρήσης Ελεύθερου Λογισμικού/ Λογισμικού ανοιχτού κώδικα

Πλέον όλο και περισσότερες είναι οι εταιρείες που χρησιμοποιούν τα ελεύθερα λογισμικά ανοιχτού κώδικα για την υποστήριξη των αναγκών τους. Σε λειτουργικά συστήματα των υπολογιστών και των εξυπηρετητών, χρήση ως ενδιάμεσο λογισμικό, για το υπολογιστικό νέφος, στην τηλεφωνία και σε άλλους δικτυακούς τόπους, είναι μερικές από τις πολλές κατηγορίες όπου η ανοιχτή φύση του ελεύθερου λογισμικού υπερτερεί έναντι των ιδιοκτητών και κλειστών λογισμικών. Ευελιξία

Έλα ελεύθερο λογισμικό δίνει τη δυνατότητα σε μια εταιρία να προσφέρει ευελιξία στις λύσεις και στις τεχνικές υλοποιήσεις που την χαρακτηρίζουν. Είναι στη φύση του ανοιχτού κώδικα να προάγει την τεχνολογική ευελιξία, καθώς προσφέρει πολλαπλές οδούς για την επίλυση διαφόρων προβλημάτων που μπορεί να προκύψουν και να βοηθήσει την επιχείρηση να μην αντιμετωπίσει εμπόδια λόγω αδυναμίας τεχνικής υποστήριξης από κάποιον προμηθευτή.

Έτσι, μία εταιρεία που βασίζεται στο ελεύθερο λογισμικό για την λειτουργία της, με την πρόσληψη του κατάλληλου προσωπικού, δύναται να προσφέρει η ίδια λύση στις τεχνικές/τεχνολογικές της ανάγκες, η οποία ενδέχεται να είναι και ποιοτικότερη από την λύση που θα μπορούσε να της προσφέρει ο εκάστοτε προμηθευτής.

Ταχύτητα

Ο ανταγωνισμός μεταξύ των επιχειρήσεων ως προς την ταχύτητα εύρεσης τεχνολογικών λύσεων, επωφελείται από τα λογισμικά ανοιχτού κώδικα. Η άμεση εγκατάσταση του λογισμικού αλλά και η μείωση της γραφειοκρατίας για την επίλυση ενός επιχειρησιακού προβλήματος, είναι ένα από τα μεγάλα πλεονεκτήματα ενός λογισμικού ανοιχτού κώδικα. Κάτι τέτοιο δίνει τη δυνατότητα στην εταιρεία να παράγει αξία αμέσως. Μειώνεται έτσι το κόστος έναρξης μιας νέας επιχείρησης, δίνοντάς της την προοπτική να αναπτυχθεί και να μεγαλώσει σε κλίμακα έχοντας σύγχρονες τεχνολογίες υποδομής, με πλήρη υποστήριξη, ανεμπόδιστα.

Ικανότητα να ξεκινάς από χαμηλό επίπεδο

Έτσι μία επιχείρηση μπορεί να ξεκινάει άμεσα από ένα χαμηλό επίπεδο χρησιμοποιώντας το διαθέσιμο ελεύθερο λογισμικό και μετά να έχετε δυνατότητα να υιοθετήσει μία εμπορική λύση εάν οι επιχειρηματικές τις ανάγκες και δυνατότητες του απαιτούν. Εναλλακτικά η επιχείρηση θα μπορεί να χρησιμοποιεί το ελεύθερο λογισμικό χωρίς κάποια παρεμπόδιση για όσο χρονικό διάστημα αυτή επιθυμεί. Το ελεύθερο λογισμικό δίνει επίσης τη δυνατότητα σε μία επιχείρηση να δοκιμάσει διάφορες εναλλακτικές επιλογές και να επιλέξει την καταλληλότερη για τις ανάγκες της, χωρίς δικλείδες από την εκάστοτε εταιρία παραγωγής κλειστού λογισμικού. Ασφάλεια πληροφοριών

Στον τομέα της Ασφάλειας Συστημάτων Πληροφορικής, το εμπορικό και ελεύθερο λογισμικό ανοιχτού κώδικα κυριαρχεί μιας και πλέον η υποστήριξη και η ανταπόκριση της κοινότητας, όσον αφορά την επίλυση προβλημάτων ασφαλείας, είναι αρκετά υψηλή. Η δυνατότητα της κοινότητας να ανακαλύπτει και να επιλύει προβλήματα στην ασφάλεια που μπορεί να προκύψουν, σε ένα λογισμικό ανοιχτού κώδικα, είναι ένα από τα πιο βασικά πλεονεκτήματα του συγκριτικά με τους περιορισμούς ενός λογισμικού κλειστού κώδικα όπου ειδήμονες για την εύρεση της λύσης, σε αυτή τη περίπτωση, είναι μόνο όσοι ασχολήθηκαν με την ανάπτυξή του στην εταιρεία που το προμηθεύει. Κόστος συντήρησης

Εκτός από την ασφάλεια, η κοινότητα φροντίζει για την διαρκή συντήρηση και βελτίωση του ήδη υπάρχοντος κώδικα λογισμικού. Αυτό είναι επίσης ένα από τα πολύ βασικά πλεονεκτήματα του ανοιχτού κώδικα. Εν αντιθέσει, μία εταιρεία που παράγει λογισμικό είναι υπεύθυνη στο ακέραιο για τη συντήρηση και την ανάπτυξη του λογισμικού, κάτι που ενδέχεται να υποβαθμίσει την ποιότητα των υπηρεσιών της. Το ελεύθερο λογισμικό επωφελεί και την εκπαίδευση. Εκτός από την ηθική διαπαιδαγώγηση που προσδίδει η ύπαρξη ενός ελεύθερου λογισμικού σε χώρους πανεπιστημίων και σχολείων όπου οι μαθητές ή φοιτητές διδάσκονται την αξία του αλτρουισμού και τους κοινωνικής προσφοράς, εξοικειώνονται με τις λειτουργίες του ηλεκτρονικού υπολογιστή μέσω ενός λογισμικού υψηλής ποιότητας.

2.6 Λογισμικό Ανοιχτού κώδικα για WebGIS

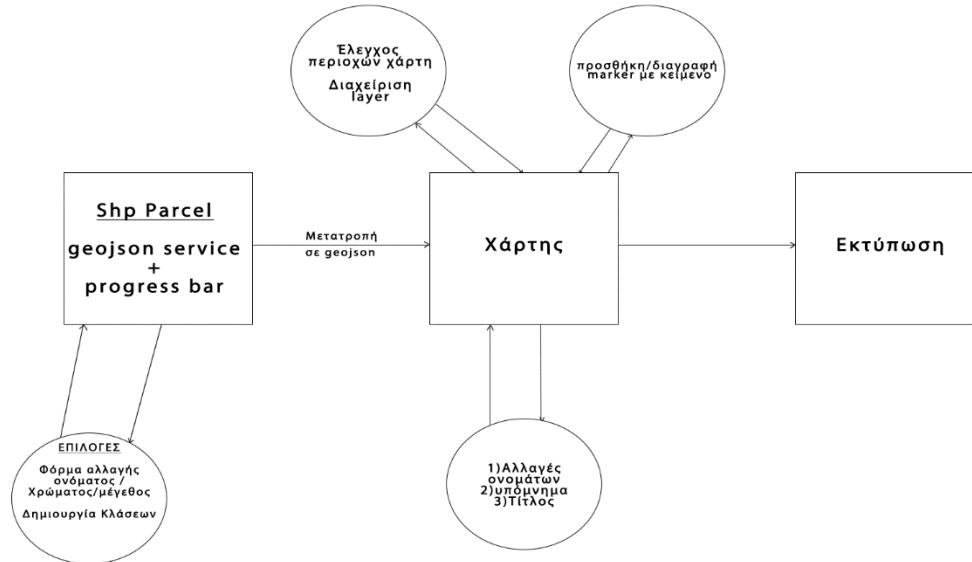
Πολλοί θα έχουν διαφορετικές απόψεις σχετικά με τη συζήτηση ανοιχτού κώδικα έναντι ιδιόκτητου λογισμικού, από χρήστες που πιστεύουν ότι τα προϊόντα ανοιχτού κώδικα δεν μπορούν καν να συγκριθούν με τους ιδιόκτητους ομολόγους τους με χρήστες που αισθάνονται ακριβώς το αντίθετο. Παρά τις απόψεις που επικρατούν και στις δύο πλευρές για το ζήτημα, το λογισμικό ανοιχτού κώδικα είναι μια δύναμη που πρέπει να υπολογιστεί και αξίζει την προσοχή της κοινότητας των GIS ως θέμα συζήτησης, έρευνας και πειραματισμού. Υπάρχει ένα αυξανόμενο επιχείρημα στον τομέα του WBGIS που υποστηρίζει την αποτελεσματικότερη τυποποίηση της μεταφοράς δεδομένων GIS σε πολλές πλατφόρμες, σε παγκόσμια κατανοητές γλώσσες και πρωτόκολλα που αποτελεί τεράστια κινητήρια δύναμη πίσω από τα περισσότερα έργα ανοιχτού κώδικα. Αυτά τα Ανοιχτά Πρότυπα (Open Standards (OS)) παρέχουν έναν οδηγό και μια καλά καθορισμένη δομή για τον τρόπο με τον οποίο τα δεδομένα GIS μπορούν να κοινοποιούνται αποτελεσματικά σε ένα ετερογενές παγκόσμιο δίκτυο υπολογιστών. Τα Ανοιχτά Πρότυπα υιοθετούν πολλά επιτυχημένα προϊόντα ανοιχτού κώδικα, αυξάνοντας έτσι την πιθανότητά τους για ευρεία υιοθέτηση. Το Open Geospatial Consortium (OGC) είναι ένας ανεξάρτητος οργανισμός για την ανάπτυξη προτύπων για γεωχωρικές εφαρμογές και υπηρεσίες. Μεταξύ των 311

προϊόντων που συμμορφώνονται ή εφαρμόζονται στις 29 Ιανουαρίου 2005 , η ποσότητα των προϊόντων ανοιχτού κώδικα είναι αρκετά εκτεταμένη, συμπεριλαμβανομένων των MapServer και PostGIS του Πανεπιστημίου της innesota (UMN). Αυτά τα προϊόντα ανοιχτού κώδικα υπάρχουν παράλληλα με πολλές ιδιόκτητες εφαρμογές λογισμικού από τους κορυφαίους προμηθευτές GIS, συμπεριλαμβανομένων των Environmental Systems Research Institute (ESRI), Autodesk και General Electric (GE). Αυτό είναι σαφές απόδειξη ότι η κίνηση GIS ανοιχτού κώδικα είναι σε πλήρη ισχύ και δεν επιβραδύνεται σύντομα.

Ο Tyler Mitchell (2005), συγγραφέας του βιβλίου *Web Mapping Illustrated*, είναι ο πρώτος που δημοσίευσε ένα βιβλίο που εστιάζει αποκλειστικά στην χαρτογράφηση ιστού ως πρακτική της Πληροφορικής στο πλαίσιο των τεχνολογιών ανοιχτού κώδικα. Ο Mitchell επισημαίνει ότι ένας νέος κόσμος συνεργασίας ανοιχτού κώδικα μαζί με μια υγιή δέσμευση από τη βιομηχανία καλύπτει την ανάγκη και τη ζήτηση για χωρικές τεχνολογίες με ρυθμό ανάπτυξης που δεν μπορεί να συγκριθεί στον κλάδο. Γράφει αυτό το βιβλίο όχι ως επαγγελματίας GIS, αλλά υποθέτοντας ότι ο αναγνώστης δεν έχει ακούσει ποτέ τον όρο. Η εστίασή του στις τεχνολογίες που είναι άμεσα διαθέσιμες στο ευρύ κοινό μέσω έργων ανοιχτού κώδικα όπως το MapServer και το PostGIS με τρόπο που επιδιώκει, όπως το θέτει, "να συμπληρώσει τα κενά της υπάρχουσας τεκμηρίωσης και να απαντήσει στις κοινές ερωτήσεις του νέου χρήστη" . Ο Mitchell έχει την άποψη ότι οι τεχνολογίες ανοιχτού κώδικα είναι βιώσιμες εναλλακτικές λύσεις στο ιδιόκτητο λογισμικό GIS, ειδικά στο WEBGIS και στον τομέα χαρτογράφησης. Υποστηρίζει τη φιλοσοφία ανοιχτού κώδικα της ανάπτυξης λογισμικού, όπου μια κοινότητα από προγραμματιστές που συμβάλλουν στο έργο είναι ένα επιτυχημένο μοντέλο για την εξέλιξη ενός προϊόντος λογισμικού και πιστεύει ότι τα περισσότερα έργα προωθούν τη διαλειτουργικότητα εφαρμόζοντας ανοιχτά πρότυπα και σε πολλές περιπτώσεις τα εφαρμόζουν γρηγορότερα από τους ιδιόκτητους ομολόγους τους. Παρέχει επίσης λεπτομερείς οδηγίες εγκατάστασης για το MapServer και το PostGIS και παρέχει χρήσιμες συμβουλές για τη χρήση των προϊόντων. Τα μαθήματά του σχετικά με τα αρχεία χαρτών που υποστηρίζονται από το MapServer βοηθούν επίσης στην ομαλή μετάβαση στη χρήση του προϊόντος το οποίο οποιοσδήποτε έχει υπόβαθρο στην επιστήμη των υπολογιστών και το GIS θα μπορούσε να πάρει εύκολα. Η δουλειά του Mitchell είναι μια σημαντική απόδοση στον τομέα του ανοιχτού κώδικα WebGIS και αυτή τη στιγμή είναι η μεγαλύτερη δημοσίευση σε αυτό το συγκεκριμένο θέμα. (Mitchell, 2005)

Κεφάλαιο 3. Μεθοδολογία

3.1 Ανάλυση κώδικα



Εικόνα 4 Αρχιτεκτονική Κώδικα

Ο κώδικα βασίστηκε σε open source πακέτα όπου η ανάγνωση και η επεξεργασία των shp αρχείων γίνεται από το shapfile.js πακέτο, η εμφάνιση του χάρτη και η παρουσίαση γίνεται από το leaflet.js και η αποτύπωση γίνεται από ένα plug in αρχείο του leaflet.js πακέτου

Ο κώδικα χωρίζεται σε 3 βασικά μέρη όπως παρουσιάζεται στην εικόνα 4:

1) Shp Parcel

Το shp parcel παίρνει SHP αρχεία με σημεία ή επιφάνειας στο χάρτη και τα μετατρέπει σε GeoJSON για να τα εισάγει στον χάρτη.

Σε αυτό το σημείο υπάρχει ένας βρόγχος (loop) που χρησιμεύει στην εισαγωγή πληροφοριών. Σε αυτό το σημείο υπάρχει η δυνατότητα εισαγωγής ονομασίας των δεδομένων, για την δημιουργία υπομνήματος και χρωματικής επιλογής. Ο κώδικας ελέγχει το αρχείο εισαγωγής και αν στον πίνακα χαρακτηριστικών υπάρχει αριθμητική στήλη δίνει την δυνατότητα στον χρήστη αν επιθυμεί να δημιουργήσει έναν αριθμό κλάσεων (από 1 έως 9) ίσου πλήθους (equal count) με δυνατότητα ονομασίας και χρωματικής διαβάθμισης για κάθε μία κλάση ξεχωριστά.

Περιέχεται επίσης ένα progress bar που δείχνει την πρόοδο εισαγωγής των αρχείων shp για την παρουσίαση στο χάρτη.

2) Χάρτης

Ο χάρτης είναι το κύριο σημείο του UI της εφαρμογής. Αφού εισαχθούν τα αρχεία από το Shp Parcel μέσω της διαδικασίας που περιγράψαμε στο προηγούμενο βήμα απεικονίζονται σε αυτόν τα δεδομένα που ο χρήστης επιθυμεί. Στον layer παρουσίασης του χάρτη υπάρχουν loops όπου μας δίνουν την δυνατότητα: α)εισαγωγής τίτλου με μορφοποίηση, β)διαχείριση των layers, δηλαδή παρουσίαση/απόκρυψη των layers που αντιστοιχούν σε κλάσεις στην περιοχή του υπομνήματος και εμφάνιση πίνακα χαρακτηριστικών για την περιοχή που επιθυμεί ο χρήστης, γ) δυνατότητα εισαγωγής σημαδιών (markers) με κείμενο.

Στο χάρτη υπάρχει το σύμβολο του βορρά.

3) Εκτύπωση

Αφού σχηματιστεί η τελική μορφή του χάρτη με τα δεδομένα που ο χρήστης επιθυμεί, προχωράει στο τελευταίο και τρίτο μέρος του κώδικα που είναι η εκτύπωση. Στο σημείο αυτό δίνεται η δυνατότητα στον χρήστη να εξάγει σε μορφή pdf την τελική μορφή του χάρτη.

Η διαδικασία στο κώδικα πραγματοποιείτε με την παραπάνω σειρά. Στην ουσία υπάρχουν 2 layers το οποίο το ένα layer είναι τα δεδομένα και η επεξεργασία των αρχείων και το άλλο layer είναι η παρουσίαση.

Το GeoJSON είναι ένα format ανοιχτού κώδικα που έχει σχεδιαστεί για την αναπαράσταση απλών γεωγραφικών χαρακτηριστικών, μαζί με τα μη χωρικά χαρακτηριστικά τους. Βασίζεται στην γλώσσα προγραμματισμού JavaScript (JSON). Έχει την δυνατότητα να επεξεργαστεί δεδομένα που περιλαμβάνουν σημεία (διευθύνσεις και τοποθεσίες), συμβολοσειρές γραμμών (δρόμους, αυτοκινητόδρομους και όρια), πολύγωνα (χώρες, επαρχίες, εκτάσεις γης) και συνδυασμούς των παραπάνω.

3.2 Πακέτα για την δημιουργία του WebGIS site

3.1.1 Leaflet

Το Leaflet μια βιβλιοθήκη JavaScript ανοιχτού κώδικα, κορυφαία στο είδος της, για δημιουργία διαδραστικών χαρτών φιλικούς προς κινητά. Έχει μέγεθος 39 KB gzipped JS + 4 KB gzipped CSS code, έχει όλες τις δυνατότητες χαρτογράφησης που χρειάζονται οι περισσότεροι προγραμματιστές.

Το Leaflet είναι σχεδιασμένη με βάση την απλότητα, τη χρηστικότητα και την απόδοση. Αποτελεσματική είναι η λειτουργία του με όλες τις μεγάλες πλατφόρμες για επιτραπέζιους υπολογιστές και κινητές συσκευές, εκμεταλλευόμενοι τα HTML5 και CSS3 σε μοντέρνα προγράμματα περιήγησης ακόμα και σε παλαιά μοντέλα. Μπορεί να επεκταθεί με ένα τεράστιο ποσό προσθηκών, και χαρακτηρίζεται από ένα εύκολο, ευπαρουσίαστο και ορθά τεκμηριωμένο API, καθώς και έναν απλό και αναγνώσιμο πηγαίο κώδικα στον οποίο ο κάθε χρήστης δύναται να συνεισφέρει. (Vladimir Agafonkin)

Εδώ παρουσιάζετε ο κώδικας:

```
var map = L.map('map').setView([51.505, -0.09], 13);
```

```
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
  attribution: '&copy; <a
href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors'
}).addTo(map);

L.marker([51.5, -0.09]).addTo(map)
  .bindPopup('A pretty CSS3 popup.<br> Easily customizable.')
  .openPopup();
```

3.1.2 Shapefile.js

Το συγκεκριμένο πακέτο προορίζεται για την αναγνώριση και αποτύπωση των shapefile αρχείων στο χάρτη. Διαβάζει zip αρχεία που περιέχουν όλα τα αρχεία shp.

Το shapefile είναι μια απλή, μη τυπολογική μορφή για την αποθήκευση της γεωμετρικής θέσης και την απόδοση πληροφοριών γεωγραφικών χαρακτηριστικών. Το σύνολο των γεωγραφικών χαρακτηριστικών που περιέχονται σε ένα σε ένα shapefile αναπαρίστανται από σημεία, από γραμμές ή πολύγωνα (ολόκληρες περιοχές). Ο χώρος εργασίας που περιέχουν τα shapefiles μπορεί αποτελείται από πίνακες (dBASE), που έχουν τη δυνατότητα να αποθηκεύσουν πρόσθετα χαρακτηριστικά τα οποία με την σειρά τους να συνδέονται με τα χαρακτηριστικά των shapefile αρχείων. ('shapefile-js @ github.com', 2020)

3.1.3 jqquery

Bootstrap: Το Bootstrap είναι ένα πλαίσιο για ανάπτυξη ιστοσελίδων front-end. Κάνει την ανάπτυξη ιστού πιο γρήγορη και ευκολότερη. Περιλαμβάνει πρότυπα σχεδίασης που βασίζονται σε HTML και CSS για διάφορους σχεδιασμούς απόκρισης front-end, καθώς και προαιρετικά πρόσθετα JavaScript. ('Getbootstrap.Com', 2020)

JQuery UI: Το JQuery UI είναι μια συλλογή γραφικών στοιχείων και θεμάτων GUI που υλοποιούνται με μια βιβλιοθήκη JavaScript. Χρησιμοποιείται για σχεδιασμό ιστοσελίδων back-end. (JQuery, 2018)

Διαφορά μεταξύ Bootstrap και JQuery UI:

1. ΤΥΠΟΣ ΠΛΑΙΣΙΟΥ: Το Bootstrap είναι ένα πλαίσιο για την ανάπτυξη εφαρμογών ιστού και ιστοσελίδων front-end. Παρέχεται ως δωρεάν πλαίσιο ανοιχτού κώδικα σε άτομα. Αναπτύχθηκε και σχεδιάστηκε από την βασική ομάδα Bootstrap. Είναι γραμμένο σε HTML, CSS και JavaScript, Ενώ, Το JQuery UI είναι ένα πλαίσιο ανοιχτού κώδικα για να σχεδιάσετε την πλευρά του πελάτη ή να κάνετε scripting back-end HTML. Είναι επίσης δωρεάν και εύκολο στη χρήση. Αναπτύχθηκε και σχεδιάστηκε από την ομάδα The JQuery. Είναι γραμμένο σε JavaScript.
2. ΒΑΣΕΙΣ ΜΕ ΤΗ ΧΡΗΣΗ: Το Bootstrap είναι διάσημο για την ανάπτυξη ανταποκρινόμενων ιστοτόπων και έργων για κινητά. Χρησιμοποιώντας το Bootstrap μπορείτε να αναπτύξετε μια δυνατότητα στην οποία μπορείτε να εργαστείτε με κάθε μέγεθος της οθόνης καθώς παρέχει συνέπεια σε όλα τα προγράμματα περιήγησης και σε διαφορετικές συσκευές, Ενώ, Το JQuery UI χρησιμοποιείται για την ανάπτυξη διεπαφής

χρήστη που βασίζεται σε HTML, ώστε οι ιστότοποι να ανταποκρίνονται περισσότερο σε όλα τα προγράμματα περιήγησης. Λειτουργεί σε βιβλιοθήκη JavaScript.

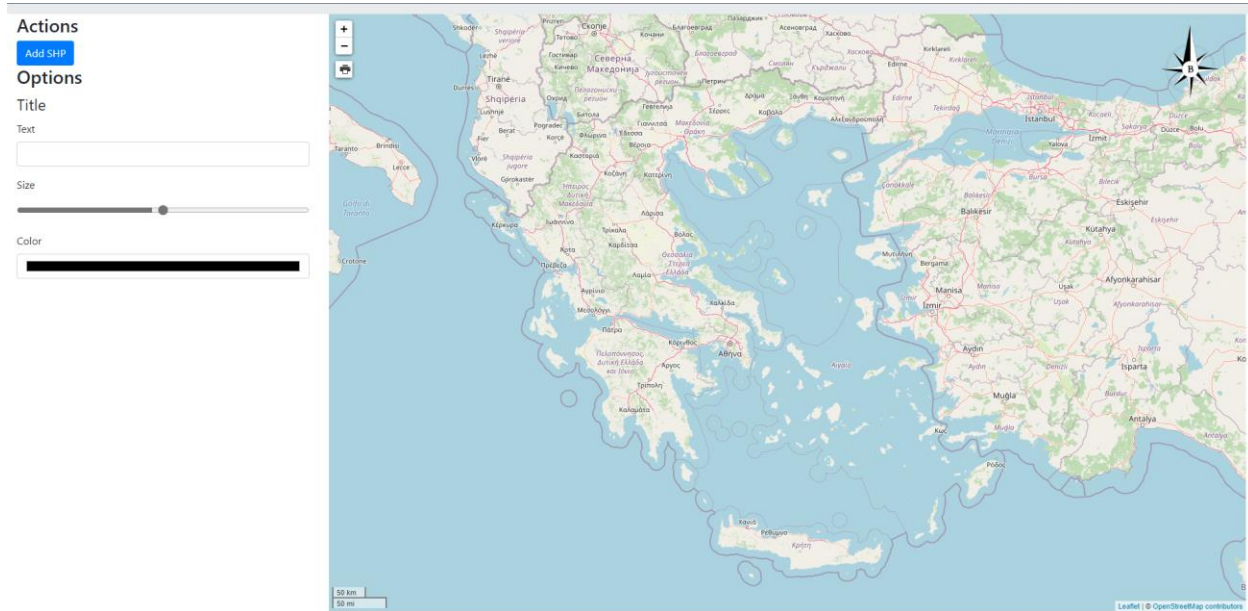
3. ΒΑΣΕΙ ΛΕΙΤΟΥΡΓΙΩΝ: Το Bootstrap περιέχει πολλά στοιχεία JavaScript γνωστά ως πρόσθετα JQuery, Ενώ, Το JQuery UI διαθέτει βιβλιοθήκη προ-γραπτών συναρτήσεων.
4. ΒΑΣΕΙ ΕΜΦΑΝΙΣΗΣ: Το Bootstrap φαίνεται να είναι πολύ επαγγελματικό ακόμα και όταν χρησιμοποιείτε τα βασικά πρότυπα, Ενώ, Το JQuery UI φαίνεται ηλικιωμένο και παλιό και δίνει καλύτερη εμφάνιση σε προγράμματα περιήγησης παλαιών εκδόσεων.

Για την χρήση του προγράμματος θα χρειαστεί να κατεβάσετε τα αρχεία που βρίσκονται στο παρακάτω ιστότοπο. Στην συνέχεια θα βρείτε το αρχείο με την ονομασία index.html θα το ανοίξετε με οποιοδήποτε πρόγραμμα browser και θα ακολουθήσετε τις διαδικασίες που αναγράφονται στο κεφάλαιο 4 για την δημιουργία ενός θεματικού χάρτη.

Δεν απαιτούνται ιδιαίτερες υπολογιστικές απαιτήσεις λόγω του ότι το πρόγραμμα έχει σχεδιαστεί κατάλληλα για υπολογιστές με χαμηλά χαρακτηριστικά.

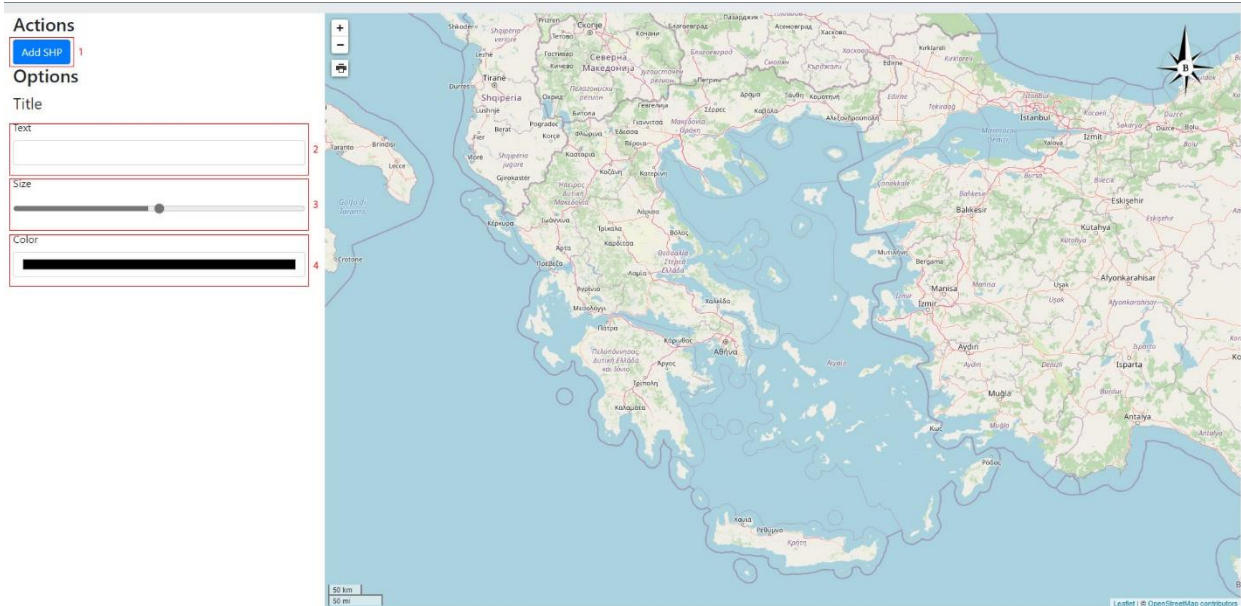
Κεφάλαιο 4. Αποτελέσματα

4.1 Η Εφαρμογή webGIS



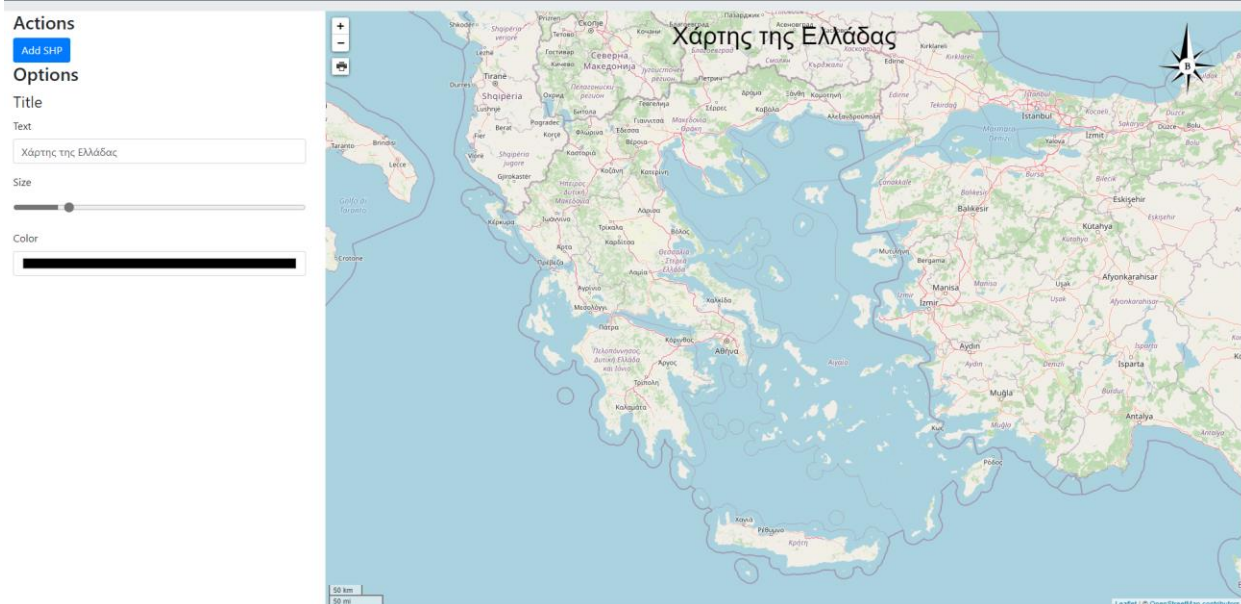
Εικόνα 5 Το UI της εφαρμογής

Η εικόνα 5 είναι η πρώτη εικόνα που αντικρίζει ένας χρήστης, όταν επισκεφθεί την ιστοσελίδα της εφαρμογής με έναν web-browser, το βασικό περιβάλλον εργασίας μέσω του οποίου θα γίνεται η χρήση των εργαλείων ανάλυσης GIS που προσφέρει η πλατφόρμα. Η εμφάνιση των εργαλείων της πλατφόρμας στη σελίδα, η τοποθέτησή τους σε αυτήν καθώς και το στυλ τους προγραμματίζεται με γλώσσα HTML και CSS. Η λειτουργικότητά τους βασίζεται στη γλώσσα JavaScript και των βιβλιοθηκών της OpenLayers και JQuery και διαφόρων add-ons. Στο αριστερό μέρος παραθέτονται εργαλεία οργανωμένα σε μία κάθετη στήλη, προς επιλογή υποσυνόλου δεδομένων της βάσης, τεχνικές ανάλυσής τους, επιλογές τρόπων παρουσίασης, καθώς και το βασικό κομβίο που εκτελεί τις εντολές που έχει επιλέξει ο χρήστης. Το μεγαλύτερο μέρος της σελίδας καλύπτεται από τον παραγόμενο χάρτη. Σκοπός της επιλογής αυτής είναι το αποτέλεσμα των εργασιών που πραγματοποιούνται να είναι εύκολα ευκρινές, χωρίς άμεση απαίτηση μεγέθυνση της εικόνας.



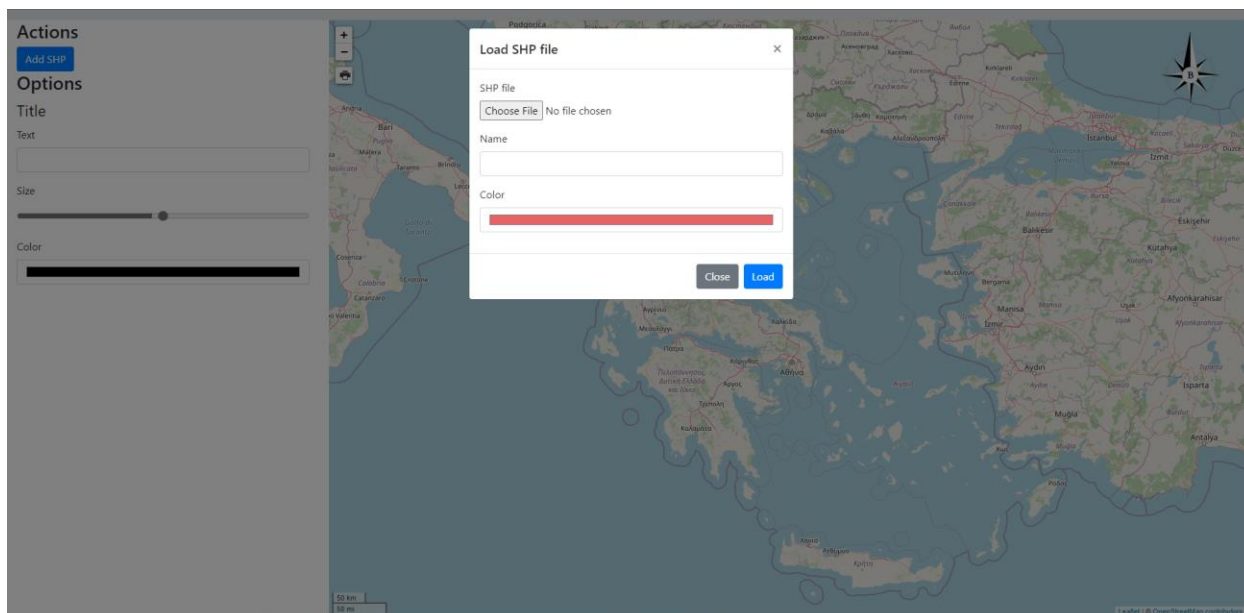
Εικόνα 6 Τα εργαλεία της εφαρμογής

Στην εικόνα 6 παρουσιάζονται οι επιλογές που υπάρχουν στον ιστό. Στην επιλογή 1 (Add SHP) προορίζεται για την προσθήκη αρχείου shaperefile και την αποτύπωση αυτού στον χάρτη. Θα αναλυθεί παρακάτω για την επιλογή αυτή. Στην επιλογή 2 προσθέτου τον τίτλο στο χάρτη όπως φαίνεται στην εικόνα 7 που θα φαίνεται κατά την αποθήκευση και αποτύπωση πάνω στον χάρτη.



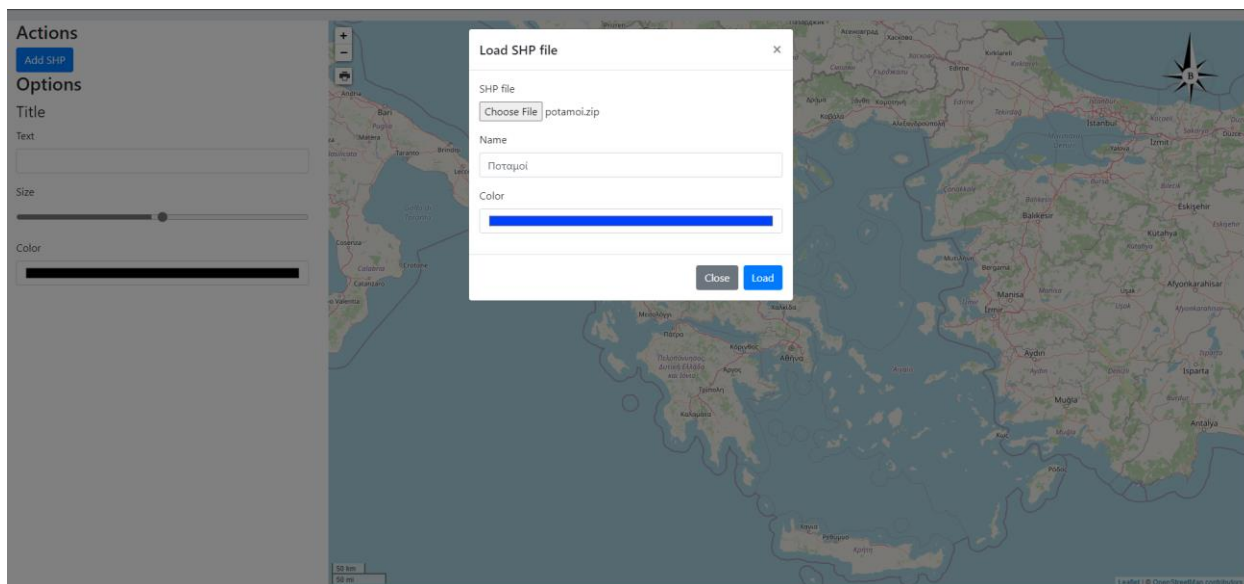
Εικόνα 7 Εισαγωγή τίτλου στην εφαρμογή

Με την επιλογή 2 (size) επιλέγουμε το μέγεθος του τίτλου πάνω στον χάρτη και με την επιλογή 3 (color) το χρώμα που θα έχει ο τίτλος.



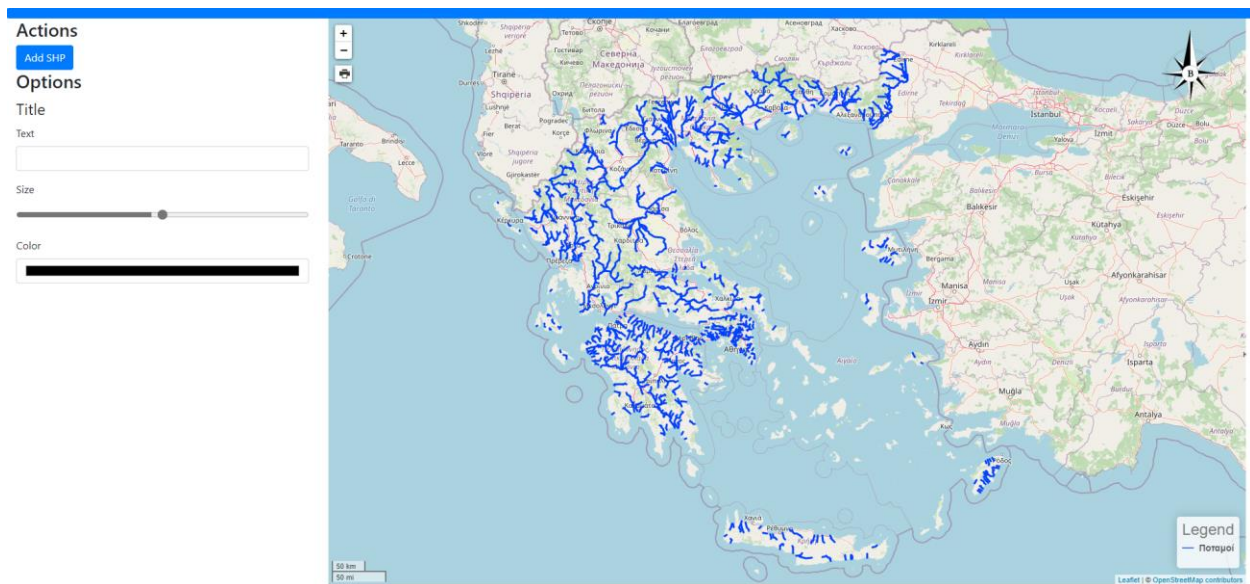
Εικόνα 8 Εισαγωγή αρχείου Shapefile

Όταν πατάμε την εντολή 1 (Add SHP) εμφανίζεται ένα καινούργιο παράθυρο όπως φαίνεται στην εικόνα 8 που μας ζητάει να επιλέξουμε το zip αρχείο όπου περιέχει το shapfile στην συνέχεια δίνουμε ένα όνομα που θα εμφανίζεται στο υπόμνημα και το χρώμα που θα αποτυπώνεται πάνω στο χάρτη. Σε περίπτωση που δεν επιλέξουμε όνομα βάζει αυτόματα από το όνομα του zip αρχείου και τυχαίο χρώμα.



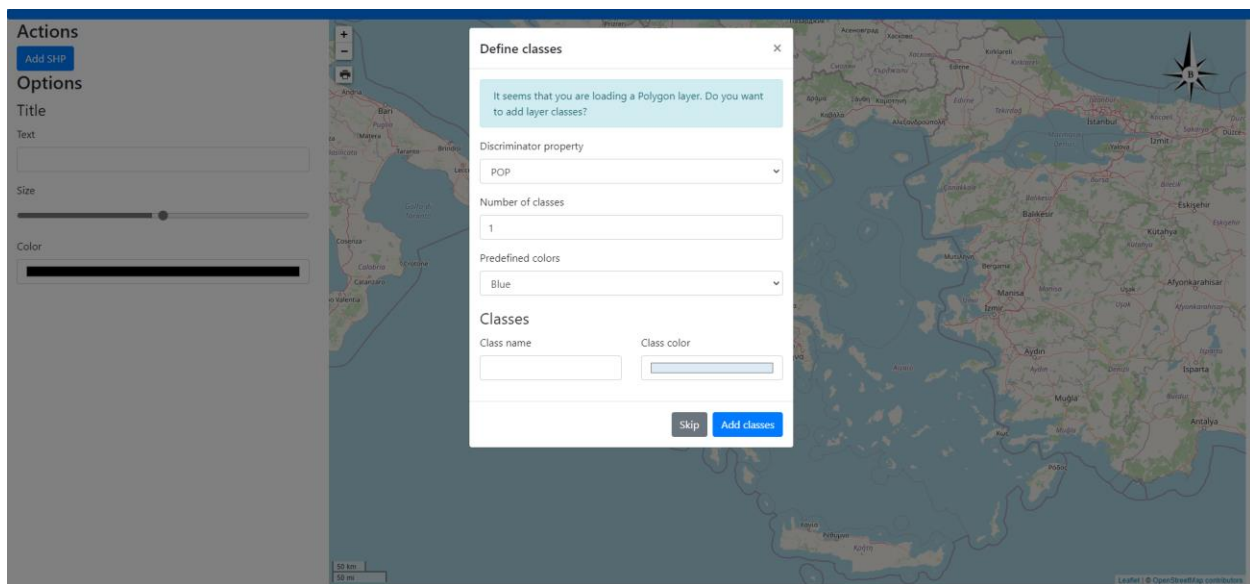
Εικόνα 9 Επιλογή χρώματος και ονόματος στο αρχείο εισαγωγής

Στο παραπάνω εικόνα 9 εισάγουμε τους ποταμούς της Ελλάδας δίνοντας όνομα «Παταμοί» και βάζοντας χρώμα μπλε



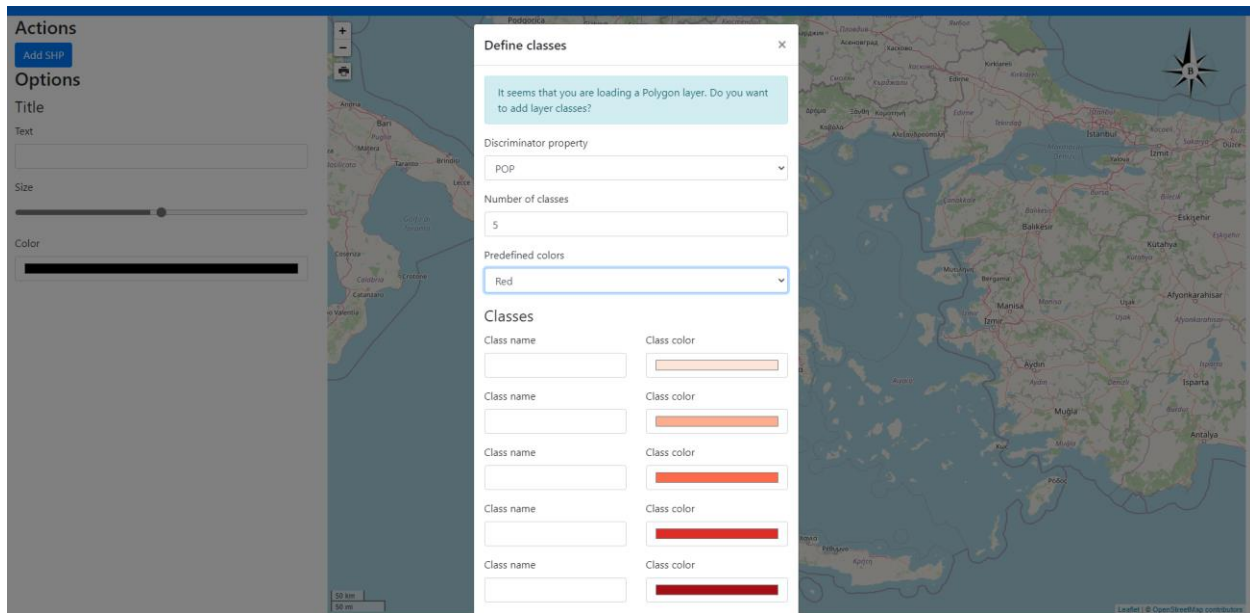
Εικόνα 10 Παρουσίαση αρχείου εισαγωγής στην εφαρμογή

Παρατηρούμε στην εικόνα 10 ότι έχει εμφανίσει τους ποταμούς σύμφωνα με τα στοιχεία που περιέχει το shp αρχείο. Κάτω δεξιά εμφανίζεται το υπόμνημα.



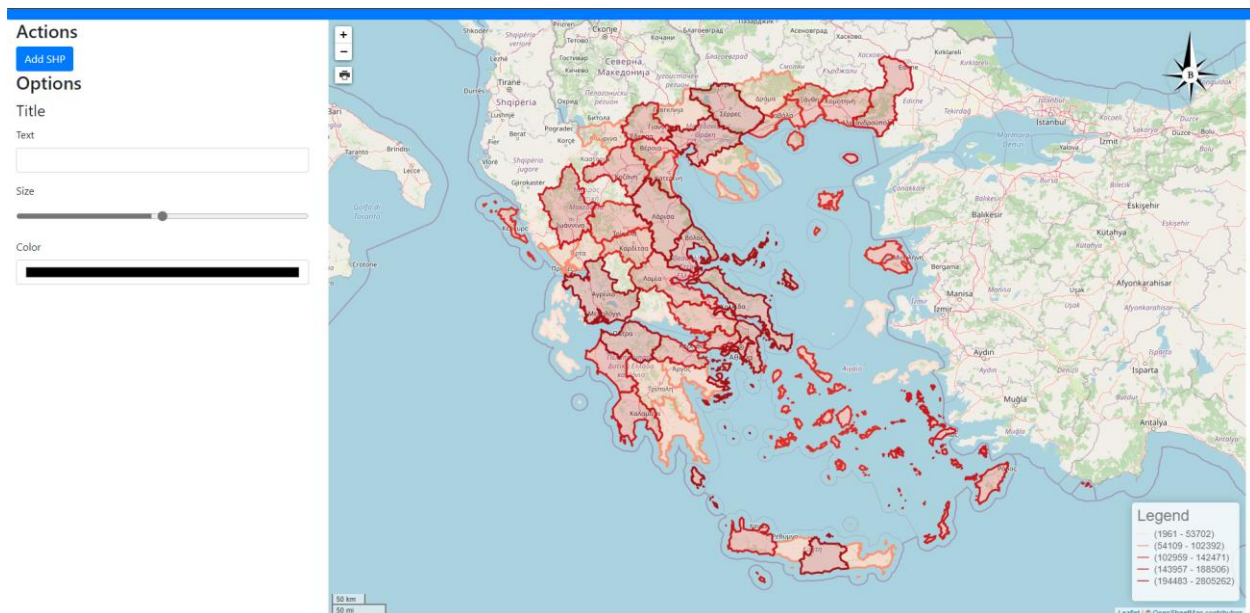
Εικόνα 11 Δημιουργία χωρικής ταξινόμησης

Στο παραπάνω παράδειγμα στην εικόνα 11 έχουμε προσθέσει του Νομούς ΟΚΧΕ έχοντας την δυνατότητα να δημιουργήσουμε κλάσεις όπως φαίνεται στην εικόνα 12 σύμφωνα με αυτό που θέλουμε να παρουσιάσουμε.



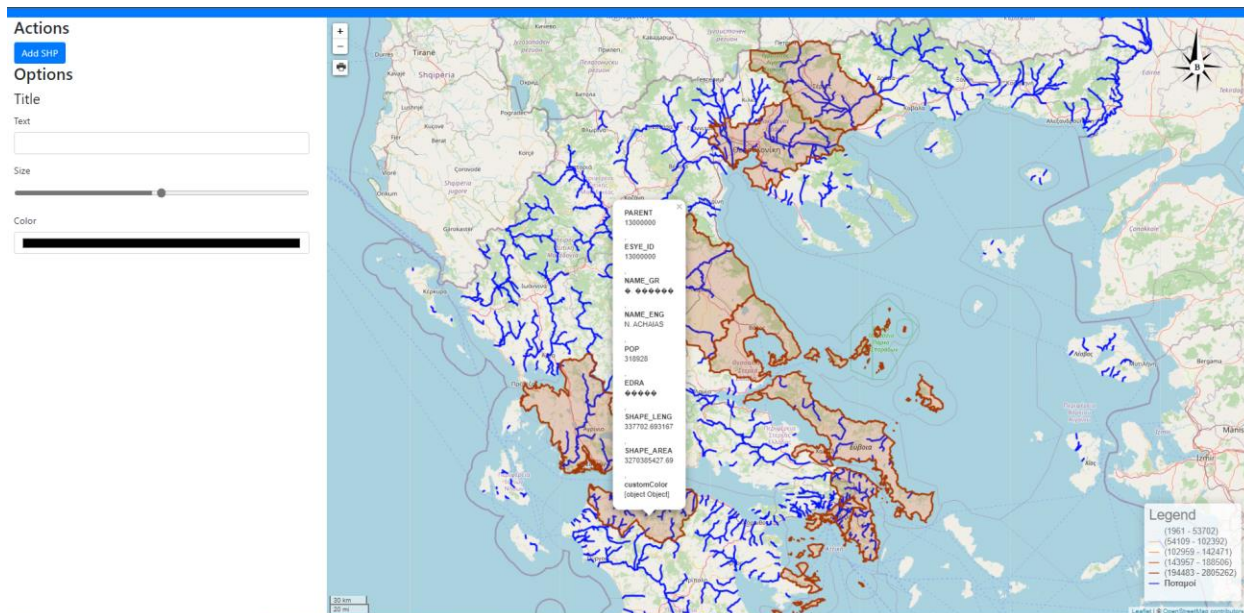
Εικόνα 12 Δημιουργία χωρικής ταξινόμησης

Επιλέγοντας την δημιουργία 5 κλάσεων με βάση το POP δηλαδή τον πληθυσμό μας δίνεται η δυνατότητα στην επιλογή του χρώματος και του ονόματος της κάθε κλάσης.



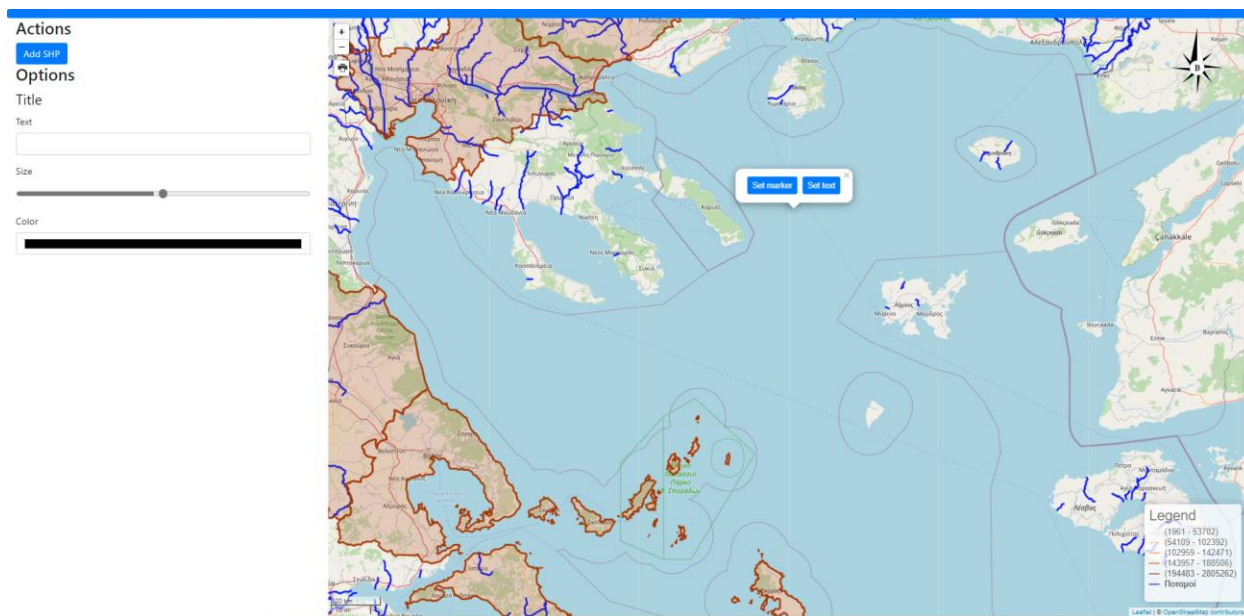
Εικόνα 13 Παρουσίαση χωρικής κατανομής

Παρατηρούμε στην εικόνα 13 ότι έχει γίνει αποτύπωση 5 διαφορετικών κλάσεων ανάλογα με τον πληθυσμό. Στον υπόμνημα του χάρτη φαίνονται οι κυμάνσεις των τιμών των κλάσεων και η αντιστοιχισή τους με τους κατάλληλους χρωματισμούς



Εικόνα 14 Εμφάνιση πληροφοριών από τα αρχεία shapefile

Πατώντας πάνω στο σημείο όπου θέλουμε να αντλήσουμε όπως φαίνεται στην εικόνα 14 ανοίγει ένα παράθυρο όπου εμφανίζει τις πληροφορίες της περιοχής που περιέχει το αρχείου SHP.



Εικόνα 15 Εισαγωγή πινέζας και κειμένου σε οποιοδήποτε σημείο.

Επιπλέον πατώντας σε οποιοδήποτε σημείο πάνω στον χάρτη έχουμε την δυνατότητα να προσθέσουμε ένα σημείο ή να γράψουμε κάποιο κείμενο όπως φαίνεται στην εικόνα 15.

Τέλος τις πληροφορίες που αποτυπώνονται πάνω στον χάρτη έχουμε την δυνατότητα να την αποθηκεύσουμε σε PDF αρχείο.

4.2 Σύγκριση λογισμικών WebGIS

4.2.1 Esri ArcGIS Server

Η Εταιρεία ESRI, που είναι μιας από τις πλέον γνωστές και μεγάλες εταιρείες παραγωγής λογισμικού για διαχείριση γεωγραφικής πληροφορίας (GIS Software) και δραστηριοποιείται στο χώρο πάνω από 30 χρόνια. Η πρώτη διανομή χαρτών με τη χρήση διαδικτύου παρατηρήθηκε μετά από την εμφάνιση του ArcView IMS και του MapObject IMS. . Λόγω των περιορισμών όμως στις τεχνολογίες του δικτύου αλλά και των λογισμικών GIS, τα παραπάνω προϊόντα παρουσίασαν σημαντικά προβλήματα σταθερότητας των διακομιστών των χαρτών (map server) αλλά και στην ευελιξία δόμησης εξειδικευμένων εφαρμογών. Με αφορμή τα παραπάνω μειονεκτήματα, η εταιρεία ESRI, παρουσιάζει το 2000 μία ολοκαίνουργια εφαρμογή που μπορεί να σταθεί ανεξάρτητα από τα άλλα λογισμικά της, έχει μεγαλύτερη ευελιξία στην δόμηση εφαρμογών και σταθερότητα στους διακομιστές χαρτών, τον ArcIMS. Σήμερα, η εταιρεία ESRI προσφέρει τρία διαφορετικά προϊόντα για δόμηση εφαρμογών διαδικτύου, μεγαλώνοντας την γκάμα δυνατοτήτων της οικογένειας του ESRI ArcGIS. (ESRI, 2005) Αυτά είναι τα:

1. ArcIMS
2. ArcGIS Image Server
3. ArcGIS Server

Στην παρούσα εργασία, επιλέχθηκε το λογισμικό του ArcGIS Server και συγκεκριμένα η έκδοση enterprise ή αλλιώς advanced (πρόκειται για την πλήρη έκδοση που μπορεί να υποστηρίξει όλες τις ικανότητες του λογισμικού).

Σημαντικά σημεία για το συγκεκριμένο λογισμικό αναλύονται παρακάτω:

- Συγκέντρωση του λειτουργικού συστήματος στους απαραίτητους servers και διανομή σε πλήθος χρηστών μέσω δικτύου
- Ισχυρά εργαλεία χωρικής ανάλυσης
- Ανοιχτή διαλειτουργική πλατφόρμα λειτουργίας
- Ικανότητα δημιουργίας εξειδικευμένων εφαρμογών χρησιμοποιώντας .NET ή Java
- Μειωμένο κόστος απόκτησης και πρόσβασης στο σύστημα
- Ευέλικτο σύστημα ανάλογα με τις εκάστοτε ανάγκες, μπορεί να εξυπηρετήσει πλήθος απαιτήσεων ή μία μικρή επιχείρηση.
- Out of the box λειτουργικότητα που παρέχεται σε τρία διαφορετικά επίπεδα (basic – standard – advanced)
- Εύκολη παραγωγή και δημοσίευση χαρτών με παροχή πλήθους υπηρεσιών διαδικτύου
- Άμεση σε πραγματικό χρόνο διαχείριση χωρικών δεδομένων τόσο επίπεδο περιγραφικών χαρακτηριστικών όσο και σε επίπεδο γραφικών
- Ύπαρξη πλήθους έτοιμων επεκτάσεων ανάλογα με τις ανάγκες της εφαρμογής

Όσο αφορά τα τεχνολογικά – συστατικά μέρη του ArcGIS Server αποτελείται από τα ακόλουθα:

- GIS Server
- Clients
- ArcGIS Desktop content authors
- Web Server
- Data server
- Manager and ArcCatalog administrators

GIS Server

Ο GIS Server διαχειρίζεται πόρους για το GIS όπως οι χάρτες, οι σφαίρες, τα εργαλεία για την γεωκωδικοποίηση αλλά και τα συστήματα για την εύρεση διευθύνσεων. Παράλληλα τα παρέχει υπό τη μορφή υπηρεσιών στις διάφορες εφαρμογές. Στην περίπτωση που ένας χρήστης συντάξει μια απαίτηση που ενεργοποιεί μια υπηρεσία, ο GIS Server απαντά σε αυτή και τη δίνει στο χρήστη.

Clients

Οι χρήστες του συστήματος (πάντοτε μέσα από την κατάλληλη εφαρμογή), μπορεί να είναι χρήστες στο διαδίκτυο (web clients), κινούμενοι (mobile clients) ακόμα και άτομα που χρησιμοποιούν μία desktop εφαρμογή και συνδέονται με τη βοήθεια του πρωτοκόλλου HTTP στις υπηρεσίες του παγκόσμιου ιστού ή υπηρεσίες ενός τοπικού δικτύου (LAN ή WAN).

Web Server

Ο Web Server αποθηκεύει τις υπηρεσίες και τις εφαρμογές διαδικτύου (web applications and services) οι οποίες χρησιμοποιούν τους πόρους που βρίσκονται αποθηκευμένοι στον GIS Server.

Data Server

Ο server των δεδομένων , γνωστός και ως data server, περιλαμβάνει πόρους του GIS που δεσμεύονται υπό τη μορφή υπηρεσιών στον GIS Server. Οι πόροι είναι χάρτες, βάσεις δεδομένων ,εργαλεία γεωκωδικοποίησης, αλλά και εργαλεία εύρεσης διευθύνσεων. Συνήθως χρησιμοποιείται μία βάση δεδομένων που βασίζεται στις σχέσεις, έτσι ώστε να παρέχει ευελιξία, ασφάλεια, ευστάθεια και καλή απόδοση στην διαχείριση και χρήση των δεδομένων.

Manager and ArcCatalog administrators

Ο Manager είναι μία web εφαρμογή που προσφέρει διαχείριση του GIS, δημιουργία εφαρμογών διαδικτύου και διαχείριση αυτών και δημοσίευση χαρτών του ArcGIS Explorer στον server. Ο ArcCatalog μπορεί να χρησιμοποιηθεί για την προσθήκη συνδέσεων στους GIS Servers είτε για γενική χρήση είτε για διαχείριση του διακομιστή. Παρέχει επίσης ένα πλαίσιο εργασίας, που χρησιμοποιείται μόνο από εξειδικευμένους χρήστες (GIS Professionals), για την δημοσίευση πόρων του GIS ως υπηρεσίες του GIS.

Web Services

Οι πέντε κύριες υπηρεσίες που παρέχονται από το λογισμικό είναι οι ακόλουθες:

1. Map services
2. Globe services
3. Geoprocessing services
4. Geocoding services
5. Geodata services

- Όσο αφορά την διαχείριση των βάσεων δεδομένων πραγματοποιείται μέσω του ArcSDE, ένα ισχυρό εργαλείο που επικεντρώνεται στη δημιουργία και στη διαχείριση βάσεων δεδομένων του server. Οι βάσεις δεδομένων που υποστηρίζει το ArcGIS είναι: IBM DB2
- Informix
- Oracle
- SQL Server

Τέλος τα αρχεία που υποστηρίζονται είναι:

- ArcINFO
- DGN
- OLE DB TABLES
- SDE layers
- DWG
- DXF
- PC ArcINFO coverages
- SDC
- VPF

Σε δεδομένα κανονικοποιημένης μορφής (raster), υποστηρίζεται μεγάλος αριθμός δομών μερικές από τις οποίες είναι:

- Compressed ARC Digitized Raster Graphics (CADRG)
- ADRG Image (.IMG)
- ADRG Legend (.LGG)
- ADRG Overview (.OVR)
- DIGEST
- ArcSDE raster
- BSB charts
- Controlled Image Base (CIB)
- ER Mapper (.ERS)
- ECW
- ERDAS IMAGINE (.IMG)

- ERDAS Lan (.Lan)
- ERDAS (.GIS)
- ERDAS Raw (.RAW)
- BIL

Όσο αφορά τα αρχεία που μπορεί να εξαγάγει το συγκεκριμένο πρόγραμμα αναλύεται παρακάτω:

Για raster δεδομένα:

- Arcii
- Float
- Point
- Polygon
- Polyline

Για μεταδεδομένα:

- HTML
- XML
- FGDC CSDGM (XML)
- FAQ-style HTML
- SGML
- Structured text

Εξαγωγή σε CAD δεδομένα:

- DGN
- DWG
- DXF

Επιπλέον επιτρέπεται η εξαγωγή των feature classes σε coverage.Επιτρέπεται η εξαγωγή των πινακοποιημένων δεδομένων σε:

- dBASE
- INFO
- OLE
- DB
- GDB

Τέλος υποστηρίζεται η εξαγωγή εικόνων σε:

- ESRI GRID
- ERDAS IMAGINE
- TIFF

4.2.2 Autodesk MapGuide

Η εταιρεία Autodesk, η πρωτοπόρος εταιρεία παραγωγής σχεδιαστικών προγραμμάτων, εισήχθη στον χώρο παραγωγής προγραμμάτων GIS στα μέσα της δεκαετίας του 90, αντιλαμβανόμενη την παρουσία ενός τεράστιου αριθμού ψηφιακών χαρτών που είχαν δημιουργηθεί σε μορφή .dwg με την σειρά των σχεδιαστικών προγραμμάτων της Autodesk. Η εταιρεία έχει δημιουργήσει τα τελευταία χρόνια ένα λογισμικό ανοιχτού κώδικα, το MapGuide open source. Πρόκειται για μία πλατφόρμα βασισμένη στο διαδίκτυο που παρέχει την δυνατότητα ανάπτυξης και διαχείρισης εφαρμογών χαρτογράφησης μέσω του παγκόσμιου ιστού (web mapping applications) αλλά την ανάπτυξη χωρικών υπηρεσιών (geospatial web services). Το λογισμικό του MapGuide πρωτοεμφανίστηκε το 1995 σαν Argus MapGuide από την εταιρεία Argus Technologies. Το φθινόπωρο του 1996 παρουσιάστηκε η πρώτη έκδοση του Autodesk MapGuide, η έκδοση 2.0. Η πρώτη έκδοση του λογισμικού παρουσιάστηκε τον Νοέμβριο του 2005. Σήμερα είναι διαθέσιμη η έκδοση 1.1 και αναμένεται η έκδοση 1.2. Στην παρούσα εργασία η έκδοση που χρησιμοποιήθηκε είναι η έκδοση 1.1. (Architecture, 2011)

Σημαντικά σημεία του προγράμματος αυτού είναι:

- Γρήγορη και ευέλικτη πλατφόρμα ανάπτυξης εφαρμογών
- Open source λογισμικό που παρέχεται με πολύ μικρό κόστος στους χρήστες
- Πρόσβαση σε χωρικά δεδομένα τόσο διανυσματικής όσο και κανονικοποιημένης δομής +
- Υποστήριξη ποικίλων δομών δεδομένων χωρίς μετατροπή αυτών
- Παροχή μίας πλούσιας παλέτας εργαλείων για χωρική ανάλυση
- Περιλαμβάνονται viewers που λειτουργούν σε οποιονδήποτε browser και σε οποιαδήποτε πλατφόρμα λειτουργικού συστήματος

Το σύστημα αποτελείται από: το Mapguide server, το Mapguide viewer, το Mapguide studio, το Mapguide web studio και το Mapguide site administrator.

Mapguide server

Ο server του MapGuide είναι αυτός που διατηρεί τις υπηρεσίες του MapGuide Open Source (MapGuide Open Source services) και ανταποκρίνεται στα ερωτήματα που θέτει ο εκάστοτε πελάτης μέσω του πρωτοκόλλου TCP/IP. Οι υπηρεσίες που παρέχονται από το σύστημα είναι:

- Site Service
- Drawing Service
- Resource Service
- Feature Service

Mapguide viewer

Παρέχονται δύο ειδών viewer:

1. DWF viewer
2. AJAX viewer

Mapguide studio

Το MapGuide Studio διαχειρίζεται όλα τα θέματα προετοιμασίας των χαρτών και των χωρικών δεδομένων με στόχο την χρήση τους και την δημοσίευσή τους μέσω του MapGuide Open Source.

Mapguide site administrator

Πρόκειται για μία εφαρμογή διαχείρισης μίας ιστοσελίδας και των διακομιστών (servers) αυτής. Η συγκεκριμένη εφαρμογή χρησιμοποιεί ένα κατάλληλο interface με το από οποιονδήποτε web browser.

Στο λογισμικό του MapGuide Open Source όλα τα δεδομένα αποθηκεύονται στα λεγόμενα MapGuide Repositories εκτός από αυτά που είναι αποθηκευμένα σε εξωτερικές βάσεις δεδομένων. Από τις τελευταίες εκδόσεις του MapGuide (MapGuide Open Source 2.0 και MapGuide 6.5 υποστηρίζονται οι ακόλουθες βάσεις δεδομένων:

- Oracle
- SQL Server
- Microsoft Access
- Dbase και
- IBM DB2 με την κατάλληλη παραμετροποίηση

Αποκλειστική χρήση των FDO (Feature Data Objects) για την πρόσβαση τόσο διανυσματικών όσο και κανονικοποιημένων δεδομένων. Συγκεκριμένα οι προεπιλεγμένες δομές δεδομένων που υποστηρίζονται είναι οι ακόλουθες:

- Υποστήριξη των διανυσματικών δομών δεδομένων ESRI SHP SDF, AutoCAD dwg.
- Υποστήριξη κανονικοποιημένων δομών δεδομένων μέσω GDAL
- Υποστήριξη OGC WMS και WFS
- Αποκλειστική χρήση του DWF viewer για εφαρμογές υλοποιημένες πάνω σε περιβάλλον CAD.

Υποστηρίζονται οι ακόλουθοι τύποι raster δομής:

- ECW
- MrSID
- TIFF
- JPEG

Το συγκεκριμένο πρόγραμμα υποστηρίζει την εξαγωγή εικόνων σε:

- PNG
- GIF

Ικανότητα εξαγωγής ολόκληρου του χάρτη ή τμήματος αυτού. Απόδοση γραφικών με χρήση του DWF viewer. Υποστηρίζεται η ικανότητα δόμηση εργαλείων για την εξαγωγή δεδομένων στην δομή που επιθυμεί ο κατασκευαστής της ιστοσελίδας.

Κριτήρια	Το λογισμικό μου	ESRI ArcGIS server	Autodesk
Εισαγωγή δεδομένων	Υποστηρίζει δεδομένα: <ul style="list-style-type: none"> • Zip αρχεία που περιέχουν όλα τα δεδομένα shp. 	Υποστηρίζονται δεδομένα των ακόλουθων δομών: <ul style="list-style-type: none"> • ArcINFO • DGN • DWG • DXF • OLE DB TABLES • SDC • PC ArcINFO coverages • Shapefiles • TIN • SDE layers • Text files (.TXT) 	Αποκλειστική χρήση των FDO (Feature Data Objects) για την πρόσβαση τόσο διανυσματικών όσο και κανονικοποιημένων δεδομένων. Συγκεκριμένα οι προεπιλεγμένες δομές δεδομένων που υποστηρίζονται είναι οι ακόλουθες: <ul style="list-style-type: none"> • Υποστήριξη των διανυσματικών δομών δεδομένων ESRI SHP SDF, AutoCAD dwg. • Υποστήριξη κανονικοποιημένων δομών δεδομένων μέσω GDAL • Υποστήριξη OGC WMS και WFS

Κριτήρια	Το λογισμικό μου	ESRI ArcGIS server	Autodesk
Μέτρηση απόστασης (Measure distance)	–	✓	✓
Μέτρηση περιοχής (Measure area)	–	✓	✓
Εμφάνιση Συντεταγμένων (Coordinate display)	✓	✓	✓
Ορισμός μονάδων μέτρησης (Set measure units)	–	✓	✓

Εισαγωγή σημείου και κειμένου στο χάρτη	✓	✓	✓
Επιλογή βάση σημείου (Select by point)	✓	–	✓
Επιλογή βάση γραμμής (Select by line)	✓	–	✓
Επιλογή βάση τετραγώνου (Select by rectangle)	✓	–	✓
Επιλογή βάση πολυγώνου (Select by polygon)	✓	–	✓
Μεγέθυνση, σμίκρυνση (Zoom in, zoom out)	✓	✓	✓
Μεγέθυνση σε καθορισμένη περιοχή (Zoom area)	✓	✓	✓
Μεγέθυνση σε όλο το εύρος της περιοχής του χάρτη (Zoom full extends)	✓	✓	✓
Γενικός χάρτης (Overview map)	✓	✓	✓
Ανανέωση, επαναφόρτωση ιστοσελίδας (Reload site)	✓	✓	✓

4.3.3 Συζήτηση

Το λογισμικό μου	ESRI ArcGIS server	Autodesk
Άμεση προβολή του περιβάλλοντος εργασίας	Χρειάζεται να δημιουργήσεις κωδικό πρόσβασης	Χρειάζεται να δημιουργήσεις κωδικό πρόσβασης
Λόγω της τοπικής λειτουργίας γίνεται άμεσα η εισαγωγή και η εμφάνιση των αρχείων	Λόγω της online λειτουργίας εξαρτάται από την ταχύτητα του διαδικτύου για την προσθήκη και εμφάνιση των αρχείων	Δουλεύει μόνο με βάση δεδομένων
Απλότητα στην εισαγωγή του ονόματος και της απεικόνισης του αρχείου και η άμεση δημιουργία υπομνήματος που παρουσιάζονται στο χάρτη	Μετά την εισαγωγή του αρχείου οι μη έμπειροι χρήστες θα δυσκολευτούν να αλλάξουν το όνομα και την απεικόνισή του	-
Δεν γίνεται να αποθηκευτεί η εργασία που πραγματοποιείται	Αποθήκευση διαδικτυακά της εργασίας	Αποθήκευση διαδικτυακά της εργασίας

Πίνακας 3 Σύγκριση Λογισμικών

Στον πίνακα 3 παρουσιάζεται η σύγκριση του λογισμικού που δημιουργήσαμε με άλλα διαδικτυακά λογισμικά. Στην διαδικτυακή εφαρμογή που δημιουργήσαμε έχουμε την δυνατότητα για άμεση προβολή του περιβάλλον εργασίας σε σχέση με τα άλλα δύο λογισμικά που χρειάζεται να πραγματοποιηθεί η δημιουργία κάποιου κωδικού πρόσβασης. Λόγο του ότι για την λειτουργία της διαδικτυακής εφαρμογής δεν χρειάζεται να βρίσκεσαι σε κάποιο δίκτυο δηλαδή δουλεύει τοπικά στον ηλεκτρονικό υπολογιστή η εισαγωγή του αρχείου επεξεργασία γίνεται άμεσα χωρίς να επηρεάζετε η εισαγωγή του από την ταχύτητα του διαδικτύου σου. Επιπλέον, κατά την εισαγωγή του αρχείου που επιθυμούμε να επεξεργαστούμε γίνεται κατευθείαν η επιλογή του ονόματος του αρχείου και το χρώμα εμφάνισης με αποτέλεσμα να γίνεται απευθείας η δημιουργία υπομνήματος στο χάρτη. Τέλος το αρνητικό που υπάρχει στην συγκεκριμένη εφαρμογή είναι ότι δεν γίνεται να αποθηκευτεί η εργασία σου και να την συνεχίσεις οποιαδήποτε στιγμή επιθυμείς.

Κεφάλαιο 5. Συμπεράσματα

Οι χρήστες έχουν πολλές φορές ανάγκη από περισσότερο εξειδικευμένες εργασίες, όπως η δημιουργία θεματικών χαρτών. Οι υπάρχουσες εφαρμογές για το σκοπό αυτό δεν είναι ιδιαίτερα φιλικές προς τον χρήστη και απαιτούν εξειδικευμένες γνώσεις. Στην συγκεκριμένη διπλωματική θα δημιουργήσουμε μια διαδικτυακή εφαρμογή όπου θα αποτελεί ένα μέσον προ βάσιμο και εύκολο στην χρήση για κάθε πολίτη ώστε να τα επεξεργαστεί και να παρουσιάσει τα γεωχωρικά δεδομένα.

Στην παρούσα διπλωματική εργασία παρουσιάστηκε η ανάπτυξη μιας διαδικτυακής εφαρμογής για την οπτικοποίηση γεωγραφικών δεδομένων και η προσπάθεια μελέτης και ανάλυσης των υπηρεσιών Γεωγραφικών συστημάτων πληροφοριών μέσω διαδικτύου (WebGIS services) μέσα από μια διαδικασία σύγκρισης ανάμεσα σε δυο εταιρίες παραγωγής λογισμικών και τα αντίστοιχα προϊόντα τους. Αφορά τις εταιρίες ESRI και Autodesk και τα αντίστοιχα λογισμικά τους ArcGIS Server και MapGuide Open Source.

Στο κεφάλαιο 4 πραγματοποιήθηκε η παρουσίαση και σύγκριση των δυο λογισμικών παραγωγής ΓΣΠ με την διαδικτυακή εφαρμογή. Η σύγκριση πραγματοποιήθηκε σε επίπεδο εισαγωγής, διαχείρισης, ανάλυσης και απόδοσης των δεδομένων και των λειτουργιών που παρέχουν αυτά τα συστήματα. Μέσα από όλη την παρουσία και σύγκριση προέκυψαν τα ακόλουθα συμπεράσματα όσο αφορά την εφαρμογή:

- Όσο αφορά την διαδικτυακή εφαρμογή για την οπτικοποίηση γεωγραφικών δεδομένων είναι μια απλούστατη εφαρμογή ανοιχτού κώδικα που μπορεί να αναπτυχθεί και να εξελιχθεί. Σίγουρα δεν μπορεί να συγκριθεί με μεγάλες εταιρίες αλλά για την παρουσίαση των δεδομένων που εισάγουμε και η βασική τους παρουσίαση αλλάζοντας χρώμα και μέγεθος αλλά και η εξαγωγή του σε αρχείο PDF είναι πολύ σημαντική για μια καινούργια αρχή.
- Το λογισμικό της ESRI είναι μία κλιμακούμενη εφαρμογή όπου υπάρχει μία βασική έκδοση με την δυνατότητα επέκτασης των λειτουργιών ανάλογα με τις απαιτήσεις του συστήματος που δομείται. Υπάρχει πληθώρα επεκτάσεων που είναι διαθέσιμες από την εταιρεία αλλά και οι οποίες μπορούν να προσαρμοστούν περαιτέρω με απλές προγραμματιστικές μεθόδους σε κάποια από τις γλώσσες που υποστηρίζονται.
- Στο τρίτο και τελευταίο λογισμικό (MapGuide Open Source) τα πράγματα είναι πιο ρευστά καθώς όλα χρήζουν την ανάγκη καλών προγραμματιστικών γνώσεων. Εντούτοις υπάρχουν έτοιμες βιβλιοθήκες και εγχειρίδια για την εύκολη δημιουργία μίας ιστοσελίδας αλλά και πάλι δεν υπάρχει το φιλικό περιβάλλον των Windows, όλα είναι γραμμένα σε κώδικα. Μία πρώτη προσπάθεια απλοποίησης των λειτουργιών αποτελεί το MapGuide Studio το οποίο παρέχει τυποποιημένες εντολές για την δόμηση μίας ιστοσελίδας.

Όσο αφορά το κόστος για την απόκτηση και λειτουργίας του κάθε λογισμικού η διαδικτυακή εφαρμογή που δημιουργήσαμε δεν βρίσκεται κάπου διαθέσιμη για απόκτηση είναι για τοπική χρήση και εννοείτε ότι βρίσκεται σε δωρεάν μορφή. Από τα λογισμικά όπου πραγματοποιήθηκε η σύγκριση το οικονομικότερο είναι προφανώς το Autodesk MapGuide Open Source όπου είναι ένα δωρεάν λογισμικό και η μόνη επιβάρυνση είναι η γλώσσα προγραμματισμού που θα

χρησιμοποιηθεί. Η ακριβότερη εφαρμογή στην συγκεκριμένη περίπτωση είναι το ArcGIS Server της ESRI με την ύπαρξη μιας βασικής έκδοσης και κλιμακούμενο κόστος ανάλογα με τις επεκτάσεις που επιθυμεί κανείς για την δόμηση της εφαρμογής.

Σχετικά με την πλατφόρμα εγκατάστασης και λειτουργίας των λογισμικών τα λογισμικά της ESRI αλλά και της Autodesk μπορούν να εγκατασταθούν σε λειτουργικά συστήματα Windows αλλά και Linux. Η διαδικτυακή εφαρμογή που δημιουργήσαμε λόγο του ότι δουλεύει σε φυλλομετρητή μπορεί να λειτουργήσει και σε λογισμικό Windows και σε Linux αλλά και σε MacOS.

Οι απαιτήσεις τεχνικών υλικών είναι χαμηλές και όλα τα λογισμικά μπορούν να εγκατασταθούν ακόμα και σε παλαιούς υπολογιστές (Pentium III) που διαθέτουν την απαραίτητη μνήμη και αποθηκευτικό χώρο. Βέβαια είναι γεγονός ότι όσο πιο γρήγορος είναι ο επεξεργαστής, όσο πιο καινούργια η κάρτα γραφικών, η μνήμη και ο αποθηκευτικός χώρος, το σύστημα λειτουργεί πολύ καλύτερα και πιο αποδοτικά.

Όσον αφορά τις βάσεις δεδομένων τα περισσότερα προβλήματα που είχαν οι προηγούμενες εκδόσεις για την μη υποστήριξη κάποιων βασικών βάσεων δεδομένων έχουν ξεπεραστεί και πλέον βρισκόμαστε σε μία κατάσταση όπου υποστηρίζονται και από τα τρία λογισμικά οι περισσότερες διαδεδομένες βάσεις δεδομένων. Παράλληλα παρέχονται ευκολόχρηστα εργαλεία διαχείρισης αυτών, ιδιαίτερα μάλιστα στην περίπτωση του ArcGIS SERVER όπου υπάρχει διαθέσιμο το ArcSDE.

Στην διαλειτουργικότητα των δεδομένων και των δομών που υποστηρίζονται, το πάνω χέρι παρουσιάζεται να έχει η ESRI με την επέκταση Data Interoperability Extension σύμφωνα με την οποία μπορούν να υποστηριχτούν πάνω από 70 χωρικές δομές δεδομένων για εισαγωγή και πάνω από 50 για εξαγωγή.

Οι viewers που μπορούν να χρησιμοποιήσουν οι χρήστες δεν φαίνεται να είναι πρόβλημα αφού αν δεν υποστηρίζεται κάποιος από αυτούς μπορεί να αντληθεί το κατάλληλο plug-in ή ActiveX Control για την ομαλή λειτουργία του συστήματος.

Τέλος όσον αφορά την επεξεργασία και τα εργαλεία διαχείρισης και ανάλυσης, φαίνεται να υπερτερεί η εταιρεία ESRI με την παροχή ποικίλων και ισχυρών εργαλείων για την δόμηση, ανάλυση και διαχείριση των δεδομένων. Ένα από τα ισχυρότερα εργαλεία της είναι η «on-line» επεξεργασία χωρικών αντικειμένων όσον αφορά τις γεωμετρικές τους ιδιότητες όσο και τις περιγραφικές τους, λειτουργία η οποία δεν παρέχεται από τα άλλα λογισμικά. Εν τούτοις θα μπορούσε να πραγματοποιηθεί στο MapGuide Open Source με την συμβολή ισχυρών προγραμματιστικών γνώσεων.

Όσο αφορά την επέκταση της διαδικτυακής εφαρμογής θα μπορούσαμε να προσθέσουμε εκτός από διανυσματικά δεδομένα να παρουσιάζει στον χάρτη και δεδομένα κανονικοποιημένης μορφής (raster) σε μορφή TIF ή JPEG. Επιπλέον να μπορεί να γίνει αποθήκευση των πληροφοριών που αποτυπώνονται στο χάρτη σε εικόνα εκτός από PDF.

Βιβλιογραφία

- Architecture, A. (2011) ‘Autodesk’, (March 2010).
- Avenue, M. (1810) *HTML and JavaScript for Visual Learners, Change*.
- Brooks, D. R. (2007) *An Introduction to HTML and JavaScript, An Introduction to HTML and JavaScript*. doi: 10.1007/978-1-84628-657-5.
- ESRI (2005) ‘What is ArcGIS 9.1?’, *Analyst, The*.
- Fu, P. (2015) ‘Getting to know WebGIS’, p. 390.
- ‘Getbootstrap.Com’ (2020). Available at: <http://getbootstrap.com/>.
- Gis, E. (2008) *Encyclopedia of GIS, Choice Reviews Online*. doi: 10.5860/choice.45-5931.
- Harper, E. (2006) ‘Open-Source Technologies in Web-Based Gis and Mapping’, *Master’s Thesis, Northwest Missouri State University, Available at: [http://www.nwmissouri.edu/library/theses/HarperErik/Erik Harper Final Thesis.pdf](http://www.nwmissouri.edu/library/theses/HarperErik/Erik%20Harper%20Final%20Thesis.pdf)*.
- JQuery (2018) ‘Jquery.Com’. Available at: <https://jquery.com/>.
- Ken Arnold, James Gosling, D. H. (2005) ‘The Java Programming Language’, p. 928.
- Mitchell, T. (2005) *Web Mapping Illustrated*.
- Putz, S. (1994) ‘Interactive information services using World-Wide Web hypertext’, *Computer Networks and ISDN Systems*, 27(2), pp. 273–280. doi: 10.1016/0169-7552(94)90141-4.
- ‘shapefile-js @ github.com’ (2020). Available at: <https://github.com/calvinmetcalf/shapefile-js>.
- Smith, A. (1986) *This is a reproduction of a library book that was digitized by Google as part of an ongoing effort to preserve the information in books and make it universally accessible. https://books.google.com, Oxford University. Available at: https://books.googleusercontent.com/books/content?req=AKW5QaeZb8cCzWI4I5L5LPBCISJgAXxJxciOie9Nj7iGbkyHgkrSKm9_f_eQZBmoPJ-LQQeri7eZdQbiBrACMUvmaq8k5jh-CyOT97vEdjCbC9lZkltWgy565ZnSag8m-cDJYIEGvgQupUrfpQ872PWSk2UuDKmw2c-wbLtlqk2HWF21P_ywlLi4L6KmLa7khGsKsla*.
- Veenendaal, B. (2011) *Advances in Web-based GIS, Mapping Services and Applications, Advances in Web-based GIS, Mapping Services and Applications*. doi: 10.1201/b15452.
- Vladimir Agafonkin (2020) ‘Leafletjs.Com’. Available at: <https://leafletjs.com/>.
- Παναγιώτα, Α. Ν. (2016) ‘Συγκριση Λογισμικών WebGis’, (1), pp. 124–128.
- Συλαίου, Σ. (2012) ‘Επισκόπηση εφαρμογών της Εθελοντικής Γεωγραφικής Πληροφορίας με έμφαση στην Πολιτισμική Κληρονομιά’, pp. 15–22.

Παράρτημα - Κώδικας

Index.js Παρουσίαση του κώδικα:

```
// modules are defined as an array
// [ module function, map of requires ]
//
// map of requires is short require name -> numeric require
//
// anything defined in a previous bundle is accessed via the
// orig method which is the require for previous bundles
parcelRequire = (function (modules, cache, entry, globalName) {
  // Save the require from previous bundle to this closure if any
  var previousRequire = typeof parcelRequire === 'function' && parcelRequire;
  var nodeRequire = typeof require === 'function' && require;

  function newRequire(name, jumped) {
    if (!cache[name]) {
      if (!modules[name]) {
        // if we cannot find the module within our internal map or
        // cache jump to the current global require ie. the last bundle
        // that was added to the page.
        var currentRequire = typeof parcelRequire === 'function' && parcelRequire;
        if (!jumped && currentRequire) {
          return currentRequire(name, true);
        }
      }

      // If there are other bundles on this page the require from the
      // previous one is saved to 'previousRequire'. Repeat this as
      // many times as there are bundles until the module is found or
```

```

// we exhaust the require chain.
if (previousRequire) {
  return previousRequire(name, true);
}

// Try the node require function if it exists.
if (nodeRequire && typeof name === 'string') {
  return nodeRequire(name);
}

var err = new Error('Cannot find module \'' + name + '\'');
err.code = 'MODULE_NOT_FOUND';
throw err;
}

localRequire.resolve = resolve;
localRequire.cache = {};

var module = cache[name] = new newRequire.Module(name);

modules[name][0].call(module.exports, localRequire, module, module.exports, this);
}

return cache[name].exports;

function localRequire(x){
  return newRequire(localRequire.resolve(x));
}

```

```
function resolve(x){
  return modules[name][1][x] || x;
}
}
```

```
function Module(moduleName) {
  this.id = moduleName;
  this.bundle = newRequire;
  this.exports = {};
}
```

```
newRequire.isParcelRequire = true;
newRequire.Module = Module;
newRequire.modules = modules;
newRequire.cache = cache;
newRequire.parent = previousRequire;
newRequire.register = function (id, exports) {
  modules[id] = [function (require, module) {
    module.exports = exports;
  }, {}];
};
```

```
var error;
for (var i = 0; i < entry.length; i++) {
  try {
    newRequire(entry[i]);
  } catch (e) {
```

```

// Save first error but execute all entries
if (!error) {
    error = e;
}
}
}

if (entry.length) {
    // Expose entry point to Node, AMD or browser globals
    // Based on https://github.com/ForbesLindesay/umd/blob/master/template.js
    var mainExports = newRequire(entry[entry.length - 1]);

    // CommonJS
    if (typeof exports === "object" && typeof module !== "undefined") {
        module.exports = mainExports;

    // RequireJS
    } else if (typeof define === "function" && define.amd) {
        define(function () {
            return mainExports;
        });

    // <script>
    } else if (globalName) {
        this[globalName] = mainExports;
    }
}

```

```

// Override the current require with this new one
parcelRequire = newRequire;

if (error) {
  // throw error from earlier, _after updating parcelRequire_
  throw error;
}

return newRequire;
})({"../node_modules/parcel-bundler/src/builtins/bundle-
url.js":[function(require,module,exports) {
var bundleURL = null;

function getBundleURLCached() {
  if (!bundleURL) {
    bundleURL = getBundleURL();
  }

  return bundleURL;
}

function getBundleURL() {
  // Attempt to find the URL of the current script and use that as the base URL
  try {
    throw new Error();
  } catch (err) {
    var matches = (" + err.stack).match(/(https?|file|ftp|chrome-extension|moz-
extension):\\V[^\\n]+/g);

```

```

    if (matches) {
        return getBaseURL(matches[0]);
    }
}

return '/';
}

function getBaseURL(url) {
    return (" + url).replace(/^(?:https?|file|ftp|chrome-extension|moz-extension):\\\.+\)[^/]+$/, '$1') + '/';
}

exports.getBundleURL = getBundleURLCached;
exports.getBaseURL = getBaseURL;
},{ }], "../node_modules/parcel-bundler/src/builtins/css-loader.js":[function(require,module,exports) {
var bundle = require('./bundle-url');

function updateLink(link) {
    var newLink = link.cloneNode();

    newLink.onload = function () {
        link.remove();
    };

    newLink.href = link.href.split('?')[0] + '?' + Date.now();
    link.parentNode.insertBefore(newLink, link.nextSibling);
}

```

```

var cssTimeout = null;

function reloadCSS() {
  if (cssTimeout) {
    return;
  }

  cssTimeout = setTimeout(function () {
    var links = document.querySelectorAll('link[rel="stylesheet"]');

    for (var i = 0; i < links.length; i++) {
      if (bundle.getBaseURL(links[i].href) === bundle.getBundleURL()) {
        updateLink(links[i]);
      }
    }

    cssTimeout = null;
  }, 50);
}

module.exports = reloadCSS;
},{"/bundle-url":"./node_modules/parcel-bundler/src/builtins/bundle-
url.js"},{"/node_modules/parcel-bundler/src/builtins/hmr-
runtime.js":function(require,module,exports) {
var global = arguments[3];
var OVERLAY_ID = '__parcel__error__overlay__';
var OldModule = module.bundle.Module;

```

```

function Module(moduleName) {
  OldModule.call(this, moduleName);
  this.hot = {
    data: module.bundle.hotData,
    _acceptCallbacks: [],
    _disposeCallbacks: [],
    accept: function (fn) {
      this._acceptCallbacks.push(fn || function () {});
    },
    dispose: function (fn) {
      this._disposeCallbacks.push(fn);
    }
  };
  module.bundle.hotData = null;
}

module.bundle.Module = Module;
var checkedAssets, assetsToAccept;
var parent = module.bundle.parent;

if ((!parent || !parent.isParcelRequire) && typeof WebSocket !== 'undefined') {
  var hostname = "" || location.hostname;
  var protocol = location.protocol === 'https:' ? 'wss' : 'ws';
  var ws = new WebSocket(protocol + '://' + hostname + ':' + "62580" + '/');

  ws.onmessage = function (event) {
    checkedAssets = {};
    assetsToAccept = [];
  };
}

```



```

var data = JSON.parse(event.data);

if (data.type === 'update') {
  var handled = false;
  data.assets.forEach(function (asset) {
    if (!asset.isNew) {
      var didAccept = hmrAcceptCheck(global.parcelRequire, asset.id);

      if (didAccept) {
        handled = true;
      }
    }
  }); // Enable HMR for CSS by default.

  handled = handled || data.assets.every(function (asset) {
    return asset.type === 'css' && asset.generated.js;
  });

  if (handled) {
    console.clear();
    data.assets.forEach(function (asset) {
      hmrApply(global.parcelRequire, asset);
    });
    assetsToAccept.forEach(function (v) {
      hmrAcceptRun(v[0], v[1]);
    });
  } else if (location.reload) {
    // `location` global exists in a web worker context but lacks `.reload()` function.

```

```

    location.reload();
  }
}

if (data.type === 'reload') {
  ws.close();

  ws.onclose = function () {
    location.reload();
  };
}

if (data.type === 'error-resolved') {
  console.log('[parcel] ✨ Error resolved');
  removeErrorOverlay();
}

if (data.type === 'error') {
  console.error('[parcel] 🚫 ' + data.error.message + '\n' + data.error.stack);
  removeErrorOverlay();
  var overlay = createErrorOverlay(data);
  document.body.appendChild(overlay);
}
};
}

function removeErrorOverlay() {
  var overlay = document.getElementById(OVERLAY_ID);

```

```


if (overlay) {
  overlay.remove();
}
}

```

```

function createErrorOverlay(data) {
  var overlay = document.createElement('div');
  overlay.id = OVERLAY_ID; // html encode message and stack trace

  var message = document.createElement('div');
  var stackTrace = document.createElement('pre');
  message.innerText = data.error.message;
  stackTrace.innerText = data.error.stack;

  overlay.innerHTML = '<div style="background: black; font-size: 16px; color: white; position: fixed; height: 100%; width: 100%; top: 0px; left: 0px; padding: 30px; opacity: 0.85; font-family: Menlo, Consolas, monospace; z-index: 9999;">' + '<span style="background: red; padding: 2px 4px; border-radius: 2px;">ERROR</span>' + '<span style="top: 2px; margin-left: 5px; position: relative;">

```

```

function getParents(bundle, id) {
  var modules = bundle.modules;

  if (!modules) {
    return [];
  }

```

```

}

var parents = [];
var k, d, dep;

for (k in modules) {
  for (d in modules[k][1]) {
    dep = modules[k][1][d];

    if (dep === id || Array.isArray(dep) && dep[dep.length - 1] === id) {
      parents.push(k);
    }
  }
}

if (bundle.parent) {
  parents = parents.concat(getParents(bundle.parent, id));
}

return parents;
}

function hmrApply(bundle, asset) {
  var modules = bundle.modules;

  if (!modules) {
    return;
  }

```

```
if (modules[asset.id] || !bundle.parent) {  
  var fn = new Function('require', 'module', 'exports', asset.generated.js);  
  asset.isNew = !modules[asset.id];  
  modules[asset.id] = [fn, asset.deps];  
} else if (bundle.parent) {  
  hmrApply(bundle.parent, asset);  
}  
}
```

```
function hmrAcceptCheck(bundle, id) {  
  var modules = bundle.modules;  
  
  if (!modules) {  
    return;  
  }  
  
  if (!modules[id] && bundle.parent) {  
    return hmrAcceptCheck(bundle.parent, id);  
  }  
  
  if (checkedAssets[id]) {  
    return;  
  }  
  
  checkedAssets[id] = true;  
  var cached = bundle.cache[id];  
  assetsToAccept.push([bundle, id]);
```

```

if (cached && cached.hot && cached.hot._acceptCallbacks.length) {
  return true;
}

return getParents(global.parcelRequire, id).some(function (id) {
  return hmrAcceptCheck(global.parcelRequire, id);
});
}

function hmrAcceptRun(bundle, id) {
  var cached = bundle.cache[id];
  bundle.hotData = {};

  if (cached) {
    cached.hot.data = bundle.hotData;
  }

  if (cached && cached.hot && cached.hot._disposeCallbacks.length) {
    cached.hot._disposeCallbacks.forEach(function (cb) {
      cb(bundle.hotData);
    });
  }

  delete bundle.cache[id];
  bundle(id);
  cached = bundle.cache[id];
}

```

```
if (cached && cached.hot && cached.hot._acceptCallbacks.length) {  
  cached.hot._acceptCallbacks.forEach(function (cb) {  
    cb();  
  });  
  
  return true;  
}  
  
}  
},{}],{},{["../node_modules/parcel-bundler/src/builtins/hmr-runtime.js"], null)  
//# sourceMappingURL=/index.js.map
```