



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Αγρονόμων και Τοπογράφων Μηχανικών

ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΔΙΑΘΕΣΗΣ
ΓΕΩΧΩΡΙΚΩΝ ΔΕΔΟΜΕΝΩΝ CROWD-SOURCING, ΜΕ
ΤΕΧΝΟΛΟΓΙΕΣ ΑΝΟΙΚΤΩΝ ΔΕΔΟΜΕΝΩΝ.

DEVELOPMENT OF A WEB-BASED SYSTEM FOR
SUPPLYING CROWD-SOURCED GEOSPATIAL DATA,
USING OPEN DATA TECHNOLOGIES.

Αγγελόπουλος Παύλος ΑΜ:06108621

Καταβενάκη Μαριέττα ΑΜ:06108622

Ιούλιος 2014

Περίληψη

Η παρούσα εργασία αφορά το σχεδιασμό και την ανάπτυξη μιας διαδικτυακής εφαρμογής η οποία προσθέτει λειτουργικότητα σε μια χωρική βάση δεδομένων η οποία δημιουργήθηκε για αυτό το σκοπό. Όλα τα λογισμικά που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας είναι Ελεύθερα Λογισμικά / Λογισμικά Ανοιχτού Κώδικα.

Οι χρήστες έχουν τη δυνατότητα να ανεβάσουν ή να κατεβάσουν χωρικά δεδομένα ή να τα δουν σε χάρτες. Μπορούν να επιλέξουν ως χάρτη υποβάθρου ανάμεσα σε Open Street Map, Google Map ή Google Satellite.

Πιο συγκεκριμένα, σχεδιάστηκε και υλοποιήθηκε μία βάση δεδομένων για να αποθηκεύει συνολα δεδομένων με τις θέσεις σημείων ενδιαφέροντος, που συλλέγονται με τη μέθοδο του crowd-sourcing. Η βάση κατασκευάστηκε με το PostgreSQL, ένα Σύστημα διαχείρισης Βάσεων Δεδομένων που προσθέτει υποστήριξη για γεωγραφικά αντικείμενα (επέκταση PostGIS) σε αντικείμενο-σχεσιακή βάση δεδομένων.

Στη συνέχεια, χρησιμοποιήθηκε ο Geoserver, μέσω του οποίου οπτικοποιούνται και διαμοιράζονται τα δεδομένα μέσω των υποστηριζόμενων υπηρεσιών Web Map Service, Web Feature Service.

Η διεπαφή της εφαρμογής με το χρήστη, έχει κατασκευαστεί με HTML5, CSS και JavaScript. Επίσης, με JavaScript γίνονται τα αιτήματα στο Geoserver και δημιουργείται ο δυναμικός χάρτης με τους σταθμούς.

Η διάθεση και η λήψη των δεδομένων έγινε με php σε ξεχωριστές ιστοσελίδες

Abstract

This thesis concerns the design and development of a web application that adds functionality to a spatial database which was created for this purpose. All software used for the development of this thesis is free software / open source software.

Users can upload or download spatial data or they can see them on a map. They can choose for basemap between Open Street Map, Google Map or Google Satellite.

More specifically, a spatial database was designed and implemented for storing datasets containing the position of points of interest that were collected via crowd-sourcing technique. Database has been created with PostgreSQL, a database management system that adds support for geographic objects (PostGIS extension) to an object-relational database.

Geoserver was used for data sharing and map creation through supported services, Web Map Service and Web Feature Service

User Interface was created with HTML5, CSS and JavaScript code. Also, JavaScript is used for Geoserver requests and dynamic map creation.

Data Upload and Download has been developed with php code in separate web pages

ΠΕΡΙΕΧΟΜΕΝΑ

1. Εισαγωγή	8
1.1 Αντικείμενο	8
1.2 Στόχοι	8
1.3 Ευχαριστίες	8
2. Γενικά	9
2.1 Σχετικά με τα Ελεύθερα / Ανοιχτά Δεδομένα	9
2.2 Σχετικά με το Ελεύθερο Λογισμικό / Λογισμικό Ανοιχτού Κώδικα	12
2.3 Σχετικά με το Crowdsourcing (Πληθοπορισμό)	15
3. Η αρχιτεκτονική της εφαρμογής	17
3.1 Λογισμικά – τεχνολογίες που χρησιμοποιήθηκαν	17
3.1.1 PostgreSQL	17
3.1.2 PostGIS	19
3.1.3 GeoServer	20
3.1.4 HTML5 – JavaScript – AJAX	21
3.1.5 PHP	25
3.2 Υλοποίηση της εφαρμογής	26
3.2.1 Γενικά	26
3.2.2 Σχεδιασμός της Βάσης Δεδομένων	27

3.2.3 Διασύνδεση με τον GeoServer	33
3.2.4 Σχεδιασμός της διεπαφής χρήστη (User Interface)	41
3.2.5 Ενημέρωση της Βάσης Δεδομένων	47
3.2.6 Λήψη αρχείων δεδομένων	52
3.3 Διαγράμματα UML	52
3.4 Σύνοψη	55
4. Παρουσίαση της εφαρμογής	56
4.1 Παρουσίαση της αρχικής σελίδας	56
4.2 Παρουσίαση της σελίδας για upload δεδομένων	61
4.3 Παρουσίαση της σελίδας για download δεδομένων	63
5. Συμπεράσματα	67
6. Βιβλιογραφία – πηγές	68
7. ΠΑΡΑΡΤΗΜΑ	71
7.1 Ο κώδικας του αρχείου OpenDataProject.html	71
7.2 Ο κώδικας του αρχείου OpenDataProject.js	76
7.3 Ο κώδικας του αρχείου OpenDataProject.css	83
7.4 Ο κώδικας του αρχείου OpenDataUploadForm.php	87
7.5 Ο κώδικας του αρχείου OpenDataUploadForm.css	91
7.6 Ο κώδικας του αρχείου FileReader.js	91
7.7 Ο κώδικας του αρχείου OpenDataDBInserts.php	93
7.8 Ο κώδικας του αρχείου OpenDataDownloadForm.php	99
7.9 Ο κώδικας του αρχείου OpenDataDownloadForm.js	101
7.10 Ο κώδικας του αρχείου FileDownload.php	102

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1. Διαχρονικά η υποστήριξη HTML5 από τους κυριότερους browser	22
Εικόνα 2 . Διαφορά στην αρχιτεκτονική κλασσικού – AJAX μοντέλου σε Web εφαρμογές	24
Εικόνα 3. Διαφορά στην ροή κλασσικού(synchronous) – AJAX (asynchronous) μοντέλου σε Web εφαρμογές	24
Εικόνα 4. Αρχιτεκτονική της εφαρμογής	26
Εικόνα 5. Δομή των αρχείων της εφαρμογής	27
Εικόνα 6. Διάγραμμα της Βάσης Δεδομένων	31
Εικόνα 7. GeoServer – Αρχική σελίδα	34
Εικόνα 8. GeoServer – Δημιουργία νέου Workspace	34
Εικόνα 9. GeoServer – Δημιουργία νέου Workspace	35
Εικόνα 10. GeoServer – Ρυθμίσεις παραμέτρων του νέου Workspace	35
Εικόνα 11. GeoServer – Δημιουργία νέου Data Store	36
Εικόνα 12. GeoServer – Επιλογή πηγής δεδομένων	36
Εικόνα 13. GeoServer – Ρυθμίσεις παραμέτρων σύνδεσης στη βάση δεδομένων	37
Εικόνα 14. Add new layer	38
Εικόνα 15. Add new layer	38
Εικόνα 16. Ρύθμιση παραμέτρων της καρτέλας Data για το layer poisTable	39
Εικόνα 17. Ρύθμιση παραμέτρων της καρτέλας Publishing για το layer poisTable	40
Εικόνα 18. Menu bar του site Open Knowledge Foundation του παραρτήματος Ελλάδος	43
Εικόνα 19. Διάγραμμα χρήσης της εφαρμογής	53
Εικόνα 20. Διάγραμμα ροής της εφαρμογής	54

Εικόνα 21. Αρχική σελίδα της εφαρμογής	56
Εικόνα 22. Αρχική σελίδα της εφαρμογής, επιλογή χάρτη υποβάθρου OSM	57
Εικόνα 23. Αρχική σελίδα της εφαρμογής, επιλογή χάρτη υποβάθρου GoogleMaps	57
Εικόνα 24. Αρχική σελίδα της εφαρμογής, επιλογή χάρτη υποβάθρου GoogleSatellite	58
Εικόνα 25. Προβολή των dataset στο χάρτη	58
Εικόνα 26. Εμφάνιση / απόκρυψη layer με τη χρήση του OpenLayers widget	59
Εικόνα 27. Ανάκτηση πληροφοριών για τα σημεία ενδιαφέροντος	60
Εικόνα 28. Ανάκτηση πληροφοριών για σημεία ενδιαφέροντος από πολλά datasets	60
Εικόνα 29. Ανακατεύθυνση στη σελίδα για upload συνόλου δεδομένων	61
Εικόνα 30. Σελίδα για upload δεδομένων	62
Εικόνα 31. Επιλογή κατηγορίας για το dataset	62
Εικόνα 32. Δημιουργία νέας κατηγορίας για το dataset	62
Εικόνα 33. Υποχρεωτικά στοιχεία για τον Uploader	63
Εικόνα 34. Προαιρετικά στοιχεία για τον Uploader	63
Εικόνα 35. Υποχρεωτική συμπλήρωση για το πεδίο όνομα dataset	64
Εικόνα 36. Υποχρεωτική υπόδειξη αρχείου pois	64
Εικόνα 37. Προαιρετικά στοιχεία για το dataset	65
Εικόνα 38. Υπόδειγμα αρχείου pois	65
Εικόνα 39. Σελίδα για download δεδομένων	66

1. Εισαγωγή

1.1 Αντικείμενο

Η διπλωματική αυτή εργασία εκπονήθηκε στα πλαίσια του προγράμματος σπουδών της Σχολής Αγρονόμων και Τοπογράφων Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου και εμπίπτει στα γνωστικά αντικείμενα των μαθημάτων των προγραμματιστικών τεχνικών, της γεωπληροφορικής, των βάσεων δεδομένων, καθώς και της χαρτογραφίας. Αφορά στο σχεδιασμό και κατασκευή μιας διαδικτυακής εφαρμογής η οποία προσθέτει λειτουργικότητα σε μια διαδικτυακή χωρική Βάση Δεδομένων.

Πιο συγκεκριμένα, οι χρήστες της εφαρμογής έχουν τη δυνατότητα να ανεβάσουν (upload), να κατεβάσουν (download) ή να δουν σε χάρτες διάφορα σημεία ενδιαφέροντος, κατάλληλα ταξινομημένα ανά σύνολο δεδομένων (dataset). Με αυτό το τρόπο, μπορούν να μοιραστούν με άλλους γεωγραφικά δεδομένα καθώς και να επωφεληθούν οι ίδιοι από δεδομένα άλλων. Η βάση δεδομένων εμπλουτίζεται βασιζόμενη στην αρχή του πληθοπορισμού (Crowdsourcing).

Ο χρήστης της εφαρμογής με την επίσκεψη του στην ιστοσελίδα, βλέπει σε ένα τμήμα τα διαθέσιμα σύνολα δεδομένων (datasets) από τα οποία μπορεί να επιλέξει. Σε ένα δεύτερο τμήμα υπάρχει χάρτης, πάνω στον οποίο εμφανίζονται τα σημεία. Η προεπιλογή είναι χάρτης Open Street Map (OSM), αλλά μπορεί να επιλέξει ως υπόβαθρο είτε Google Map ή Google Satellite. Σε ένα τρίτο τμήμα μπορεί να δει τις πληροφορίες που περιέχονται στη βάση για κάθε σημείο ενδιαφέροντος, κάνοντας κλικ. Οι παραπάνω υπηρεσίες υλοποιήθηκαν σύμφωνα με τα standard Web Map Service (WMS) και Web Feature Service (WFS).

1.2 Στόχοι

Στόχος της συγκεκριμένης εφαρμογής, αποτελεί η εύκολη πρόσβαση μέσω του παγκόσμιου ιστού (Web) σε δημόσια – ανοιχτή πληροφορία, ώστε να είναι δυνατή η αξιοποίηση της από κάθε ενδιαφερόμενο. Για τη δημιουργία της χρησιμοποιήθηκαν δεδομένα τα οποία διατίθενται ελεύθερα από την ιστοσελίδα poi Index – Database (<http://index.pois.gr/>).

Επίσης, σε όλα τα στάδια της υλοποίησης χρησιμοποιήθηκαν ελεύθερα λογισμικά, τόσο αυτά που αποτελούν συστατικά μέρη της εφαρμογής, όσο και τα εργαλεία ανάπτυξης.

1.3 Ευχαριστίες

Νιώθουμε την ανάγκη να ευχαριστήσουμε όλους τους καθηγητές και τους συμφοιτητές μας στη Σχολή Αγρονόμων και Τοπογράφων Μηχανικών για την συμπαράσταση και τη βοήθεια τους, σε όλο τον κύκλο σπουδών.

Ιδιαίτερη μνεία πρέπει να γίνει για τον καθηγητή κ. Βασίλειο Βεσκούκη, αφού χωρίς την υποστήριξη και βοήθεια του η εργασία αυτή δεν θα είχε ποτέ ολοκληρωθεί. Η συμβολή του υπήρξε καθοριστική σε όλα τα στάδια εκπόνησης της, ενώ ήταν πάντα διαθέσιμος να προσφέρει τη βοήθεια του, όποτε ζητήθηκε, ακόμα και σε ημέρες και ώρες εκτός των επαγγελματικών του υποχρεώσεων.

Τέλος, είναι αυτονόητο πως για οποιαδήποτε παράλειψη ή λάθος της παρούσας εργασίας η ευθύνη βαρύνει αποκλειστικά και μόνο εμάς.

2. Γενικά

2.1 Σχετικά με τα Ελεύθερα / Ανοιχτά Δεδομένα

“Ανοιχτά Δεδομένα είναι τα Δεδομένα που μπορούν να χρησιμοποιηθούν ελεύθερα, να διαμοιραστούν και να επεξεργαστούν από οποιονδήποτε, οπουδήποτε, για οποιονδήποτε σκοπό.” [Open Knowledge Foundation, Παράρτημα Ελλάδος]

Η σημασία καθώς και η αξία των δεδομένων έγκειται στην αξιοποίηση τους από τους ενδιαφερόμενους. Καθιστώντας ένα σύνολο δεδομένων ανοιχτό δηλαδή ελεύθερο προς αξιοποίηση προσδίδεται σε αυτό προστιθέμενη αξία. Έτσι, το ίδιο σύνολο δεδομένων είναι δυνατόν να αξιοποιηθεί με πολλούς και διαφορετικούς τρόπους, από πολλούς ανθρώπους ή μηχανές.

Ανοιχτά μπορούν να είναι όλα τα δεδομένα από οποιαδήποτε πηγή και για οποιοδήποτε θέμα. Ο καθένας μπορεί να δημοσιεύσει τις πληροφορίες που θέλει με μια Ανοιχτή Άδεια για ελεύθερη χρήση για το κοινό καλό. Μέχρι τώρα, ελεύθερα δεδομένα δίνουν οι κυβερνήσεις και οι κοινωνικοί φορείς, όπως προϋπολογισμούς και χάρτες ή οι ερευνητές που μοιράζονται τα αποτελέσματα των ερευνών τους και τις δημοσιεύσεις τους. Παρόλα αυτά σιγά σιγά όλοι οι οργανισμοί απελευθερώνουν τα δεδομένα τους όπως εταιρίες, πανεπιστήμια, Μη Κυβερνητικές Οργανώσεις, startups, κοινωφελείς οργανισμοί, κοινότητες και ιδιώτες.

Πλέον υπάρχουν ανοιχτά δεδομένα σε διάφορους τομείς όπως μετακίνηση, επιστήμη, προϊόντα, εκπαίδευση, χάρτες, νομοθεσία, βιβλιοθήκες, οικονομικά, ανάπτυξη, σχέδιο, επιχειρηματικότητα, επενδύσεις κτλ. Ανοιχτά μπορούν να είναι δεδομένα, αρχεία αλλά και εικόνες, βιβλία, κείμενα, ταινίες και μουσική. Γίνεται αντιληπτό ότι στόχος του Ανοιχτού Κινήματος είναι ο ελεύθερος διαμοιρασμός πληροφοριών και υλικού προς όφελος της κοινωνίας για ανάπτυξη, ελεύθερη διακίνηση γνώσης, καινοτομίας. Ο πλήρης ορισμός της Ανοιχτότητας (Open Definition) έχει δύο κύρια χαρακτηριστικά:

Νομική Υπόσταση (Legal Openness): Τα Ανοιχτά Δεδομένα θα πρέπει να διατίθενται νόμιμα, με δικαίωμα περαιτέρω επεξεργασίας και διανομής. Αυτό επιτυγχάνεται με χρήση μιας Ανοιχτής Άδειας, για παράδειγμα Creative Commons, η οποία επιτρέπει ελεύθερη πρόσβαση στο υλικό ή με το να διατίθενται το περιεχόμενο ελεύθερα στον Δημόσιο Τομέα (Public Domain).

Τεχνολογική Υπόσταση (Technical Openness): Δεν πρέπει να υπάρχουν τεχνολογικά εμπόδια στην χρήση των δεδομένων. Αντίθετα, τα Ανοιχτά Δεδομένα θα πρέπει να διανέμονται σε μορφή που να επιτρέπει την επεξεργασία τους από υπολογιστή (Bulk)

Κύρια χαρακτηριστικά των Ανοιχτών Δεδομένων είναι η ελεύθερη χρήση τους για επεξεργασία και διανομή από οποιονδήποτε, ακόμα και η ελεύθερη εμπορική χρήση. Τα Ανοιχτά Δεδομένα πρέπει να διατίθενται σε όγκο (bulk) και ποσότητα και να είναι είτε δωρεάν είτε σε λογική τιμή για επαναχρησιμοποίηση. Λογική τιμή θεωρείται η τιμή κόστους αντιγραφής των δεδομένων. Οι πληροφορίες πρέπει να είναι σε ψηφιακή μορφή, κατά προτίμηση διαθέσιμες για κατέβασμα μέσω του Διαδικτύου και εύκολα επεξεργάσιμες από υπολογιστή αλλιώς οι χρήστες δεν μπορούν να επωφεληθούν πλήρως από την ανοιχτότητα ώστε να δημιουργήσουν κάτι καινούριο. Τα δικαιώματα του χρήστη πάνω στα Ανοιχτά Δεδομένα πρέπει να είναι η χρήση, η επαναχρησιμοποίηση, η διανομή και η επεξεργασία τους μαζί με άλλα δεδομένα ώστε να είναι εύκολη και η αναδιανομή και των ερευνητικών αποτελεσμάτων.

Ο ορισμός της Ανοιχτότητας δεν επιτρέπει περιορισμούς στην χρήση των Ανοιχτών Δεδομένων αλλά ενθαρρύνει την Αναφορά στο Δημιουργό ή στην Πηγή και στον τρόπο επεξεργασίας όπως φυσικά και την Παρόμοια Διανομή δηλαδή την διανομή του καινούριου υλικού με παρόμοια Ανοιχτή Άδεια. Η αναφορά στο Δημιουργό είναι ο μόνος και κύριος κανόνας που διέπει το Ανοιχτό Κίνημα καθώς και η διάθεση του περιεχομένου πάντα Ανοιχτά.

Τρεις είναι οι κύριες αρχές της Ανοιχτότητας και οι λόγοι για τους οποίους άλλωστε είναι τόσο σημαντικό^[1]:

1. Ανοιχτή και Ελεύθερη Διαθεσιμότητα και Πρόσβαση

2. Επαναχρησιμοποίηση και Αναδιανομή

3. Παγκόσμια Συμμετοχή χωρίς όρια

Ο ορισμός της Ανοιχτότητας δημιουργήθηκε το 2005 από το Ίδρυμα Ανοιχτής Γνώσης με τη συμμετοχή πολλών ανθρώπων. Ο ορισμός βασίστηκε στον Ορισμό Ανοιχτού Λογισμικού από το Open Source Initiative που επαναχρησιμοποιήθηκε βάση των αρχών και των πρακτικών της Κοινότητας Ανοιχτού Λογισμικού που έχουν αναπτυχθεί τα τελευταία χρόνια για το Λογισμικό και εφαρμόζονται πλέον στα δεδομένα και στο περιεχόμενο.

Πιο αναλυτικά, σύμφωνα με τον ορισμό της Ανοιχτής Γνώσης όπως εξειδικεύεται σχετικά με τα δεδομένα και παρουσιάζεται στην σελίδα του Open Knowledge Foundation ο ορισμός της Ανοιχτότητας προϋποθέτει τα παρακάτω^[2]:

1. Πρόσβαση. Το έργο (ο όρος έργο θα χρησιμοποιείται εδώ για να δηλώνει το επίμαχο στοιχείο γνώσεως – στη συγκεκριμένη περίπτωση τα δεδομένα) θα είναι διαθέσιμο στο σύνολό του και σε κόστος όχι μεγαλύτερο από ένα εύλογο κόστος αντιγραφής, κατά προτίμηση για δωρεάν λήψη μέσω του διαδικτύου. Το έργο επίσης θα πρέπει να είναι διαθέσιμο σε μορφή κατάλληλη και επιδεκτική τροποποιήσεων.

2. Αναδιανομή. Η άδεια (ο όρος αναφέρεται στη νόμιμη άδεια υπό την οποία διατίθενται τα δεδομένα και στην περίπτωση κατά την οποία δεν υφίσταται άδεια, αναφέρεται συνακόλουθα στο προκαθορισμένο νομικό πλαίσιο εντός του οποίου διατίθενται τα δεδομένα), δεν θα θέτει περιορισμούς για την πώληση ή δωρεάν διανομή των δεδομένων, είτε αφ' εαυτού είτε ως μέρος συνόλου αποτελούμενης από δεδομένα προερχόμενα από διαφορετικές πηγές. Η άδεια δεν θα απαιτεί καταβολή πνευματικών δικαιωμάτων ή άλλου τέλους για τέτοιου τύπου πώληση ή διανομή.

3. Επαναχρησιμοποίηση. Η άδεια πρέπει να προβλέπει τη δυνατότητα τροποποιήσεων καθώς και τη δημιουργία δευτερογενών δεδομένων και να επιτρέπει αυτά να διανέμονται υπό τους ίδιους όρους με τα πρωτότυπα. Η άδεια μπορεί να επιβάλλει κάποια μορφή προϋποθέσεων απόδοσης και ακεραιότητας. Βλ. παρακάτω αρχές 5 (Απόδοση) και 6 (Ακεραιότητα).

4. Απουσία τεχνολογικών περιορισμών. Τα δεδομένα πρέπει να παρέχονται σε τέτοια μορφή ώστε να μην τίθενται τεχνολογικά εμπόδια στην πραγματοποίηση των παραπάνω δραστηριοτήτων. Αυτό μπορεί να επιτευχθεί με την προσφορά τους σε ανοιχτή μορφή δεδομένων, δηλαδή σε κάποια της οποίας οι προδιαγραφές είναι δημόσια και ελεύθερα διαθέσιμες και δεν υπάρχουν χρηματικοί ή άλλοι περιορισμοί για τη χρήση της.

5. Απόδοση. Η άδεια μπορεί να προϋποθέτει ως όρο για αναδιανομή και επαναχρησιμοποίηση την απόδοση στους συντελεστές και δημιουργούς του έργου. Εάν ο όρος επιβάλλεται, δεν μπορεί να είναι επαχθής. Παραδείγματος χάριν, εάν απαιτείται απόδοση, θα πρέπει να συνοδεύεται το έργο από έναν κατάλογο όλων όσων χρήζουν αποδόσεως.

6. Ακεραιότητα. Η άδεια μπορεί να προϋποθέτει ως όρο για αναδιανομή του έργου σε τροποποιημένη μορφή το τελικό έργο να έχει διαφορετικό τίτλο ή διαφορετικό αριθμό έκδοσης από το πρωτότυπο έργο.

7. Καμιά διάκριση εις βάρος πρόσωπων ή ομάδων. Η άδεια δεν πρέπει να μεροληπτεί εναντίον προσώπου ή ομάδας προσώπων.

8. Καμιά διάκριση εις βάρος πεδίων δραστηριότητας. Η άδεια δεν πρέπει να περιορίζει χρήση του έργου σε συγκεκριμένο πεδίο δραστηριότητας. Παραδείγματος χάριν, δεν μπορεί να εμποδίζει τη χρήση του έργου στο εμπόριο ή για στρατιωτική έρευνα.

9. Διανομή άδειας. Τα δικαιώματα που συνδέονται με το έργο πρέπει να ισχύουν για όλους αυτούς στους οποίους αναδιανέμεται το έργο χωρίς ανάγκη εκτελέσεως περαιτέρω άδειας.

10. Η άδεια δεν πρέπει να περιορίζεται σε ένα μόνο σύνολο δεδομένων. Τα δικαιώματα που συνδέονται με το έργο πρέπει να μην εξαρτώνται από το εάν το έργο αποτελεί μέρος συγκεκριμένου συνόλου. Εάν το έργο αποσπάται από αυτό και χρησιμοποιείται ή διανέμεται εντός των όρων της άδειάς του, όλοι αυτοί στους οποίους το έργο αναδιανέμεται θα έπρεπε να έχουν τα ίδια δικαιώματα με εκείνα που παραχωρούνται σε σύνδεση με το πρωτότυπο σύνολο δεδομένων.

11. Η άδεια δεν πρέπει να περιορίζει τη διανομή άλλων έργων. Η άδεια δεν πρέπει να θέτει περιορισμούς σε άλλα έργα που διανέμονται μαζί με το εξουσιοδοτημένο έργο. Παραδείγματος χάριν, η άδεια δεν μπορεί να απαιτεί να είναι ανοιχτά και όλα τα υπόλοιπα έργα που διανέμονται με το ίδιο μέσο.

Ο Tim Berners-Lee, ο εφευρέτης του Παγκόσμιου Ιστού και των Συνδεδεμένων Δεδομένων, πρότεινε ένα σχήμα 5 αστέρων για τα Ανοικτά Δεδομένα. Προβλέπονται συγκεκριμένα βήματα για την απόκτηση κάθε αστεριού, έτσι ώστε αυτά να μπορούν να συνδέονται μεταξύ τους και να γίνουν πιο χρήσιμα. Σύμφωνα με το σχήμα αυτό, το οποίο βρίσκει εφαρμογή στο Semantic Web, τα βήματα για την απόκτηση κάθε αστεριού είναι^[3] :

* διάθεση των δεδομένων στον Ιστό (με οποιαδήποτε μορφή) κάτω από μια ανοικτή άδεια.

** το παραπάνω και επιπλέον τη διάθεση των δεδομένων ως δομημένα δεδομένα (π.χ., σε μορφή Excel αντί για σαρωμένη εικόνα ενός πίνακα).

*** τα παραπάνω και επιπλέον χρήση μη ιδιόκτητων μορφών (π.χ., CSV αντί για Excel).

**** τα παραπάνω και επιπλέον χρήση URI για την ταυτοποίηση των δεδομένων ώστε οι χρήστες να μπορούν να δείχνουν τα δεδομένα.

***** τα παραπάνω και επιπλέον σύνδεση των δεδομένων σε άλλα δεδομένα ώστε να παρέχεται το σημασιολογικό τους πλαίσιο.

2.2 Σχετικά με το Ελεύθερο Λογισμικό / Λογισμικό Ανοιχτού Κώδικα

Το ελεύθερο λογισμικό, όπως ορίζεται από το Ίδρυμα Ελευθέρου Λογισμικού (Free Software Foundation), είναι λογισμικό που μπορεί να χρησιμοποιηθεί, αντιγραφεί, μελετηθεί, τροποποιηθεί και αναδιανεμηθεί χωρίς περιορισμό. Η ελευθερία από τέτοιους περιορισμούς είναι βασικό στοιχείο στην ιδέα του "ελεύθερου λογισμικού", έτσι ώστε το αντίθετο του ελεύθερου λογισμικού είναι το ιδιόκτητο λογισμικό, και όχι το λογισμικό που πωλείται για κέρδος, όπως το εμπορικό λογισμικό^[4].

Το Ελεύθερο λογισμικό παρέχει στους χρήστες την ελευθερία να εκτελούν, αντιγράψουν, διανέμουν, μελετούν, τροποποιούν και βελτιώνουν το Ελεύθερο λογισμικό. Για την ακρίβεια, αναφέρεται σε τέσσερις βασικές ελευθερίες^[5]:

Ελευθερία 0. Την ελευθερία κάθε χρήστη να εκτελεί το πρόγραμμα για οποιονδήποτε σκοπό.

Ελευθερία 1. Την ελευθερία κάθε χρήστη να έχει τη δυνατότητα να μελετά τον τρόπο λειτουργίας του προγράμματος και να το προσαρμόζει στις ανάγκες του. Η πρόσβαση στον πηγαίο κώδικα είναι προϋπόθεση για να ισχύει κάτι τέτοιο.

Ελευθερία 2. Την ελευθερία κάθε χρήστη να έχει τη δυνατότητα να αναδιανέμει αντίγραφα του προγράμματος ώστε να βοηθά άλλους χρήστες.

Ελευθερία 3. Την ελευθερία κάθε χρήστη να έχει τη δυνατότητα να βελτιώνει το πρόγραμμα και να δημοσιεύει τις βελτιώσεις που έχει κάνει στο ευρύ κοινό, ώστε να

επωφεληθεί ολόκληρη η κοινότητα. Η πρόσβαση στον πηγαίο κώδικα είναι προϋπόθεση για να ισχύει κάτι τέτοιο.

Ένα πρόγραμμα θεωρείται ελεύθερο λογισμικό όταν οι χρήστες του έχουν όλες τις παραπάνω ελευθερίες. Επομένως, θα πρέπει να είναι ελεύθεροι να αναδιανέμουν αντίγραφα, με ή χωρίς τροποποιήσεις, δωρεάν ή χρεώνοντας για την διανομή, στον οποιονδήποτε και οπουδήποτε. Το να είναι ελεύθεροι να κάνουν όλα τα παραπάνω σημαίνει, μεταξύ άλλων, πως δεν χρειάζεται να ζητήσουν εξουσιοδότηση ή να πληρώσουν κάποιον ώστε να λάβουν την ανάλογη άδεια. Θα πρέπει επίσης να έχουν την ελευθερία να κάνουν τροποποιήσεις και να τις χρησιμοποιούν, χωρίς να χρειάζεται να το αναφέρουν. Αν δημοσιεύσουν τις αλλαγές τους, δεν είναι υποχρεωμένοι να ειδοποιήσουν κάποιο συγκεκριμένο άτομο.

Η ελευθερία της χρήσης ενός προγράμματος σημαίνει πως δίδεται η ελευθερία σε κάθε άτομο ή επιχείρηση να το χρησιμοποιήσει σε κάθε είδους υπολογιστικό σύστημα, για κάθε είδος εργασίας χωρίς να είναι υποχρεωμένο να επικοινωνήσει εκ των προτέρων με τον προγραμματιστή ή με κάποια άλλη οντότητα. Σε αυτή την ελευθερία, είναι η άποψη του χρήστη που έχει σημασία, και όχι η άποψη του κατασκευαστή. Οι χρήστες είναι ελεύθεροι να εκτελούν το πρόγραμμα για τους δικούς τους λόγους, και αν το διανέμουν σε οποιονδήποτε άλλο, τότε και εκείνος είναι ελεύθερος να το εκτελεί για τους δικούς του λόγους, δίχως να έχει ο πάροχος το δικαίωμα να του υποβάλει δικές του κυρώσεις.

Η ελευθερία της αναδιανομής αντιγράφων θα πρέπει να περιλαμβάνει εκτελέσιμες (executable) μορφές του προγράμματος, καθώς και τον πηγαίο κώδικα, τόσο για την τροποποιημένη όσο και για την αρχική έκδοση του προγράμματος. (Η διανομή προγραμμάτων σε εκτελέσιμη μορφή είναι απαραίτητη για εγκατεστημένα ελεύθερα λειτουργικά συστήματα). Θεωρείται δεκτό όταν δεν υπάρχει τρόπος να δημιουργηθεί η εκτελέσιμη μορφή για κάποιο συγκεκριμένο πρόγραμμα (από τη στιγμή που μερικές γλώσσες δεν υποστηρίζουν κάτι τέτοιο), αλλά θα πρέπει να υπάρχει η ελευθερία αναδιανομής τέτοιων μορφών σε περίπτωση που ο χρήστης την αναπτύξει ή βρει κάποιο νέο τρόπο να τις δημιουργήσει.

Για να ισχύουν πρακτικά οι τέσσερις βασικές ελευθερίες, και να μπορούν να δημοσιεύθούν βελτιωμένες εκδόσεις, θα πρέπει να υπάρχει πρόσβαση στον πηγαίο κώδικα του προγράμματος. Επομένως, η πρόσβαση στον πηγαίο κώδικα είναι απαραίτητη προϋπόθεση στο ελεύθερο λογισμικό. Μία σημαντική μέθοδος τροποποίησης ενός προγράμματος είναι η συγχώνευση του με διαθέσιμες ελεύθερες υπορουτίνες και υποπρογράμματα (modules). Εάν η άδεια του προγράμματος αναφέρει ότι δεν διατίθεται το δικαίωμα συγχώνευσης με ένα υπάρχον υποπρόγραμμα, για παράδειγμα, εάν απαιτεί ο χρήστης να κατέχει τα πνευματικά δικαιώματα (copyright holder) για οποιονδήποτε κώδικα έχει προσθέσει, τότε η άδεια είναι πολύ περιοριστική ώστε να μπορεί να θεωρηθεί ελεύθερη (δηλαδή άδεια ελεύθερου λογισμικού).

Μπορεί κάποιος να έχει προμηθευτεί αντίγραφα ελεύθερου λογισμικού χωρίς κάποια οικονομική επιβάρυνση ή ακόμα και να έχει πληρώσει για να τα προμηθευτεί. Ανεξαρτήτως όμως από τον τρόπο προμήθειας, θα πρέπει πάντα να έχει την ελευθερία να τροποποιεί το λογισμικό, ακόμα και να πουλάει αντίγραφα του. Ελεύθερο λογισμικό δεν σημαίνει μη-εμπορικό. Ένα ελεύθερο πρόγραμμα θα πρέπει να είναι διαθέσιμο για εμπορική χρήση, εμπορική ανάπτυξη ή εμπορική διανομή. Η εμπορική ανάπτυξη του ελεύθερου λογισμικού

δεν είναι ασυνήθιστη. Αντιθέτως, τέτοιου είδους ελεύθερο λογισμικό είναι πολύ σημαντικό να υπάρχει, καθώς προσφέρει προστιθέμενη αξία σε αυτό.

Κανόνες για την διαδικασία πακεταρίσματος (packaging) μίας τροποποιημένης έκδοσης είναι αποδεκτοί όταν δεν περιορίζουν ουσιαστικά την ελευθερία δημοσίευσης των διαφόρων τροποποιημένων εκδόσεων. Κανόνες όπως για παράδειγμα εάν διατεθεί το πρόγραμμα με τον τάδε τρόπο θα πρέπει να διατεθεί επίσης και με τον δεινά τρόπο είναι αποδεκτοί, με την ίδια προϋπόθεση. Επίσης είναι αποδεκτό σε μία άδεια ελεύθερου λογισμικού να υπάρχουν και άλλοι είδους απαιτήσεις, όπως για παράδειγμα εάν κάποιος προγραμματιστής έχει διαθέσει μία τροποποιημένη έκδοση ενός προγράμματος και κάποιος προηγούμενος κατασκευαστής ζητήσει αντίγραφο του, θα πρέπει να του αποστείλει ένα, ή να δηλώσει τον εαυτό του στις μετατροπές που έκανε.

Συμπερασματικά, όπως αναφέρει ο Richard Stallman, ιδρυτής του μη-κερδοσκοπικού οργανισμού Ιδρύμα Ελεύθερου Λογισμικού (FSF-Free Software Foundation), για το ελεύθερο λογισμικό ισχύουν τα παρακάτω^[6]:

Σχεδόν όλο το λογισμικό ανοιχτού κώδικα είναι ελεύθερο λογισμικό. Οι δύο όροι περιγράφουν σχεδόν την ίδια κατηγορία λογισμικού. Αλλά αντιπροσωπεύουν κάποιες απόψεις βασισμένες σε βασικές διαφορετικές αξίες. Ο ανοιχτός κώδικας είναι μία μεθοδολογία ανάπτυξης. Το ελεύθερο λογισμικό είναι ένα κοινωνικό κίνημα. Για το κίνημα ελεύθερου λογισμικού, το ελεύθερο λογισμικό είναι μία ηθική προσαγωγή, επειδή μόνο το ελεύθερο λογισμικό σέβεται την ελευθερία του χρήστη. Σε αντίθεση, η φιλοσοφία του ανοιχτού κώδικα σε σχέση με θέματα για το πως να κάνει το λογισμικό "καλύτερο"--με καθαρά πρακτικό τρόπο μόνο. Λέγεται πως το μη ελεύθερο λογισμικό είναι μία χειρίστη λύση. Για το κίνημα του ελεύθερου λογισμικού, πάντως, το μη ελεύθερο λογισμικό είναι ένα κοινωνικό πρόβλημα, και η κίνηση του να ελευθερώσουμε το λογισμικό είναι η λύση.

Αντίστοιχα, στην ιστοσελίδα της Ελληνικής Εταιρίας Ελεύθερου Λογισμικού / Λογισμικού Ανοιχτού Κώδικα (ΕΛ/ΛΑΚ), αναφέρονται τα εξής^[7]:

«Ελεύθερο Λογισμικό / Λογισμικό Ανοιχτού Κώδικα (ΕΛ/ΛΑΚ) είναι το λογισμικό που ο καθένας μπορεί ελεύθερα να χρησιμοποιεί, να αντιγράψει, να διανέμει και να τροποποιεί ανάλογα με τις ανάγκες του. Είναι ένα εναλλακτικό μοντέλο ανάπτυξης και χρήσης λογισμικού που βασίζεται στην ελεύθερη διάθεση του πηγαίου κώδικα, το οποίο παρέχει τη δυνατότητα αλλαγών ή βελτιώσεων ώστε να καλύπτονται οι ανάγκες αυτού που το χρησιμοποιεί. Με βάση αυτή τη φιλοσοφία δημιουργήθηκε μια μεγάλη κοινότητα χρηστών και προγραμματιστών, οι οποίοι συνεργάζονται για τη συνεχή βελτίωση του λογισμικού, παρέχοντας γνώσεις και εργασία. Σήμερα λειτουργεί ένα παγκόσμιο ανοικτό δίκτυο προγραμματιστών, οι οποίοι παράλληλα αναπτύσσουν και διορθώνουν τον κώδικα των προγραμμάτων, κυκλοφορώντας ταχύτατα νέες βελτιωμένες εκδόσεις λογισμικού. Με αυτό τον τρόπο συμβάλλουν καθημερινά στην δημιουργία νέων κοινών αγαθών. Το Διαδίκτυο αποτελεί το βασικό τρόπο πρόσβασης στο διαθέσιμο Ελεύθερο Λογισμικό. Η σχέση του Διαδικτύου και ΕΛ/ΛΑΚ, υπήρξε εξ αρχής στενή και αμφίδρομη. Η εξάπλωση του ΕΛ/ΛΑΚ έχει στηριχθεί στην ευρεία χρήση του Διαδικτύου και η διαδικασία ανάπτυξης και λειτουργίας του Διαδικτύου βασίζεται, κατά κύριο λόγο, σε ΕΛ/ΛΑΚ. Τέλος και το Διαδίκτυο νέας γενιάς αναπτύσσεται με ανοιχτό λογισμικό. Το Λογισμικό Ανοιχτού Κώδικα, με συνεχείς βελτιώσεις και αυξημένη πλέον φιλικότητα προς το χρήστη, κερδίζει διαρκώς νέους φίλους παγκοσμίως. Στην εκπαίδευση, στη δημόσια διοίκηση και στις επιχειρήσεις, οι ενδιαφερόμενοι ενημερώνονται και αποκτούν ιδιαίτερα ελκυστικά εργαλεία, αξιόπιστα, σταθερά στη λειτουργία, και απαλλαγμένα από τα σημαντικά κόστη απόκτησης και συνεχούς αναβάθμισης που απαιτούν τα κλειστά λογισμικά. Έτσι πλέον όλο και

πιο πολλοί πόροι διατίθενται στην τεχνική υποστήριξη με σημαντικά οφέλη για την τοπική και εθνική οικονομία».

2.3 Σχετικά με το Crowdsourcing (Πληθοπορισμός)

Το Crowdsourcing ή πληθοπορισμός στα ελληνικά, αποτελεί νέα τάση της σημερινής εποχής, κυρίως λόγω της ευρείας εξάπλωσης της τεχνολογίας στο κοινωνικό σύνολο. Σύμφωνα με τον ορισμό που έδωσαν οι Estellés-Arolas και González Ladrón-de-Guevara, ισχύουν τα εξής^[8]:

«Ο πληθοπορισμός είναι μία μορφή συλλογικής διαδικτυακής δραστηριότητας στην οποία ένα άτομο, ένα ίδρυμα, ένας μη κερδοσκοπικός οργανισμός ή μία εταιρεία προτείνει σε μία ομάδα ατόμων με ποικίλλες γνώσεις, ετερογένεια και αριθμό, μέσω μίας ανοιχτής πρόσκλησης, να αναλάβουν εθελοντικά μια εργασία. Η ανάληψη της εργασίας, η οποία ποικίλλει σε πολυπλοκότητα και στο βαθμό στον οποίο είναι χωρισμένη και στην οποία το πλήθος πρέπει να συμμετάσχει με προσωπική εργασία, χρήματα, γνώση, εμπειρία, περιλαμβάνει πάντοτε αμοιβαίο όφελος και για τις δύο πλευρές. Οι χρήστες λαμβάνουν την ικανοποίηση κάποιας ανάγκης τους, είτε αυτή είναι οικονομική, είτε κοινωνική αναγνώριση, προσωπική ικανοποίηση, ανάπτυξη ατομικών ικανοτήτων σε κάποιο τομέα, ενώ ο εκκινητής της πρωτοβουλίας (πληθοποριστής) αποκτά και χρησιμοποιεί προς όφελός του, αυτά που έχει συνεισφέρει ο χρήστης στο εγχείρημα, τα οποία εξαρτώνται από τη δραστηριότητα που έχει αναλάβει ο χρήστης.»

Ο όρος crowdsourcing προέρχεται από την ένωση των λέξεων crowd (πλήθος) και outsourcing (εξωτερική ανάθεση εργασιών) και χρησιμοποιήθηκε για πρώτη φορά από τον Jeff Howe τον Ιούνιο του 2006 στην έκδοση του Wired με τίτλο "The Rise of Crowdsourcing"^{[9][10]}. Ο ίδιος, εξηγεί πως επειδή η εξέλιξη της τεχνολογίας έχει οδηγήσει σε φθηνότερες καταναλωτικές ηλεκτρονικές συσκευές, η απόσταση μεταξύ επαγγελματιών και ερασιτεχνών έχει ελαχιστοποιηθεί. Οι εταιρείες μπορούν πλέον να αξιοποιήσουν τα ταλέντα του κοινού και ο Howe σημειώνει πως «δεν πρόκειται για outsourcing, είναι crowdsourcing.»

Επίσης, καθόρισε πως η έννοια του crowdsourcing εξαρτάται ουσιαστικά από το γεγονός πως εφόσον πρόκειται για ανοικτή πρόσκληση σε ένα άγνωστο πλήθος ατόμων, συγκεντρώνει αυτούς που είναι οι πιο ικανοί για να αναλάβουν καθήκοντα, να λύσουν πολύπλοκα προβλήματα και να συνεισφέρουν με τις πλέον σχετικές και φρέσκιες ιδέες. Για παράδειγμα, το κοινό μπορεί να προσκληθεί να αναπτύξει μια νέα τεχνολογία, να αναλάβει μια εργασία σχεδίασης (γνωστό και ως συνεργατικός ή δημοκρατικός σχεδιασμός και καταναμεμένος συνεργατικός σχεδιασμός), να εξειδικεύσει ή να υλοποιήσει τα βήματα ενός αλγορίθμου (ανθρωπογενής υπολογιστική), ή να βοηθήσει στην αποτύπωση, συστηματοποίηση και ανάλυση δεδομένων μεγάλου μεγέθους.

Μια λιγότερο εμπορική προσέγγιση δόθηκε από τον Henk van Ess το Σεπτέμβριο του 2010: «Το Crowdsourcing είναι η δρομολόγηση της επιθυμίας των ειδικών να λύσουν ένα πρόβλημα και να μοιραστούν ελεύθερα την απάντησή του με όλους».

Ο πληθοπορισμός είναι ένα μοντέλο καταναμεμένης επίλυσης προβλημάτων και λειτουργίας. Με την τυπική έννοια του όρου, τα προβλήματα δημοσιεύονται σε ένα άγνωστο πλήθος χρηστών, μέσω μιας ανοιχτής πρόσκλησης για την παροχή λύσεων. Οι χρήστες - οι οποίοι αναφέρονται και ως το πλήθος, δηλαδή crowd - σχηματίζουν κοινότητες στο Διαδίκτυο και υποβάλουν λύσεις. Το πλήθος αξιολογεί επίσης τις λύσεις, επιλέγοντας τις καλύτερες.

Αυτές οι λύσεις επιλέγονται από την αρχική οντότητα που έθεσε το πρόβλημα, ενώ τα φυσικά πρόσωπα που της πρότειναν, ανταμείβονται κάποιες φορές. Σε ορισμένες περιπτώσεις, αυτή η εργασία ανταμείβεται είτε χρηματικά, είτε με βραβεία, είτε με αναγνώριση. Ο πληθοπορισμός μπορεί να παράγει λύσεις από ερασιτέχνες ή εθελοντές που δουλεύουν στον

ελεύθερο χρόνο τους, ή από ειδικούς και μικρές επιχειρήσεις που είναι άγνωστες στο διοργανωτή. Ο Jeff Howe έχει διακρίνει τέσσερις τύπους στρατηγικών για crowdsourcing^[10]:

1. Crowdfunding (συγκέντρωση χρηματικών κεφαλαίων)

2. Crowdcreation (συλλογική δημιουργία)

3. Crowdvoting (συλλογική ψηφοφορία)

4. Crowd wisdom (συλλογική ευφυΐα).

Αντίστοιχα, τα πλεονεκτήματα του crowdsourcing είναι τα παρακάτω:

- Τα προβλήματα μπορούν να επιλυθούν με συγκριτικά μικρό κόστος και συχνά πολύ γρήγορα.
- Η ανταμοιβή γίνεται με βάση τα αποτελέσματα ή δεν υπάρχει καθόλου
- Ο οργανισμός μπορεί να αξιοποιήσει ένα μεγαλύτερο εύρος δεξιοτήτων από αυτό που διαθέτει^[11].
- Ακούγοντας το πλήθος, οι οργανισμοί αποκτούν άμεση επίγνωση των επιθυμιών των πελατών τους.

Στον αντίποδα, ο πληθοπορισμός έχει δεχθεί κριτική για διάφορους λόγους, όπως για το ότι δεν παράγει πάντα ποιοτικά αποτελέσματα, ή επειδή χρησιμοποιείται για την ανάθεση φθηνής ή ακόμη και δωρεάν εργασίας.

Το crowdsourcing προσφέρει επιπλέον τη δυνατότητα να αποτελέσει ένα μηχανισμό επίλυσης προβλημάτων για κυβερνήσεις και μη κερδοσκοπική χρήση. Η διαφορά σε σχέση με την αξιοποίηση του crowdsourcing για επιχειρηματικούς σκοπούς είναι πως ο αρχικός δημιουργός της διαδικασίας επίλυσης του προβλήματος (κυβέρνηση, μη κερδοσκοπικοί οργανισμοί) και οι συμμετέχοντες στη διαδικασία (πολίτες), έχουν αμοιβαία και κοινά οφέλη. Κατά συνέπεια, οι πολίτες έχουν συνεισφέρει τη συλλογική εθελοντική εργασία τους και ωφελούνται οι ίδιοι βελτιώνοντας την ποιότητα ζωής τους, καθώς βοηθούν στην επίλυση του προβλήματος.

Συμπερασματικά, η ανάπτυξη των συνεργατικών υπηρεσιών στο Διαδίκτυο, η κουλτούρα της εθελοντικής συμμετοχής, τα κοινωνικά δίκτυα και τα έργα ανοικτού περιεχομένου και δεδομένων, οδηγούν σε μια νέα συλλογική πραγματικότητα στο Διαδίκτυο. Εθελοντές, αναλαμβάνουν συλλογικά να εκτελέσουν ένα έργο ή να προσφέρουν μια υπηρεσία, μοιράζοντας μεταξύ τους την προσπάθεια και προσφέροντας ελεύθερα τα αποτελέσματά της. Δηλαδή το πλήθος μπορεί να προσφέρει τη συλλογική εμπειρία και ευφυΐα για το κοινό καλό. Αυτή η παγκόσμια δυναμική συμμετοχής και αλληλοβοήθειας, είναι ευκαιρία για να παρακαμφθούν αργοκίνητες και πολύπλοκες γραφειοκρατικές διαδικασίες ή/και να αντιμετωπιστεί η έλλειψη πόρων, ώστε να δοθούν άμεσες λύσεις σε σημαντικά προβλήματα, ενισχύοντας παράλληλα την αλληλεγγύη και την κοινωνική συνοχή.

3. Η αρχιτεκτονική της εφαρμογής

3.1 Λογισμικά – τεχνολογίες που χρησιμοποιήθηκαν

Όλα τα λογισμικά που χρησιμοποιήθηκαν για την εκπόνηση της συγκεκριμένης εργασίας, είναι Ελεύθερα Λογισμικά / Λογισμικά Ανοιχτού Κώδικα. Πρόκειται δηλαδή για λογισμικά τα οποία μπορούν να χρησιμοποιηθούν, αντιγραφούν, μελετηθούν, τροποποιηθούν και αναδιανεμηθούν χωρίς περιορισμό, σύμφωνα με τα όσα ορίζει το το Ίδρυμα Ελευθέρου Λογισμικού (Free Software Foundation, FSB). Παρακάτω ακολουθεί η περιγραφή και τα χαρακτηριστικά τους.

3.1.1 PostgreSQL

Η PostgreSQL αποτελεί ένα ανοιχτού κώδικα Σύστημα Διαχείρισης αντικειμενο-σχεσιακής Βάσης Δεδομένων (ΣΔΒΔ) με πολλές και συνεχώς αυξανόμενες δυνατότητες. Η ανάπτυξη του ήδη διαρκεί πάνω από 20 χρόνια και βασίζεται σε μια αποδεδειγμένα καλή αρχιτεκτονική η οποία εξασφαλίζει την αξιοπιστία, την ακεραιότητα δεδομένων και την ορθή λειτουργία του. Τρέχει σε όλα τα βασικά λειτουργικά συστήματα, όπως Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), και Windows. Επίσης, είναι ACID συμβατό (ACID compliant), δηλαδή συμμορφώνεται με το παρακάτω σύνολο ιδιοτήτων:

- **Atomicity / Ατομικότητα** : Η Ατομικότητα απαιτεί η τροποποίηση που θα γίνει στην ΒΔ να τηρεί τον κανόνα όλα ή τίποτα. Κάθε ατομική συναλλαγή έχει το χαρακτηριστικό πως αν ένα μέρος της αποτύχει, αποτυγχάνει όλη η συναλλαγή και η ΒΔ μένει όπως ήταν πριν εκτελεστεί η συναλλαγή. Είναι πολύ κρίσιμο το ΣΔΒΔ να διατηρεί την ατομικότητα κάθε συναλλαγής ανεξάρτητα από το είδος της εφαρμογής, του ΣΔΒΔ, του λειτουργικού συστήματος ή αστοχίες του υλικού. Μια ατομική συναλλαγή δεν μπορεί να υποδιαιρεθεί, και πρέπει να επεξεργάζεται ολόκληρη ή καθόλου. Έτσι, οι χρήστες είναι εξαλείφουν την πιθανότητα μη ολοκληρωμένων ή ημιτελών συναλλαγών.
- **Consistency / Συνέπεια** : Η ιδιότητα της Συνέπειας διασφαλίζει ότι η ΒΔ διατηρείται σε μια συνεπή κατάσταση, δηλαδή ότι κάθε συναλλαγή θα οδηγεί την βάση δεδομένων από την μια συνεπή κατάσταση στην άλλη. Για παράδειγμα αν ένα πεδίο ενός πίνακα είναι μόνο για ακέραιους αριθμούς τότε το ΣΔΒΔ μπορεί είτε να απορρίψει απόπειρες για είσοδο δεκαδικών αριθμών είτε να τους στρογγυλοποιήσει. Και οι δυο αυτές ενέργειες διατηρούν την συνέπεια.
- **Isolation / Απομόνωση** : Η Απομόνωση αναφέρεται στην απαίτηση ότι καμία ενέργεια δεν μπορεί να γίνει σε δεδομένα τα οποία τροποποιούνται εκείνη την στιγμή από μια συναλλαγή η οποία δεν έχει ακόμα ολοκληρωθεί.
- **Durability / Μονιμότητα** : Η Μονιμότητα εγγυάται στον χρήστη το ΣΔΒΔ ότι αν τελειώσει μια συναλλαγή επιτυχώς τότε τα αποτελέσματα της δεν θα χαθούν. Οι αλλαγές που έχει κάνει η συναλλαγή δεν θα χαθούν ακόμα και αν κρυστάλλει το σύστημα.

Η PostgreSQL έχει ολοκληρωμένη υποστήριξη για foreign keys, joins, views, triggers, καθώς και για stored procedures σε διάφορες γλώσσες προγραμματισμού. Συμπεριλαμβάνει τους περισσότερους τύπους δεδομένων SQL:2008, συμπεριλαμβανομένων INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, και TIMESTAMP. Επίσης, υποστηρίζει

αποθήκευση μεγάλων αντικειμένων (binary objects), όπως εικόνες, ήχοι ή video. Διαθέτει ακόμα περιβάλλοντα προγραμματισμού (APIs) για περισσότερες από δώδεκα γλώσσες προγραμματισμού όπως C, C++, Java, Perl, Python, Ruby, Tcl και άλλες, καθώς και υποστήριξη για την πλατφόρμα .NET και το πρότυπο ODBC.

Επιπρόσθετα, η PostgreSQL υλοποιεί εξεζητημένα χαρακτηριστικά όπως Multi-Version Concurrency Control (MVCC), point in time recovery, tablespaces, asynchronous replication, nested transactions (savepoints), online/hot backups, a sophisticated query planner/optimizer, write ahead logging for fault tolerance. Υποστηρίζει επίσης διεθνή σετ χαρακτήρων, Unicode, κωδικοποίηση χαρακτήρων σε πολλά bytes καθώς και δυνατότητα ταξινόμησης δεδομένων ανεξάρτητα από το locale. Έχει ολοκληρωμένη υποστήριξη για subqueries (συμπεριλαμβανομένων subselects μέσα από το FROM), read-committed και serializable transaction isolation levels.

Άλλα προηγμένα χαρακτηριστικά της PostgreSQL είναι τα table inheritance, rules systems και database events. Το table inheritance (κληρονομικότητα πινάκων) προσθέτει μια αντικειμενοστραφή διάσταση στην δημιουργία πινάκων, επιτρέποντας στους σχεδιαστές βάσεων δεδομένων να δημιουργούν νέους πίνακες από άλλους χρησιμοποιώντας τους ως βάση. Το rules system, (το οποίο επίσης καλείται the query rewrite system), επιτρέπει στον σχεδιαστή να δημιουργήσει κανόνες που ορίζουν συγκεκριμένες λειτουργίες για έναν πίνακα ή για ένα view. Το events system αποτελεί ένα interprocess communication system στο οποίο μηνύματα και events μπορούν να μεταδοθούν μεταξύ πελατών (clients) χρησιμοποιώντας τις LISTEN και NOTIFY εντολές, επιτρέποντας από την απλή peer to peer επικοινωνία ως ένα εξελιγμένο συντονισμό βασισμένο σε database events. Εφόσον τα notifications μπορεί να προέρχονται από triggers και stored procedures, οι clients μπορούν να επιβλέπουν λειτουργίες όπως: update, insert ή delete πινάκων όταν αυτά γίνονται.

Στην βασική βιβλιοθήκη συναρτήσεων της PostgreSQL συμπεριλαμβάνονται εκατοντάδες built-in συναρτήσεις οι οποίες καλύπτουν από βασικές μαθηματικές συναρτήσεις και διαχείριση συμβολοσειρών έως κρυπτογραφία. Triggers και stored procedures μπορούν να γράφουν σε c και να φορτωθούν μέσα στην βάση δεδομένων ως βιβλιοθήκη, επιτρέποντας μεγάλη ευελιξία στην επέκταση των δυνατοτήτων της βάσης. Παρομοίως η PostgreSQL περιλαμβάνει framework που επιτρέπει τον ορισμό και την δημιουργία custom data types καθώς και βοηθητικές συναρτήσεις και τελεστές (operators) που θα περιγράψουν την λειτουργία τους.

Η PostgreSQL αποτελεί ένα πλήρες σχεσιακό σύστημα που υποστηρίζει πολλαπλά σχήματα ανά database, ενώ ο κατάλογος (πληροφορίες σχετικά με τους πίνακες, στήλες, views, κ.α.) είναι διαθέσιμος μέσω του Information Schema, όπως ορίζεται στο SQL standard. Όσον αφορά στα data integrity χαρακτηριστικά, συμπεριλαμβάνονται primary keys, foreign keys με υποστήριξη restricting και cascading updates/deletes, check constraints, unique constraints, και not null constraints. Επίσης, έχει αρκετά προηγμένα χαρακτηριστικά όπως auto-increment columns μέσω sequences και LIMIT/OFFSET που επιτρέπουν την επιστροφή partial result sets. Όσον αφορά τα indexes υποστηρίζει compound, unique, partial, και functional indexes τα οποία μπορούν να χρησιμοποιήσουν οποιονδήποτε από τους B-tree, R-tree, hash, ή Gist αλγόριθμους.

Το GiST (Generalized Search Tree) indexing αποτελεί ένα προηγμένο σύστημα το οποίο συνδυάζει ένα μεγάλο εύρος από διαφορετικούς αλγόριθμους ταξινόμησης και αναζήτησης όπως B-tree, B+-tree, R-tree, partial sum trees, ranked B+-trees και αρκετούς ακόμα. Επίσης, διαθέτει interface το οποίο επιτρέπει τόσο την δημιουργία custom data types

όσο και επεκτάσιμους τρόπους ερωτήσεων (query) για την αναζήτηση τους. Πάνω στο GiST και την PostgreSQL έχει στηριχθεί και η επέκταση PostGIS.

Η PostgreSQL μπορεί να διαχειριστεί εύκολα μεγάλους αριθμούς ταυτόχρονων χρηστών καθώς και μεγάλο όγκο δεδομένων. Υπάρχουν άλλωστε ενεργές εγκαταστάσεις σε περιβάλλοντα παραγωγής που διαχειρίζονται πάνω από 4 terabytes δεδομένων. Μερικές γενικές οριακές τιμές συμπεριλαμβάνονται στον παρακάτω πίνακα:

Limit	Value
Maximum Database Size	Unlimited
Maximum Table Size	32 TB
Maximum Row Size	1.6 TB
Maximum Field Size	1 GB
Maximum Rows per Table	Unlimited
Maximum Columns per Table	250 - 1600 depending on column types
Maximum Indexes per Table	Unlimited

Τέλος, ο πηγαίος κώδικας της PostgreSQL είναι διαθέσιμος κάτω από την ελεύθερη open source άδεια, BSD license. Αυτή η άδεια δίνει την δυνατότητα χρήσης, μετατροπής και διανομής της PostgreSQL σε οποιαδήποτε μορφή, ανοιχτού ή κλειστού κώδικα. Η PostgreSQL δεν είναι μόνο μια δυνατή βάση δεδομένων ικανή να τρέχει μέσα σε επιχειρήσεις, είναι μια πλατφόρμα ανάπτυξης πάνω στην οποία δύναται να γίνει ανάπτυξη in-house, web ή εμπορικών εφαρμογών τα οποία χρειάζονται RDBMS.

3.1.2 PostGIS

Το PostGIS είναι μία επέκταση της PostgreSQL για να υποστηρίξει χωρικά δεδομένα, σύμφωνα με το πρότυπο του OGC (Open Geospatial Consortium). Παρέχει ειδικούς τελεστές για τη σύνταξη ερωτημάτων, λειτουργίες συνάθροισης επάνω σε χωρικά δεδομένα καθώς και χωρικές συναρτήσεις. Επιτρέπει επίσης την ανάθεση προβολικών συστημάτων στα χωρικά δεδομένα. Τέλος, μπορεί να χρησιμοποιηθεί για να οπτικοποιηθούν τα δεδομένα μέσω ειδικών εφαρμογών όπως το QuantumGIS και ο GeoServer.

Πιο συγκεκριμένα, το PostGIS ενεργοποιεί την PostgreSQL, καθιστώντας την χωρική βάση δεδομένων, επιτρέποντάς της να χρησιμοποιηθεί σε Γεωγραφικά Συστήματα Πληροφοριών (GIS) και εφαρμογές διαδικτυακής χαρτογράφησης, κατ' αντιστοιχία με τον τρόπο που η Oracle Spatial ενεργοποιεί χωρικά την βάση δεδομένων Oracle. Επίσης, είναι σταθερό, γρήγορο, συμβατό με τα διεθνή πρότυπα και παρέχει εκατοντάδες χωρικές συναρτήσεις. Όλα τα παραπάνω, την έχουν καταστήσει ως μία εκ των δημοφιλέστερων αυτή τη στιγμή χωρική βάση δεδομένων ανοιχτού κώδικα. Η PostGIS χρησιμοποιείται από ποικίλους οργανισμούς ανά τον κόσμο, περιλαμβανομένων υψηλού ρίσκου κυβερνητικών υπηρεσιών και οργανισμών, αποθηκεύοντας terabytes δεδομένων και εξυπηρετώντας

εκατομμύρια διαδικτυακές κλήσεις ανά ημέρα. Παρακάτω, αναφέρονται συνοπτικά κάποια από τα βασικά χαρακτηριστικά της επέκτασης PostGIS:

- Εκατοντάδες χωρικές λειτουργίες και εργαλεία χωρικής ανάλυσης όπως ζώνες επιρροής, ένωση, τομή, επίθεση, απόσταση και πολλά περισσότερα
- Ολοκλήρωση συναλλαγών ACID
- Χωρικοί κατάλογοι R-Tree
- Υποστήριξη πολλών ταυτόχρονων χρηστών
- Κλείδωμα σε επίπεδο γραμμής
- Δυνατότητα αντιγραφής
- Στεγανοποίηση
- Ασφάλεια με βάση ρόλους
- Χώροι πινάκων, σχήματα βάσης

3.1.3 GeoServer

Ο GeoServer είναι ένας open source Web Server που επιτρέπει στους χρήστες του να διαχειρίζονται και να μοιράζουν γεωχωρικά δεδομένα. Είναι γραμμένος στη γλώσσα προγραμματισμού Java και από το 2001 που ξεκίνησε το Geoserver Project αναπτύσσεται συνεχώς από μια ευρεία κοινότητα χρηστών και προγραμματιστών σε όλο τον κόσμο. Συμμορφώνεται πλήρως με τα πρότυπα του Open Geospatial Consortium (OGC) και προσφέρει υπηρεσίες σε χωρική πληροφορία χρησιμοποιώντας τα open standards για πολλά από τα οποία αποτελεί και την υλοποίηση αναφοράς.

Ο GeoServer χρησιμοποιείται για να αξιοποιηθούν δεδομένα στα οποία υπάρχει χωρική πληροφορία (vector ή raster) και υποστηρίζει μεγάλο πλήθος μορφών όπως:

- PostGIS
- Oracle Spatial
- ArcSDE
- DB2
- MySQL
- Shapefiles
- GeoTIFF
- GTOPO30
- ECW, MrSID
- JPEG2000

Στα χωρικά δεδομένα ο GeoServer προσφέρει υπηρεσίες όπως Web Map Service (WMS), Web Feature Service (WFS), Web Coverage Service (WCS) και άλλες. Υποστηρίζει ένα τεράστιο πλήθος από συστήματα αναφοράς, τόσο παγκόσμια όσο και τοπικά (γεωκεντρικά και γεωδαιτικά), ενώ επιτρέπει on-the-fly transformation από ένα σύστημα σε άλλο. Η υπηρεσία WMS αφορά τη θέαση (προβολή) των χωρικών δεδομένων σε χάρτη και αυτό μπορεί να γίνει σε οποιαδήποτε από τις παρακάτω μορφές:

- AtomPub

- GIF
- GeoRSS
- GeoTiff / GeoTiff 8-bits
- JPEG
- KML compressed / KML network link / KML plain
- OpenLayers
- PDF
- PNG / PNG 8-bit
- SVG
- Tiff / Tiff 8-bits

Αντίστοιχα, η υπηρεσία WFS υποστηρίζει αναζήτηση και ανάκτηση των δεδομένων, με γεωγραφικά ή μη κριτήρια. Επίσης, ο GeoServer υλοποιεί και την υπηρεσία transactional Web Feature Service (WFS-T) η οποία επιτρέπει επίσης την δημιουργία, διαγραφή και τροποποίηση δεδομένων. Η υπηρεσία υλοποιείται στις παρακάτω μορφές:

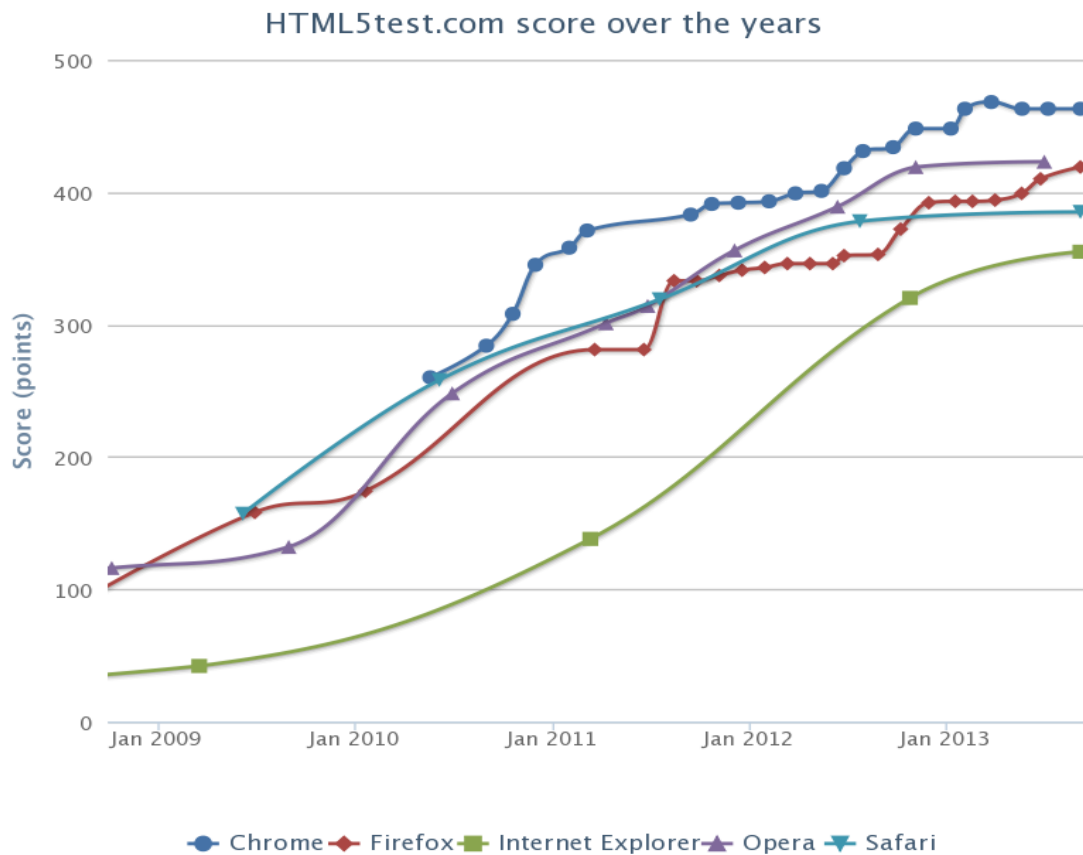
- CSV
- GML2 / GML2-GZIP / GML3.1 / GML3.2
- GeoJSON
- Shapefile
- application/gml+xml

Ο GeoServer είναι το λογισμικό που δέχεται τα αιτήματα του χρήστη της εφαρμογής και αποκρίνεται αναλόγως με τα περιεχόμενα της βάσης δεδομένων.

3.1.4 HTML5 – JavaScript – AJAX

Η HTML (HyperText Markup Language) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες και η HTML5 είναι η εξέλιξη της, η οποία αναμένεται στο μέλλον να την αντικαταστήσει πλήρως. Γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες, (οι οποίες συνήθως λειτουργούν ανά ζεύγη για παράδειγμα <html> </html>), μέσα στο περιεχόμενο της ιστοσελίδας. Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ. Οι ιστοσελίδες ουσιαστικά είναι *.html αρχεία, τα οποία οι browser τα διαβάζουν και ανάλογα μορφοποιούν την ιστοσελίδα. Ο browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει και να μορφοποιήσει το περιεχόμενο της σελίδας.

Η HTML5 αποτελεί το μελλοντικό πρότυπο στο σχεδιασμό των ιστοσελίδων όπως σχεδιάζεται και υλοποιείται από το World Wide Web Consortium (W3C) και το Web Hypertext Application Technology Working Group (WHATWG). Αποτελεί την 5η κατά σειρά βελτίωση της HTML και προσφέρει πολλά πλεονεκτήματα στους σχεδιαστές, όπως η μείωση της ανάγκης για plug-in και η υποστήριξη multimedia αντικειμένων. Ακόμα δεν έχει υιοθετηθεί πλήρως ως πρότυπη γλώσσα γραφής ιστοσελίδων (ακόμα χρησιμοποιείται η HTML 4.01), αλλά ήδη υποστηρίζεται από τους περισσότερους browser αφού ουσιαστικά η HTML5 αποτελεί υπερσύνολο της HTML. Στην Εικόνα 34 φαίνεται η υποστήριξη HTML5 στους πιο συνηθισμένους browser διαχρονικά.



Εικόνα 1. Διαχρονικά η υποστήριξη HTML5 από τους κυριότερους browser

Η HTML5 μαζί με τη JavaScript δίνει στους προγραμματιστές τη δυνατότητα να σχεδιάζουν και να αναπτύσσουν εφαρμογές οι οποίες τρέχουν σε όλες τις συσκευές, υπολογιστές, tablets ή Smartphones, ανεξαρτήτως μεγέθους και τύπου οθόνης ή Λειτουργικού Συστήματος.

Η JavaScript (JS) είναι διερμηνευόμενη (interpreted) γλώσσα προγραμματισμού για διαδικτυακές εφαρμογές. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται. Πρόκειται για γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης.

Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστραφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού. Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων — τέτοια παραδείγματα είναι τα έγγραφα PDF, οι εξειδικευμένοι browsers (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets).

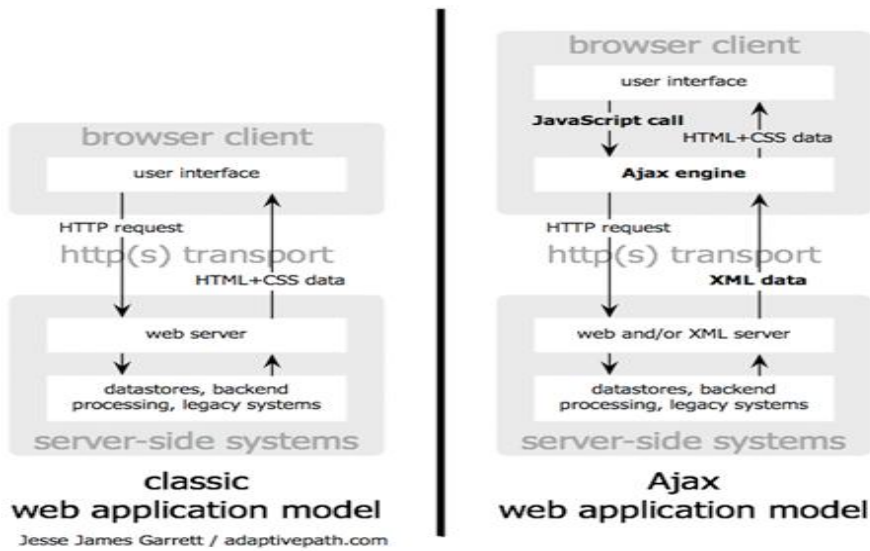
Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), δηλαδή το browser του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα JavaScript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή (server), αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (server-side γλώσσες προγραμματισμού). Η χρήση της JavaScript στο διακομιστή εμφανίζεται πάλι σήμερα, με τη διάδοση του Node.js, ενός μοντέλου προγραμματισμού βασισμένο στα γεγονότα (events).

Η JavaScript δημιουργήθηκε από τον Brendan Eich της εταιρείας Netscape με την επωνυμία Mocha. Αργότερα, μετονομάστηκε σε LiveScript, και τελικά σε JavaScript, κυρίως επειδή η ανάπτυξή της επηρεάστηκε περισσότερο από τη γλώσσα προγραμματισμού Java (καθώς και οι δύο είναι αντικειμενοστραφείς γλώσσες προγραμματισμού). LiveScript ήταν το επίσημο όνομα της γλώσσας όταν για πρώτη φορά κυκλοφόρησε στην αγορά, με το πρόγραμμα περιήγησης Netscape Navigator 2.0, τον Σεπτέμβριο του 1995. Από LiveScript μετονομάστηκε σε JavaScript σε μια κοινή ανακοίνωση με την εταιρεία Sun Microsystems στις 4 Δεκεμβρίου 1995, όταν επεκτάθηκε στην έκδοση του προγράμματος περιήγησης στο Web, Netscape εκδοχή 2.0B3. Έτσι, απέκτησε μεγάλη επιτυχία ως γλώσσα στην πλευρά του πελάτη (client-side) για εκτέλεση κώδικα σε ιστοσελίδες, και περιλήφθηκε σε διάφορα προγράμματα περιήγησης. Η εταιρεία Microsoft ονόμασε την εφαρμογή της σε JScript για να αποφύγει δύσκολα θέματα εμπορικών σημάτων και την συμπεριέλαβε στο πρόγραμμα Internet Explorer 3.0, το οποίο κυκλοφόρησε τον Αύγουστο του 1996.

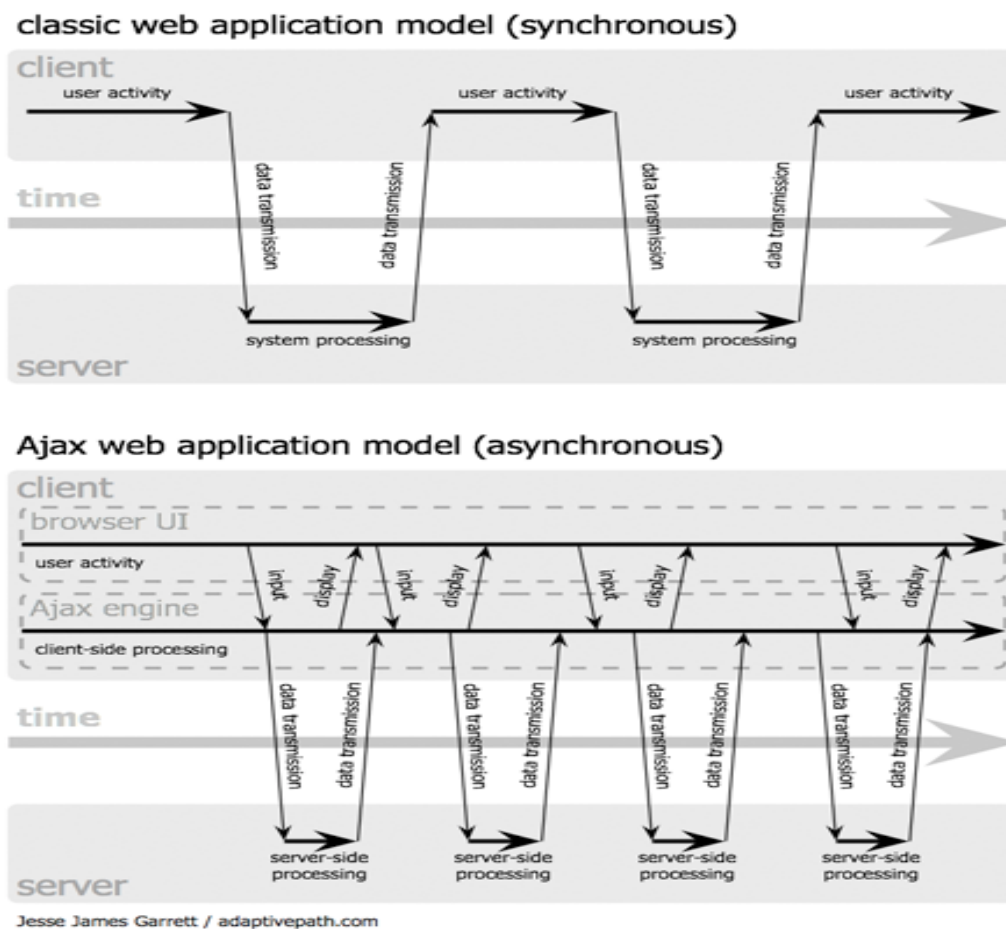
Η JavaScript είναι πλέον μία από τις πιο δημοφιλείς γλώσσες προγραμματισμού ηλεκτρονικών υπολογιστών στον παγκόσμιο ιστό (Web). Αρχικά, όμως, πολλοί επαγγελματίες προγραμματιστές υποτίμησαν τη γλώσσα διότι το κοινό της ήταν ερασιτέχνες συγγραφείς ιστοσελίδων και όχι επαγγελματίες προγραμματιστές. Με τη χρήση της τεχνολογίας AJAX, η γλώσσα επέστρεψε στο προσκήνιο και έδωσε πολύ περισσότερες δυνατότητες στους προγραμματιστές.

Με τη τεχνική AJAX (Asynchronous JavaScript And Xml) οι διαδικτυακές εφαρμογές έχουν τη δυνατότητα να στείλουν ή να ανακτήσουν δεδομένα από το server στο παρασκήνιο χωρίς καμία παρέμβαση στην εμφάνιση ή τη συμπεριφορά της ιστοσελίδας. Αυτό συμβαίνει μέσω των αντικειμένων XMLHttpRequest και το μεγάλο πλεονέκτημα αυτής της τεχνικής είναι πως ελαχιστοποιεί τις περιπτώσεις που η ιστοσελίδα κολλάει και δεν ανταποκρίνεται όσο περιμένει την απάντηση από το server.

Στις Εικόνες 35 – 36 (<http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications>) φαίνεται η διαφορά στην αρχιτεκτονική μιας διαδικτυακής εφαρμογής με την τεχνολογία AJAX σε σχέση με το κλασικό μοντέλο, καθώς και η διαφορά μεταξύ synchronous και asynchronous αιτημάτων στο διακομιστή.



Εικόνα 2 . Διαφορά στην αρχιτεκτονική κλασσικού – AJAX μοντέλου σε Web εφαρμογές



Εικόνα 3. Διαφορά στην ροή κλασσικού(synchronous) – AJAX (asynchronous) μοντέλου σε Web εφαρμογές

3.1.5 PHP

Η PHP αποτελεί τη δημοφιλέστερη και πιο διαδεδομένη server-side γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από συμβατό web server (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML. Αυτό, πρακτικά σημαίνει πως ο server διαβάζει τα αρχεία PHP τα οποία πρέπει να είναι αποθηκευμένα εκεί, και τα μεταφράζει σε κώδικα HTML τον οποίο εμφανίζει ο φυλλομετρητής (browser) στη πλευρά του πελάτη.

Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. *.php, *.php4, *.html κ.ά.). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται κώδικα PHP.

Η ιστορία της PHP ξεκινά από το 1995, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες.

Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, βασιζόμενη αυτή τη φορά στη γλώσσα C και αριθμώντας περισσότερα από 50.000 web sites που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και τα πρώτα snapshots της επερχόμενης PHP 6 , για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει.

Σήμερα περισσότερα από το 35% των ιστοσελίδων του Διαδικτύου , χρησιμοποιούν scripts γραμμένα με τη γλώσσα PHP, ενώ το υπόλοιπο 65% το μοιράζονται στατικές σελίδες HTML και όλες οι άλλες γλώσσες προγραμματισμού.

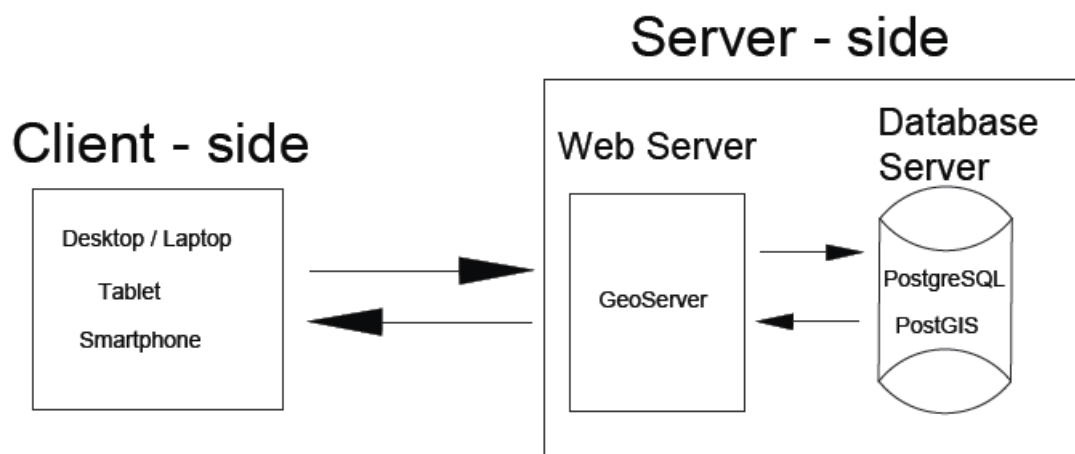
Πρόκειται για μια εξέλιξη που ο ίδιος ο Rasmus Lerdorf σε πρόσφατη συνέντευξή του δήλωσε ότι δεν περίμενε όταν, δημιουργούσε τις πρώτες γραμμές κώδικα PHP. Τόνισε όμως ότι η PHP δεν θα είχε γίνει τόσο δημοφιλής αν η εξέλιξή της είχε παραμείνει προσωπική του προσπάθεια και δεν είχε βοηθηθεί από τους Andi Gutmans, Zeev Suraski και την εθελοντική συμμετοχή προγραμματιστών από ολόκληρο τον κόσμο. Τα περισσότερα web sites επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP.

Με τις γλώσσες Perl και C/C++ στις οποίες έχει τις ρίζες της, η PHP έχει εξαιρετική ομοιότητα ως προς τον τρόπο σύνταξης, αλλά και πολλές εντολές της. Ο κώδικας PHP μιας ιστοσελίδας περικλείεται από τα tags `<?php` και `>`. Εναλλακτικά, μπορούν να χρησιμοποιηθούν τα tags `<script language=php>` και `</script>`, όμως αυτή η τελευταία μέθοδος δεν χρησιμοποιείται συχνά, καθώς η χρήση των tags `<?php` και `>` είναι πιο απλή και έχει επικρατήσει. Την απεικόνιση κειμένου αναλαμβάνουν οι εντολές `print` ή `echo`, οι οποίες ουσιαστικά μετατρέπουν τον κώδικα σε HTML tags που εμφανίζονται στον πελάτη.

3.2 Υλοποίηση της εφαρμογής

3.2.1 Γενικά

Η υλοποίηση της εφαρμογής συνίσταται στο σχεδιασμό και κατασκευή της βάσης δεδομένων, στη διασύνδεση της με τον GeoServer, και στη συνέχεια τη δημιουργία της διεπαφής με το χρήστη (User Interface) καθώς και των αλγορίθμων που προσδίδουν τη λειτουργικότητα. Στην Εικόνα 37 φαίνονται τα συστατικά μέρη της εφαρμογής.



Εικόνα 4. Αρχιτεκτονική της εφαρμογής

Η εφαρμογή αποτελείται από δέκα αρχεία καθένα από τα οποία εκτελεί συγκεκριμένες λειτουργίες. Πιο συγκεκριμένα, υπάρχουν αρχεία html, css, javascript και php τα οποία είναι αποθηκευμένα στους φακέλους www του Apache EnterpriseDB Server και του GeoServer. Οι δύο αυτοί servers, σε τυπική εγκατάσταση τρέχουν παράλληλα μεν, αλλά σε διαφορετικές θύρες(ports).

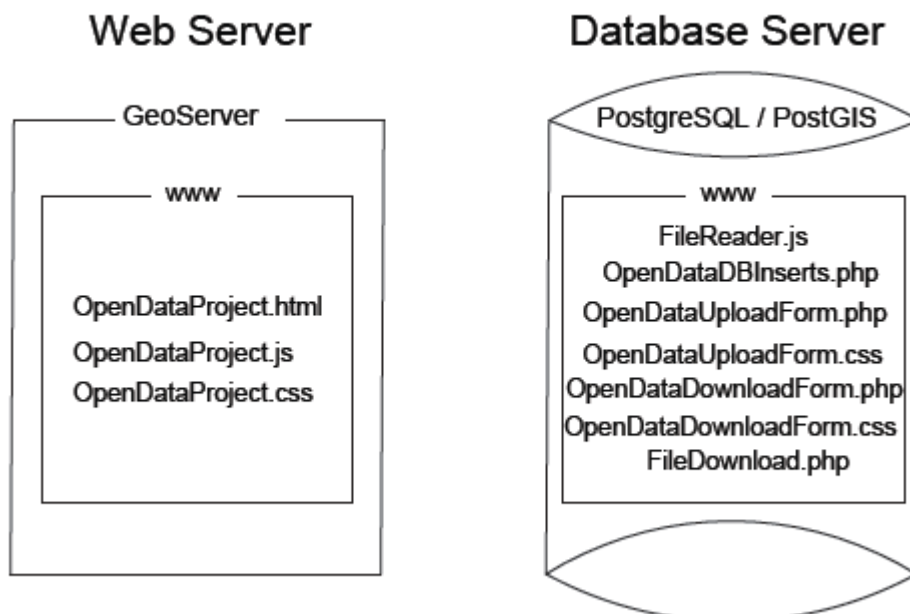
Στο φάκελο www του Database Server υπάρχουν τα αρχεία τα οποία είναι υπεύθυνα για upload και download των δεδομένων ενώ στο φάκελο www του GeoServer τα αρχεία που εμφανίζουν την αρχική σελίδα της εφαρμογής. Αυτό συμβαίνει καθώς, για λόγους ασφαλείας, δεν επιτρέπονται απ'ευθείας αιτήματα από τον ένα server στον άλλο (cross-server requests), και έτσι, μέσω υπερσυνδέσμων (links) από την αρχική σελίδα ο χρήστης μεταβαίνει στον server της Postgres όποτε χρειάζεται να ανεβάσει ή να κατεβάσει δεδομένα.

Τα αρχεία που ενημερώνουν ή κάνουν ερωτήματα στη βάση δεδομένων είναι αρχεία php καθώς είναι καθαρή server – side γλώσσα προγραμματισμού και προσφέρει πληθώρα συναρτήσεων για αυτό το σκοπό.

Αντίθετα, πολλές από τις συναρτήσεις που υλοποιούν τους εκάστοτε αλγόριθμους που είναι απαραίτητοι για την λειτουργία της εφαρμογής είναι γραμμένοι σε JavaScript, δεδομένου ότι είναι πανίσχυρη και πάρα πολύ γρήγορη. Η JavaScript διαθέτει μεθόδους για πάρα πολλές λειτουργίες οι οποίες εάν είχαν γραφεί σε php θα απαιτούσαν πολλές γραμμές κώδικα με ό,τι συνεπάγεται αυτό ειδικά σε σχέση με την αποσφαλμάτωση.

Τέλος, το User Interface δημιουργήθηκε με κώδικα HTML5 και η μορφοποίηση της εμφάνισης έγινε με αρχεία css τα οποία αποτελούν βέλτιστη πρακτική όσον αφορά το styling των ιστοσελίδων.

Παρακάτω, φαίνεται σχηματικά η δομή των αρχείων της εφαρμογής



Εικόνα 5. Δομή των αρχείων της εφαρμογής

Στο φάκελο www του GeoServer, είναι αποθηκευμένα τα αρχεία που αποτελούν την αρχική σελίδα της εφαρμογής, καθώς εκεί κατευθύνεται ο επισκέπτης της ιστοσελίδας. Σε αυτήν την περίπτωση, ο GeoServer ανακτά τα περιεχόμενα της βάσης δεδομένων μέσω της τεχνικής AJAX (XmlHttpRequests) και τα επεξεργάζεται κατάλληλα με JavaScript.

3.2.2 Σχεδιασμός της Βάσης Δεδομένων

Η Βάση Δεδομένων αποτελείται από έξι πίνακες που περιέχουν δεδομένα για τους uploaders (πίνακας contributorTable), τα dataset (πίνακας datasetTable), τα σημεία ενδιαφέροντος (πίνακας poisTable), τις κατηγορίες των dataset (πίνακας categoryTable) καθώς και τις λέξεις-κλειδιά (πίνακας attributeTable). Επειδή σύμφωνα με το σχεδιασμό, τα dataset έχουν σχέση πολλά προς πολλά με τις λέξεις κλειδιά, δημιουργήθηκε ένας έκτος πίνακας συσχετίσεων (πίνακας dsaTable). Από όλους τους παραπάνω πίνακες, ο μόνος που περιέχει χωρική πληροφορία (στήλη με type geometry) είναι ο πίνακας για τα σημεία ενδιαφέροντος.

Οι ορισμοί της Βάσης Δεδομένων ακολουθούν παρακάτω:

```
-- Database: "GeoDB"
-- DROP DATABASE "GeoDB";
CREATE DATABASE "GeoDB"
  WITH OWNER = postgres
       ENCODING = 'UTF8'
       TABLESPACE = pg_default
       LC_COLLATE = 'Greek_Greece.1253'
       LC_CTYPE = 'Greek_Greece.1253'
       CONNECTION LIMIT = -1;
ALTER DATABASE "GeoDB"
  SET search_path = "$user", public, topology, tiger;
```

```

-- Schema: mydb
-- DROP SCHEMA mydb;
CREATE SCHEMA mydb
  AUTHORIZATION postgres;

-----
-- Table: mydb."attributeTable"
-- DROP TABLE mydb."attributeTable";
CREATE TABLE mydb."attributeTable"
(
  attributeid integer NOT NULL DEFAULT
nextval('mydb."attributeTable_addtributeid_seq"'::regclass),
  attributename character varying(100) NOT NULL,
  CONSTRAINT "attributeTable_pkey" PRIMARY KEY (attributeid)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE mydb."attributeTable"
  OWNER TO postgres;

-----
-- Table: mydb."categoryTable"
-- DROP TABLE mydb."categoryTable";
CREATE TABLE mydb."categoryTable"
(
  categoryid serial NOT NULL,
  categorydescription character varying(200) NOT NULL,
  categoryparentid integer,
  CONSTRAINT "categoryTable_pkey" PRIMARY KEY (categoryid),
  CONSTRAINT "FK_categoryParentID" FOREIGN KEY (categoryparentid)
  REFERENCES mydb."categoryTable" (categoryid) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE mydb."categoryTable"
  OWNER TO postgres;
-- Index: mydb."fk_categoryTable_categoryTable1_idx"
-- DROP INDEX mydb."fk_categoryTable_categoryTable1_idx";
CREATE INDEX "fk_categoryTable_categoryTable1_idx"
  ON mydb."categoryTable"
  USING btree
  (categoryparentid);

-----
-- Table: mydb."contributorTable"
-- DROP TABLE mydb."contributorTable";
CREATE TABLE mydb."contributorTable"
(
  contributorid serial NOT NULL,
  contributornickname character varying(100) NOT NULL,
  contributoremail character varying(100),

```

```

contributorfirstname character varying(100),
contributorlastname character varying(100),
confributorfacebook character varying(100),
contributortwitter character varying(100),
contributorwebpageurl character varying(200),
contributorregdate date,
contributorstatus character varying(20) NOT NULL,
CONSTRAINT "contributorTable_pkey" PRIMARY KEY (contributorid)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE mydb."contributorTable"
  OWNER TO postgres;

-----
-- Table: mydb."datasetTable"
-- DROP TABLE mydb."datasetTable";
CREATE TABLE mydb."datasetTable"
(
  dataset_id serial NOT NULL,
  datasetname character varying(100) NOT NULL,
  datasetdescr text,
  datasetdate date NOT NULL,
  dataseticon character varying(100),
  datasetpoiicon character varying(100),
  contributortable_contributorid integer NOT NULL,
  categorytable_categoryid integer NOT NULL,
  CONSTRAINT "datasetTable_pkey" PRIMARY KEY (dataset_id),
  CONSTRAINT "FK_categoryTable_categoryID" FOREIGN KEY
(contributortable_contributorid)
  REFERENCES mydb."categoryTable" (categoryid) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "FK_contributorTable_contributorID" FOREIGN KEY
(contributortable_contributorid)
  REFERENCES mydb."contributorTable" (contributorid) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE mydb."datasetTable"
  OWNER TO postgres;
-- Index: mydb."fk_datasetTable_categoryTable1_idx"
-- DROP INDEX mydb."fk_datasetTable_categoryTable1_idx";
CREATE INDEX "fk_datasetTable_categoryTable1_idx"
  ON mydb."datasetTable"
  USING btree
  (categorytable_categoryid);
-- Index: mydb."fk_datasetTable_contributorTable1_idx"
-- DROP INDEX mydb."fk_datasetTable_contributorTable1_idx";
CREATE INDEX "fk_datasetTable_contributorTable1_idx"
  ON mydb."datasetTable"
  USING btree
  (contributortable_contributorid);

```

```

-----
-- Table: mydb."dsaTable"
-- DROP TABLE mydb."dsaTable";
CREATE TABLE mydb."dsaTable"
(
  attributetable_attributeid integer NOT NULL,
  datasettable_dataset_id integer NOT NULL,
  attrvalue character varying(200) NOT NULL,
  CONSTRAINT "dsaTable_pkey" PRIMARY KEY (attributetable_attributeid,
datasettable_dataset_id),
  CONSTRAINT "FK_dsaTable_attributeTable1" FOREIGN KEY
(attributetable_attributeid)
  REFERENCES mydb."attributeTable" (attributeid) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT "FK_dsaTable_datasetTable1" FOREIGN KEY
(datasettable_dataset_id)
  REFERENCES mydb."datasetTable" (dataset_id) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE mydb."dsaTable"
  OWNER TO postgres;
-- Index: mydb."fk_dsaTable_attributeTable1_idx"
-- DROP INDEX mydb."fk_dsaTable_attributeTable1_idx";
CREATE INDEX "fk_dsaTable_attributeTable1_idx"
  ON mydb."dsaTable"
  USING btree
  (attributetable_attributeid);
-- Index: mydb."fk_dsaTable_datasetTable1_idx"
-- DROP INDEX mydb."fk_dsaTable_datasetTable1_idx";
CREATE INDEX "fk_dsaTable_datasetTable1_idx"
  ON mydb."dsaTable"
  USING btree
  (datasettable_dataset_id);

-----
-- Table: mydb."poisTable"
-- DROP TABLE mydb."poisTable";
CREATE TABLE mydb."poisTable"
(
  poisid serial NOT NULL,
  poisdescr character varying(200),
  poisdataset character varying(20),
  poisgeom geometry(Point,4326),
  poidataset_id integer NOT NULL,
  CONSTRAINT "poisTable_pkey" PRIMARY KEY (poisid),
  CONSTRAINT "FK_poiDataset_id" FOREIGN KEY (poidataset_id)
  REFERENCES mydb."datasetTable" (dataset_id) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE

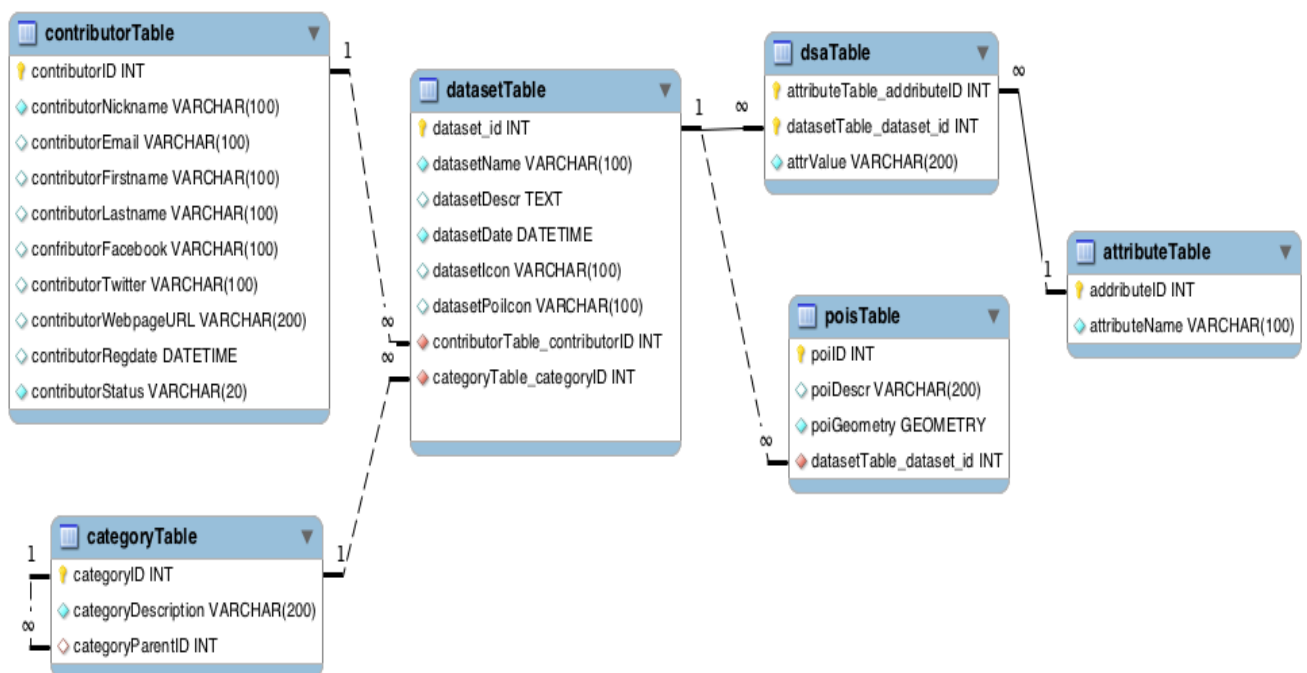
```

```

);
ALTER TABLE mydb."poiTable"
  OWNER TO postgres;
-- Index: mydb."fk_poiTable_datasetTable_idx"
-- DROP INDEX mydb."fk_poiTable_datasetTable_idx";
CREATE INDEX "fk_poiTable_datasetTable_idx"
  ON mydb."poiTable"
  USING btree
  (poidataset_id);

```

Σχηματικά, ο σχεδιασμός της βάσης δεδομένων φαίνεται στην παρακάτω εικόνα, ενώ στη συνέχεια αναλύονται οι βασικοί πίνακες ξεχωριστά με αναφορά στα πεδία τους.



Εικόνα 6. Διάγραμμα της Βάσης Δεδομένων

datasetTable : Ο συγκεκριμένος πίνακας περιέχει όλες τις πληροφορίες που αφορούν κάθε dataset, δηλαδή ένα σύνολο από σημεία ενδιαφέροντος (pois). Οι στήλες αυτού του πίνακα είναι οι παρακάτω:

dataset_id (απαραίτητο): Μοναδικό αναγνωριστικό κάθε εγγραφής(πρωτεύον κλειδί), τύπου serial δηλαδή μεγάλος ακέραιος ο οποίος αυξάνει αυτόματα κατά ένα σε κάθε νέα καταχώρηση.

datasetname (απαραίτητο): Το όνομα του dataset, όπως το καταχωρεί ο χρήστης στη σελίδα upload. Συμβολοσειρά μήκους εκατό χαρακτήρων

datasetdescr (προαιρετικό): Η περιγραφή του dataset, όπως τη δίνει ο χρήστης στη σελίδα upload. Συμβολοσειρά χωρίς όριο μήκους (κείμενο)

datasetdate (προστίθεται αυτόματα): Η ημερομηνία που έγινε upload το dataset.

`dataseticon` (προαιρετικό): Το εικονίδιο που θα έχει το `dataset`. Συμβολοσειρά μήκους εκατό χαρακτήρων όπου αποθηκεύεται το όνομα της εικόνας

`datasetroiicon` (προαιρετικό): Το εικονίδιο που θα έχει κάθε ροί αυτού του `dataset`. Συμβολοσειρά μήκους εκατό χαρακτήρων όπου αποθηκεύεται το όνομα της εικόνας

`contributortable_contributorid` (προστίθεται αυτόματα): Πρόκειται για ξένο κλειδί, το οποίο «δείχνει» το πρωτεύον κλειδί στον πίνακα `contributortable`, δηλαδή την εγγραφή που αφορά τον uploader του συγκεκριμένου `dataset`.

`categorytable_categoryid` (προστίθεται αυτόματα): Πρόκειται για ξένο κλειδί, το οποίο «δείχνει» το πρωτεύον κλειδί στον πίνακα `categorytable`, δηλαδή την εγγραφή που αφορά την κατηγορία του συγκεκριμένου `dataset`.

contributorTable : Ο συγκεκριμένος πίνακας περιέχει όλες τις πληροφορίες που αφορούν κάθε uploader, τα στοιχεία του ατόμου που ανέβασε το `dataset`. Οι στήλες αυτού του πίνακα είναι οι παρακάτω:

`contributorid` (απαραίτητο) : Μοναδικό αναγνωριστικό κάθε εγγραφής(πρωτεύον κλειδί), τύπου `serial`.

`contributornickname`(απαραίτητο): Το ψευδώνυμο του uploader. Συμβολοσειρά μήκους εκατό χαρακτήρων

`contributoremail`(προαιρετικό): Το email του uploader. Συμβολοσειρά μήκους εκατό χαρακτήρων. Ελέγχεται αν είναι έγκυρο e-mail (αν περιέχει το σύμβολο @)

`contributorfirstname`(προαιρετικό): Το όνομα του uploader. Συμβολοσειρά μήκους εκατό χαρακτήρων.

`contributorlastname`(προαιρετικό): Το επίθετο του uploader. Συμβολοσειρά μήκους εκατό χαρακτήρων.

`contributorfacebook`(προαιρετικό): Ο λογαριασμός facebook του uploader. Συμβολοσειρά μήκους εκατό χαρακτήρων. Ελέγχεται αν είναι έγκυρο (αν περιέχει το http).

`contributortwitter`(προαιρετικό): Ο λογαριασμός twitter του uploader. Συμβολοσειρά μήκους εκατό χαρακτήρων. Ελέγχεται αν είναι έγκυρο (αν περιέχει το http).

`contributorwebpageurl`(προαιρετικό): Ιστοσελίδα του uploader. Συμβολοσειρά μήκους διακοσίων χαρακτήρων. Ελέγχεται αν είναι έγκυρο (αν περιέχει το http).

`contributorregdate`(προστίθεται αυτόματα) : Η ημερομηνία που καταχωρήθηκε ο uploader.

`contributorstatus` (απαραίτητο): Το status του uploader Συμβολοσειρά μήκους είκοσι χαρακτήρων

poisTable: Ο συγκεκριμένος πίνακας περιέχει τα σημεία ενδιαφέροντος. Οι στήλες αυτού του πίνακα είναι οι παρακάτω:

`poisid` (απαραίτητο): Μοναδικό αναγνωριστικό κάθε εγγραφής(πρωτεύον κλειδί), τύπου `serial`.

`poisdescr`(προαιρετικό): Η περιγραφή του ροί, όπως υπάρχει στο αρχείο upload. Συμβολοσειρά μήκους διακοσίων χαρακτήρων.

`poisgeom`(απαραίτητο): Η γεωμετρία point στο Σύστημα Αναφοράς WGS'84

roidataset_id (προστίθεται αυτόματα): Ξένο κλειδί, το οποίο «δείχνει» το πρωτεύον κλειδί στον πίνακα datasetTable, δηλαδή το dataset στο οποίο ανήκει κάθε σημείο.

categoryTable: Ο συγκεκριμένος πίνακας περιέχει τις κατηγορίες που ταξινομούνται τα dataset. Οι στήλες αυτού του πίνακα είναι οι παρακάτω:

categoryid (προστίθεται αυτόματα) : Μοναδικό αναγνωριστικό κάθε εγγραφής (πρωτεύον κλειδί), τύπου serial.

categorydescription (προαιρετικό): Η περιγραφή της κατηγορίας, όπως τη δίνει ο χρήστης στη σελίδα upload. Συμβολοσειρά μήκους διακοσίων χαρακτήρων.

categoryparentid(προαιρετικό): Ξένο κλειδί, σε περίπτωση που αποτελεί υποκατηγορία, δείχνει στην κύρια κατηγορία.

Όπως έχει ήδη αναφερθεί, η διαχείριση της βάσης δεδομένων γίνεται από τα αρχεία php που είναι αποθηκευμένα στο φάκελο www της βάσης δεδομένων. Πιο συγκεκριμένα, την ενημέρωση της βάσης, δηλαδή την εισαγωγή νέων εγγραφών, την υλοποιεί το αρχείο OpenDataUploadForm.php το οποίο με τη σειρά του καλεί τα OpenDataUploadForm.css, FileReader.js και OpenDataDBInserts.php. Το αρχείο css φροντίζει για τη μορφοποίηση και το styling της σελίδας upload, το αρχείο js περιέχει κώδικα σε javascript για επεξεργασία του αρχείου που ανέβασε ο χρήστης, ενώ στο OpenDataDBInserts.php είναι γραμμένα τα queries για την εισαγωγή των δεδομένων .

Αντίστοιχα, τα αρχεία OpenDataDownloadForm.php, OpenDataDownloadForm.css FileDownload.php είναι υπεύθυνα για τα downloads του χρήστη. Το πρώτο εμφανίζει τα διαθέσιμα σύνολα δεδομένων για να επιλέξει ενώ το τελευταίο τα γράφει σε ένα νέο αρχείο για κατέβασμα

3.2.3 Διασύνδεση με τον GeoServer

Το επόμενο βήμα, μετά την ολοκλήρωση της βάσης δεδομένων, είναι η σύνδεση της με τον GeoServer. Με αυτό τον τρόπο γίνεται δυνατή η αξιοποίηση της βάσης από τη διαδικτυακή εφαρμογή, μέσω αιτημάτων που γίνονται στον GeoServer, είτε με την τεχνική AJAX είτε με queries γραμμένα στις γλώσσες CQL (Common Query Language) και ECQL(Extended Common Query Language). Πρόκειται για γλώσσες που έχουν αναπτυχθεί από την OGC (Open Geospatial Consortium) σε συμφωνία με το πρότυπο WFS, και ουσιαστικά επιτρέπουν στον προγραμματιστή να κάνει αιτήματα στο server, εφαρμόζοντας συγκεκριμένα φίλτρα στα δεδομένα. Η διαδικασία είναι αντίστοιχη με τα συνηθισμένα queries της sql αλλά η σύνταξη είναι εντελώς διαφορετική, όπως και οι εντολές που υποστηρίζονται.

Για να γίνουν τα παραπάνω, πρέπει πρώτα να κάνει publish ο GeoServer τους πίνακες της βάσης ως layers, δίνοντας τις απαραίτητες πληροφορίες. Επίσης, ο χρήστης μπορεί να καταχωρίσει λέξεις-κλειδιά για τους πίνακες ή μεταδεδομένα, ή να δημιουργήσει δικά του στυλ για την απεικόνιση των δεδομένων.

Η όλη διαδικασία συνδεσης είναι πολύ απλή και γρήγορη καθώς την κάνει ιδιαίτερα εύκολη το γραφικό περιβάλλον του λογισμικού. Παρακάτω, φαίνεται σε βήματα, ο τρόπος διασύνδεσης της βάσης δεδομένων με το Geoserver.

The screenshot shows the GeoServer interface. On the left is a navigation menu with sections: **About & Status** (Server Status, GeoServer Logs, Contact Information, About GeoServer), **Data** (Layer Preview, Workspaces, Stores, Layers, Layer Groups, Styles), **Services** (WCS, WFS, WMS), and **Settings** (Global, JAI, Coverage Access). The main content area is titled 'Welcome' and contains the following information:

- 3 Layers (Add layers)
- 1 Stores (Add stores)
- 1 Workspaces (Create workspaces)

Service Capabilities:

WCS	1.0.0
	1.1.1
WFS	1.0.0
	1.1.0
	2.0.0
WMS	1.1.1
	1.3.0
TMS	1.0.0
WMS-C	1.1.1
WMTS	1.0.0

Security warnings:

- Please read the file C:\Program Files\GeoServer 2.2.1\data_dir\security\masterpw.info and remove it afterwards. This file is a **security risk**.
- Please remove the file C:\Program Files\GeoServer 2.2.1\data_dir\security\users.properties.old because it contains user passwords in plain text. This file is a **security risk**.
- The default user/group service should use digest password encoding.
- The administrator password for this server has not been changed from the default. It is **highly** recommended that you change it now. [Change it](#)
- No strong cryptography available, installation of the unrestricted policy jar files is recommended

Εικόνα 7. GeoServer – Αρχική σελίδα

Βήμα 1ο : Αφού γίνει σύνδεση στο Geoserver (username = admin, password=geoserver οι default τιμές) πρέπει να οριστεί ένα νέο Workspace. Αυτό γίνεται επιλέγοντας **Workspaces** από την αντίστοιχη επιλογή στο μενού Data στα αριστερά και στη συνέχεια το κουμπί **Add new Workspace** όπως φαίνεται στις εικόνες 42α και 42β.

The screenshot shows the GeoServer 'Workspaces' page. The navigation menu is the same as in the previous screenshot. The main content area is titled 'Workspaces' and contains the following information:

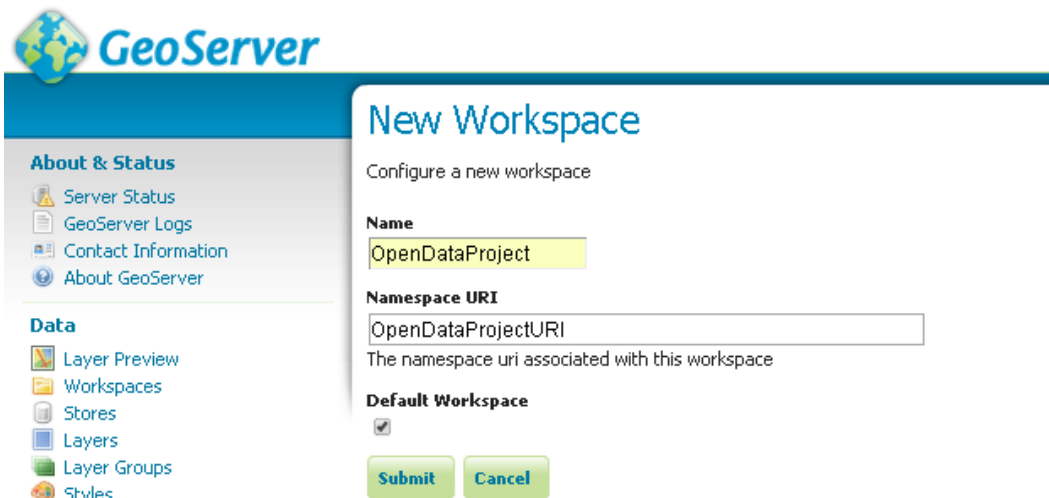
- Manage GeoServer workspaces
- Add new workspace
- Remove selected workspace(s)

Results 1 to 1 (out of 1 items)

Workspace Name
perinpx002

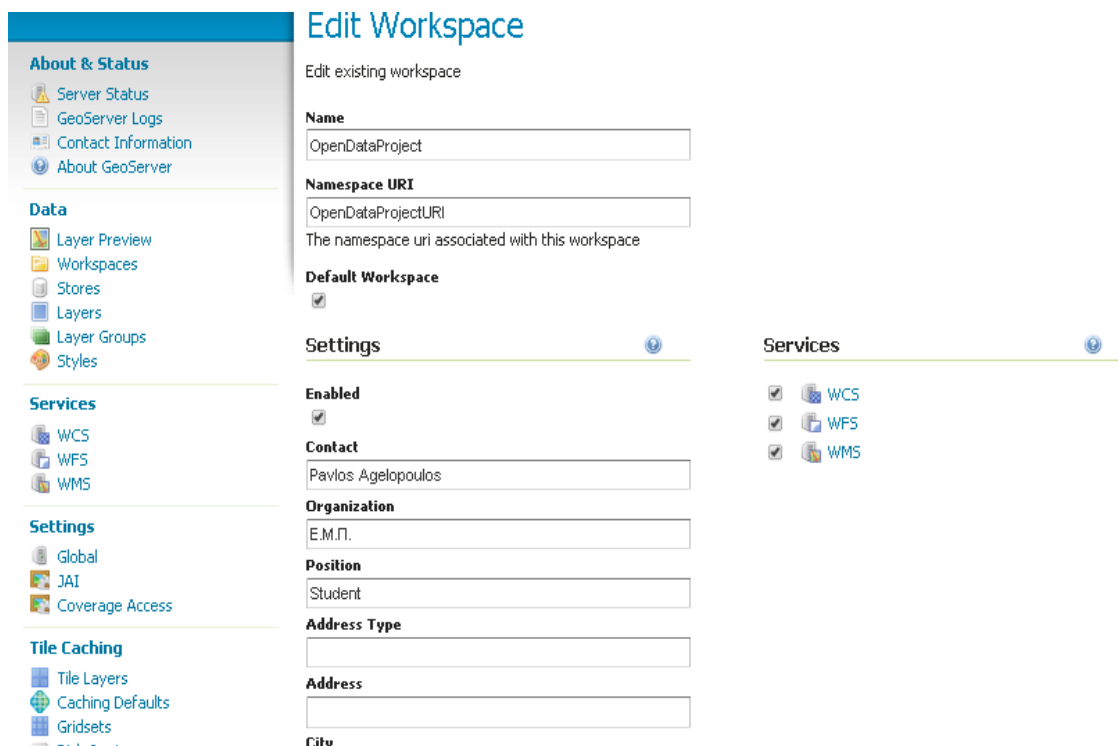
Results 1 to 1 (out of 1 items)

Εικόνα 8. GeoServer – Δημιουργία νέου Workspace



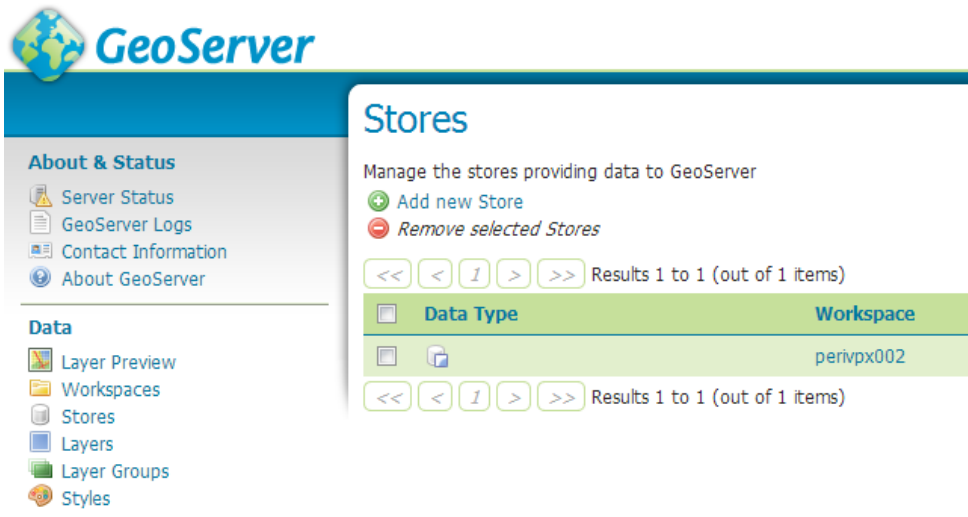
Εικόνα 9. GeoServer – Δημιουργία νέου Workspace

Στη συνέχεια, κάνοντας κλικ στο νέο Workspace που δημιουργήθηκε - ώστε να κάνει Edit - ο χρήστης μπορεί να το ενεργοποιήσει και να επιλέξει τις υπηρεσίες (Services) που επιθυμεί να υποστηρίζονται όπως WMS, WFS, WCS (Εικόνα 43). Επίσης, σε αυτή τη σελίδα μπορεί να επιλέξει την κωδικοποίηση χαρακτήρων (UTF-8) αλλά και να προσθέσει τις προσωπικές του πληροφορίες. Αφού συμπληρώσει τα απαιτούμενα πεδία (τα περισσότερα είναι προαιρετικά) πατάει Save για να αποθηκεύσει τις αλλαγές.

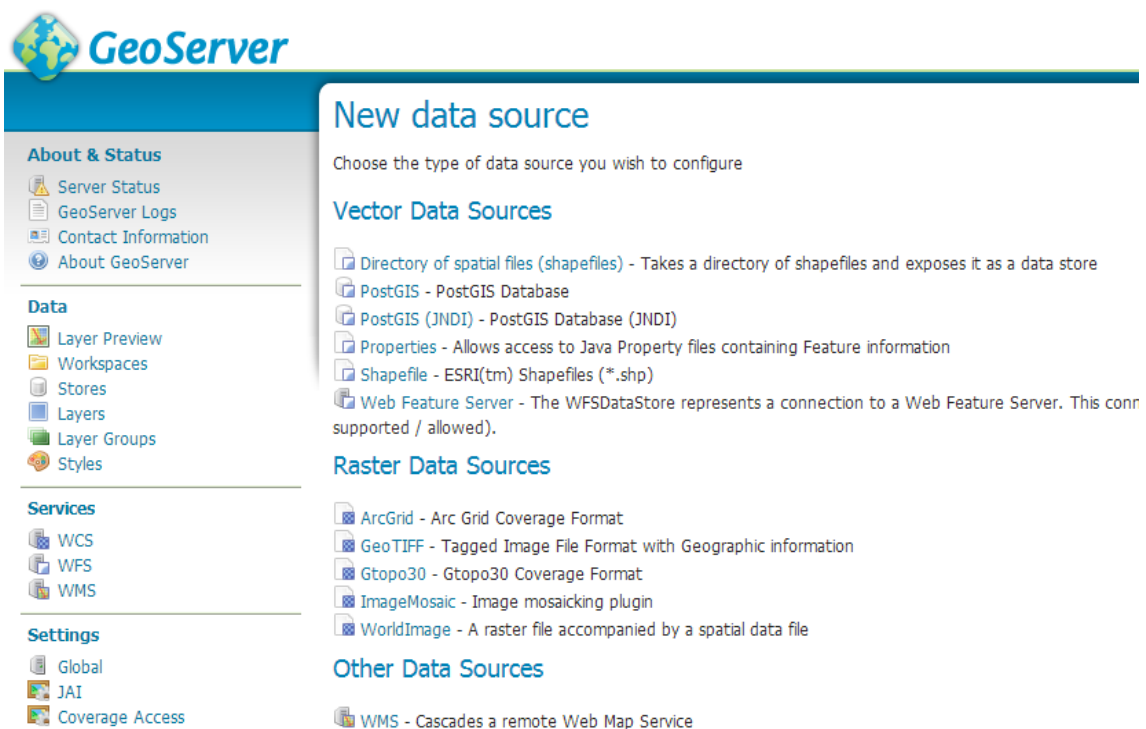


Εικόνα 10. GeoServer – Ρυθμίσεις παραμέτρων του νέου Workspace

Βήμα 2ο : Στη συνέχεια, επιλέγοντας από το μενού Data την επιλογή **Stores**, με το κουμπί **Add new Store** (Εικόνα 44) ο χρήστης καλείται να επιλέξει την προέλευση των δεδομένων. Από τις διαθέσιμες επιλογές για τη συγκεκριμένη εργασία επιλέγει από την κατηγορία Vector Data Sources την επιλογή **PostGIS – PostGIS Database** (Εικόνα 45). Στη συνέχεια, όπως φαίνεται στην Εικόνα 46 καλείται να συμπληρώσει τις απαραίτητες παραμέτρους ώστε να συνδεθεί με τη βάση. Για την λειτουργία της εφαρμογής είναι απαραίτητο να ονομάσει το Workspace ως **OpenDataProject**, και να επιλέξει την επιλογή **Expose primary keys** καθώς χρησιμοποιούνται οι τιμές αυτές από τον κώδικα JavaScript. Αφού συμπληρώσει τα απαραίτητα, πατάει Save.



Εικόνα 11. GeoServer – Δημιουργία νέου Data Store



Εικόνα 12. GeoServer – Επιλογή πηγής δεδομένων

New Vector Data Source

Add a new vector data source

PostGIS
PostGIS Database

Basic Store Info

Workspace *
OpenDataProject ▼

Data Source Name *
OpenDataProject

Description
Διπλωματική Ε.Μ.Π.

Enabled

Connection Parameters

host *
localhost

port *
5432 port

database
GeoDB

schema
mydb

user *
postgres

passwd

Namespace *
OpenDataProjectURI
 Expose primary keys

max connections
10

min connections
1

fetch size
1000

Connection timeout
20

validate connections

Primary key metadata table

Session startup SQL

Session close-up SQL

Loose bbox

Estimated extends

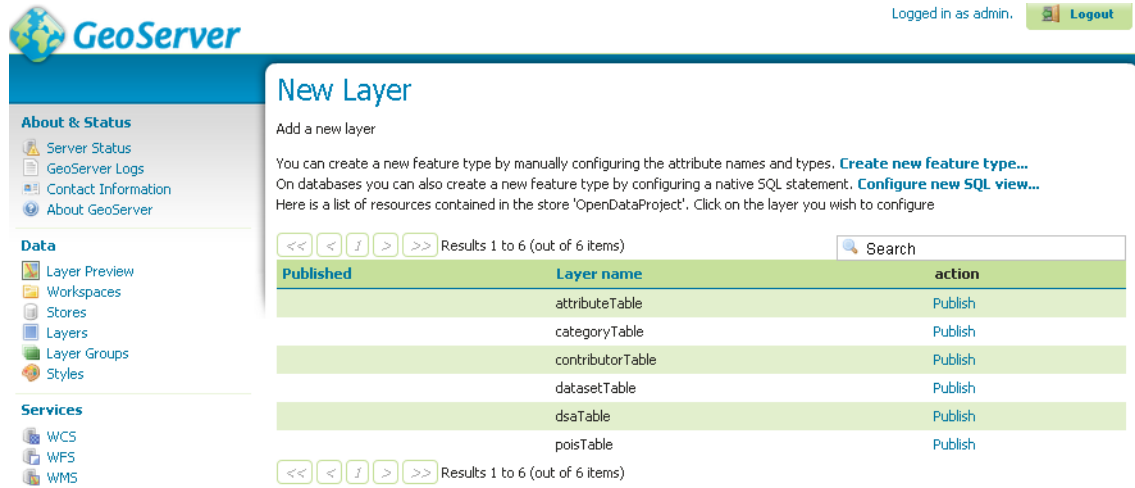
preparedStatements

Max open prepared statements
50

encode functions

Εικόνα 13. GeoServer – Ρυθμίσεις παραμέτρων σύνδεσης στη βάση δεδομένων

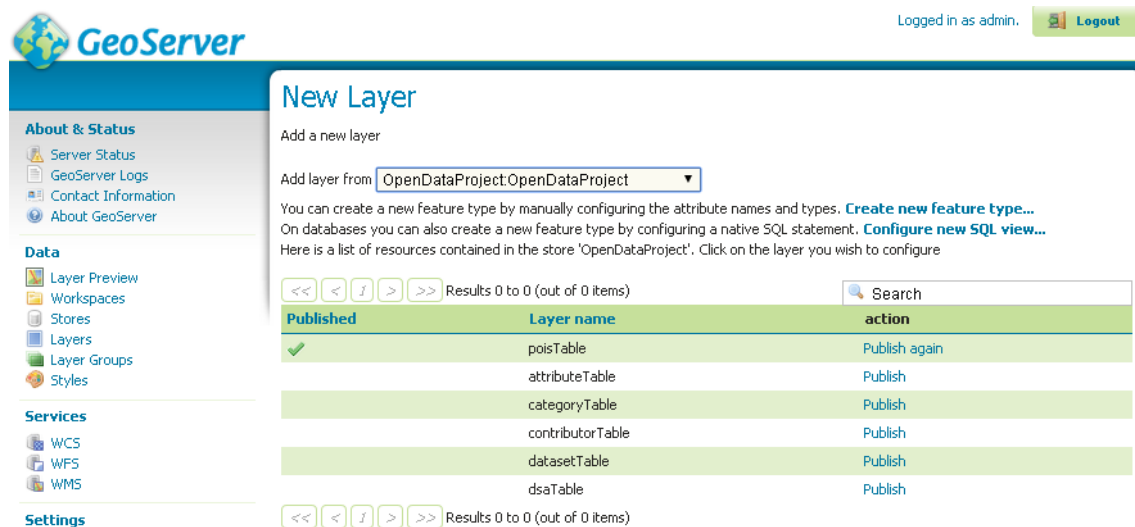
Βήμα 3ο :Στη συνέχεια, ο χρήστης καλείται να επιλέξει ποιους από τους πίνακες της βάσης δεδομένων ενδιαφέρει να εισάγει στον GeoServer (Εικόνα 47α). Στην συγκεκριμένη περίπτωση όλοι οι πίνακες πρέπει να γίνουν publish. Αυτό μπορεί να γίνει επίσης από την επιλογή **Add a New Resource** από το μενού Layers όπως φαίνεται στην Εικόνα 47β παρακάτω



The screenshot shows the 'New Layer' page in GeoServer. The left sidebar contains navigation menus for 'About & Status', 'Data', and 'Services'. The main content area is titled 'New Layer' and includes instructions on how to create a new feature type. Below the instructions is a table listing resources from the 'OpenDataProject' store. The table has three columns: 'Published', 'Layer name', and 'action'. All resources are marked as 'Published' and have a 'Publish' action button.

Published	Layer name	action
	attributeTable	Publish
	categoryTable	Publish
	contributorTable	Publish
	datasetTable	Publish
	dsaTable	Publish
	poisTable	Publish

Εικόνα 14. Add new layer



The screenshot shows the 'New Layer' page in GeoServer. The left sidebar is the same as in the previous image. The main content area is titled 'New Layer' and includes instructions on how to create a new feature type. Below the instructions is a dropdown menu for 'Add layer from' set to 'OpenDataProject:OpenDataProject'. Below the dropdown is a table listing resources from the 'OpenDataProject' store. The table has three columns: 'Published', 'Layer name', and 'action'. The 'poisTable' resource is selected, indicated by a green checkmark in the 'Published' column and a 'Publish again' action button. Other resources have 'Publish' action buttons.

Published	Layer name	action
✓	poisTable	Publish again
	attributeTable	Publish
	categoryTable	Publish
	contributorTable	Publish
	datasetTable	Publish
	dsaTable	Publish

Εικόνα 15. Add new layer

Στη συνέχεια, πρέπει να γίνει ρύθμιση παραμέτρων για κάθε layer. Αυτό αφορά κυρίως το Σύστημα Αναφοράς (Coordinate Reference System) και το Bounding Box και έχουν χρησιμότητα σε layers για τα οποία υπάρχει χωρική πληροφορία (συντεταγμένες). Όσον αφορά το πρώτο, ο GeoServer δίνει τη δυνατότητα επιλογής από ένα πολύ μεγάλο πλήθος διαφορετικών συστημάτων αναφοράς, τόσο για το εγγενές σύστημα (Native) όσο και για δεδηλωμένο (Declared), ώστε να είναι δυνατόν on-the-fly transformation. Στη συγκεκριμένη εργασία – για τον πίνακα poisTable – επιλέγεται ως σύστημα αναφοράς το WGS84 με κωδικό EPSG:4326. Το Bounding Box αποτελεί ένα παραλληλόγραμμο μέσα στο οποίο περικλείονται όλα τα στοιχεία του layer, και ο GeoServer μπορεί να το υπολογίσει αυτόματα με την επιλογή **Compute from data**. Σύστημα Αναφοράς και Bounding Box είναι απαραίτητο να οριστούν ακόμα και για πίνακες για τους οποίους δεν είναι διαθέσιμη χωρική πληροφορία για να μπορέσει ο GeoServer να κάνει publish τα layer. Σε αυτή την περίπτωση –για τους υπόλοιπους

πίνακες – ως σύστημα αναφοράς ορίστηκε και πάλι το WGS84 ενώ η επιλογή Compute from data δίνει άκυρες τιμές (min X,Y = -1, max X,Y =0). Στις Εικόνες 48α και 48β παρακάτω φαίνεται η ρύθμιση των παραμέτρων για το layer poisTable των καρτελών Data και Publishing.

Edit Layer
Edit layer data and publishing

OpenDataProject:poisTable
Configure the resource and publishing information for the current layer

Data | Publishing | Dimensions | Tile Caching

Basic Resource Info

Name
poisTable

Title
poisTable

Abstract

Keywords

Current Keywords
features
poisTable
Remove selected

New Keyword

Vocabulary
Add Keyword

Metadata links
No metadata links so far
Add link *Note only FGDC and TC211 metadata links show up in WMS 1.1.1 capabilities*

Coordinate Reference Systems

Native SRS
EPSG:4326 [EPSG:WGS 84...](#)

Declared SRS
EPSG:4326 [Find... EPSG:WGS 84...](#)

SRS handling
Force declared

Bounding Boxes

Native Bounding Box

Min X	Min Y	Max X	Max Y
19,92003	35,00899	28,22324	41,50319

[Compute from data](#)

Lat/Lon Bounding Box

Min X	Min Y	Max X	Max Y
19,92003	35,00899	28,22324	41,50319

[Compute from native bounds](#)

Feature Type Details

Property	Type	Nullable	Min/Max Occurrences
poisid	Integer	false	1/1
poisdescr	String	true	0/1
poisdataset	String	true	0/1
poisgeom	Point	true	0/1
poidataset_id	Integer	false	1/1

[Reload feature type](#) ⚠ ...

Save Cancel

Εικόνα 16. Ρύθμιση παραμέτρων της καρτέλας Data για το layer poisTable

Edit Layer
Edit layer data and publishing

OpenDataProject:poisTable
Configure the resource and publishing information for the current layer

Data | **Publishing** | **Dimensions** | **Tile Caching**

Edit Layer

Name
poisTable

Enabled

Advertised

HTTP Settings

Response Cache Headers

Cache Time (seconds)
[input field]

WFS Settings

Per-Request Feature Limit
0

Maximum number of decimals
0

WMS Settings

Queryable

Default Style
point

Additional Styles

Available Styles	Selected Styles
<ul style="list-style-type: none"> burg capitals cite_lakes dem giant_polygon grass green line poi point 	

Default Rendering Buffer
[input field]

Default WMS Path
[input field]

Attribution Text
[input field]

Attribution Link
[input field]

Logo URL
[input field]

Logo Content Type
[input field]

Logo Image Width
0

Logo Image Height
0

[Auto-detect image size and type](#)

KML Format Settings

Default Regionating Attribute
Choose One

Default Regionating Method
Choose One

Features Per Regionated Tile
[input field]

Save **Cancel**

Εικόνα 17. Ρύθμιση παραμέτρων της καρτέλας Publishing για το layer poisTable

3.2.4 Σχεδιασμός της διεπαφής χρήστη (User Interface)

Μετά τη διασύνδεση της βάσης δεδομένων με το GeoServer ήταν απαραίτητο να δημιουργηθεί γραφικό περιβάλλον τέτοιο ώστε να είναι εύκολο για κάθε χρήστη, ακόμα και για αυτούς που δεν είναι ιδιαίτερα εξοικειωμένοι με τη χρήση νέων τεχνολογιών, να χρησιμοποιούν την εφαρμογή. Επίσης, κρίθηκε σκόπιμο, η εφαρμογή να είναι ανεξάρτητη από τα λειτουργικά συστήματα ώστε να είναι συμβατή με όλες τις συσκευές οι οποίες έχουν πρόσβαση στο διαδίκτυο, συμπεριλαμβανομένων smartphones, tablets, ή laptops και pc.

Ο καλύτερος τρόπος για να γίνει αυτό είναι μέσω τεχνολογιών Web, οι οποίες είναι platform-independent και ραγδαία αναπτυσσόμενες. Παρόλαυτά όμως έπρεπε να αντιμετωπιστούν προβλήματα σχετικά με την συμβατότητα των διαφόρων web browsers. Το γραφικό περιβάλλον σχεδιάστηκε εξ ολοκλήρου με HTML5 ενώ οι αλγόριθμοι λειτουργίας με JavaScript και όπου ήταν απαραίτητη η αλληλεπίδραση με τη βάση δεδομένων με php. Την αρχική σελίδα της εφαρμογής αποτελούν τρία αρχεία τα οποία βρίσκονται αποθηκευμένα στο φάκελο www του GeoServer. Το βασικό αρχείο είναι το OpenDataProject.html το οποίο καλεί τα OpenDataProject.js και OpenDataProject.css. Επίσης, η εφαρμογή καλεί μία σειρά εξωτερικών αρχείων και βιβλιοθήκες από το web.

Πιο σημαντική από αυτές είναι η βιβλιοθήκη openlayers, που αποτελεί project του Open Source Geospatial Foundation. Η συγκεκριμένη είναι γραμμένη σε JavaScript, διατίθεται ελεύθερα και προσφέρει στους προγραμματιστές πλήθος εργαλείων για χαρτογραφικές εφαρμογές. Μεταξύ άλλων, παρέχει έτοιμες προς χρήση συναρτήσεις για μετασχηματισμούς συντεταγμένων και υπολογισμούς σε χαρτογραφικές προβολές, ενώ υποστηρίζει σχεδόν το σύνολο των γνωστών προβολικών συστημάτων παγκοσμίως. Ακόμα, προσφέρει έτοιμα πολλά controls για χάρτες όπως για παράδειγμα zoom bars και υπομνηματα για τα layers που γίνονται overlay σε χαρτογραφικά υπόβαθρα.

Επίσης, καλούνται βιβλιοθήκες οι οποίες προσφέρουν διαδικτυακούς, δυναμικούς χάρτες, όπως είναι οι Open Street Map, GoogleMaps και GoogleSatellite. Εκτός από τα παραπάνω, καλούνται αρχεία και βιβλιοθήκες τα οποία χρησιμοποιούνται στο site του ελληνικού παραρτήματος του Open Knowledge Foundation. Παρακάτω φαίνεται το απόσπασμα του κώδικα στο αρχείο OpenDataProject.html όπου γίνονται οι κλήσεις προς εξωτερικά αρχεία και βιβλιοθήκες

```
<link rel='stylesheet' id='admin-bar-css'  
href='http://okfn.gr/wp-includes/css/admin-bar.min.css?ver=3.6'  
type='text/css' media='all'  
>  
<link rel='stylesheet' id='bp-admin-bar-css'  
href='http://okfn.gr/wp-content/plugins/buddypress/bp-core/css/admin-  
bar.css?ver=1.6.4' type='text/css' media='all'  
>
```

```

<link rel='stylesheet' id='okf-panel-master-frontend-css'
href='http://okfn.gr/wp-content/plugins/okf-panel-
master/assets/css/frontend.css' type='text/css' media='all'
/>

<link rel='stylesheet' id='okf-wordpress-footer-frontend-css'
href='http://okfn.gr/wp-content/plugins/okf-wordpress-
footer/assets/css/frontend.css' type='text/css' media='all'
/>

<!-- nagios_check timestamp:1399826556 -->
<!-- nagios_db_check sitename:OKFN Greece -->
<link rel="stylesheet" href=http://okfn.gr/wp-content/themes/wordpress-theme-
okfn/style.css?1391101935
/>

<!-- Theme Settings -->
<link rel="stylesheet" href=http://okfn.gr/wp-content/themes/wordpress-theme-
okfn/css/sub-header.css?ver=1
/>

<link rel="stylesheet" href=http://okfn.gr/wp-content/themes/wordpress-theme-
okfn/css/grey.css?ver=1.1.2
/>

<link rel="shortcut icon" href=http://assets.okfn.org/p/okfn/img/favicon.ico
/>

<!-- OpenLayers -->
<script src="http://openlayers.org/dev/OpenLayers.js">
</script>

<!-- OpenStreetMap base layer js -->
<script src="http://www.openstreetmap.org/openlayers/OpenStreetMap.js">
</script>

<!-- Google Maps -->
<script src="http://maps.google.com/maps/api/js?sensor=false">
</script>

<!-- myscript -->
<script src="OpenDataProject.js" defer="defer" type="text/javascript">
</script>

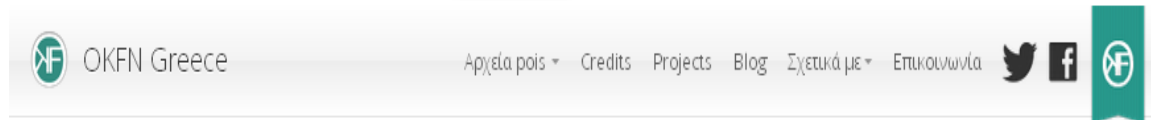
<!-- mycss -->
<link rel="stylesheet" href="OpenDataProject.css"/>

```

Εδώ πρέπει να αναφερθεί πως η εφαρμογή εμφωλεύθηκε στη σελίδα του site του Open Knowledge Foundation του παραρτήματος Ελλάδος. Διατηρήθηκε δηλαδή το menu της σελίδας του εν λόγω site και προστέθηκαν σε αυτή οι λειτουργίες της εφαρμογής. Έτσι, όλες οι υπόλοιπες επιλογές του menu (πλην της επιλογής Αρχεία pois) δείχνουν στους συνδεσμούς του site Open Knowledge Foundation. Αυτό συμβαίνει χωρίς η σελίδα της εφαρμογής να είναι

αποθηκευμένη στο Server που βρίσκεται το site, απλώς έχει χρησιμοποιηθεί το menu bar της σελίδας και οι υπερσυνδεσμοί στις υπόλοιπες επιλογές δείχνουν εκεί.

Στην παρακάτω εικόνα φαίνεται το menu επιλογών της αρχικής σελίδας όπως εμφανίζεται στην εφαρμογή.



Εικόνα 18. Menu bar του site Open Knowledge Foundation του παραρτήματος Ελλάδος

Η αρχική σελίδα της εφαρμογής, εκτός από το header της σελίδας και το footer, αποτελείται από ένα τμήμα (div με id="MyDiv") το οποίο περιέχει τρία τμήματα (divs) τα οποία παρατίθενται πλάι-πλάι. Το πρώτο από αυτά, (με id="Datasets"), καταλαμβάνει το 25% του πλάτους της σελίδας και περιέχει τα διαθέσιμα datasets που είναι αποθηκευμένα στη βάση δεδομένων. Παρακάτω φαίνεται απόσπασμα από το αρχείο OpenDataProject.html όπου ορίζεται το συγκεκριμένο τμήμα.

```
<div id="Datasets" style="width:25%;height:100%;float:left;overflow:auto;">
  <b>Datasets</b>
</div>
```

Στο τμήμα αυτό αναγράφονται τα datasets που ανακτώνται από το GeoServer μέσω της τεχνικής AJAX. Πιο συγκεκριμένα, έχει γραφεί η συνάρτηση fillDatasetDiv στο αρχείο OpenDataProject.js η οποία κάνει XmlHttpRequests και αφού επεξεργαστεί την απόκριση του GeoServer εμφανίζει εκεί το εικονίδιο του dataset (αν έχει ανεβάσει τέτοιο χρήστης που το έκανε upload), το όνομα και την περιγραφή του. Αυτό γίνεται εφικτό μέσω της υπηρεσίας WFS (Web Feature Service). Η συγκεκριμένη συνάρτηση καλείται με τρεις παραμέτρους:

- το url που γίνονται τα WFS requests,
- το όνομα του layer που γίνονται τα requests και αντιστοιχεί στον πίνακα της βάσης,
- ένα Array στο οποίο περιέχονται ως ετικέτες τα ονόματα των πεδίων που θα εμφανίζονται πριν τις τιμές.

Πιο κάτω φαίνεται η κλήση και ο ορισμός της εν λόγω συνάρτησης.

```
fillDatasetDiv(url, datasetLayer, datasetTabCols);
```

```
function fillDatasetDiv(url, DbTab, TabColsArr) {
  var xmlDoc;
  var request;

  if(window.XMLHttpRequest){ // code for IE7+, Firefox, Chrome, Opera, Safari
    request = new XMLHttpRequest();
  } else // code for IE6, IE5
    request=new ActiveXObject("Microsoft.XMLHTTP");
```

```

}
request.open("GET", url, true);
request.onreadystatechange = function() {
    if (request.readyState == 4 && request.status == 200) {

        var txt = request.responseText;
        if (window.DOMParser){
            parser=new DOMParser();
            xmlDoc=parser.parseFromString(txt,"text/xml");
        }else {
            // Internet Explorer
            xmlDoc=new ActiveXObject("Microsoft.XMLDOM");
            xmlDoc.async=false;
            xmlDoc.loadXML(txt);
        }

        var DbTabFull= layerNamespace+": "+DbTab;
        var x=xmlDoc.getElementsByTagName(DbTabFull);
        var htmlText="<form>";

        if (x.length == 0) {
            //Kwdikas gia Chrome

            x = xmlDoc.getElementsByTagName(DbTab);
            for (var i=0;i<x.length;i++){
                htmlText=htmlText+('<fieldset style="border-
color:#585858;border-style:groove;border-width:5px;padding:5;">');

                var fid=x[i].getAttribute("fid");
                fid=fid.split(".")[1];

                htmlText= htmlText+imgDiv+
(x[i].getElementsByTagName(datasetpoiicon)[0].childNodes[0].nodeValue)+
'"alt="Not Available" width="20" height="20" style="display:block; margin:
auto;">' ;

                for(var ii=0;ii<TabColsArr.length;ii++){

                    htmlText=htmlText+ "<b>" +
datasetTabLabels[ii] + "</b>";

                    htmlText = htmlText +
(x[i].getElementsByTagName(TabColsArr[ii])[0].childNodes[0].nodeValue);

                    htmlText = htmlText + ("<hr
style='margin-top:0;margin-bottom:0;'>");
                }//end loop Columns

                var name =
x[i].getElementsByTagName(TabColsArr[0])[0].childNodes[0].nodeValue;

                htmlText = htmlText+'<input type="checkbox"
name="'+name+'" id="'+ fid + '"onclick=handleClick(this);> <b>Εμφάνιση στο
χάρτη</b><br>';

                htmlText = htmlText + ("</fieldset>");

            } //End looping xmlDocResponse

```

```

    }else { //Kwdikas gia Firefox

        for (var i=0;i<x.length;i++){
            htmlText = htmlText + ('<fieldset style="border-
color:#585858;border-style:groove;border-width:5px;padding:5">');

            var fid = x[i].getAttribute("fid");
            fid = fid.split(".")[1];
            htmlText = htmlText + imgDiv +
(x[i].getElementsByName(layerNamespace+"."+datasetpoiicon)[0].childNodes[0
].nodeValue)+"alt="Not Available" width="20" height="20"
style="display:block; margin: auto;">' ;

            for(var ii=0;ii<TabColsArr.length;ii++){

                htmlText = htmlText +
"<b>"+datasetTabLabels[ii]+ "</b>";

                htmlText = htmlText +
(x[i].getElementsByName(layerNamespace+"."+TabColsArr[ii])[0].childNodes[0
].nodeValue);

                htmlText = htmlText + ("<hr
style='margin-top:0;margin-bottom:0;'>");
            }//end loop Columns

            var name =
x[i].getElementsByName(layerNamespace+"."+TabColsArr[0])[0].childNodes[0].
nodeValue;

            htmlText = htmlText +'<input type="checkbox"
name="'+name+'" id="'+ fid + '"onclick=handleClick(this);><b>Εμφάνιση στο
χάρτη</b><br>';

            htmlText = htmlText + ("</fieldset>");

        }//End looping xmlDocResponse

    }//end_elegxos Chrome/Firefox

    htmlText = htmlText + ("</form>");
    document.getElementById('Datasets').innerHTML = htmlText;

} //end if ola ok
}; //ENDrequest.onload

request.send(null);
} //endfunction fillDatasetDiv

```

Το δεύτερο κατά σειρά τμήμα, το οποίο καταλαμβάνει το 50% της σελίδας, περιέχει το χάρτη, καθώς και τρία άλλα υποτμήματα που εμφανίζουν τις συντεταγμένες της θέσης του ποντικιού (λ,φ γωνίες στο γεωκεντρικό σύστημα αναφοράς WGS'84) και την κλίμακα θέασης του χάρτη. Ακολουθεί ο ορισμός τους στο αρχείο html.

```
<div id="content" style="width:50%;height:100%;float:left;">
```

```

ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΔΙΑΘΕΣΗΣ ΓΕΩΧΩΡΙΚΩΝ ΔΕΔΟΜΕΝΩΝ
CROWD-SOURCING ΜΕ ΤΕΧΝΟΛΟΓΙΕΣ ΑΝΟΙΚΤΩΝ ΔΕΔΟΜΕΝΩΝ.

```

```

<div id="map"></div>
<div id="wrapper">
  <div id="location"> location</div>
  <div id="lf">λ,φ :</div>
  <div id="scale"></div>
</div>
</div>

```

Η προεπιλογή για τον χάρτη υποβάθρου είναι χάρτης Open Street Map (OSM) αλλά ο χρήστης μπορεί να επιλέξει και μεταξύ Google Maps Google Satellite. Επίσης, στο χάρτη χρησιμοποιούνται και widgets από τη βιβλιοθήκη openlayers.

Τέλος, το τρίτο τμήμα το οποίο καταλαμβάνει το υπολοιπούμενο 25% του πλάτους της σελίδας, δημιουργείται δυναμικά εφόσον ο χρήστης κάνει κλικ πάνω σε κάποιο από τα σημεία ενδιαφέροντος και εμφανίζει τις πληροφορίες που περιέχονται στη βάση δεδομένων για το συγκεκριμένο σημείο. Αυτό πραγματοποιείται μέσω της συνάρτησης GetFeatureInfo που προσφέρει το WMS (Web Mapping Service) μέσω του GeoServer, ενώ έχει γραφεί μία συνάρτηση JavaScript η οποία παίρνει την απόκριση και αφού την επεξεργαστεί την εμφανίζει στο συγκεκριμένο τμήμα. Παρακάτω φαίνεται το συγκεκριμένο τμήμα στο αρχείο html.

```

<div id="resultsWfs" style="width:25%;height:100%;float:left;overflow:auto;">
  <div id="nodelist"> <em>Click on the map to get feature info</em></div>
</div>

```

Η συνάρτηση, η οποία εμφανίζει τα στοιχεία που αφορούν το σημείο ενδιαφέροντος πάνω στο οποίο έκανε κλικ ο χρήστης, παίρνει ως παράμετρο την απόκριση του GeoServer και παρουσιάζεται παρακάτω.

```

function setHTML(response) {

  var txt=response.responseText;
  var xmlDoc;

  if (window.DOMParser){
    parser=new DOMParser();
    xmlDoc=parser.parseFromString(txt,"text/xml");
  } else { // Internet Explorer
    xmlDoc=new ActiveXObject("Microsoft.XMLDOM");
    xmlDoc.async=false;
    xmlDoc.loadXML(txt);
  }

  var DbTab = "poisTable";
  var DbTabFull= layerNamespace+": "+DbTab;
  var htmlText="<table border='1' style='border-color:#585858;border-style:groove;border-width:5px;padding:0'>";
  var x=xmlDoc.getElementsByTagName(DbTabFull);

```

```

        if (x.length == 0) {
            x = xmlDoc.getElementsByTagName(DbTab);
            for (var i=0;i<x.length;i++){
                htmlText=htmlText+("<tr padding:0><td>");

htmlText=htmlText+(x[i].getElementsByTagName(poisColumn1)[0].childNodes[0].no
deValue);
                htmlText=htmlText+("</td><td>");
                htmlText = htmlText +
(x[i].getElementsByTagName(poisColumn2)[0].childNodes[0].nodeValue);
                htmlText = htmlText+("</td></tr>");
            }
            htmlText = htmlText+("</table>");
        }else {
            for (var i=0;i<x.length;i++){
                htmlText = htmlText+("<tr padding:0><td>");
                htmlText = htmlText +
(x[i].getElementsByTagName(layerNamespace+": "+poisColumn1)[0].childNodes[0].n
odeValue);
                htmlText = htmlText+("</td><td>");
                htmlText = htmlText +
(x[i].getElementsByTagName(layerNamespace+": "+poisColumn2)[0].childNodes[0].n
odeValue);
                htmlText = htmlText+("</td></tr>");
            }

        }

    } //end_elegxos Chrome/Firefox

    htmlText = htmlText+("</table>");
    document.getElementById('nodelist').innerHTML = htmlText;

}; //end_setHTML

```

3.2.5 Ενημέρωση της Βάσης Δεδομένων

Εκτός από την αρχική σελίδα, έχουν δημιουργηθεί άλλες δύο ιστοσελίδες για την εφαρμογή. Η πρώτη, αφορά ανέβασμα (upload) και η δεύτερη κατέβασμα (download) δεδομένων από τους χρήστες. Οι δύο αυτές σελίδες δεδομένου ότι σχετίζονται με τη βάση δεδομένων περιέχουν κώδικα php και είναι αποθηκευμένες στο φάκελο www του Server της Postgres (Apache EnterpriseDB).

Ο κώδικας για τη λειτουργία upload περιέχεται σε τέσσερα διαφορετικά αρχεία, τα OpenDataUploadForm.php, OpenDataUploadForm.css, OpenDataDBInserts.php και FileReader.js. Το βασικό αρχείο, αυτό στο οποίο κατευθύνεται ο χρήστης προκειμένου να ανεβάσει στη βάση δεδομένα και με τη σειρά του καλεί όλα τα υπόλοιπα, είναι το πρώτο. Στο συγκεκριμένο αρχείο περιέχονται κυρίως ετικέτες HTML5 όπου ο χρήστης καλείται να συμπληρώσει τις απαραίτητες πληροφορίες για τη βάση δεδομένων. Ο κώδικας php αφορά απλώς και μόνο στην ανάκτηση από τη βάση των ήδη καταχωρισμένων κατηγοριών datasets για την περίπτωση που αυτό που θέλει να αναβάσει εντάσσεται σε κάποιο από αυτά.

Πιο συγκεκριμένα, η ιστοσελίδα upload χωρίζεται σε δύο τμήματα, ένα για τα στοιχεία που αφορούν τον uploader και ένα δεύτερο για τα στοιχεία που αφορούν το dataset. Για λόγους ευκολίας, αρχικά στην σελίδα εμφανίζονται μόνο τα απαραίτητα πεδία που πρέπει να συμπληρώσει ο χρήστης ώστε να ανεβάσει δεδομένα. Τα προαιρετικά είναι σε divs τα οποία είναι κρυφά (attribute hidden) και εμφανίζονται μόνο στη περίπτωση που το επιθυμεί ο χρήστης. Αυτό γίνεται με τη βοήθεια συναρτήσεων που είναι γραμμένες στο αρχείο JavaScript FileReader.js.

Τα στοιχεία που αφορούν τον uploader είναι αυτά που εκτέθηκαν αναλυτικά στο κεφάλαιο που παρουσιάστηκε ο πίνακας contributorTable της βάσης δεδομένων. Η επιλογή μεταξύ των απαραίτητων και προαιρετικών πληροφοριών γίνεται με τη βοήθεια ενός HTML <select> tag κάθε αλλαγή του οποίου ενεργοποιεί τη συνάρτηση personalDivChange() η οποία εμφανίζει ή αποκρύπτει τα κρυφά πεδία.

Αντίστοιχα, λειτουργεί και το τμήμα που αφορά τις πληροφορίες για το dataset. Σε αυτή την περίπτωση ο χρήστης πρέπει να συμπληρώσει τα απαραίτητα στοιχεία για το σύνολο δεδομένων (όπως αυτά εκτέθηκαν στο πίνακα datasetTable στο αντίστοιχο κεφάλαιο) και επιλέγει την κατηγορία στην οποία εντάσσεται. Έτσι, γίνεται ερώτηση στη βάση δεδομένων με κώδικα rhr (αυτός είναι και ο λόγος που το αρχείο έχει κατάληξη rhr) και τα αποτελέσματα εμφανίζονται σε ένα select tag ώστε να επιλέξει. Η προεπιλογή είναι Χωρίς Κατηγορία, ενώ υπάρχει και η επιλογή Νέα Κατηγορία, για την περίπτωση που θέλει να καταχωρίσει καινούργια. Τότε εμφανίζεται ένα textbox όπου γράφει το όνομα της νέας κατηγορίας το οποίο αποθηκεύεται στη βάση. Για τη λειτουργία της εφαρμογής, η επιλογή Νέα Κατηγορία είναι απαραίτητο να έχει την τιμή 2 στη στήλη categoryid.

Όσον αφορά το αρχείο pois αυτό υποδεικνύεται με την ετικέτα <input> η οποία έχει την ιδιότητα file για το πεδίο type. Το αρχείο μόλις φορτωθεί διαβάζεται με κώδικα JavaScript και τα περιεχόμενα του αποθηκεύονται σε ένα κρυφό div (DataFromFileDB2) το οποίο έχει δημιουργηθεί για αυτό το σκοπό. Αυτό γίνεται ώστε όταν η φόρμα αποσταλεί να υπάρχουν τα δεδομένα διαθέσιμα για τον κώδικα rhr, έτοιμα προς επεξεργασία. Η επεξεργασία κειμένου είναι σαφώς ευκολότερη με JavaScript απότι με rhr Θεωρείται επίσης ασφαλέστερος τρόπος αποφυγής επιθέσεων κατά της βάσης δεδομένων (injection attacks) από επίδοξους hackers, οι οποίοι θα μπορούσαν να ενσωματώσουν στο αρχείο

Στις προαιρετικές πληροφορίες σχετικά με το dataset συγκαταλέγονται το σύστημα αναφοράς των συντεταγμένων και η μορφή γραμμογράφησης του αρχείου pois. Και αυτά τα στοιχεία βρίσκονται σε κρυφό τμήμα της σελίδας και εμφανίζονται αν ο χρήστης κάνει κλικ στο αντίστοιχο κουμπί.

Η προεπιλογή για το σύστημα αναφοράς είναι το γεωκεντρικό σύστημα WGS'84 όπου οι συντεταγμένες είναι γωνίες της μορφής λ,φ αλλά παρέχεται η δυνατότητα οι συντεταγμένες να δίνονται ως Χ,Ψ στο Ελληνικό Γεωδαιτικό Σύστημα Αναφοράς ΕΓΣΑ'87. Έτσι, αξιοποιείται και η πολύ σημαντική λειτουργία της PostGIS για on the fly μετασχηματισμούς συστημάτων αναφοράς. Η επιλογή γίνεται με radio button όπου το id αντιστοιχεί στο SRID του επιλεγμένου συστήματος και αξιοποιείται από το query που γεμίζει τη βάση.

Όσον αφορά τη γραμμογράφιση του αρχείου, υποστηρίζονται οι παρακάτω μορφές:

- λ , φ , data
- φ , λ , data
- data , λ , φ

Η προεπιλογή είναι κάθε γραμμή να είναι της μορφής λ,φ,data. Το λ αφορά την τετμημένη, το φ την τεταγμένη και τα data ό,τι πληροφορίες περιέχονται στο αρχείο σχετικά με το συγκεκριμένο σημείο ενδιαφέροντος. Και σε αυτή την περίπτωση, η επιλογή γίνεται με τη χρήση radio buttons και η επιλογή μεταφέρεται στο query.

Το τελευταίο προαιρετικό στοιχείο για κάθε dataset αφορά στο εικονίδιο που εμφανίζεται την αρχική σελίδα της εφαρμογής. Υποστηρίζεται πλήθος format, όπως για παράδειγμα png, jpeg, gif κ.α. αλλά υπάρχει περιορισμός στο μέγεθος της εικόνας στα 20KB. Η εικόνα που ανεβάζει ο χρήστης αποθηκεύεται στο φάκελο uploads που πρέπει να βρίσκεται μαζί με τα υπόλοιπα αρχεία της εφαρμογής στο φάκελο www του Apache EnterpriseBD.

Μόλις ο χρήστης συμπληρώσει όλες τις απαραίτητες πληροφορίες (η ιδιότητα required στα HTML tags το εξασφαλίζει αυτό) και κάνει submit τη φόρμα, καλείται το αρχείο OpenDataDBInserts.php. Αυτό, παραλαμβάνει τις τιμές που έχει δώσει ο χρήστης στη φόρμα και συνθέτει τα queries που ενημερώνουν τους πίνακες της βάσης δεδομένων. Δεδομένου ότι οι πίνακες είναι αλληλοσυνδεόμενοι με ξένα κλειδιά, έχουν γραφεί συναρτήσεις σε php οι οποίες μόλις ενημερώσουν τη βάση με εντολές insert επιστρέφουν το αναγνωριστικό της συγκεκριμένης εγγραφής, ώστε αυτό να εισαχθεί σε άλλους πίνακες.

Έτσι, μετά από κάθε insert query αμέσως μετά πραγματοποιείται ένα select max ώστε να ανακτηθεί το id της τελευταίας εγγραφής. Το γεγονός ότι τα πρωτεύοντα κλειδιά των πινάκων είναι serial (δηλαδή ακέραιοι οι οποίοι αυξάνουν αυτόματα κατά ένα σε κάθε νέα εγγραφή) το εξασφαλίζει αυτό. Παρακάτω φαίνεται η συνάρτηση php InsertTable_ReturnFK η οποία καλείται με τρεις παραμέτρους (τη σύνδεση με τη βάση δεδομένων και τα 2 sql statements) και η οποία επιστρέφει το id της εγγραφής.

```
function InsertTable_ReturnFK($db, $sql, $sql1){
    $val="";
    $ret = pg_query($db, $sql);
    if(!$ret){
        echo pg_last_error($db);
        echo '<p>';
    }
    $ret = pg_query($db, $sql1);
    if(!$ret){
        echo pg_last_error($db);
        echo '<p>';
    }else {
        $val = pg_fetch_result($ret, 0, 0);
    }
    return $val;
}
```

Είναι προφανές πως για να λειτουργήσει το πρόγραμμα σωστά πρέπει οι πίνακες να ενημερώνονται με συγκεκριμένη σειρά. Έτσι, αρχικά ενημερώνεται ο πίνακας contributorTable και ανακτάται το αναγνωριστικό της εγγραφής το οποίο αποθηκεύεται σε μεταβλητή για να χρησιμοποιηθεί μετά στο πίνακα datasetTable. Στη συνέχεια, ενημερώνεται ο πίνακας categoryTable και ομοίως αποθηκεύεται το id της εγγραφής.

Ο επόμενος πίνακας ο οποίος ενημερώνεται ο datasetTable ο οποίος χρησιμοποιεί ως ξένα κλειδιά τα id των εγγραφών των δύο προηγούμενων. Το πρόβλημα σε αυτόν τον πίνακα ήταν με το εικονίδιο αφού όταν χρησιμοποιούσε για το path το όνομα του αρχείου της εικόνας και αυτό ήταν με ελληνικούς χαρακτήρες η εικόνα που αποθηκευόταν στο φάκελο upload είχε αλλόκοτους χαρακτήρες και δεν εμφανιζόταν. Για να αντιμετωπιστεί αυτό, η εικόνα μετονομάζεται με βάση κάποια random συνάρτηση και με αυτό το όνομα εισάγεται στον πίνακα (στήλη datasetpoiicon) και αποθηκεύεται στον φάκελο upload.

Τέλος, αφού ενημερωθεί και ο πίνακας datasetTable επιστρέφεται το id της εγγραφής και χρησιμοποιείται για κάθε εγγραφή αυτού του dataset στον πίνακα poisTable. Για τα queries αυτού του πίνακα χρησιμοποιούνται τα δεδομένα του αρχείου pois που ανέβασε ο χρήστης σε συνδιασμό με τις πληροφορίες σχετικά με το σύστημα αναφοράς και τη γραμμογράφηση του αρχείου. Σε κάθε περίπτωση, οι συντεταγμένες μετασχηματίζονται και το σημείο αποθηκεύεται ως τύπος geometry στο σύστημα WGS'84. Παρακάτω φαίνεται το τμήμα του κώδικα που ενημερώνει τον εν λόγω πίνακα.

```
//ARXH Block gia poisTable
$sql="";
$sqlPoisTab1="INSERT INTO
".$schema.".".$poisTable.$poisTableColNamesStr." VALUES ('";
if("$grammografisi"==1){
    foreach($grammesArxeioly as $data){
        $sqlStrPoiValues="";
        $dataArr=explode(",",$data,3);
        $lamda=$dataArr[0];
        $fi=$dataArr[1];
        $desc=rtrim($dataArr[2]);
        $desc = str_replace(","," ", $desc);
        $sqlStrPoiValues =
        $desc."','".$datasetname."','."ST_SetSRID(ST_MakePoint($lamda, $fi),
        $srid)".",".$poidataset_id.'');";
        $sql .= $sqlPoisTab1.$sqlStrPoiValues;

    }//end_foreach
    unset($value);
}elseif("$grammografisi"==2){
    foreach($grammesArxeioly as $data){
        $sqlStrPoiValues="";
        $dataArr=explode(",",$data,3);
        $lamda=$dataArr[1];
        $fi=$dataArr[0];
```

```
                $desc=rtrim($dataArr[2]);
                $desc = str_replace(", ", " ", $desc);
                $sqlStrPoiValues =
$desc.'"','"$datasetname.'"','"ST_SetSRID(ST_MakePoint($lamda, $fi),
$srid)'".'',"$poidataset_id.'"');";
                $sql .= $sqlPoisTab1.$sqlStrPoiValues;

                }//end_foreach
                unset($value);
            }else{
                foreach($grammesArxeioy as $data){
                    $sqlStrPoiValues="";
                    $dataArr=explode(",", $data);
                    $arrCount = count($dataArr);
                    $lamda=$dataArr[$arrCount-2];
                    $fi=rtrim($dataArr[$arrCount-1]);
                    $desc="";

                    for( $i=0; $i<arrCount-3; $i++ ){
                        $desc = $dataArr[i]." ";
                    }
                    $sqlStrPoiValues =
$desc.'"','"$datasetname.'"','"ST_SetSRID(ST_MakePoint($lamda, $fi),
$srid)'".'',"$poidataset_id.'"');";
                    $sql .= $sqlPoisTab1.$sqlStrPoiValues.'';

                    }//end_foreach
                    unset($value);
                }

                echo'<p>';
echo $sql.'<p>';

                $ret = pg_query($db, $sql);
                if(!$ret){
                    echo pg_last_error($db);
                    echo'<p>';
                } else {
                    echo "<h3>". "Η βάση δεδομένων ενημερώθηκε με
επιτυχία\n"."</h3>". "<p>";
                    echo "<p>";
                }

                }//An ola ok
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

                pg_close($db);
            }
    }
```

3.2.6 Λήψη αρχείων δεδομένων

Αντίστοιχα, για τη λήψη αρχείων δεδομένων έχουν δημιουργηθεί τρία αρχεία, τα `OpenDataDownloadForm.php`, `OpenDataDownloadForm.js` και `FileDownload.php`. Και σε αυτή την περίπτωση, το βασικό αρχείο για αυτή τη λειτουργία είναι το πρώτο που καλεί και τα υπόλοιπα. Ο χρήστης, με την επίσκεψη του στην ιστοσελίδα βλέπει όλα τα datasets τα οποία μπορεί να επιλέξει (ένα ή περισσότερα μαζί), και να τα κατεβάσει στον υπολογιστή του.

Οι πληροφορίες που μπορεί να δει γι' αυτά είναι :

- Όνομα dataset
- Περιγραφή
- Ημερομηνία που έγινε upload το αρχείο
- Το nickname του uploader.

Τα διαθέσιμα datasets επιστρέφονται μετά από query που κάνει η σελίδα στον πίνακα `poisTable` με cross-check τους πίνακες `datasetTable` και `contributorTable`.

Κάτω από το κάθε σύνολο δεδομένων υπάρχει ένα checkbox που ο χρήστης πρέπει να επιλέξει για να κατεβάσει αυτά που τον ενδιαφέρουν. Οι επιλογές του, και πιο συγκεκριμένα τα id των datasets που επέλεξε διαχωρισμένα με κόμμα, αποθηκεύονται ως συμβολοσειρά σε ένα κρυφό div της σελίδας. Και σε αυτή την περίπτωση ο κώδικας για τις λειτουργίες αυτές έχει βρίσκεται στο αρχείο `JavaScript`.

Έτσι, μόλις ο χρήστης κάνει submit τη φόρμα, καλείται το αρχείο `FileDownload.php` το οποίο παραλαμβάνει το κρυφό τμήμα της σελίδας που περιέχει ως string τα id των επιλεγμένων datasets και αφού το επεξεργαστεί – δηλαδή τα διαχωρίσει – συνθέτει τη συνθήκη `WHERE` του query που απευθύνεται στον πίνακα `poisTable` και αφορά τα σημεία ενδιαφέροντος που ανήκουν στα σύνολα δεδομένων που επέλεξε. Στη συνέχεια αφού κάνει το select statement και πάρει την απόκριση από τη βάση, απλώς προσθέτει τα κατάλληλα headers ώστε τα αποτελέσματα να γραφούν σε αρχείο και να μεφανιστεί στο χρήστη προτροπή αποθήκευσης στον υπολογιστή του.

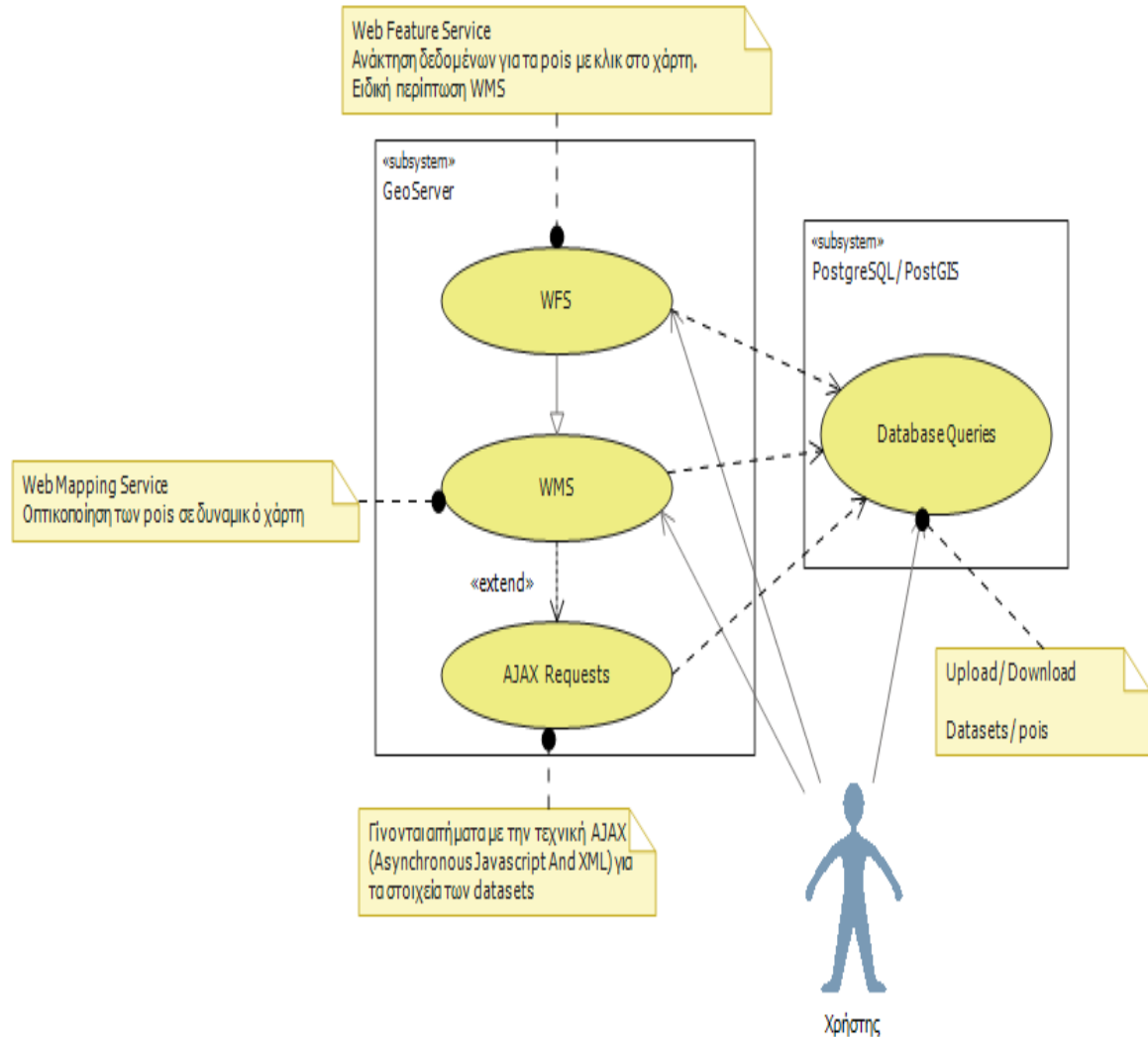
Τα αρχεία που κατεβάζει ο χρήστης είναι *.csv (η default ονομασία των αρχείων είναι `OpenDataProject-download.csv`) με κωδικοποίηση UTF-8, και τα περιεχόμενα του είναι σημεία ενδιαφέροντος της μορφής λ , φ , data με τις συντεταγμένες στο WGS'84.

3.3 Διαγράμματα UML

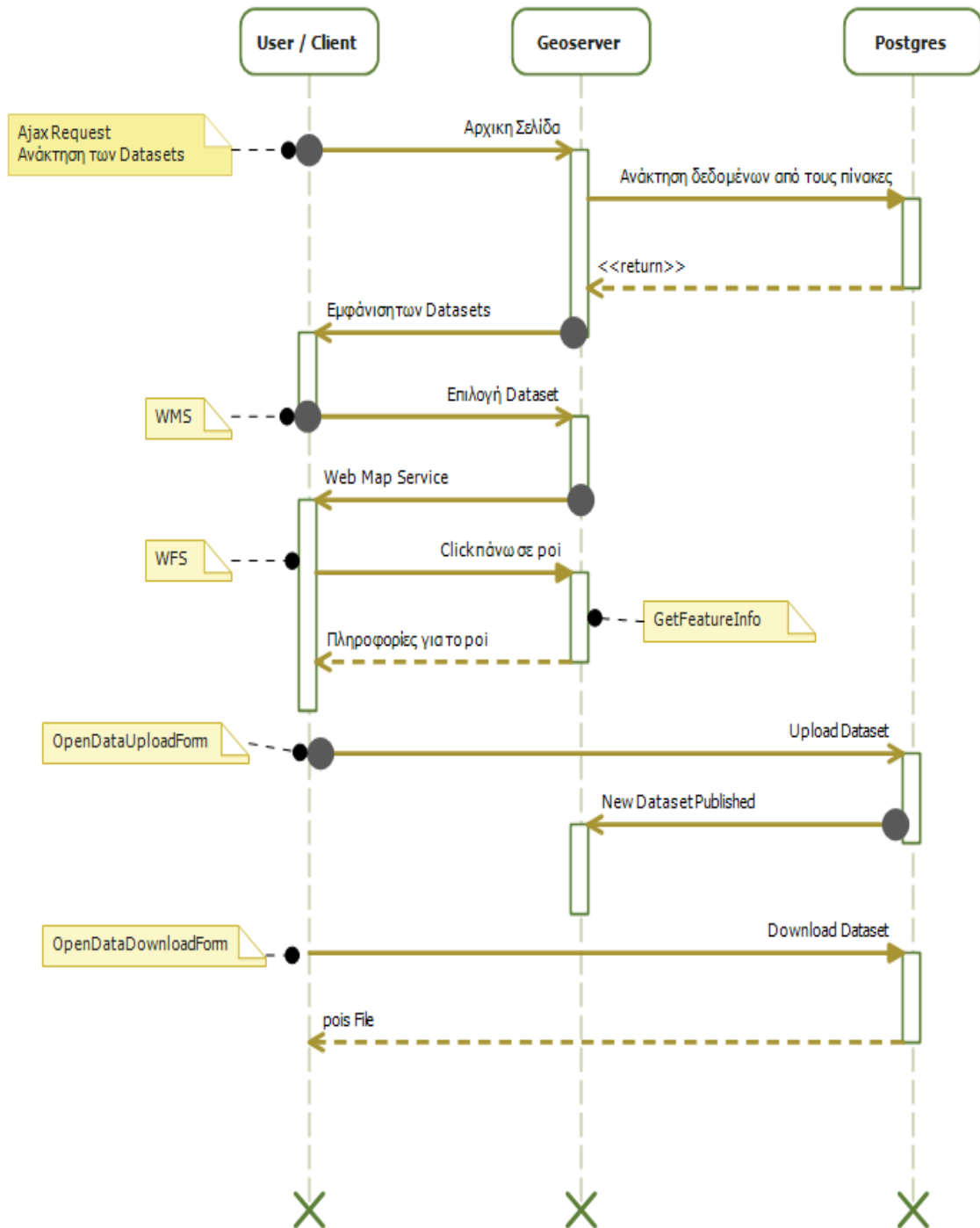
Σε αυτό το κεφάλαιο εμφανίζονται το διάγραμμα περιπτώσεων χρήσης (Use Case Diagram) και το διάγραμμα συνέχειας (Sequence Diagram) όπου φαίνονται σχηματικά όσα εκτέθηκαν παραπάνω.

Το διάγραμμα χρήσης εξετάζει όλες τις δυνατές περιπτώσεις χρήσης της εφαρμογής, τόσο από χρήστες – άτομα, όσο και από τα υπόλοιπα λογισμικά που απαρτίζουν το πρόγραμμα. Το διάγραμμα συνέχειας παρουσιάζει τη λογική ροή του προγράμματος για κάθε περίπτωση χρήσης.

Τα διαγράμματα δημιουργήθηκαν με το Microsoft Visual Studio, το οποίο είναι εμπορικό λογισμικό, αλλά θα μπορούσαν να κατασκευαστούν με οποιοδήποτε σχεδιαστικό πρόγραμμα.



Εικόνα 19. Διάγραμμα χρήσης της εφαρμογής



Εικόνα 20. Διάγραμμα ροής της εφαρμογής

3.4 Σύνοψη

Συμπερασματικά, η εφαρμογή υλοποιήθηκε με αλληλεπίδραση μιας βάσης δεδομένων, του GeoServer, και γραφικού περιβάλλοντος κατασκευασμένου με τεχνολογίες Web.

Όλα τα σημεία ενδιαφέροντος αποθηκεύονται στο πίνακα poisTable, και διαφοροποιούνται ανά dataset ανάλογα με την τιμή της στήλης poisdataset_id το οποίο είναι ξένο κλειδί και δείχνει στον πίνακα datasetTable που κρατάει στοιχεία για τα datasets. Αυτό δίνει μεγάλη ευελιξία στην εφαρμογή, καθώς δεν χρειάζεται να δημιουργούνται νέοι πίνακες σε κάθε upload, απλώς αποθηκεύονται στον ίδιο πίνακα όλα τα σημεία. Στη συνέχεια, με φιλτράρισμα του πίνακα αυτού, απομονώνονται τα επιθυμητά κάθε φορά pois.

Ο GeoServer αναλαμβάνει να κάνει publish τους πίνακες της βάσης δεδομένων ώστε να προσφέρει σε αυτούς συγκεκριμένες υπηρεσίες. Αυτές που χρησιμοποιούνται στην εφαρμογή είναι οι Web Mapping Service και Web Feature Service.

Τα αρχεία που απαρτίζουν την εφαρμογή είναι HTML, JavaScript, CSS, και PHP. Με HTML κατασκευάστηκε το User Interface, το styling του οποίου έγινε με κανόνες μορφοποίησης CSS.

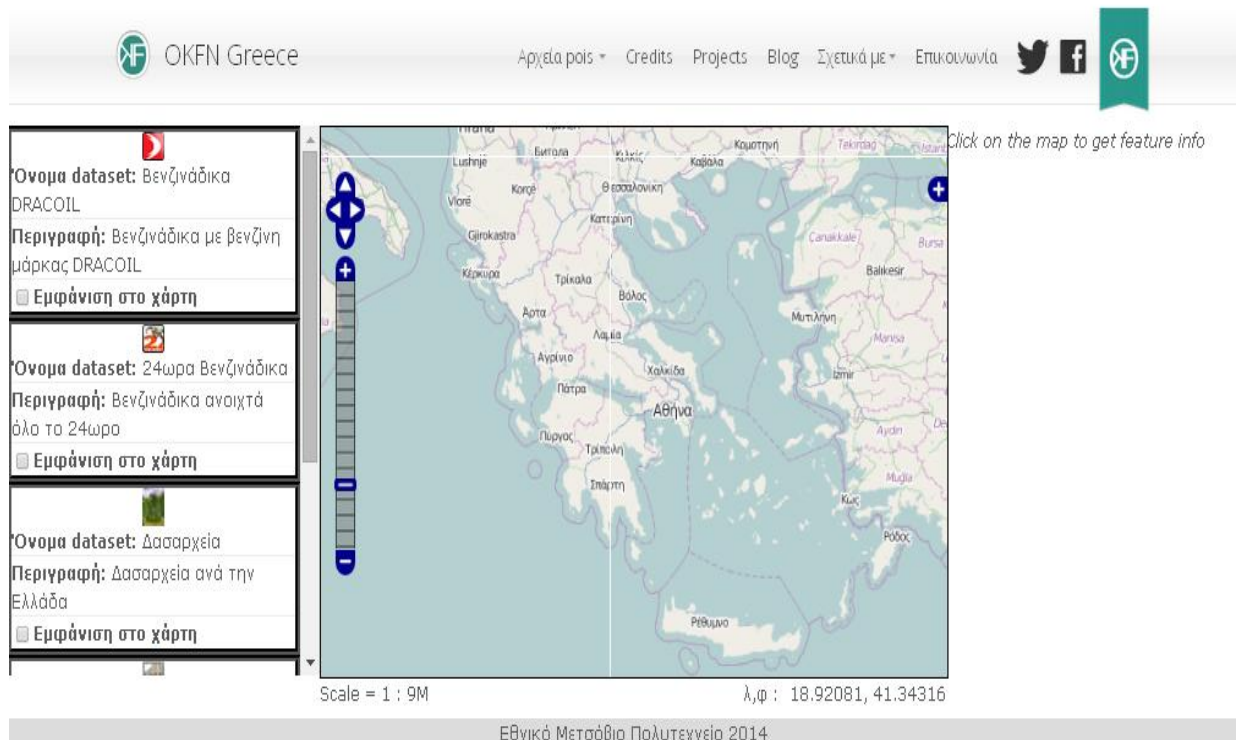
Σε JavaScript έχει γραφεί σχεδόν το σύνολο των αλγορίθμων λειτουργίας της εφαρμογής (flow logic), τουλάχιστον ότι δεν χρειάζοταν αλληλεπίδραση με τη βάση δεδομένων.

Τέλος, με Php έγινε οτιδήποτε δεν ήταν δυνατόν να γίνει με JavaScript, και ειδικότερα ότι αφορούσε ενέργειες προς ή από τη βάση δεδομένων. Η php αντιθέτα από την JavaScript είναι server-side γλώσσα, οπότε ότι χρειάστηκε να γίνει στην πλευρά του πελάτη έγινε με Javascript, ενώ ότι έπρεπε να γίνει στο server έγινε με php.

4. Παρουσίαση της εφαρμογής

4.1 Παρουσίαση της αρχικής σελίδας

Στο συγκεκριμένο κεφάλαιο παρουσιάζεται η εφαρμογή σε κάθε στάδιο λειτουργίας. Στην παρακάτω εικόνα φαίνεται η οθόνη της αρχικής σελίδας της εφαρμογής.



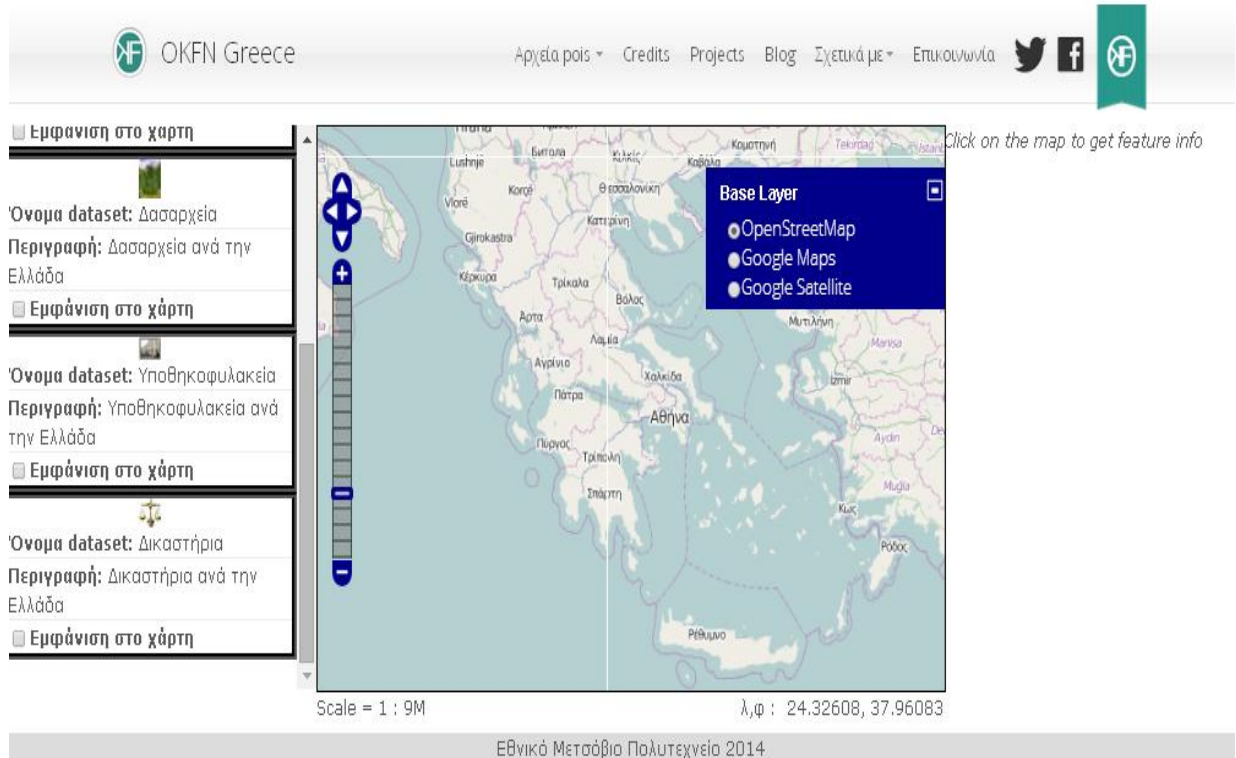
Εικόνα 21. Αρχική σελίδα της εφαρμογής

Όπως φαίνεται και παραπάνω, στην αρχική σελίδα ο χρήστης βλέπει σε ένα τμήμα τα datasets που περιέχονται στη βάση δεδομένων, και δίπλα ένα χάρτη στον οποίο εμφανίζονται αυτά που επιλέγει να δει.

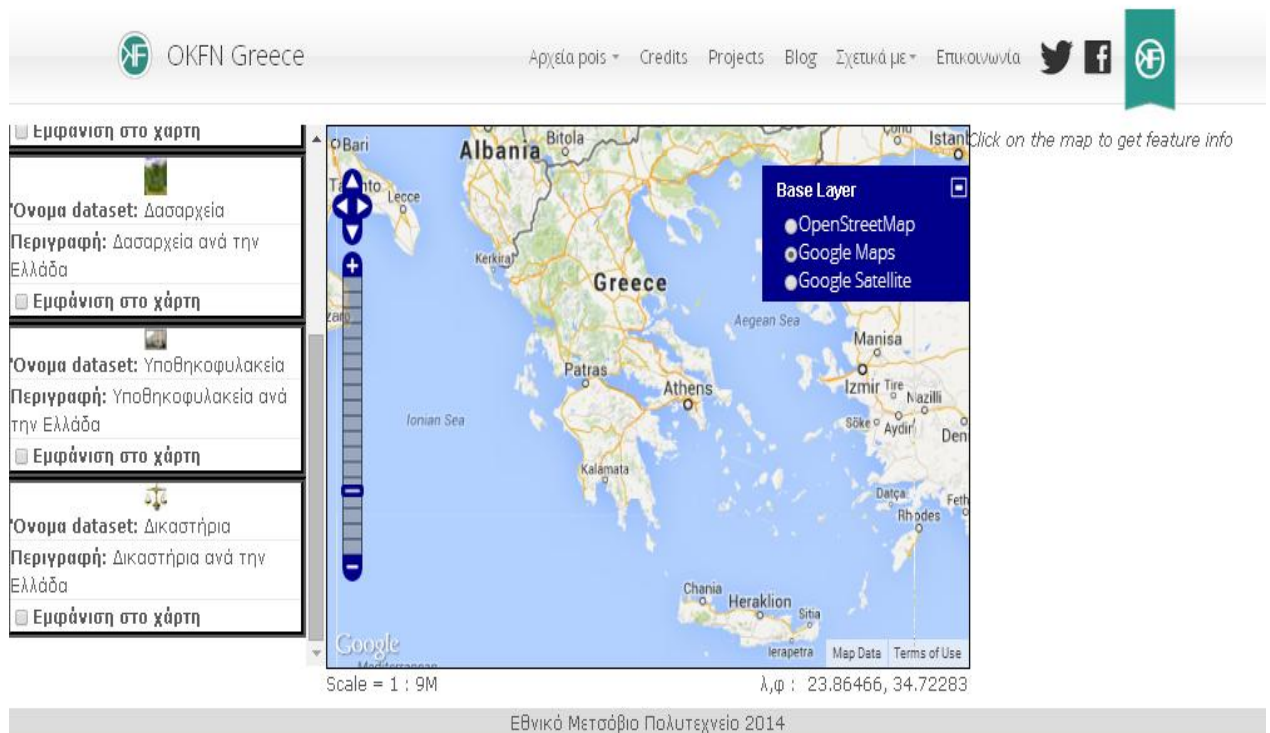
Οι πληροφορίες για τα σύνολα δεδομένων ανακτώνται μέσω της υπηρεσίας WFS που προσφέρει ο GeoServer. Από όλες τις πληροφορίες που περιέχονται στη βάση για κάθε σύνολο δεδομένων, εμφανίζονται μόνο το εικονίδιο, το όνομα και η περιγραφή. Για την παρουσίαση τους τοποθετούνται σε HTML <fieldset> tags.

Ως υπόβαθρο προεπιλογή είναι χάρτης OpenStreetMap (OSM), αλλά μπορεί να επιλέξει και χάρτες Google. Η επιλογή γίνεται με το widget από το OpenLayers με το οποίο επίσης μπορεί να εμφανίζει και να αποκρύβει τα επιλεγμένα layers. Ακόμα, στο κάτω μέρος του χάρτη ο χρήστης βλέπει την κλίμακα θέασης, καθώς και τις συντεταγμένες (λ , φ) της θέσης του κέρσορα του ποντικιού όπως κινείται πάνω στο χάρτη.

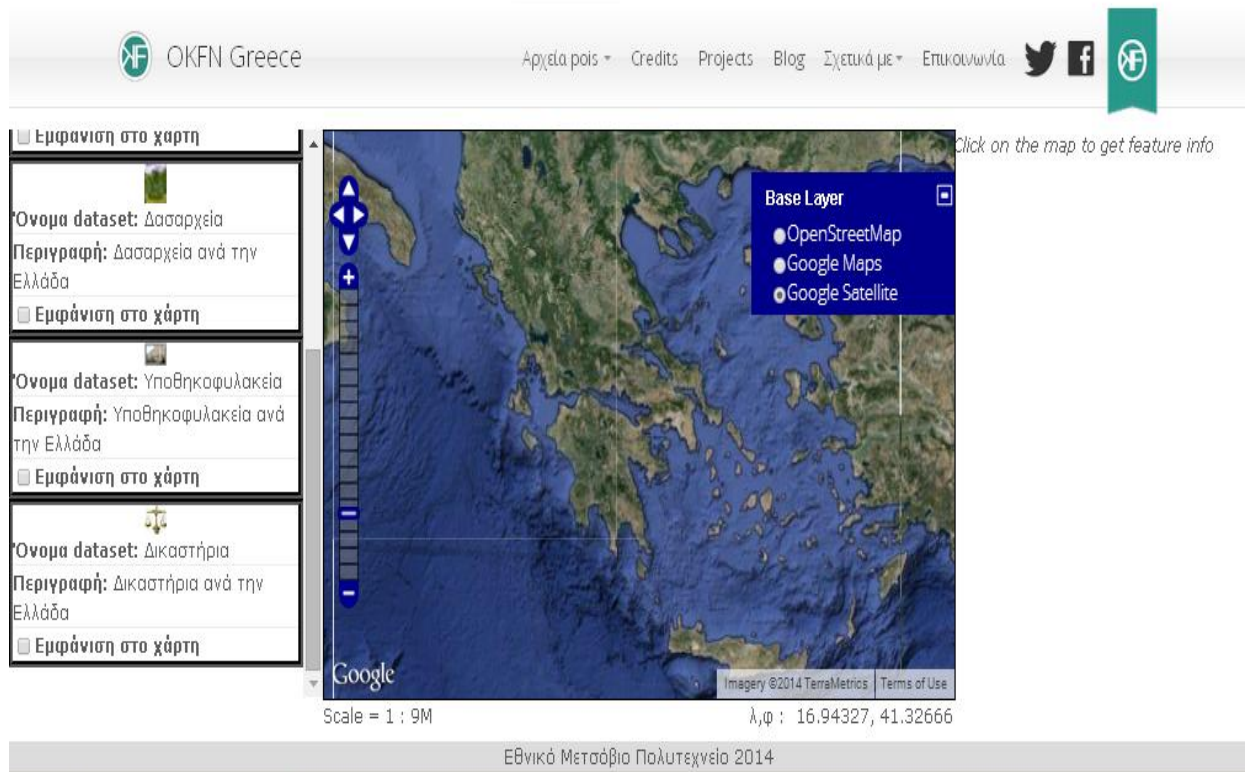
Στις παρακάτω εικόνες φαίνονται οι εναλλακτικοί χάρτες υποβάθρου της αρχικής σελίδας.



Εικόνα 22. Αρχική σελίδα της εφαρμογής, επιλογή χάρτη υποβάθρου OSM

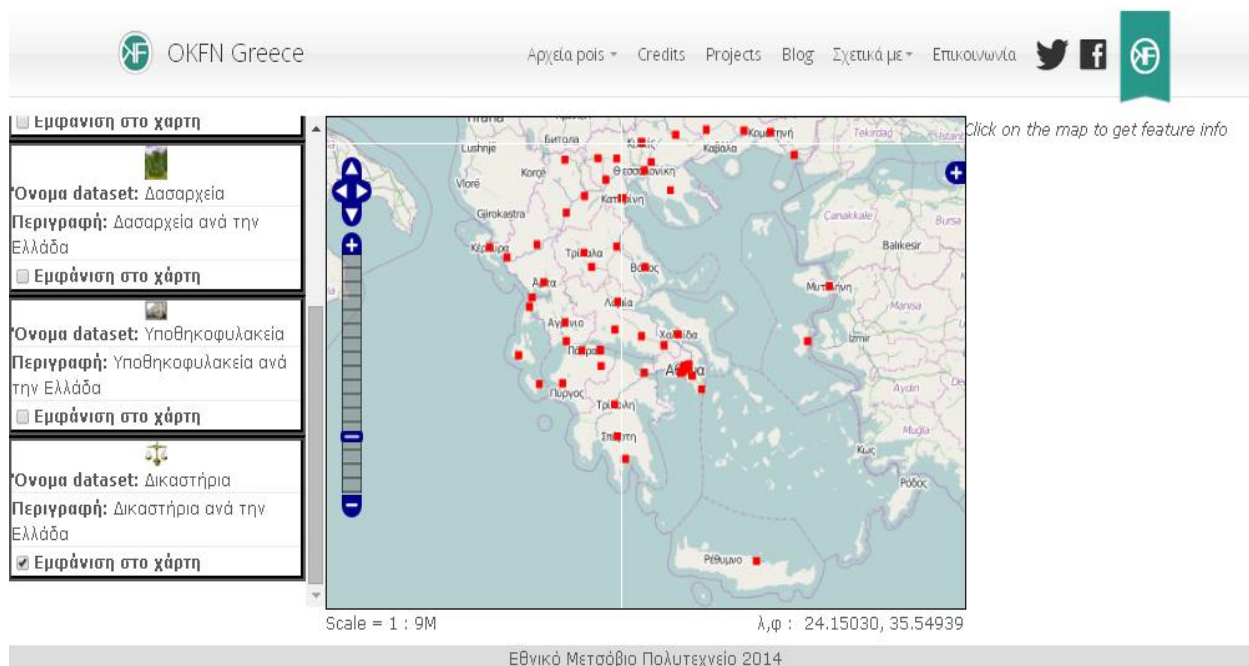


Εικόνα 23. Αρχική σελίδα της εφαρμογής, επιλογή χάρτη υποβάθρου GoogleMaps



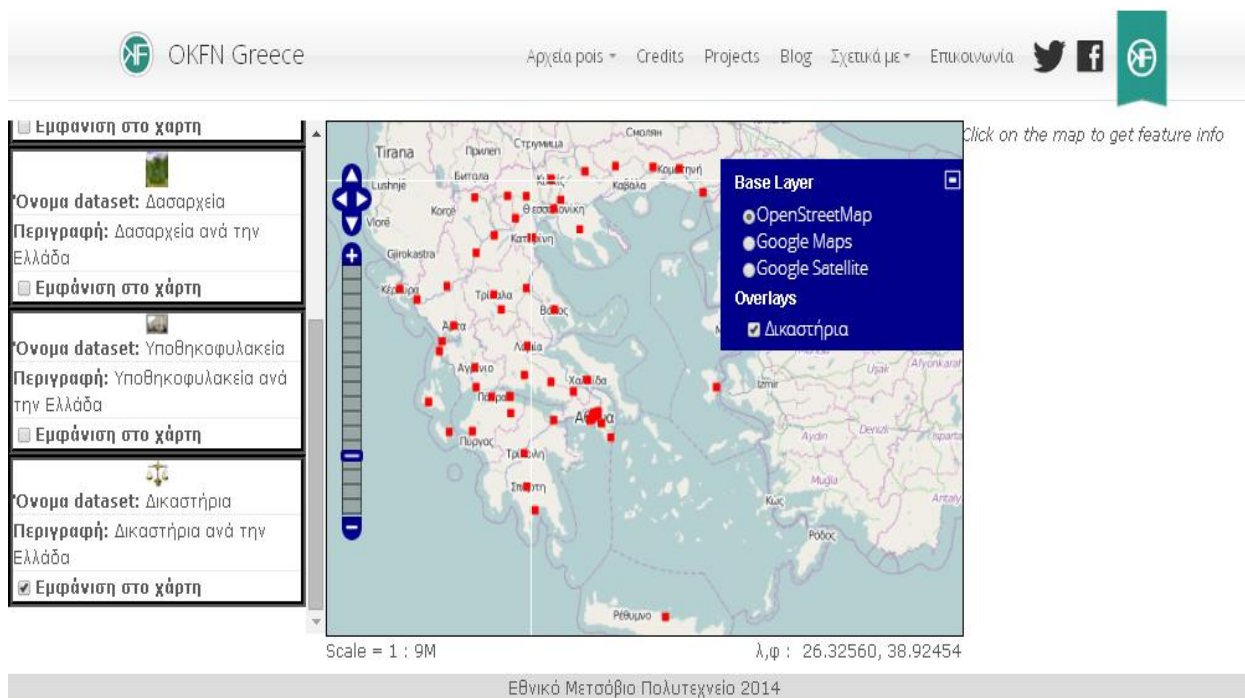
Εικόνα 24. Αρχική σελίδα της εφαρμογής, επιλογή χάρτη υποβάθρου GoogleSatellite

χρήστης ενδιαφέρεται να δει στο χάρτη τις θέσεις των pois αρκεί να επιλέξει το checkbox που βρίσκεται κάτω από το συγκεκριμένο dataset. Τότε εμφανίζονται στο χάρτη κόκκινες κουκκίδες που αντιπροσωπεύουν τα συγκεκριμένα σημεία ενδιαφέροντος



Εικόνα 25. Προβολή των dataset στο χάρτη

Επίσης, μπορεί αν θέλει να εμφανίζει / αποκρύβει τα φορτωμένα datasets από το χάρτη, ως overlay layers με τη χρήση του OpenLayers widget.



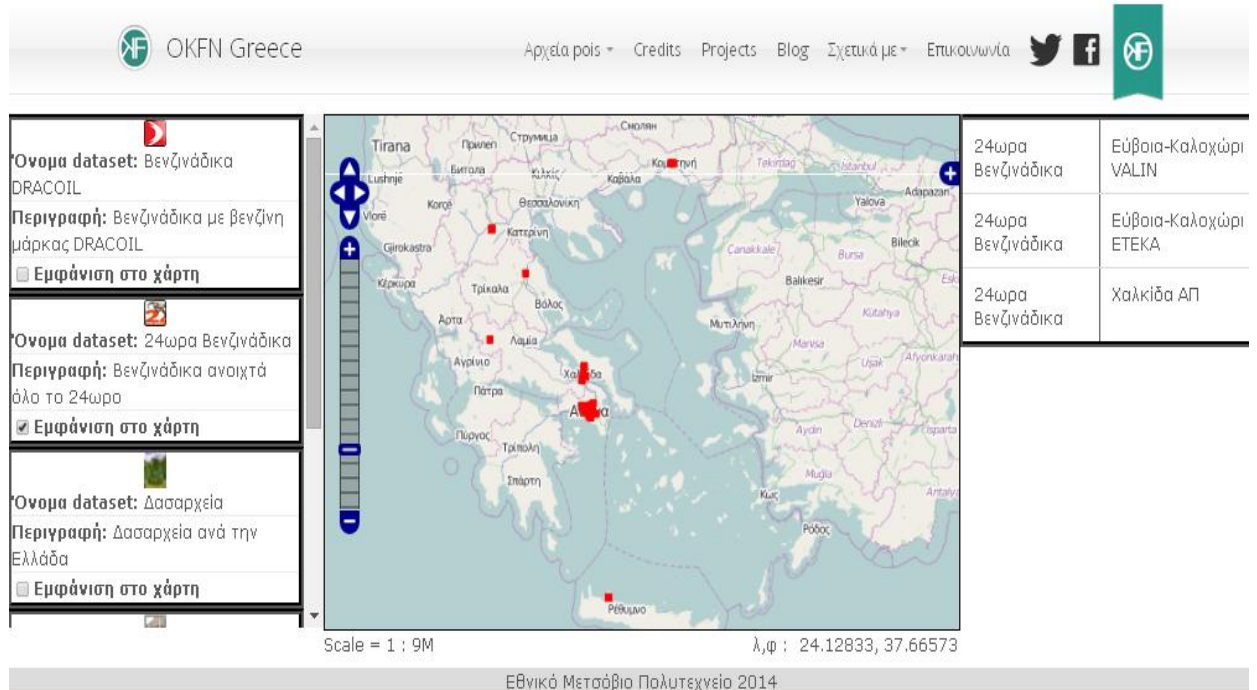
Εικόνα 26. Εμφάνιση / απόκρυψη layer με τη χρήση του OpenLayers widget

Για κάθε σημείο ενδιαφέροντος του οποίου το dataset είναι φορτωμένο στην σελίδα, ο χρήστης έχει τη δυνατότητα να ανακτήσει τις πληροφορίες που περιέχονται στη βάση δεδομένων σχετικά με αυτό. Αυτό γίνεται με απλό κλικ στο σημείο επάνω στο χάρτη και αποτελεί υλοποίηση της υπηρεσίας WMS που παρέχει ο GeoServer με την λειτουργία GetFeatureInfo.

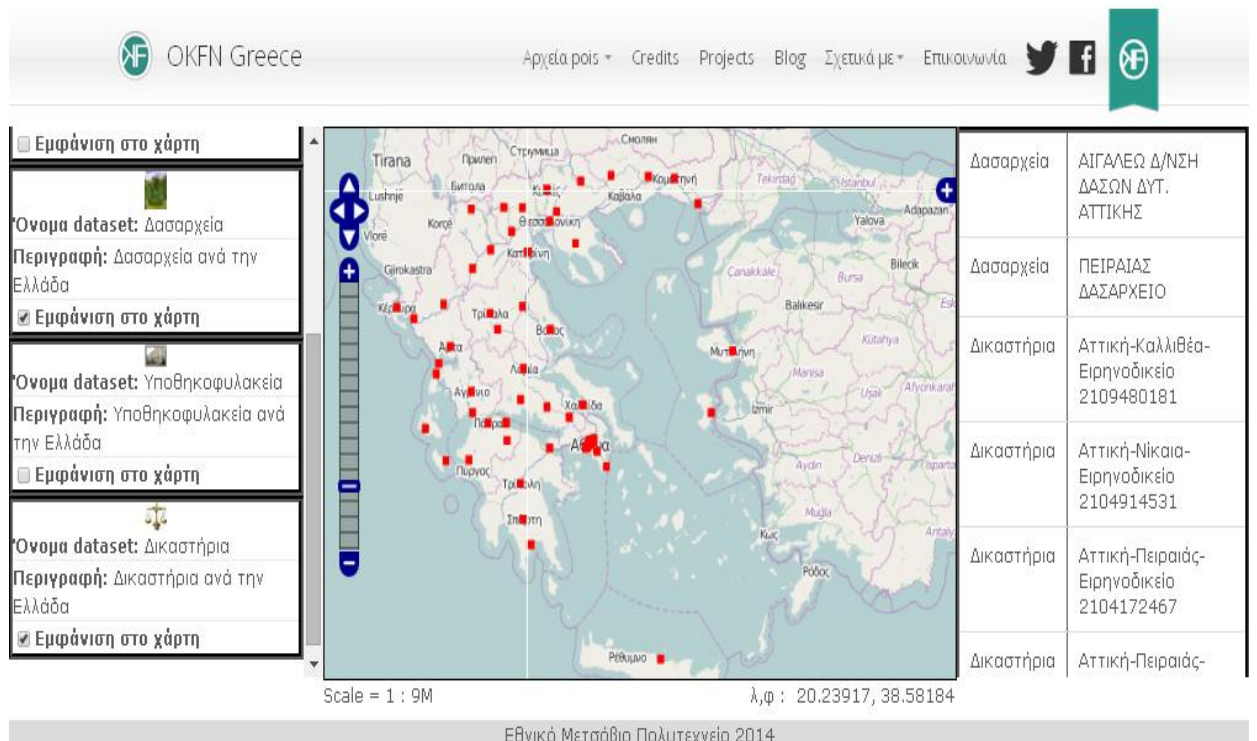
Όταν το κλικ αντιστοιχεί σε περισσότερα του ενός σημεία, για παράδειγμα στην περίπτωση πολύ μικρού zoom (δηλαδή μεγάλου συντελεστή κλίμακας) ή στην περίπτωση που τα σημεία ενδιαφέροντος απέχουν πολύ λίγο μεταξύ τους και το κλικ τα περιλαμβάνει μαζί, τότε εμφανίζονται όλες οι εγγραφές. Ανάλογα με τον αριθμό των σημείων που αφορά το κλικ, η απόκριση ίσως αργεί λίγο να εμφανιστεί στην οθόνη.

Οι πληροφορίες αυτές απεικονίζονται στο δεξί μέρος της σελίδας δομημένες σε μορφή πίνακα HTML <table> tag. Ο πίνακας αυτός έχει δύο στήλες, στην πρώτη αναγράφεται το όνομα του dataset στο οποίο ανήκει το ροί και στην δεύτερη οι πληροφορίες που περιέχονται στη βάση για αυτό. Για να σβηστούν οι εγγραφές από την οθόνη, αρκεί ο χρήστης να κάνει κλικ σε κάποιο κενό σημείο του χάρτη.

Στις παρακάτω εικόνες παρουσιάζεται η εν λόγω υπηρεσία.



Εικόνα 27. Ανάκτηση πληροφοριών για τα σημεία ενδιαφέροντος

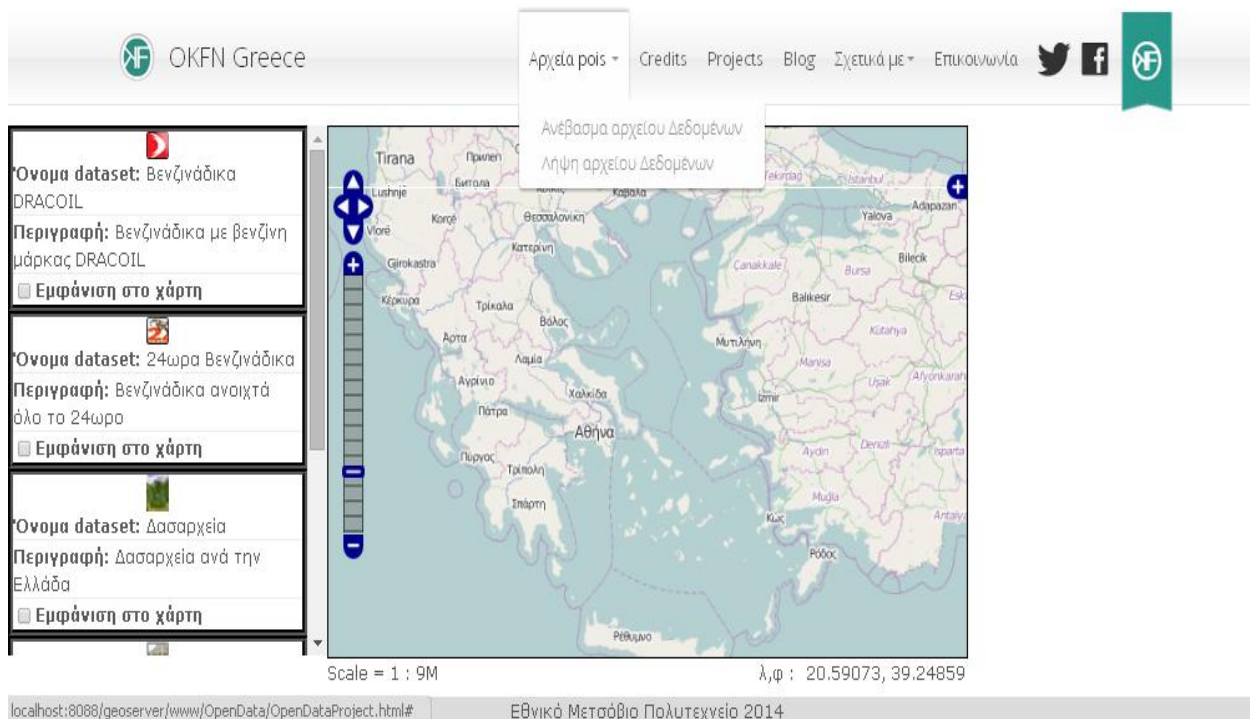


Εικόνα 28. Ανάκτηση πληροφοριών για σημεία ενδιαφέροντος από πολλά datasets

4.2 Παρουσίαση της σελίδας για upload δεδομένων

Στην αρχική σελίδα υπάρχουν σύνδεσμοι που κατευθύνουν το χρήστη στη σελίδα όπου μπορεί να ανεβάσει αρχεία δεδομένων και να καταχωρίσει στη βάση το δικό του dataset.

Πιο συγκεκριμένα, όπως φαίνεται και στην παρακάτω εικόνα, από την επιλογή του menu Αρχεία pois πρέπει να επιλέξει «Ανέβασμα αρχείου Δεδομένων» και θα κατευθυνθεί στο Server της βάσης δεδομένων.



Εικόνα 29. Ανακατεύθυνση στη σελίδα για upload συνόλου δεδομένων

Εκεί καλείται να συμπληρώσει κάποια στοιχεία στη φόρμα που βλέπει σχετικά με τον ίδιο, (δηλαδή τον uploader), και κάποια σχετικά με το dataset που θα ανεβάσει.

Οι πληροφορίες διακρίνονται σε υποχρεωτικές και προαιρετικές. Τις πρώτες πρέπει να τις συμπληρώσει διαφορετικά δεν μπορεί να γίνει υποβολή της φόρμας. Όσον αφορά τις προαιρετικές πληροφορίες κάποιες από αυτές έχουν default τιμές και κάποιες άλλες αποστέλονται κενές.

Η HTML5 προσφέρει μηχανισμούς ελέγχου (ιδιότητα required στα ανάλογα πεδία) που ενημερώνουν το χρήστη στην περίπτωση που δεν έχει συμπληρώσει κάποια από τα απαραίτητα πεδία ώστε να εξασφαλιστεί πως η φόρμα δεν θα αποσταλεί αν δεν έχουν συγκεντρωθεί οι απαραίτητες πληροφορίες. Αυτό είναι πολύ σημαντικό, δεδομένου ότι πρόκειται για στοιχεία που αντιστοιχούν σε υποχρεωτικές τιμές των πινάκων της βάσης δεδομένων και σε αντίθετη περίπτωση η εφαρμογή δεν θα λειτουργούσε.

Στις παρακάτω εικόνες φαίνεται η σελίδα upload, τα υποχρεωτικά και τα προαιρετικά πεδία της φόρμας.

Ανέβασμα Αρχείου Δεδομένων

Πληροφορίες για τον Uploader

Απαραίτητες πληροφορίες ▾

Ψευδώνυμο :

Στοιχεία που αφορούν το Dataset [DB2]

ΚΑΤΗΓΟΡΙΑ :

Όνομα Dataset :

Περιγραφή Dataset :

Αρχείο pois No file chosen

Εμφάνιση ρυθμίσεων αρχείου

Εικόνα 30. Σελίδα για upload δεδομένων

Στοιχεία που αφορούν το Dataset [DB2]

ΚΑΤΗΓΟΡΙΑ :

Όνομα Dataset :

Περιγραφή Dataset :

Αρχείο pois No file chosen

Εμφάνιση ρυθμίσεων αρχείου

Εικόνα 31. Επιλογή κατηγορίας για το dataset

Στοιχεία που αφορούν το Dataset [DB2]

ΚΑΤΗΓΟΡΙΑ :

Όνομα Dataset :

Περιγραφή Dataset :

Αρχείο pois No file chosen

Εμφάνιση ρυθμίσεων αρχείου

Εικόνα 32. Δημιουργία νέας κατηγορίας για το dataset

Ανέβασμα Αρχείου Δεδομένων

Πληροφορίες για τον Uploader

Απαραίτητες πληροφορίες ▾

Ψευδώνυμο :

Στοιχεία που αφορούν το αρχείο : ! Please fill out this field.

ΚΑΤΗΓΟΡΙΑ :

Όνομα Dataset :

Περιγραφή Dataset :

Αρχείο ποis No file chosen

Εμφάνιση ρυθμίσεων αρχείου

Εικόνα 33. Υποχρεωτικά στοιχεία για τον Uploader

Πληροφορίες για τον Uploader

Περισσότερες πληροφορίες ▾

Ψευδώνυμο :

E-mail :

Όνομα :

Επώνυμο :

Facebook :

Twitter :

Ιστοσελίδα :

Εικόνα 34. Προαιρετικά στοιχεία για τον Uploader

Ανέβασμα Αρχείου Δεδομένων

Πληροφορίες για τον Uploader

Απαραίτητες πληροφορίες ▾

Ψευδώνυμο : pois Web Site

Στοιχεία που αφορούν το Dataset [DB2]

ΚΑΤΗΓΟΡΙΑ : Χωρίς Κατηγορία ▾

Όνομα Dataset :

Περιγραφή Dataset :

Αρχείο pois No file chosen

Εμφάνιση ρυθμίσεων αρχείου

Εικόνα 35. Υποχρεωτική συμπλήρωση για το πεδίο όνομα dataset

Ανέβασμα Αρχείου Δεδομένων

Πληροφορίες για τον Uploader

Απαραίτητες πληροφορίες ▾

Ψευδώνυμο : pois Web Site

Στοιχεία που αφορούν το Dataset [DB2]

ΚΑΤΗΓΟΡΙΑ : Χωρίς Κατηγορία ▾

Όνομα Dataset : test

Περιγραφή Dataset :

Αρχείο pois No file chosen

Εμφάνιση ρυθμίσεων αρχείου

Εικόνα 36. Υποχρεωτική υπόδειξη αρχείου pois

Στοιχεία που αφορούν το Dataset [DB2]

ΚΑΤΗΓΟΡΙΑ :

Όνομα Dataset :

Περιγραφή Dataset :

Αρχείο pois No file chosen

Απόκρυψη ρυθμίσεων αρχείου

Σύστημα Αναφοράς

λ , φ [WGS 84]

X , Ψ [ΕΓΣΑ 87]

Γραμμογράφηση αρχείου

λ ή X , φ ή Y , data

φ ή Y , λ ή X , data

data , λ ή X , φ ή Y

Εικονίδιο dataset: No file chosen

Εικόνα 37. Προαιρετικά στοιχεία για το dataset

Στην εικόνα που ακολουθεί φαίνεται ένα υπόδειγμα αρχείου pois που θα μπορούσε να γίνει upload στη βάση δεδομένων (πηγή: http://index.pois.gr/pocard.asp?POI_ID=396, τράπεζες AlphaBank, format:Garmin csv,)

Line	POI ID	Latitude	Longitude	Name	Category	POI ID
1	21.40961	38.62349	"Αγρινιο Τρικουπη Χαρ. 11"	" "	" "	
2	25.340385	37.434805	"ΑΕΡΟΔΡΟΜΙΟΥ ΜΥΚΟΝΟΥ (794)"	" "	" "	
3	23.42727	37.74639	"Αιγίνα Λεωφορος Δημοκρατίας"	" "	" "	
4	22.08913	38.25092	"Αιγιο Γερμανου 10-12"	" "	" "	
5	20.883884	38.785544	"Αιτωλοακαρνανία Πάλαιρος ΑΤΜ"	" "	" "	
6	21.40360	38.62026	"Αιτωλοακαρνανία-Αγρίνιο"	"2641048627"	" "	
7	21.40880	38.62434	"Αιτωλοακαρνανίας-Αγρίνιο"	"2641037050-Πρώην Εμπορική"	" "	
8	21.42930	38.36900	"Αιτωλοακαρνανίας-Μεσολόγγι"	"2631050230-Πρώην Εμπορική"	" "	
9	21.83420	38.39492	"Αιτωλοακαρνανίας-Ναύπακτος"	"2634038740-Πρώην Εμπορική"	" "	
10	25.89002	40.84845	"Αλεξανδρουπολη Λεωφ.Δημοκρατίας 1"	" "	" "	
11	25.87730	40.84608	"Αλεξανδρουπολη Λεωφ.Δημοκρατίας 238"	" "	" "	
12	22.75798	39.18063	"Αλμυρος Βασ.Κωνσταντίνου 113"	" "	" "	
13	21.35029	37.79601	"Αμαλιαδα Ερμου 6"	" "	" "	
14	21.16670	38.86341	"Αμφιλοχια Ν.Στρατου 37"	" "	" "	
15	22.37731	38.52709	"Αμφισσα Ιωαννου Γατου 6"	" "	" "	
16	24.93759	37.83718	"Ανδρος"	" "	" "	
17	22.58579	38.48049	"Αραχωβα Πλατεια Λακκα"	" "	" "	
18	23.16089	37.37470	"Αργολίδα-Κρανίδι"	"2754022253"	" "	
19	23.14379	37.32820	"Αργολίδα-Πόρτο Χέλι"	"2754051862"	" "	

Εικόνα 38. Υπόδειγμα αρχείου pois

4.3 Παρουσίαση της σελίδας για download δεδομένων

Κατ' αντιστοιχία με τη σελίδα upload έχει δημιουργηθεί και σελίδα download. Ο χρήστης για να κατευθυνθεί εκεί πρέπει να επιλέξει από το menu Αρχεία pois στην αρχική σελίδα, «Λήψη αρχείου Δεδομένων».

Εκεί, εμφανίζονται όλα τα datasets που περιέχει η βάση δεδομένων, ώστε να επιλέξει αυτά που των ενδιαφέρουν. Τα datasets εμφανίζονται δομημένα ως <fieldset> με label το όνομα τους και για το καθένα αναγράφεται η περιγραφή, η ημερομηνία upload καθώς και το nickname του uploader. Για να τα κατεβάσει ο χρήστης, αρκεί να επιλέξει το checkbox για το καθένα. Είναι δυνατή η λήψη πολλών datasets σε ένα αρχείο, αν είναι επιλεγμένα περισσότερα του ενός σύνολα δεδομένων. Μόλις επιλέξει τα datasets που τον ενδιαφέρουν αρκεί να πατήσει το κουμπί «Λήψη αρχείου» και θα εμφανιστεί ένα παράθυρο που θα τον προτρέπει να αποθηκεύσει το αρχείο csv στον υπολογιστή του.

Παρακάτω στην εικόνα φαίνεται η σελίδα για την λειτουργία download.

Διαθέσιμα Datasets για Download

<p>Όνομα Dataset: Βενζινόδικα DRACOIL Περιγραφή: Βενζινόδικα με βενζίνη μάρκας DRACOIL Ημερομηνία Upload: 2014-07-26 Uploader: pois Web Site</p> <p><input type="checkbox"/> Επιλογή</p>
<p>Όνομα Dataset: 24ωρα Βενζινόδικα Περιγραφή: Βενζινόδικα ανοιχτά όλο το 24ωρο Ημερομηνία Upload: 2014-07-26 Uploader: pois Web Site</p> <p><input type="checkbox"/> Επιλογή</p>
<p>Όνομα Dataset: Δασαρχεία Περιγραφή: Δασαρχεία ανά την Ελλάδα Ημερομηνία Upload: 2014-07-26 Uploader: pois Web Site</p> <p><input type="checkbox"/> Επιλογή</p>
<p>Όνομα Dataset: Υποθηκοφυλακεία Περιγραφή: Υποθηκοφυλακεία ανά την Ελλάδα Ημερομηνία Upload: 2014-07-26 Uploader: pois Web Site</p> <p><input type="checkbox"/> Επιλογή</p>
<p>Όνομα Dataset: Δικαστήρια Περιγραφή: Δικαστήρια ανά την Ελλάδα Ημερομηνία Upload: 2014-07-26 Uploader: pois Web Site</p> <p><input type="checkbox"/> Επιλογή</p>

[Επιστροφή στην Αρχική Σελίδα](#)

Εικόνα 39. Σελίδα για download δεδομένων

5. Συμπεράσματα

Συνοψίζοντας, οι τεχνολογίες που χρησιμοποιούνται σε αυτή την εφαρμογή αποτελούν κάποιες από τις πλέον σύγχρονες λύσεις για την δημιουργία διαδικτυακών εφαρμογών σχετικές με τη διάθεση δεδομένων και υπηρεσιών πάνω σε αυτά. Η ίδια εφαρμογή θα μπορούσε να κατασκευαστεί με τη χρήση άλλων γλωσσών προγραμματισμού ή χρησιμοποιώντας διαφορετική βάση δεδομένων ή λογισμικό Server.

Έγινε επιλογή να χρησιμοποιηθούν ελεύθερα λογισμικά, δωρεάν διαθέσιμα σε οποιοδήποτε ενδιαφερόμενο. Τα συγκεκριμένα, δεν υστερούν σε τίποτα από τα εμπορικά λογισμικά, ενώ υπόκεινται σε συνεχείς βελτιώσεις και αναβαθμίσεις από πολυπληθείς κοινότητες προγραμματιστών και χρηστών διεθνώς. Παράλληλα, το γεγονός ότι είναι επίσης Λογισμικά Ανοιχτού Κώδικα τα καθιστά πιο εύκολα αξιοποιήσιμα από τρίτες εφαρμογές.

Είναι αναμφισβήτητο γεγονός πως η αξία των δεδομένων συναρτάται με την πληροφωρία που προκύπτει από την επεξεργασία τους, οπότε είναι σημαντικό να εξασφαλιστεί η απρόσκοπτη πρόσβαση κάθε ενδιαφερόμενου σε αυτά. Πολύ περισσότερο σήμερα που η δημόσια διοίκηση υποχρεώνεται από ευρωπαϊκές οδηγίες να διαθέσει ελεύθερα δεδομένα και υπάρχει πλούσιο υλικό προς αξιοποίηση.

Επίσης, η ραγδαία πρόοδος της τεχνολογίας, που έχει κάνει προσιτά τα προϊόντα της σε όλο το φάσμα της κοινωνίας, δίνει τη δυνατότητα στον καθένα να συγκεντρώνει, να διαμοιράζει και να επεξεργάζεται δεδομένα, με τρόπο εύκολο και γρήγορο, κάνοντας με αυτό τον τρόπο το crowdsourcing μία συνεχώς αναπτυσσόμενη τάση. Έτσι, είναι δυνατόν με κατάλληλη επεξεργασία των δεδομένων και την προσθήκη λειτουργιών και υπηρεσιών σε αυτά να δημιουργηθεί προστιθέμενη αξία καθιστώντας τα ακόμα πιο χρήσιμα, για ακόμα μεγαλύτερο κομμάτι του πληθυσμού. Άλλωστε, πέρα από όλα τα άλλα το crowdsourcing έχει σαφή συλλογική διάσταση, ιδιαίτερα όταν αξιοποιείται προς όφελος του κοινωνικού συνόλου. Με αυτό τον τρόπο τα οφέλη της προσπάθειας επιστρέφουν στους πολίτες.

Είναι σαφές ότι υπάρχουν επισφάλειες, ιδιαίτερα όσον αφορά την ποιότητα και την αξιοπιστία των δεδομένων. Σε κάθε περίπτωση όμως, τα οφέλη υπερβαίνουν τα μειονεκτήματα, ενώ με συγκεκριμένες τεχνικές θα μπορούσε να ελεγχθεί και να βελτιωθεί κατά πολύ η ποιότητα του αποτελέσματος, δηλαδή των παρεχόμενων δεδομένων.

Μία τέτοια λύση θα μπορούσε να είναι η αξιολόγηση των δεδομένων από τους χρήστες ούτως ώστε με την πάροδο ικανού χρονικού διαστήματος να υπάρχει σαφή εικόνα για την ποιότητα τους. Αυτό αποτελεί ένα από τα αντικείμενα μελλοντικής βελτίωσης της εφαρμογής σε επόμενο χρόνο.

Επίσης, η αξιοποίηση του Geolocation API της JavaScript θα μπορούσε να προσφέρει πιο εξειδικευμένη πληροφόρηση στους χρήστες εμφανίζοντας τους τα σημεία που βρίσκονται πιο κοντά στη θέση τους. Η συγκεκριμένη υλοποίηση είναι σχετικά απλή και θα προσέφερε πολλαπλάσια οφέλη.

Τέλος, χρειάζεται να γίνει περαιτέρω δουλειά για την κατηγοριοποίηση των δεδομένων, αφού είναι ο καλύτερος τρόπος για να είναι αυτά εύκολα ανακαλύψιμα.

6. Βιβλιογραφία – πηγές

- [1] <http://www.creativecommons.gr/?p=1529>
- [2] <http://opendefinition.org/od/ellinika/#sthash.Kq9KDHZy.dpuf>
- [3] <http://swu.auth.gr/el/glossary#glos-page-4>
- [4] http://el.wikipedia.org/wiki/%CE%95%CE%BB%CE%B5%CF%8D%CE%B8%CE%B5%CF%81%CE%BF_%CE%BB%CE%BF%CE%B3%CE%B9%CF%83%CE%BC%CE%B9%CE%BA%CF%8C
- [5] <http://www.gnu.org/philosophy/free-sw.el.html>
- [6] <http://www.gnu.org/philosophy/open-source-misses-the-point.el.html>
- [7] <http://mathe.ellak.gr/?p=6>
- [8] E. Estellés-Arolas & F. González Ladrón-de- Guevara.
Towards an integrated crowdsourcing definition. Journal of Information Science (JCR: 1,41), 2012
- [9] David Whitford (2007-03-22).
«Hired Guns on the Cheap». Fortune Small Business
- [10] Jeff Howe (June 2006).
«The Rise of Crowdsourcing»
- [11] Noveck, Simone. (2009).
Wiki Government: How Technology Can Make Government Better, Democracy Stronger, and Citizens More Powerful, σελ. 63
- Βασίλειος Βεσκούκης,
"Τεχνολογία λογισμικού I", Ελληνικό Ανοικτό Πανεπιστήμιο, Πάτρα, 2000,
- Βασίλειος Βεσκούκης,
"Τεχνολογία λογισμικού II, Τόμος Β' Σχεδιασμός Λογισμικού", Ελληνικό Ανοικτό Πανεπιστήμιο, Πάτρα, 2008

Θεοδωρίδης Γ.,
Σημειώσεις σε ΓΕΩΓΡΑΦΙΚΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ (GIS), Πειραιάς 2012

Θεοδωρίδης Γ.,
Σημειώσεις ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ, Πειραιάς 2011
Βόντας Μ., Ταμπάκης Π.,
Εργαστηριακή Διάλεξη στα Γεωγραφικά Πληροφοριακά Συστήματα – PostGIS Πειραιάς 2012

Στεφανάκης Ε. 2009,
Τεχνολογίες Δημοσιοποίησης Χαρτογραφικού Περιεχομένου στον Παγκόσμιο Ιστό. Αθήνα,
ΕΚΔΟΣΕΙΣ ΝΕΩΝ ΤΕΧΝΟΛΟΓΙΩΝ

Janowicz, K., Schade, S., Broring, A., Kezler, C., Maue, P., Stasch, C. (2010),
Semantic Enablement for Spatial Data Infrastructures

Επίσημος διαδικτυακός τόπος της Εταιρείας Ελεύθερου Λογισμικού / Λογισμικού Ανοιχτού Κώδικα (ΕΛ/ΛΑΚ),
<http://www.ellak.gr/>

Επίσημος διαδικτυακός τόπος OGC,
<http://www.opengeospatial.org/>

Επίσημος διαδικτυακός τόπος του OpenLayers,
<http://openlayers.org>

Επίσημος διαδικτυακός τόπος του World Wide Web Schools,
<http://www.w3schools.com>

Επίσημος διαδικτυακός τόπος για το Σύστημα Διαχείρισης Βάσεων Δεδομένων PostgreSQL,
<http://www.postgresql.org/>

Επίσημος διαδικτυακός τόπος των κειμένων τεκμηρίωσης της επέκτασης PostGIS,
<http://postgis.refractory.net/documentation/>

Επίσημος διαδικτυακός τόπος του GeoServer,
<http://geoserver.org>

OpenLayers WFS-T using a GeoServer hosted PostGIS layer,

<http://www.gistutor.com/openlayers/22-advanced-openlayers-tutorials/47-openlayers-wfs-t-using-a-geoserver-hosted-postgis-layer.html>

Επίσημος διαδικτυακός τόπος του Οργανισμού Κτηματολογίου και Χαρτογραφίσεων Ελλάδος,

<http://www.okxe.gr>

Επίσημος διαδικτυακός τόπος της Εθνικής Υποδομής Γεωχωρικών Πληροφοριών (ΕΥΓΕΠ),

<http://www.inspire.okxe.gr>

Επίσημος διαδικτυακός τόπος της Federal Geographic Data Committee των Ηνωμένων Πολιτειών Αμερικής,

<http://www.fgdc.gov/>

Guidance Note 7-3: EPSG Registry Developers Guide,

<http://www.epsg.org/guides/docs/G7-3.pdf>

epsg projection 2100 - ggrs87 / greek grid

<http://spatialreference.org/ref/epsg/2100/>

epsg projection 4326 - wgs 84

<http://spatialreference.org/ref/epsg/4326/>

7. ΠΑΡΑΡΤΗΜΑ

7.1 Ο κώδικας του αρχείου OpenDataProject.html

```

<!DOCTYPE HTML>

<html xmlns="http://www.w3.org/1999/xhtml" lang="en-US">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>OpenDataProject-poisDB</title>

    <meta name="description" content="Η Ελλάδα επίσημο παράρτημα του Open Knowledge Foundation" />

    <link rel='stylesheet' id='admin-bar-css' href='http://okfn.gr/wp-includes/css/admin-bar.min.css?ver=3.6' type='text/css' media='all' />
    <link rel='stylesheet' id='bp-admin-bar-css' href='http://okfn.gr/wp-content/plugins/buddypress/bp-core/css/admin-bar.css?ver=1.6.4' type='text/css' media='all' />
    <link rel='stylesheet' id='okf-panel-master-frontend-css' href='http://okfn.gr/wp-content/plugins/okf-panel-master/assets/css/frontend.css' type='text/css' media='all' />
    <link rel='stylesheet' id='okf-wordpress-footer-frontend-css' href='http://okfn.gr/wp-content/plugins/okf-wordpress-footer/assets/css/frontend.css' type='text/css' media='all' />
    <!-- nagios_check timestamp:1399826556 --><!-- nagios_db_check sitename:OKFN Greece -->
    <link rel="stylesheet" href="http://okfn.gr/wp-content/themes/wordpress-theme-okfn/style.css?1391101935"/>
    <!-- Theme Settings -->
    <link rel="stylesheet" href="http://okfn.gr/wp-content/themes/wordpress-theme-okfn/css/sub-header.css?ver=1"/>
    <link rel="stylesheet" href="http://okfn.gr/wp-content/themes/wordpress-theme-okfn/css/grey.css?ver=1.1.2"/>

    <style type="text/css">
      @media (min-width: 980px) {
        .navbar {
          height: 65px;
        }
        .navbar .brand {
          line-height: 65px;
        }
        .navbar .brand img {
          margin-top: 15px;
          height: 35px;
        }
        .navbar .nav > li > a {
          padding-top: 24px;
          padding-bottom: 22px;
        }
        .navbar .sub-brand {
          line-height: 60px;
          padding-top: 5px;
        }
        .header-text {
          margin-top: 15px;
        }
        .navbar .navbar-inner .social-links, .navbar .navbar-inner
        .header-search {
          margin-top: 22px;
        }
      }
    </style>
  </head>
  <body>
  </body>
</html>

```

```

        #wpadminbar {
            display: none;
        }
        html {
            margin-top: 0px !important;
        }
    </style>

    <link rel="shortcut icon" href="http://assets.okfn.org/p/okfn/img/favicon.ico"
/>

    <script type="text/javascript">
        var Okfn = Okfn || {};
        // Make this variable available to Javascript
        Okfn.theme_directory = 'http://okfn.gr/wp-content/themes/wordpress-
theme-okfn';
    </script>

    <!--[if lt IE 9]>
    <script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
    <![endif]-->

    <meta property="og:type" content="article" />
    <meta property="article:publisher" content="https://www.facebook.com/okfngreece"
/>

    <!------->

    <!-- OpenLayers -->
    <script src="http://openlayers.org/dev/OpenLayers.js"></script>
    <!-- OpenStreetMap base layer js -->
    <script
src="http://www.openstreetmap.org/openlayers/OpenStreetMap.js"></script>
    <!-- Google Maps -->
    <script src="http://maps.google.com/maps/api/js?sensor=false"></script>
    <!-- myscript -->
    <script src="OpenDataProject.js" defer="defer" type="text/javascript"></script>
    <link rel="stylesheet" href="OpenDataProject.css"/>
    <!------->

    </head>

    <body onload="init()">

        <div id="container2" style="width:100%;height:100%">

            <div id="okf-panel" class="collapse">
                <iframe src="http://assets.okfn.org/themes/okfn/okf-panel.html"
scrolling="no"></iframe>
            </div>
            <header id="okfnEpikefalida" class="has-subheader">
                <div class="navbar">
                    <div class="navbar-inner">
                        <div class="container">

                            <div class="okfn-ribbon">
                                <a href="http://okfn.org/"
data-toggle="collapse" data-target="#okf-panel" title="Part of the Open Knowledge Foundation
Network">An Open Knowledge Foundation Site</a>

                            </div>
                        </div>
                    </div>
                </div>
            </header>
        </div>
    </body>
</html>

```



```

        <a data-target=".nav-collapse" data-
toggle="collapse" class="btn btn-navbar collapsed"> <span class="icon-bar"></span> <span
class="icon-bar"></span> <span class="icon-bar"></span> </a>
        <a title="Home" class="brand"
href="http://okfn.gr"> OKFN Greece </a>

        <nav class="nav-collapse collapse">
            <div class="social-links">
                <a class="twitter"
href="https://twitter.com/okfngr">twitter</a>
                <a class="facebook"
href="http://www.facebook.com/okfngreece">facebook</a>
            </div>
            <ul id="nav" class="nav">
                <li id="menu-item-1710"
class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1710 dropdown" data-
dropdown="dropdown">
                    <a href="#"
class=" dropdown-toggle">Αρχεία pois</a>
                    <ul
class="sub-menu dropdown-menu">
                        <li
id="menu-item-1711" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-
1711">
                            <a href="http://localhost:8080/OpenData/OpenDataUploadForm.php">Ανέβασμα αρχείου
Δεδομένων</a>
                        </li>
                        <li
id="menu-item-1709" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-
1709">
                            <a href="http://localhost:8080/OpenData/OpenDataDownloadForm.php">Λήψη αρχείου
Δεδομένων</a>
                        </li>
                    </ul>
                </li>
                <li id="menu-item-1706"
class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1706">
                    <a href=#
onclick="CreditsAlert()">Credits</a>
                </li>
                <li id="menu-item-1685"
class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1685">
                    <a
href="http://okfn.gr/projects/">Projects</a>
                </li>
                <li id="menu-item-1713"
class="menu-item menu-item-type-custom menu-item-object-custom menu-item-1713">
                    <a
href="http://okfn.gr/blog-magazine/">Blog</a>
                </li>
                <li id="menu-item-1689"
class="menu-item menu-item-type-custom menu-item-object-custom menu-item-1689 dropdown" data-
dropdown="dropdown">
                    <a class="
dropdown-toggle">Σχετικά με</a>
                    <ul
class="sub-menu dropdown-menu">
                        <li
id="menu-item-1690" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-
1690">
                            <a href="http://okfn.gr/central-okfn/">Το Κεντρικό OKFN</a>
                        </li>

```

```

</li>
id="menu-item-1691" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1691">
    <a href="http://okfn.gr/about/vision/">Όραμα</a>
</li>
</li>
id="menu-item-1721" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1721">
    <a href="http://okfn.gr/team/">Η ομάδα</a>
</li>
</li>
id="menu-item-1693" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1693">
    <a href="http://okfn.gr/open-data/">Ανοιχτά Δεδομένα</a>
</li>
</li>
id="menu-item-90" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-90">
    <a href="http://okfn.gr/greeklod/">Ελληνικά Συνδεδεμένα Δεδομένα</a>
</li>
</li>
id="menu-item-1698" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1698">
    <a href="http://okfn.gr/partners/">Συνεργάτες</a>
</li>
</ul>
</li>
<li id="menu-item-1688"
class="menu-item menu-item-type-post_type menu-item-object-page menu-item-1688">
    <a
href="http://okfn.gr/contact/">Επικοινωνία</a>
</li>
</ul>
</nav>
<!-- Disabled until I've got separate
images and confirmed link addresses -->
</div>
</div>
</div>
<form action="http://okblogfarm.org/search/" method="post"
class="search-form" role="search">
    <label for="search-terms" class="accessibly-
hidden">Search for:</label>
    <input type="text" id="search-terms" name="search-
terms" value="" />
    <label for="search-which" class="accessibly-
hidden">Search these:</label>
    <select name="search-which" id="search-which"
style="width: auto">
        <option value="members">Members</option><option
value="groups">Groups</option><option value="blogs">Blogs</option><option
value="posts">Posts</option>
    </select>
    <input type="submit" name="search-submit" id="search-
submit" value="Search" />
    <input type="hidden" id="_wpnonce" name="_wpnonce"
value="a28ceb4037" />
    <input type="hidden" name="_wp_http_referer" value="/"
/>
</form><!-- #search-form -->

```

```

        <div class="sub-header">
            <div class="container">
                <div class="row">
                    <div class="span8"></div>
                    <div class="span4"></div>
                </div>
            </div>
        </div>
    </header>

    <!--           <div class="container">           --->
    <div class="strapline" id="okfnEpikefalida2">
    <!--           <div class="inner">
        Η Ελλάδα επίσημο παράρτημα του Open Knowledge
Foundation
        </div><!-- /inner           -->
    </div><!-- /strapline -->

    <!--
    <div id="container2" style="width:100%;height:100%">

    <!--overflow:scroll;-->

    <div id="MyDiv" >
        <div id="Datasets"
style="width:25%;height:100%;float:left;overflow:auto;">
            <b>Datasets</b>
        </div>
        <div id="content" style="width:50%;height:100%;float:left;">
            <div id="map"></div>
            <div id="wrapper">
                <div id="location">           location</div>
                <div id="lf">>λ,φ :</div>
                <div id="scale"></div>
            </div>
        </div><!------- close context div           ---->
        <div id="resultsWfs"
style="width:25%;height:100%;float:left;overflow:auto;">
            <div id="nodelist">           <em>Click on the map to get
feature info</em></div>
        </div>
    </div>

    <div id="footer" style="clear:both;text-align:center;background-
color:#dcdcdc;bottom: 0;">
        Εθνικό Μετσόβιο Πολυτεχνείο 2014
    </div>

    </div><!------- close container div           ---->
    <!------           </div><!-- /container -->

    <!------->

    </body>
</html>

```

7.2 Ο κώδικας του αρχείου **OpenDataProject.js**

```
/**  
 * @author Pavlos_Notebook  
 */  
////////////////////////////////////  
//                                                                global variables  
//                                                                //  
////////////////////////////////////  
var keimenoCredits="Καποιο κειμενο ";  
var map;  
var untiled;  
var tiled;  
var tiled1;  
var pureCoverage = false;  
  
var filtro;  
var filtroTab= new Array();  
var eventReg;  
  
var WGS84;  
var WGS84_google_mercator;  
var greekgrid;  
  
var urlBase = "http://localhost:8088/geoserver/";//Notebook geoserver  
var layerNamespace = "OpenDataProject";  
var poisLayer=":poisTable";  
var urlTypeNamePois = "&typeName=" + layerNamespace + poisLayer;  
var poisColumn1='poisdataset';  
var poisColumn2='poisdescr';  
  
var datasetLayer="datasetTable";  
var urlTypeNameDataset = "&typeName=" + layerNamespace + ":"+ datasetLayer;  
var datasetTabCols=["datasetname", "datasetdescr"];  
var datasetpoiicon="datasetpoiicon";  
var imgDiv=' 0) {  
        height=height-70;  
        document.getElementById("container2").style.height = height + "px";  
    }  
  
    var myDiv = document.getElementById('okf-panel'); //get #myDiv  
    var okf_panel=myDiv.clientHeight;  
    var myDiv2 = document.getElementById('okfnEpikefalida'); //get #myDiv
```

```

var okfnEpikefalidaHeigth=myDiv2.clientHeight;
var myDiv3 = document.getElementById('okfnEpikefalida2'); //get #myDiv
var okfnEpikefalidaHeigth2=myDiv3.clientHeight;
var myDiv3 = document.getElementById('footer'); //get #myDiv
var footerHeigth=myDiv3.clientHeight;

var EverythingElse= okfnEpikefalidaHeigth+okfnEpikefalidaHeigth2+footerHeigth;
var temp=(EverythingElse*100)/height;
var myHeight=100-temp;
var contHeight=height-okfnEpikefalidaHeigth-okfnEpikefalidaHeigth2-footerHeigth;
contHeight=height-EverythingElse;

document.getElementById("MyDiv").style.height = contHeight + "px";

format = 'image/png';
if (pureCoverage) {
    document.getElementById('filterType').disabled = true;
    document.getElementById('filter').disabled = true;
    document.getElementById('antialiasSelector').disabled = true;
    document.getElementById('updateFilterButton').disabled = true;
    document.getElementById('resetFilterButton').disabled = true;
    document.getElementById('jpeg').selected = true;
}

var bounds = new OpenLayers.Bounds(20.8499984741211, 35.3324966430664, 25.0800018310547,
40.7398452758789);
var options = {
    controls : [],
    maxExtent : bounds,
    maxResolution : 0.0211224555969238,
    projection : "EPSG:4326",
    displayProjection : "EPSG:4326",
    units : 'degrees'
};

map = new OpenLayers.Map('map', options);
// projections
WGS84 = new OpenLayers.Projection("EPSG:4326"); // WGS84 projection
WGS84_google_mercator = new OpenLayers.Projection("EPSG:900913"); // WGS84 Google
projection
greekgrid = new OpenLayers.Projection("EPSG:2100"); // EGSA'87 projection

//base layers
var openstreetmap = new OpenLayers.Layer.OSM();
var google_maps = new OpenLayers.Layer.Google("Google Maps", {
    numZoomLevels : 20
});
var google_satellite = new OpenLayers.Layer.Google("Google Satellite", {
    type : google.maps.MapTypeId.SATELLITE,
    numZoomLevels : 20
});

map.addLayers([openstreetmap, google_maps, google_satellite/*,ktim*/]);
map.addControl(new OpenLayers.Control.LayerSwitcher());
map.addControl(new OpenLayers.Control.PanZoomBar({
    position : new OpenLayers.Pixel(2, 15)
}));

map.addControl(new OpenLayers.Control.Navigation());
map.addControl(new OpenLayers.Control.Scale($('scale')));
map.addControl(new OpenLayers.Control.MousePosition({
    element : $('location')
}));

var mapextent = new OpenLayers.Bounds(20.8499984741211, 35.3324966430664,
25.0800018310547, 40.7398452758789).transform(WGS84, WGS84_google_mercator);
map.zoomToExtent(mapextent);

```

```

        var url2=urlBase+layerNamespace+urlWfs1+urlGetFeature+urlTypeNameDataset;
        fillDatasetDiv(url2,datasetLayer,datasetTabCols);
    }//end_init

    ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
    //                                                                                      functions
    //                                                                                      //
    ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
    /*-----*/

function getViewportHeight() {
    var h = 0;
    if(self.innerHeight){
        h = window.innerHeight;
    }
    else if(document.documentElement && document.documentElement.clientHeight){
        h = document.documentElement.clientHeight;
    }
    else if(document.body) {
        h = document.body.clientHeight;
    }
    return h;
}

/*-----*/

function setHTML(response) {

    var txt=response.responseText;
    var xmlDoc;

    if (window.DOMParser){
        parser=new DOMParser();
        xmlDoc=parser.parseFromString(txt,"text/xml");
    } else {
        // Internet Explorer
        xmlDoc=new ActiveXObject("Microsoft.XMLDOM");
        xmlDoc.async=false;
        xmlDoc.loadXML(txt);
    }

    var DbTab = "poisTable";
    var DbTabFull= layerNamespace+"."+DbTab;
    var htmlText="<table border='1' style='border-color:#585858;border-
style:groove;border-width:5px;padding:0'>";
    var x=xmlDoc.getElementsByTagName(DbTabFull);

    if (x.length == 0) {
        x = xmlDoc.getElementsByTagName(DbTab);
        for (var i=0;i<x.length;i++){
            htmlText=htmlText+("<tr padding:0><td>");

htmlText=htmlText+(x[i].getElementsByTagName(poisColumn1)[0].childNodes[0].nodeValue);
            htmlText=htmlText+("</td><td>");

htmlText=htmlText+(x[i].getElementsByTagName(poisColumn2)[0].childNodes[0].nodeValue);
            htmlText=htmlText+("</td></tr>");
        }
        htmlText=htmlText+("</table>");
    }else {
        for (var i=0;i<x.length;i++){
            htmlText=htmlText+("<tr padding:0><td>");

htmlText=htmlText+(x[i].getElementsByTagName(layerNamespace+"."+poisColumn1)[0].childNodes[0].no
deValue);
            htmlText=htmlText+("</td><td>");

```

```

htmlText=htmlText+(x[i].getElementsByTagName(layerNamespace+" "+poisColumn2)[0].childNodes[0].nodeValue);
                htmlText=htmlText+("</td></tr>");
            }

        }//end_elegxos Chrome/Firefox

        htmlText=htmlText+("</table>");
        document.getElementById('odelist').innerHTML = htmlText;

};//end_setHTML

/*-----*/

function fillDatasetDiv(url,DbTab,TabColsArr) {
    var xmlDoc;
    var request;

    if(window.XMLHttpRequest){ // code for IE7+, Firefox, Chrome, Opera,
Safari
        request = new XMLHttpRequest();
    } else{
        // code for IE6, IE5
        request=new ActiveXObject("Microsoft.XMLHTTP");
    }
    request.open("GET", url, true);
    request.onreadystatechange = function() {
        if (request.readyState == 4 && request.status == 200) {
            var txt = request.responseText;
            if (window.DOMParser){
                parser=new DOMParser();
                xmlDoc=parser.parseFromString(txt,"text/xml");
            } else {
                // Internet Explorer
                xmlDoc=new ActiveXObject("Microsoft.XMLDOM");
                xmlDoc.async=false;
                xmlDoc.loadXML(txt);
            }

            var DbTabFull= layerNamespace+" "+DbTab;
            var x=xmlDoc.getElementsByTagName(DbTabFull);
            var htmlText="<form>";

            if (x.length == 0) { //Kwdikas gia Chrome

                x = xmlDoc.getElementsByTagName(DbTab);

                for (var i=0;i<x.length;i++){
                    htmlText=htmlText+('<fieldset style="border-color:#585858;border-style:groove;border-width:5px;padding:5;">');

                    var fid=x[i].getAttribute("fid");
                    fid=fid.split(".")[1];

                    htmlText=htmlText+imgDiv+(x[i].getElementsByTagName(datasetpoiicon)[0].childNodes[0].nodeValue)+'alt="Not Available" width="20" height="20" style="display:block; margin: auto;">';

                    for(var ii=0;ii<TabColsArr.length;ii++){

                        htmlText=htmlText+

"<b>"+datasetTabLabels[ii]+ "</b>";

                    htmlText=htmlText+(x[i].getElementsByTagName(TabColsArr[ii])[0].childNodes[0].nodeValue);

```

```

                                htmlText=htmlText+("<hr style='margin-
top:0;margin-bottom:0;'>");
                                //end loop Columns

                                var
name=x[i].getElementsByName(TabColsArr[0])[0].childNodes[0].nodeValue;
                                htmlText=htmlText+'<input type="checkbox" name="'+name+'
id="'+ fid + '"onclick=handleClick(this);><b>Εμφάνιση στο χάρτη</b><br>';
                                htmlText=htmlText+("</fieldset>");
                                //End looping xmlDocResponse

                                }else {
//Kwdikas gia Firefox
                                for (var i=0;i<x.length;i++){
                                htmlText=htmlText+('<fieldset style="border-
color:#585858;border-style:groove;border-width:5px;padding:5">');
                                var fid=x[i].getAttribute("fid");
                                fid=fid.split(".")[1];

htmlText=htmlText+imgDiv+(x[i].getElementsByName(layerNamespace+": "+datasetpoiicon)[0].childN
odes[0].nodeValue)+"alt="Not Available" width="20" height="20" style="display:block; margin:
auto;">' ;

                                for(var ii=0;ii<TabColsArr.length;ii++){

                                htmlText=htmlText+
" <b>"+datasetTabLabels[ii]+ " </b>";

htmlText=htmlText+(x[i].getElementsByName(layerNamespace+": "+TabColsArr[ii])[0].childNodes[0]
.nodeValue);

                                htmlText=htmlText+("<hr style='margin-
top:0;margin-bottom:0;'>");
                                //end loop Columns

                                var
name=x[i].getElementsByName(layerNamespace+": "+TabColsArr[0])[0].childNodes[0].nodeValue;
                                htmlText=htmlText+'<input type="checkbox"
name="'+name+' id="'+ fid + '"onclick=handleClick(this);><b>Εμφάνιση στο χάρτη</b><br>';

                                htmlText=htmlText+("</fieldset>");

                                //End looping xmlDocResponse

                                //end_elegxos Chrome/Firefox

                                htmlText=htmlText+("</form>");
                                document.getElementById('Datasets').innerHTML = htmlText;

                                } //end if ola ok
}; //ENDrequest.onload

                                request.send(null);
                                //endfunction fillDatasetDiv
}

/*-----*/
function handleClick(cb) {

                                if(cb.checked){
                                filtroTab.splice(filtroTab.length,0,cb.id);
                                setUpLayerWMS(cb.name,cb.id,filtroTab);

                                } else{

                                filtroTab.splice(filtroTab.indexOf(cb.id),1);
                                map.removeLayer(map.getLayer(cb.id));
                                removeLayerEvent(cb.name,cb.id,filtroTab);

```



```

    }//end else
} //End_handleClick_function

/*-----*/
function setUpLayerWMS(name,id,poisIdTab) {
    filtro="";
    for(var i=0;i<poisIdTab.length;i++){
        filtro+='poidataset_id == '+poisIdTab[i]+' OR ';
    }
    filtro=filtro.substring(0,filtro.length-4);

    var par1={
        LAYERS : layerNamespace + poisLayer,
        CQL_FILTER: filtro,
        STYLES : '',
        format : format,
        transparent : 'true',
        tiled : true,
        tilesOrigin : map.maxExtent.left + ',' + map.maxExtent.bottom
    };
    var par2={
        buffer : 0,
        displayOutsideMaxExtent : true,
        isBaseLayer : false,
        yx : {'EPSG:4326' : true}
    };
    tiled1 = new OpenLayers.Layer.WMS(name, urlBase + layerNamespace + "/wms",par1,par2);
    tiled1.id=id;

    map.addLayer(tiled1);

    map.events.unregister('click', map,eventReg);
    map.events.register('click', map, eventReg);
} //End_setUpLayerWMS_function

/*-----*/
function removeLayerEvent(name,id,poisIdTab) {
    filtro="";
    for(var i=0;i<poisIdTab.length;i++){
        filtro+='poidataset_id == '+poisIdTab[i]+' OR ';
    }
    filtro=filtro.substring(0,filtro.length-4);

    map.events.unregister('click', map,eventReg);
    map.events.register('click', map, eventReg);
} //End_removeLayerEvent_function

/*-----*/
function eventReg(e) {
    var filtroParam=filtro;
    var mapParam=tiled1;
    document.getElementById('nodelist').innerHTML = "Loading... please wait...";
    var params = {
        REQUEST : "GetFeatureInfo",
        EXCEPTIONS : "application/vnd.ogc.se_xml",
        BBOX : map.getExtent().toBBOX(),
        SERVICE : "WMS",

```

```

INFO_FORMAT : 'application/vnd.ogc.gml/3.1.1',
////////////////////////////////////
QUERY_LAYERS : layerNamespace + poisLayer,

CQL_FILTER: filtroParam,
FEATURE_COUNT : 50,
Layers : layerNamespace + poisLayer ,
vendorParams: {propertyName:['poisdataset',
'poisdescr']},////////////////////////////////////
WIDTH : map.size.w,
HEIGHT : map.size.h,
format : format,
styles : mapParam.params.STYLES,
srs : mapParam.params.SRS
};
// wms 1.3 h wms 1.1
if (mapParam.params.VERSION == "1.3.0") {
    params.version = "1.3.0";
    params.j = parseInt(e.xy.x);
    params.i = parseInt(e.xy.y);
} else {
    params.version = "1.1.1";
    params.x = parseInt(e.xy.x);
    params.y = parseInt(e.xy.y);
}
if (mapParam.params.CQL_FILTER != null) {
    params.cql_filter = mapParam.params.CQL_FILTER;
}
if (mapParam.params.FILTER != null) {
    params.filter = mapParam.params.FILTER;
}
OpenLayers.loadURL(urlBase + layerNamespace + "/wms", params, this, setHTML,
setHTML);
OpenLayers.Event.stop(e);

} //End_eventReg_Function

/*-----*/

function CreditsAlert(){
alert(keimenoCredits);
}

```

7.3 Ο κώδικας του αρχείου OpenDataProject.css

```
/* General settings */

body {
    font-family: Verdana, Geneva, Arial, Helvetica, sans-serif;
    font-size: small;
    color: #595959;
}

/* The map and the location bar */
#map {
    clear: both;
    position: relative;
    width: 100%;
    height: 100%;
    top: 0px;
    bottom: 10px;
    border: 1px solid black;
}

#wrapper {
clear: both;
    position: relative;
    width: 100%;
    height: 10%;
bottom: 0px;
}

#location {
    float: right;
}

/* ----- */
#resultsWfs{
    overflow: auto;
}
/* ----- */

.olControlLayerSwitcher {
    position: relative;
    top: 0px;
    right: 0;
    width: 20em;
    font-family: sans-serif;
    font-weight: bold;
    margin-top: 0px;
    margin-left: 3px;
    margin-bottom: 0px;
    font-size: small;
    color: white;
    background-color: transparent;
}

.olControlLayerSwitcher .layersDiv {
    padding-top: 0px;
    padding-left: 10px;
    padding-bottom: 5px;
    padding-right: 10px;
}
```

```

        font-size: small;

        /*background-color: "00BFFF";*/
    }

    .olControlLayerSwitcher .layersDiv .baseLbl, .olControlLayerSwitcher .layersDiv .dataLbl {
        margin-top: 0px;
        margin-left: 3px;
        margin-bottom: 3px;
        font-size: small;
    }

    .olControlLayerSwitcher .layersDiv .baseLayersDiv, .olControlLayerSwitcher .layersDiv
    .dataLayersDiv {
        padding-left: 10px;
        font-size: small;
    }

    label.labelSpan.olButton {
        display:inline;
    }

    }

    .olControlLayerSwitcher .maximizeDiv, .olControlLayerSwitcher .minimizeDiv {
        width: 18px;
        height: 18px;
        top: 0px;
        right: 0;
        cursor: pointer;
        font-size: small;
    }

    }

    .olControlLayerSwitcher {
    position: absolute;
    top: 25px;
    right: 0;
    width: 20em;
    font-family: sans-serif;
    font-weight: bold;
    margin-top: 3px;
    margin-left: 3px;
    margin-bottom: 3px;
    font-size: smaller;
    color: white;
    background-color: transparent;
    text-align: left;
    -ms-touch-action: none;
    }

    /* ----- */

    /* Styles used by the default GetFeatureInfo output, added to make IE happy */
    table.featureInfo, table.featureInfo td, table.featureInfo th {
        border: 1px solid #ddd;
        border-collapse: collapse;
        margin: 0;
        padding: 0;
        font-size: 90%;
        padding: .2em .1em;
    }

    table.featureInfo th {

```

```

        padding: .2em .2em;
        font-weight: bold;
        background: #eee;
    }

    table.featureInfo td {
        background: #fff;
    }

    table.featureInfo tr.odd td {
        background: #eee;
    }

    table.featureInfo caption {
        text-align: left;
        font-size: 100%;
        font-weight: bold;
        padding: .2em .2em;
    }

/* ---- mycss ----- */

#ekt {
    background-color: #d9d9d9;
    -webkit-border-radius: 6px;
    border-radius: 6px;
    margin: 10px 0px 0px 0px;
    padding: 0px;
    border: 1px solid #d9d9d9;
}

#lf {
    float: right;
    margin-right: 10px;
}

div.menu {
    float: left;
    margin: 0 5px 0 5px;
}

div.lst {
    float: left;
    margin: 0 5px 0 5px;
}

#my{
    display: block;
}

/* ---- end mycss ----- */

/*
fieldset {
display: block;
-webkit-margin-start: 2px;
-webkit-margin-end: 2px;
-webkit-padding-before: 0.35em;
-webkit-padding-start: 0.75em;
-webkit-padding-end: 0.75em;
-webkit-padding-after: 0.625em;
}

*/

input[type="checkbox"] {
    margin: 4px 4px 4px;
}

```

```
margin-top: 4px;
margin-right-value: 4px;
margin-bottom: 4px;
margin-left-value: 4px;
margin-left-ltr-source: physical;
margin-left-rtl-source: physical;
margin-right-ltr-source: physical;
margin-right-rtl-source: physical;
line-height: normal;
cursor: pointer;
}

html,body{
    height: 100%
}

/*
footer{
    height: 100px;
    width:100%;
    position: absolute;
    left: 0;
    bottom: 0;
}
*/

#footer {
    position: fixed;
    bottom: 0;
    width: 100%;
}
```

7.4 Ο κώδικας του αρχείου OpenDataUploadForm.php

```

<!DOCTYPE HTML>
  <head>

      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
      <meta name="description" content="Η Ελλάδα επίσημο παράρτημα του Open Knowledge
Foundation" />

      <title>OpenDataProject-UploadForm</title>
      <link rel="stylesheet" href="OpenDataUploadForm.css">
      <script src="FileReader.js" defer="defer" type="text/javascript"></script>
      <meta charset="utf-8">

  </head>

  <body onload="checkFileAPI();">

<!-- -----
----- -->

      <h2> Ανέβασμα Αρχείου Δεδομένων </h2>
      <div id="my">
          <form name="myform" method="post" action="OpenDataDBInserts.php"
onsubmit="createQueries()" enctype="multipart/form-data" accept-charset="utf-8">
              <fieldset>
                  <legend><h3> Πληροφορίες για τον Uploader
</h3></legend>
                  <select id="personalInfo_sel"
onchange="personalDivChange(this.value);">
                      <option name="personalInfo" id="reqInfo"
value="reqInfo" > Απαραίτητες πληροφορίες </option>
                      <option name="personalInfo" id="allInfo"
value="allInfo"> Περισσότερες πληροφορίες </option>
                  </select>
                  <br/><br/>

                  <label for="contributornickname" style="display:inline-
block"><b>Ψευδώνυμο : </b></label>
                  <input type="text" id="contributornickname"
name="contributornickname" value="" maxlength="100" required/>
                  <br/>

                  <div class="center_div" id="personalDiv" hidden>
                      <label for="contributoremail"
style="display:inline-block">E-mail :</label>
                      <input type="email" id="contributoremail"
name="contributoremail" value="" maxlength="100"/>
                      <br/>

                      <label for="contributorfirstname"
style="display:inline-block">Όνομα : </label>
                      <input type="text" id="contributorfirstname"
name="contributorfirstname" value="" maxlength="100"/>
                      <br/>

                      <label for="contributorlastname"
style="display:inline-block">Επώνυμο : </label>
                      <input type="text" id="contributorlastname"
name="contributorlastname" value="" maxlength="100"/>

```

```

<br/>
<label for="confributorfacebook"
style="display:inline-block">Facebook : </label>
<input type="url" id="confributorfacebook"
name="confributorfacebook" value="" maxlength="100"/>
<br/>
<label for="contributortwitter"
style="display:inline-block">Twitter : </label>
<input type="url" id="contributortwitter"
name="contributortwitter" value="" maxlength="100"/>
<br/>
<label for="contributorwebpageurl"
style="display:inline-block">Ιστοσελίδα : </label>
<input type="url" id="contributorwebpageurl"
name="contributorwebpageurl" value="" maxlength="200"/>
<br/>
</div>
</fieldset>
<p/>
<fieldset>
<legend> <h3>Στοιχεία που αφορούν το Dataset [DB2]</h3>
<p/>
</div>
<?php
style="display:inline-block">ΚΑΤΗΓΟΡΙΑ : </label>';
echo'<select id="selCategory"
name="selCategory" onchange="selCategChange(this.value)">';
//////////////////////////////////// - Notebook -
////////////////////////////////////
$host = "host=localhost";
$port = "port=5432"; //NotebookUbuntuPostgresServerPort
$credentials = "user=postgres password=pavlos7"; //NotebookW7PostgresServerUser
$dbname = "dbname=GeoDB"; // DB2NotebookW7
$schema = "mydb";
////////////////////////////////////
////////////////////////////////////

/*
//////////////////////////////////// - Desktop -
////////////////////////////////////
$host = "host=localhost";
$port = "port=5431";
//NotebookUbuntuPostgresServerPort
$credentials = "user=postgres
password=pavlos7"; //NotebookW7PostgresServerUser
$dbname = "dbname=GeoDB"; // DB2NotebookW7
$schema = "mydb";
////////////////////////////////////
////////////////////////////////////
*/

```



```

$credentials" );
                                $db = pg_pconnect( "$host $port $dbname
                                if(!$db){
                                    echo "Error : Unable to open database\n";
                                } else {

                                    $sql='SELECT categoryid,
categorydescription, categoryparentid FROM mydb."categoryTable" ORDER BY categoryid';
                                    $ret = pg_query($db, $sql);

                                    while ($row = pg_fetch_row($ret)) {
                                        echo '<option
value="' . $row['0'] . '" id="' . $row['0'] . '" name="' . $row['1'] . '">' . $row['1'] . '</option>';

                                        }

                                    }

                                pg_close($db);
                                echo '</select>';

?>
                                <input type="text" id="newCategory"
name="newCategory" value="" maxlength="100" hidden />

                                </div>
                                <br/>
                                <label for="datasetname"><b>Όνομα Dataset :</b></label>
                                <input type="text" id="datasetname" name="datasetname"
value="" maxlength="100"required/>

                                <br/>
                                <label for="datasetdescr">Περιγραφή Dataset :</label>
                                <textarea rows="2" id="datasetdescr"
name="datasetdescr" value=""></textarea>
                                <br/>
                                <p/>
                                <label for="poisFile"><b>Αρχείο pois</b></label>
                                <input type="file" id="poisFile"
onchange='readText(this)' required />

                                <br/>
                                <p/>

                                <label id="butlabel"for="but">Εμφάνιση ρυθμίσεων
αρχείου</label>

                                <button id="but" type="button" value="0"
onclick="showFileSettingsDiv()">\</button>

                                <!--
                                <div class="center_div" >
                                <div class="center_div" id="FileSettingsDiv" hidden>

                                    <fieldset style="width:150px;">
                                        <legend> Σύστημα Αναφοράς </legend>

                                        <label for="4326"style="width:120px;" >
λ , φ [WGS 84] </label>
                                        <input type="radio" name="srid"
id="4326" value="4326" checked/>

                                        <br/>

                                        <label for="2100"style="width:120px;">
X , ψ [ΕΓΣΑ 87] </label>
                                        <input type="radio" name="srid"
id="2100" value="2100"/>

```

```

</fieldset>
<p/>
<fieldset style="width:180px;">
  <legend> Γραμμογράφιση αρχείου
</legend>
<label for="xydata" style="width:150px;">
  >λ ή X , φ ή Y , data </label>
  <input type="radio"
name="grammografisi" id="xydata" value="1" checked/>
  <br/>
  <label for="yxdata" style="width:150px;">
  >φ ή Y , λ ή X , data </label>
  <input type="radio"
name="grammografisi" id="yxdata" value="2" />
  <br/>
  <label for="dataxy" style="width:150px;">
  >data , λ ή X , φ ή Y </label>
  <input type="radio"
name="grammografisi" id="dataxy" value="3" />
  <br/>
</fieldset>
<p/>
  <label for="file">Εικονίδιο dataset:</label>
  <input type="file" name="file" id="file" /><br>
  <input type="submit" name="submit" value="Submit"/>
-->
</div>
<input type="hidden" id="DataFromFileDB2"
name="DataFromFileDB2" value=""/>
<br/>
<!-- </div> -->
</fieldset>
<p/>
  <input type="submit" name="submit" value="Υποβολή Φόρμας"/>
</form>
<p/>
<hr/>
</div>
</body>
</html>

```

7.5 Ο κώδικας του αρχείου OpenDataUploadForm.css

```
/* General settings */
body {
    font-family: Verdana, Geneva, Arial, Helvetica, sans-serif;
    font-size: small;
}

/* ----- */

.my_linkable_button {
    background: none repeat scroll 0 0 #075482;

    position: absolute;
    right: 0px;

    border: medium none;
    color: #FFFFFF;
    cursor: pointer;
    margin-bottom: 26px;
    padding: 8px;
}

label{
display:inline-block;
vertical-align:middle;
width:150px;
}

.center_div {
    border: 1px solid gray;
    /* color: #F0F8FF; */
    background-color: #E6E6E6
;
    text-align: left;
}
```

7.6 Ο κώδικας του αρχείου FileReader.js

```
var reader;
var butval0='\\/';
var butval1='/\\';
var butlabel0='Εμφάνιση ρυθμίσεων αρχείου';
var butlabel1='Απόκρυψη ρυθμίσεων αρχείου';

/**
 * Check for the various File API support.
 */
function checkFileAPI() {
    if (window.File && window.FileReader && window.FileList && window.Blob) {
        reader = new FileReader();
        return true;
    } else {
        alert('The File APIs are not fully supported by your browser. Fallback required.');
```

```

/**
 * read text input
 */
function readText(filePath) {
    var output = ""; //placeholder for text output
    if(filePath.files && filePath.files[0]) {
        reader.onload = function (e) {
            output = e.target.result;
            displayContents(output);
        }; //end onload()
        reader.readAsText(filePath.files[0]);
    } //end if html5 filelist support
    else if(ActiveXObject && filePath) { //fallback to IE 6-8 support via ActiveX
        try {
            reader = new ActiveXObject("Scripting.FileSystemObject");
            var file = reader.OpenTextFile(filePath, 1); //ActiveX File Object
            output = file.ReadAll(); //text contents of file
            file.Close(); //close file "input stream"
            displayContents(output);
        } catch (e) {
            if (e.number == -2146827859) {
                alert('Unable to access local files due to browser security settings. ' +
                    'To overcome this, go to Tools->Internet Options->Security->Custom Level. '
                    +
                    'Find the setting for "Initialize and script ActiveX controls not marked as
                    safe" and change it to "Enable" or "Prompt"');
            }
        }
    }
    else { //this is where you could fallback to Java Applet, Flash or similar
        return false;
    }
    return true;
}

/**
 * display content using a basic HTML replacement
 */
function displayContents(txt) {
    txt1=txt.replace(/\n/g,"<br>");
    // var e1 = document.getElementById('main');
    // e1.innerHTML = txt1; //display output in DOM

    var kryfo = document.getElementById('DataFromFileDB2');
    // kryfo.value = txt1; //display output in DOM
    kryfo.value = txt; //display output in DOM
}

function personalDivChange(){
    var e = document.getElementById("personalInfo_sel");
    var strUser = e.options[e.selectedIndex].value;
    // alert(strUser);
    var div = document.getElementById('personalDiv');
    if(strUser == "reqInfo"){
        // hide
        //div.style.visibility = 'hidden';
        // OR
        div.style.display = 'none';
    }else{
        // show
        //div.style.visibility = 'visible';
    }
}

```

```

        // OR
        div.style.display = 'block';
    }
}

function selCategChange(){
    var sel = document.getElementById('selCategory');
    var newCatText = document.getElementById('newCategory');
    /*
    alert("id= "+sel.id);
    alert("value= "+sel.value);
    alert("name= "+newCatText.name);
    */
    if(sel.value == 2){
        newCatText.style.display = 'inline-block';
    }else{
        newCatText.style.display = 'none';
    }
}

function showFileSettingsDiv(){
    var but = document.getElementById('but');
    var butlabel = document.getElementById('butlabel');
    if(but.value=="0"){
        but.value="1";
        but.innerHTML=butval1;
        butlabel.innerHTML=butlabel1;
    }else{
        but.value="0";
        but.innerHTML=butval0;
        butlabel.innerHTML=butlabel0;
    }
    var div = document.getElementById('FileSettingsDiv');
    div.style.display = 'block';
}

function createQueries(){
}

```

7.7 Ο κώδικας του αρχείου OpenDataDBInserts.php

```

<!DOCTYPE HTML>
<head>
    <title>OpenDataUploadForm</title>
    <link rel="stylesheet" href="OpenDataUploadForm.css">
    <meta charset="utf-8">
</head>
<body>

<?php

//////////////////// - Notebook -
////////////////////
    $host = "host=localhost";
    $port = "port=5432"; //NotebookUbuntuPostgresServerPort
    $credentials = "user=postgres password=pavlos7"; //NotebookW7PostgresServerUser

```

```

        $dbname      = "dbname=GeoDB";// DB2NotebookW7
        $schema      = "mydb";
////////////////////////////////////
////////////////////////////////////

        $datasetTable="datasetTable";
        $poisTable="poisTable";
        $attributeTable="attributeTable";
        $categoryTable="categoryTable";
        $contributorTable="contributorTable";
        $dsaTable="dsaTable";
        $DatasetTableColNamesStr="(datasetname, datasetdescr, datasetdate, dataseticon,
datasetpoiicon, contributortable_contributorid, categorytable_categoryid)";
        $poisTableColNamesStr="(poisdescr, poisdataset, poisgeom, poisdataset_id)";
        $contributorTableColNamesStr="(contributornickname, contributoremail,
contributorfirstname, contributorlastname, contributorfacebook, contributortwitter,
contributorwebpageurl, contributorregdate, contributorstatus)";
        $categoryTableColNamesStr="(categorydescription, categoryparentid)";
        $attributeTableColNamesStr="(attributeid, attributename)";
        $dsaTableColNamesStr="(attributetable_attributeid, datasettable_dataset_id,
attrvalue)";

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

        $datasetname=$_POST['datasetname'];
//
        $UploaderName=$_POST['UploaderName'];
        $datasetdescr=checkPOST($_POST['datasetdescr']);

        echo "datasetname= ".$datasetname."<p>";
        echo "datasetdescr= ".$datasetdescr."<p>";

        $contributornickname=$_POST['contributornickname'];
        $contributoremail=checkPOST($_POST['contributoremail']);
        $contributorfirstname=checkPOST($_POST['contributorfirstname']);
        $contributorlastname=checkPOST($_POST['contributorlastname']);
        $contributorfacebook=checkPOST($_POST['contributorfacebook']);
        $contributortwitter=checkPOST($_POST['contributortwitter']);
        $contributorwebpageurl=checkPOST($_POST['contributorwebpageurl']);
        $datasetpoiicon=checkPOST($_FILES["file"]["name"]);

        if($datasetpoiicon != null){
            $datasetpoiicon= mt_rand();
        }
////////////////////////////////////

//ARXH Block gia eikona
        $allowedExts = array("gif", "jpeg", "jpg", "png");
        $temp = explode(".", $_FILES["file"]["name"]);
        $extension = end($temp);
        $datasetpoiicon=$datasetpoiicon.'.'.$extension;
        //echo $_FILES["file"]["name"];
        if ((($_FILES["file"]["type"] == "image/gif")
            || ($_FILES["file"]["type"] == "image/jpeg")
            || ($_FILES["file"]["type"] == "image/jpg")
            || ($_FILES["file"]["type"] == "image/pjpeg")
            || ($_FILES["file"]["type"] == "image/x-png")
            || ($_FILES["file"]["type"] == "image/png"))
            && ($_FILES["file"]["size"] < 20000)
            && in_array($extension, $allowedExts)) {
            if ($_FILES["file"]["error"] > 0) {
                echo "Return Code: " . $_FILES["file"]["error"] . "<br>";
            } else {
                echo "Upload: " . $_FILES["file"]["name"] . "<br>";
                echo "Type: " . $_FILES["file"]["type"] . "<br>";
            }
        }

```

```

        echo "Size: " . ($_FILES["file"]["size"] / 1024) . " kB<br>";
        echo "Temp file: " . $_FILES["file"]["tmp_name"] . "<br>";
        if (file_exists("upload/" . $_FILES["file"]["name"])) {
            echo $_FILES["file"]["name"] . " already exists. ";
        } else {
            move_uploaded_file($_FILES["file"]["tmp_name"], "upload/" .
$datasetpoiicon);
//            echo "Stored in: " . "upload/" . $_FILES["file"]["name"];
        }
    }
} else {
    echo "Invalid file";
}
//TELOS Block

/*
$datasetname=$_POST['datasetname'];
// $UploaderName=$_POST['UploaderName'];
$datasetdescr=checkPOST($_POST['datasetdescr']);

echo "datasetname= ".$datasetname."<p>";
echo "datasetdescr= ".$datasetdescr."<p>";

$contributornickname=$_POST['contributornickname'];
$contributoremail=checkPOST($_POST['contributoremail']);
$contributorfirstname=checkPOST($_POST['contributorfirstname']);
$contributorlastname=checkPOST($_POST['contributorlastname']);
$contributorfacebook=checkPOST($_POST['contributorfacebook']);
$contributortwitter=checkPOST($_POST['contributortwitter']);
$contributorwebpageurl=checkPOST($_POST['contributorwebpageurl']);
// $datasetpoiicon=checkPOST($_FILES["file"]["name"]);

if($datasetpoiicon != null){
    $datasetpoiicon= $contributornickname.'_'.$datasetname.'_.mt_rand();
}

*/

$selCategory=$_POST['selCategory'];
if($selCategory == 2){
    $newCategory=checkPOST($_POST['newCategory']);
}else{
    $newCategory=null;
}

$srid=$_POST['srid'];

$grammografisi=$_POST['grammografisi'];

////////////////////////////////////
$DataFromFile=$_POST['DataFromFileDB2'];
$DataFromFile = str_replace('"', "", $DataFromFile);
$DataFromFile = str_replace("'", "", $DataFromFile);
$grammesArxeioy = explode("\n", $DataFromFile);

echo '$DataFromFile';
echo '<p/>';

if(strlen(end($grammesArxeioy))=='){
    $grammesArxeioy =array_slice($grammesArxeioy, 0, -1);
}
$arrayLength = count($grammesArxeioy);
echo "To arxeio exei $arrayLength grammes";
echo '<p/>';
foreach ($grammesArxeioy as $value) {

```

```

        $value = str_replace("", "", $value);
        $value = str_replace("'", "", $value);
        echo"$value."<br>";
    }
    unset($value); // break the reference with the last element
    //////////////////////////////////////

echo'<p/>';
echo"$contributornickname."<br/>";
echo"$contributoremail."<br/>";
echo"$contributorfirstname."<br/>";
echo"$contributorlastname."<br/>";
echo"$contributorfacebook."<br/>";
echo"$contributortwitter."<br/>";
echo"$contributorwebpageurl."<br/>";

echo'<p/>';
echo"$datasetpoiicon."<br/>";

echo"$selCategory."<br/>";
echo"$newCategory."<br/>";
echo'srid= '.$srid."<br/>";
echo'grammografisi= '.$grammografisi."<br/>";

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

// $strValStr=
// $sqlStrcat='INSERT INTO '.$schema.'.'.$categoryTable.$categoryTableColNamesStr.' VALUES (';

$datasetdate=date(DATE_RSS);
$status='test';

$sqlStrContValues=$contributornickname.','.$contributoremail.','.$contributorfirstname.','.$contributorlastname.','.$contributorfacebook.','.$contributortwitter.','.$contributorwebpageurl.','.$datasetdate.','.$status;

$sqlStrCont="INSERT INTO ".$schema.".'.$contributorTable.$contributorTableColNamesStr." VALUES ('".$sqlStrContValues."'");

echo'<p/>';
echo'<hr/>';
echo'<p/>';
echo $sqlStrCont;

echo'<p/>';
echo'<hr/>';
echo'<p/>';

    //////////////////////////////////////
    //-----
    //////////////////////////////////////
    $db = pg_pconnect( "$host $port $dbname $credentials" );
    if(!$db){
        echo "Error : Unable to open database\n";
    } else {
        // echo "Opened database successfully\n";
        // echo '<p>';

//ARXH Block gia contributorTable
        $sqlStrContValues =
        $contributornickname.','.$contributoremail.','.$contributorfirstname.','.$contributorlastn

```

ARXH DB Handling


```

ame.",',"$contributorfacebook.",',"$contributortwitter.",',"$contributorwebpageurl.",',"$datasetdate.",',"$status;
    $sqlStrCont = "INSERT INTO
".$schema.".contributorTable.$contributorTableColNamesStr." VALUES
('".$sqlStrContValues."");
    $sqlMaxCont = 'SELECT max("contributorid") FROM mydb.contributorTable';
    $contributortable_contributorid= InsertTable_ReturnFK($db, $sqlStrCont,
$sqlMaxCont);
//TELOS Block gia contributorTable

//ARXH Block gia categoryTable
$categorytable_categoryid= "";
$sqlStrCat="INSERT INTO ".$schema.".categoryTable.(categorydescription)."
VALUES ('".$newCategory."");
    $sqlMaxCat='SELECT max("categoryid") FROM mydb.categoryTable';

echo 'QUERIES'.'<p>';

echo $sqlStrCat.<p>';
echo $sqlMaxCat.<p>';

        if($selCategory == 2){
            $categorytable_categoryid=InsertTable_ReturnFK($db, $sqlStrCat,
$sqlMaxCat);
        }else{
            $categorytable_categoryid=$selCategory;
        }
//TELOS Block gia categoryTable

//ARXH Block gia datasetTable
    $sqlStrDatasetValues =
$datasetname.",',"$datasetdescr.",',"$datasetdate.",',"$dataseticon.",',"$datasetpoiicon.",
'".$contributortable_contributorid.",',"$categorytable_categoryid;
    $sqlStrDataset = "INSERT INTO
".$schema.".datasetTable.$DatasetTableColNamesStr." VALUES ('".$sqlStrDatasetValues."");

    $sqlMaxDataset='SELECT max("dataset_id") FROM mydb.datasetTable';

echo $sqlStrDatasetValues.<p>';
echo $sqlStrDataset.<p>';
echo $sqlMaxDataset.<p>';

        $poidataset_id= InsertTable_ReturnFK($db, $sqlStrDataset, $sqlMaxDataset);
//TELOS Block gia datasetTable

//ARXH Block gia poisTable
    $sql="";
    $sqlPoisTab1="INSERT INTO ".$schema.".poisTable.$poisTableColNamesStr."
VALUES ('";
        if("$grammografisi"==1){
            foreach($grammesArxeioly as $data){
                $sqlStrPoiValues="";
                $dataArray=explode(",",$data,3);
                $lamda=$dataArray[0];
                $fi=$dataArray[1];
                $desc=rtrim($dataArray[2]);
                $desc = str_replace(","," ", $desc);
                $sqlStrPoiValues =
$desc.",',"$datasetname.",',"$ST_SetSRID(ST_MakePoint($lamda, $fi),
$srld)".",',"$poidataset_id.");";
                $sql .= $sqlPoisTab1.$sqlStrPoiValues;
            }
        }
        unset($value);
    }elseif("$grammografisi"==2){
        foreach($grammesArxeioly as $data){

```

```

        $sqlStrPoiValues="";
        $dataArr=explode(", ", $data,3);
        $lamda=$dataArr[1];
        $fi=$dataArr[0];
        $desc=rtrim($dataArr[2]);
        $desc = str_replace(", ", " ", $desc);
        $sqlStrPoiValues =
        $desc."", ".$datasetname.", ". "ST_SetSRID(ST_MakePoint($lamda, $fi),
        $srid)", ".$poidataset_id."";
        $sql .= $sqlPoisTab1.$sqlStrPoiValues;

        }//end_foreach
        unset($value);
    }else{
        foreach($grammesArxeiroy as $data){
            $sqlStrPoiValues="";
            $dataArr=explode(", ", $data);
            $arrCount = count($dataArr);
            $lamda=$dataArr[$arrCount-2];
            $fi=rtrim($dataArr[$arrCount-1]);
            $desc="";

            for( $i=0; $i<arrCount-3; $i++ ){
                $desc = $dataArr[$i]." ";
            }
            $sqlStrPoiValues =
            $desc."", ".$datasetname.", ". "ST_SetSRID(ST_MakePoint($lamda, $fi),
            $srid)", ".$poidataset_id."";
            $sql .= $sqlPoisTab1.$sqlStrPoiValues.'';

        }//end_foreach
        unset($value);
    }

    echo '<p>';
echo $sql.'<p>';

    $ret = pg_query($db, $sql);
    if(!$ret){
        echo pg_last_error($db);
        echo '<p>';
    } else {
        echo "<h3>". "Η βάση δεδομένων ενημερώθηκε με επιτυχία\n". "</h3>". "<p>";
        echo "<p>";
    }

    }//An ola ok
    //////////////////////////////////////

    pg_close($db);

    //////////////////////////////////////
    ////////////////////////////////////// telos
    //////////////////////////////////////
function checkPOST($str){
    if($str == ''){
        return null;
    }else{
        return $str;
    }
}

//////////////////////////////////// telos checkPOST

function InsertTable_ReturnFK($db, $sql, $sql1){
    $val="";

```

```

$ret = pg_query($db, $sql);
if(!$ret){
    echo pg_last_error($db);
    echo'<p>';
}
$ret = pg_query($db, $sql1);
if(!$ret){
    echo pg_last_error($db);
    echo'<p>';
}else {
    $val = pg_fetch_result($ret, 0, 0);
}
return $val;
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////// telos InsertTable_ReturnFK

////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////// kleisimo php
?>

        <form
action="http://localhost:8088/geoserver/www/OpenData/OpenDataProject.html"method="get">
        <button type="submit" class="my_linkable_button"> Επιστροφή στην Αρχική
Σελίδα </button>
        </form>

        </body>
</html>

```

7.8 Ο κώδικας του αρχείου OpenDataDownloadForm.php

```

<!DOCTYPE HTML>
<head>
    <title>OpenDataDownloadForm</title>
    <link rel="stylesheet" href="OpenDataUploadForm.css">
    <script src="OpenDataDownloadForm.js" defer="defer"
type="text/javascript"></script>
    <meta charset="utf-8">
</head>

<body>

<?php

//////////////////////////////////////////////////////////////////////////////////////////////////////////////// - Notebook -
////////////////////////////////////////////////////////////////////////////////////////////////////////////////
        $host      = "host=localhost";
        $port      = "port=5432";           //NotebookUbuntuPostgresServerPort
        $credentials = "user=postgres password=pavlos7"; //NotebookW7PostgresServerUser
        $dbname    = "dbname=GeoDB"; // DB2NotebookW7
        $schema    = "mydb";

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

        $datasetTable="datasetTable";

```

```

    $poisTable="poisTable";
    //Οι στήλες των πινάκων. *ΕΞΑΙΡΟΥΝΤΑΙ ΤΑ id ΕΠΕΙΔΗ ΕΙΝΑΙ TYPE serial*
    $datasetTableColsArray=array("datasetname","datasetdescr","datasetdate","datasetUploader
");
    $poisTableColsArray=array("poisdescr", "poisdataset", "poisgeom","poidataset_id");

    $db = pg_pconnect( "$host $port $dbname $credentials" );
    if(!$db){
        echo "Error : Unable to open database\n";
    } //else {
//    echo "Opened database successfully\n";
//    }

    $sql =<<<EOF
    SELECT
        dataset_id, datasetname, datasetdescr, datasetdate, contributornickname,
        contributorfirstname, contributorlastname, contributoremail, confributorfacebook,
        contributortwitter, contributorwebpageurl
        FROM mydb."datasetTable", mydb."contributorTable"
        WHERE contributortable_contributorid=contributorid
    EOF;

/*
    $sql =<<<EOF
    SELECT "dataset_ID", "datasetName", "datasetDescr", "datasetUploader", "datasetDate" FROM
    "$schema"."datasetTable";
    EOF;
*/
    $ret = pg_query($db, $sql);

    $arrId = pg_fetch_all_columns($ret, 0);
    $arrName = pg_fetch_all_columns($ret, 1);
    $arrDescr = pg_fetch_all_columns($ret, 2);
    $arrDate = pg_fetch_all_columns($ret, 3);
    $arrUploader = pg_fetch_all_columns($ret, 4);
    $arrContName = pg_fetch_all_columns($ret, 5);
    $arrContLName = pg_fetch_all_columns($ret, 6);
    $arrContEmail = pg_fetch_all_columns($ret, 7);
    $arrContFB = pg_fetch_all_columns($ret, 8);
    $arrContTw = pg_fetch_all_columns($ret, 9);
    $arrContWeb = pg_fetch_all_columns($ret, 10);

    pg_close($db);

    echo '<h2>Διαθέσιμα Datasets για Download</h2>';
    echo '<p>';

    echo '<form name="DownloadForm" method="post" action="FileDownload.php">';

    for($i=0;$i<count($arrId);$i++){

        echo '<fieldset>';
        echo '<legend><b>Όνομα Dataset: </b>'. $arrName[$i]. '</legend>';
        echo '<b>Περιγραφή: </b>'. $arrDescr[$i]. '<br>';
        echo '<b>Ημερομηνία Upload: </b>'. $arrDate[$i]. '<br>';
        echo '<b>Uploader: </b>'. $arrUploader[$i]. '<br>';

        echo '<p>';
        echo '<input type="checkbox" id="'. $arrId[$i]. '"
onclick='. "'handleClick2(this);">'. '<b>Επιλογή</b>';
        echo '</fieldset>';

    } //end_for

```

```

        echo '<input type="hidden" id="selectedDatasets" name="selectedDatasets" value="">';

        echo '<p>';
        echo '<input type="submit" value="Λήψη Αρχείου" />';

echo '</form>';

//      pg_close($db);
?>

        <form
action="http://localhost:8088/geoserver/www/OpenData/OpenDataProject.html"method="get">
        <button type="submit" class="my_linkable_button"> Επιστροφή στην Αρχική
Σελίδα </button>
        </form>

        </body>
</html>

```

7.9 Ο κώδικας του αρχείου OpenDataDownloadForm.js

```

var selectionTab= new Array();
var selection="";

function handleClick2(cb) {

    if(cb.checked){
        selectionTab.splice(selectionTab.length,0,cb.id);
    } else{
        selectionTab.splice(selectionTab.indexOf(cb.id),1);
    }//end else

    tab2text(selectionTab);

} //End_handleClick_function

function tab2text(selectionTab){
    var kryfo = document.getElementById('selectedDatasets');
    var seltext="";
    if(selectionTab.length>0){
        for(var i=0;i<selectionTab.length;i++){
            seltext+=selectionTab[i]+", ";
        }
        seltext=seltext.substring(0,seltext.length-1);
    }

    kryfo.value = seltext;

//      alert(seltext);

} //End_Function_tab2text

```

7.10 Ο κώδικας του αρχείου FileDownload.php

```

<?php

//////////////////////////////////// - Notebook - //////////////////////////////////////

        $host      = "host=localhost";
        $port      = "port=5432";           //NotebookUbuntuPostgresServerPort
        $credentials = "user=postgres password=pavlos7"; //NotebookW7PostgresServerUser
        $dbname    = "dbname=GeoDB"; // DB2NotebookW7
        $schema    = "mydb";
////////////////////////////////////
        $poisTable="poisTable";

        $selectedDatasets=$_POST['selectedDatasets'];

        $selDatasetTab=explode(',',$selectedDatasets);

        $where="";
        $somecontent="";

        for($i=0;$i<count($selDatasetTab);$i++){

                $where .= ' "poidataset_id"='.$selDatasetTab[$i]." OR";

        }

        $where=substr($where, 0, -2);

        $db = pg_pconnect( "$host $port $dbname $credentials" );
        if(!$db){
                echo "Error : Unable to open database\n";
        }

        $sql='SELECT ST_X("poisTable"."poisgeom") AS X1,ST_Y("poisTable"."poisgeom") AS
Y1,poidataset,poidesdescr FROM "'. $schema.'"."poisTable" WHERE'.$where;

        $ret = pg_query($db, $sql);

        if (!$ret) {
                echo "An error occurred.\n";
                //exit;
        }

        $arr = pg_fetch_all($ret);
        for($i=0;$i<count($arr);$i++){
                $r=$arr[$i];

                $somecontent.=$r[x1].",".$r[y1].",".$r[poidataset].",".$r[poidesdescr]."\n";

        }

        pg_close($db);
////////////////////////////////////
        $filename = 'OpenDataProject-download.csv';
        $htmlcode1 = "<HTML> \n <BODY>";
        $htmlcode2 = "</BODY> \n <HTML>";
        $mycontent = $somecontent;
        !$handle = fopen($filename, 'w');
        fwrite($handle, $mycontent);
        fclose($handle);
        //echo $mycontent;

        header("Cache-Control: public");
        header("Content-Description: File Transfer");
        //header("Content-Length: ". filesize("$filename").");");

```

```
header("Content-Disposition: attachment; filename=$filename");
header("Content-Type: application/csv; charset=utf-8");
header("Content-Transfer-Encoding: binary");
echo "\xEF\xBB\xBF"; // UTF-8 BOM
//readfile($filename);

echo $mycontent;

?>
```