

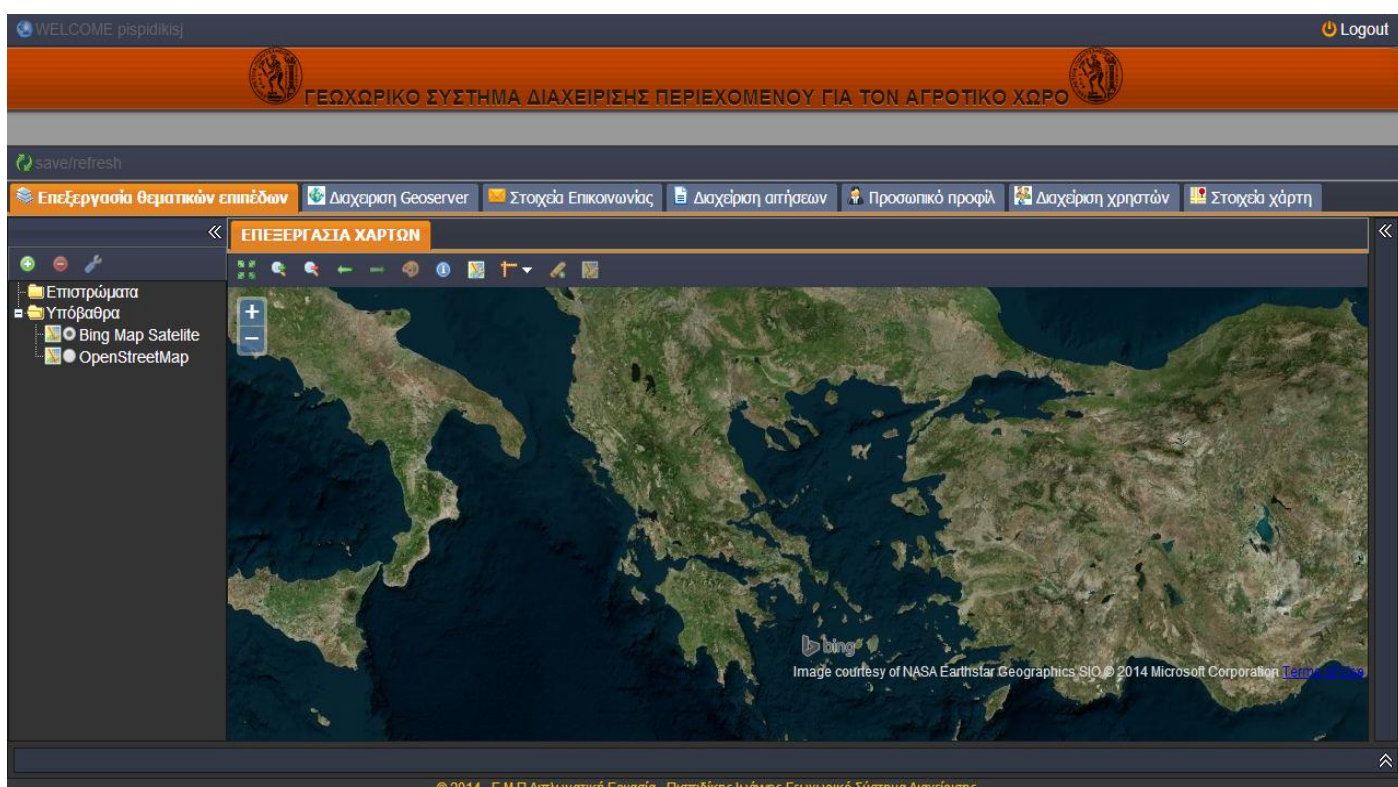


ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ
ΜΗΧΑΝΙΚΩΝ
ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ-ΚΤΗΜΑΤΟΛΟΓΙΟΥ



ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΓΕΩΧΩΡΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ ΜΕ ΑΞΙΟΠΟΙΗΣΗ ΕΛΕΥΘΕΡΩΝ ΤΕΧΝΟΛΟΓΙΩΝ ΑΝΟΙΚΤΟΥ ΚΩΔΙΚΑ

ΠΑΡΑΔΕΙΓΜΑ ΕΦΑΡΜΟΓΗΣ ΣΤΟΝ ΑΓΡΟΤΙΚΟ ΧΩΡΟ



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΠΙΣΠΙΔΙΚΗΣ ΙΩΑΝΝΗΣ

ΕΠΙΒΛΕΠΟΥΣΑ ΚΑΘΗΓΗΤΡΙΑ:
ΕΦΗ ΔΗΜΟΠΟΥΛΟΥ
ΑΝ. ΚΑΘΗΓΗΤΡΙΑ, Ε.Μ.Π



ΑΘΗΝΑ, ΟΚΤΩΒΡΙΟΣ 2014





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ
ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ-ΚΤΗΜΑΤΟΛΟΓΙΟΥ

**ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΓΕΩΧΩΡΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ ΜΕ
ΑΞΙΟΠΟΙΗΣΗ ΕΛΕΥΘΕΡΩΝ ΤΕΧΝΟΛΟΓΙΩΝ ΑΝΟΙΚΤΟΥ ΚΩΔΙΚΑ**

ΠΑΡΑΔΕΙΓΜΑ ΕΦΑΡΜΟΓΗΣ ΣΤΟΝ ΑΓΡΟΤΙΚΟ ΧΩΡΟ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΠΙΣΠΙΔΙΚΗΣ ΙΩΑΝΝΗΣ

ΕΠΙΒΛΕΠΟΥΣΑ ΚΑΘΗΓΗΤΡΙΑ:
ΕΦΗ ΔΗΜΟΠΟΥΛΟΥ
ΑΝ. ΚΑΘΗΓΗΤΡΙΑ, Ε.Μ.Π

ΤΡΙΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Έφη Δημοπούλου
Αν. Καθηγήτρια Ε.Μ.Π

Μαρίνος Κάβουρας
Καθηγητής Ε.Μ.Π

Νικόλαος Δουλάμης
Επ. Καθηγητής Ε.Μ.Π

Αθήνα, Οκτώβριος 2014

Ιωάννης Σ. Πισπιδίκης

Διπλωματούχος Αγρονόμος και Τοπογράφος Μηχανικός Ε.Μ.Π

Copyright © Ιωάννης Σ. Πισπιδίκης, 2014.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία με τίτλο **«Ανάπτυξη διαδικτυακού γεωχωρικού συστήματος διαχείρισης περιεχομένου με αξιοποίηση ελεύθερων τεχνολογιών ανοικτού κώδικα. Παράδειγμα εφαρμογής στον αγροτικό χώρο»** εκπονήθηκε στο πλαίσιο του Προγράμματος Προπτυχιακών Σπουδών του τμήματος Αγρονόμων και Τοπογράφων Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου, βασιζόμενη σε ελληνική και ξένη βιβλιογραφία, καθώς και σε πρωτογενή εμπειρική έρευνα.

Στην προσπάθεια η διπλωματική αυτή να γίνει ουσιαστική και αξιόπιστη, η αρωγή ορισμένων ατόμων φάνηκε πολύτιμη. Αρχικά, θα ήθελα να ευχαριστήσω θερμά την επιβλέπουσα καθηγήτρια κυρία Δημοπούλου Έφη, που με τις πολύτιμες συμβουλές και τις γνώσεις της με βοήθησε να πραγματοποιήσω ένα άρτιο αποτέλεσμα. Επίσης, προσωπικές ευχαριστίες θα ήθελα να αποδώσω στον κύριο Χατζηφιλίππιδη Σταύρο, Προϊστάμενο του τμήματος Τοπογραφίας, Εποικισμού και Αναδασμού της Διεύθυνσης Αγροτικής Οικονομίας και Κτηνιατρικής της Νομαρχίας Αλεξανδρούπολης, για την πολύτιμη συνεργασία του. Τέλος, ιδιαίτερα θα ήθελα να ευχαριστήσω από τα βάθη της καρδιάς μου τη σύζυγό μου για τη συμπαράστασή της καθ' όλη τη διάρκεια των σπουδών μου αλλά και για τη στήριξή της να φέρω αυτό το έργο εις πέρας.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΡΟΛΟΓΟΣ	1
ΠΕΡΙΛΗΨΗ.....	11
ABSTRACT.....	13
1. ΕΙΣΑΓΩΓΗ	15
2. ΔΙΑΔΙΚΤΥΟ ΚΑΙ ΓΕΩΓΡΑΦΙΚΑ ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΩΝ	17
2.1. Ανάπτυξη Δυναμικής Ιστοσελίδας	17
2.1.1. HTTPs request methods	18
2.1.2. PHP (HyperText Preprocessor)	19
2.1.3. JavaScript	22
2.2. Γεωγραφικά Συστήματα Πληροφοριών	26
2.2.1. Πηγές χωρικών δεδομένων	27
2.2.2. Δομές δεδομένων	28
2.2.3. Ελληνικά Γεωδαιτικά Συστήματα Αναφοράς	30
2.2.4. HEllenic POsitioning System (HEPOS)	31
2.2.5. European Petroleum Survey Group (EPSG)	33
3. WEBGIS.....	35
3.1. Αρχιτεκτονική των WebGIS	35
3.1.1. Client	35
3.1.2. Web Server και Application server	36
3.1.3. Map server	38
3.1.4. Data server	38
3.2. Πρότυπα Γεωχωρικών Υπηρεσιών Διαδικτύου	39
3.2.1. Web Map Service (WMS)	39
3.2.2. Web Feature Service (WFS)	41
3.2.3. Web Coverage Service (WCS)	43
3.2.4. Web Processing Service (WPS)	43
3.3. Μορφές Διανυσματικών Δεδομένων Διαδικτύου	44
3.3.1. Geography Markup Language (GML)	44
3.3.2. KML Encoding Standard (KML)	46
3.3.3. GeoJSON.....	51

3.4.	Οδηγία Inspire.....	53
3.4.1.	Ευρωπαϊκός Κανονισμός 1205/2008 για τα Μεταδεδομένα.....	54
3.4.2.	Νόμος 3882/2010	55
3.4.3.	Πρότυπο μεταδεδομένων ISO 19115	56
4.	ΣΥΣΤΗΜΑΤΑ ΑΝΑΠΤΥΞΗΣ WEBGIS	57
4.1.	Γεωχωρικά Εργαλεία και Ελεύθερα Λογισμικά	57
4.1.1.	ΣΔΒΔ και PostgreSQL/PostGIS	57
4.1.2.	OpenLayers	60
4.1.3.	ExtJS/GeoExt/GXP	62
4.1.4.	QuantumGIS.....	64
4.2.	Εξυπηρετητές Χαρτών Ανοικτού Κώδικα	64
4.2.1.	GeoServer	65
4.2.2.	MapServer.....	68
4.3.	Συστήματα Διαχείρισης Περιεχομένου	73
4.3.1.	OpenGeo Suite	73
4.3.2.	MapGuide OpenSource.....	77
4.3.3.	Cartaro	84
4.3.4.	GeoNode	87
5.	ΑΓΡΟΤΙΚΟΣ ΧΩΡΟΣ	91
5.1.	Ιστορική αναδρομή	91
5.1.1.	Ο αγροτικός χώρος κατά τον 19 ^ο αιώνα.....	91
5.1.2.	Ο αγροτικός χώρος κατά τον 20 ^ο αιώνα.....	92
5.2.	Μέτρα πολιτικής γης στον αγροτικό χώρο	93
5.2.1.	Διανομή.....	93
5.2.2.	Αναδασμός.....	95
5.3.	Εθνικό Κτηματολόγιο και Χαρτογράφηση (ΕΚΧΑ)	98
5.3.1.	Ένταξη Διανομών και Αναδασμών στο ΕΚΧΑ	99
5.3.2.	Περιοχή με Αναδασμό ή Διανομή μη ενταγμένης στο ΕΚΧΑ	99
6.	ΣΧΕΔΙΑΣΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ	101
6.1.	Αρχιτεκτονική Συστήματος	101
6.2.	Εγκατάσταση Συστήματος	101
6.3.	Σύστημα Διαχείρισης Περιεχομένου	104

6.3.1.	Στοιχεία χάρτη	105
6.3.2.	Διαχείριση χρηστών	105
6.3.3.	Προσωπικό προφίλ	106
6.3.4.	Διαχείριση αιτήσεων	106
6.3.5.	Στοιχεία επικοινωνίας.....	107
6.3.6.	Διαχείριση του GeoServer	107
6.3.7.	Επεξεργασία θεματικών επιπέδων.....	110
6.3.8.	Αξιοποίηση του QuantumGIS	113
6.4.	Σύστημα Αίτησης Χαρτών	115
6.4.1.	Δυνατότητα επικοινωνίας	116
6.4.2.	Διαδικασία αίτησης	117
6.5.	Αξιολόγηση Συστήματος	118
7.	ΠΑΡΑΔΕΙΓΜΑ ΣΤΗΝ ΠΕΡΙΟΧΗ ΤΟΥ ΕΒΡΟΥ	121
7.1.	Περιοχή Μελέτης και Διαθέσιμα Δεδομένα.....	121
7.2.	Παραδείγματα Λειτουργίας.....	122
7.2.1.	Σύστημα Διαχείρισης για προετοιμασία δεδομένων.....	122
7.2.2.	Εφαρμογή αίτησης χαρτών	125
7.2.3.	Σύστημα Διαχείρισης για επεξεργασία αιτήσεων	130
8.	ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΡΟΤΑΣΕΙΣ	133
9.	ΒΙΒΛΙΟΓΡΑΦΙΑ	135
	ΠΑΡΑΡΤΗΜΑ (ΣΚΕΛΕΤΟΣ ΣΥΣΤΗΜΑΤΟΣ).....	139

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 2-1: Διαδικασία PHP.	19
Εικόνα 2-2: Ροή παραδοσιακού και Ajax web application.	24
Εικόνα 2-3: Παράδειγμα χρήσης τεχνολογίας Ajax.	25
Εικόνα 2-4: Βασικές λειτουργίες ενός ΓΣΠ.	26
Εικόνα 2-5: Μέθοδοι ραδιομετρικής παρεμβολής εικόνας.	29
Εικόνα 2-6: Μορφές διανυσματικών δεδομένων.	30
Εικόνα 2-7: Οι 98 Σταθμοί Αναφοράς του HEPOS.	32
Εικόνα 3-1: Λειτουργία client.	36
Εικόνα 3-2: Λειτουργία Web server.	37
Εικόνα 3-3: Λειτουργία Web server με application server.	37
Εικόνα 3-4: Web-mapping server.	38
Εικόνα 3-5: Παράμετροι αιτήματος GetCapabilities.	40
Εικόνα 3-7: Χρήση αιτήματος GetFeatureInfo.	41
Εικόνα 3-8: Παράδειγμα δημιουργίας στοιχείου μέσω αιτήματος Transaction.	42
Εικόνα 3-9: Παράδειγμα υπηρεσίας WPS.	44
Εικόνα 3-10: Ιστότοπος spatialreference.org για το σύστημα αναφοράς WGS 84. ...	46
Εικόνα 3-11: Σχέση γεωμετρικού και ορθομετρικού υψομέτρου.	48
Εικόνα 3-12: Οπτικοποίηση γραμμής στο χώρο (1).	49
Εικόνα 3-13: Οπτικοποίηση γραμμής στο χώρο (2).	49
Εικόνα 3-14: Οπτικοποίηση γραμμής στο χώρο(3).	50
Εικόνα 3-15: Παράδειγμα χρήσης της ετικέτας region.	50
Εικόνα 4-1: OpenLayers.	61
Εικόνα 4-2: Διαθέσιμες πηγές πληροφοριών της Openlayers.	62
Εικόνα 4-3: WebGIS εφαρμογή με χρήση βιβλιοθήκης GeoExt.	63
Εικόνα 4-4: Οργάνωση θεματικών επιπέδων στον GeoServer.	67
Εικόνα 4-5: Ενσωμάτωση του GeoWebCache στην αρχιτεκτονική ενός WebGIS.	68
Εικόνα 4-6: Δομή λειτουργίας του MapServer.	69
Εικόνα 4-7: Σύνθεση χάρτη σε QGIS.	72
Εικόνα 4-8: Πακέτο λογισμικών OpenGeo Suite.	74
Εικόνα 4-9: GeoExplorer.	74
Εικόνα 4-10: Μεταφόρτωση διανυσματικών δεδομένων.	75
Εικόνα 4-11: Μορφοποίηση διανυσματικών δεδομένων.	76
Εικόνα 4-12: Διαδικασία εξαγωγής του χάρτη.	77
Εικόνα 4-13: Παράδειγμα AJAX viewer.	78
Εικόνα 4-14: Παράδειγμα fusion viewer.	78
Εικόνα 4-15: Αρχιτεκτονική MapGuide Open Source.	79
Εικόνα 4-16: Σύνδεση Maestro με MapGuide.	81
Εικόνα 4-17: Δημιουργία Βασικών φακέλων.	81
Εικόνα 4-18: Προσδιορισμός πηγής δεδομένων.	82

Εικόνα 4-19: Μορφοποίηση του στυλ των θεματικών επιπέδων.....	82
Εικόνα 4-20: Προεπισκόπηση του χάρτη.	83
Εικόνα 4-21: Τελική εφαρμογή.....	83
Εικόνα 4-22: Cartaro.....	84
Εικόνα 4-23: Παράδειγμα ιστοσελίδα του ΕΜΠ.	85
Εικόνα 4-24: Νέο content type.	85
Εικόνα 4-25: Δημιουργία θεματικού επιπέδου στον GeoServer.	86
Εικόνα 4-26: Τελική ιστοσελίδα με τον επιθυμητό χάρτη.	86
Εικόνα 4-27: Μεταφόρτωση δεδομένων και καθορισμός προσβασιμότητας.	88
Εικόνα 4-28: Δημιουργία μεταδεδομένων για τα δεδομένα.	88
Εικόνα 4-29: Μορφοποίηση του στυλ των δεδομένων.	89
Εικόνα 5-1: Απόσπασμα οριστικής διανομής έτους 1932 στον Πόρο.	95
Εικόνα 5-2: Περαιτωθέντες αναδασμοί ανά έτος 1953-2005.	97
Εικόνα 6-1: Αρχιτεκτονική συστήματος.....	101
Εικόνα 6-2: Καταχώρηση στοιχείων βάσης δεδομένων PostGIS.	102
Εικόνα 6-3: Καταχώρηση στοιχείων κύριου διαχειριστή.	103
Εικόνα 6-4: Εισαγωγή στο σύστημα διαχείρισης περιεχομένου.	104
Εικόνα 6-5: Καθορισμός ζώνης ορατότητας των δεδομένων.	105
Εικόνα 6-6: Διαχείριση χρηστών.....	106
Εικόνα 6-7: Διαχείριση αιτήσεων.	106
Εικόνα 6-8: Επεξεργασία αίτησης.....	107
Εικόνα 6-9: Workspace.	108
Εικόνα 6-10: Store.....	108
Εικόνα 6-11: Διαγραφή θεματικού επιπέδου.	109
Εικόνα 6-12: Δημιουργία PostGIS store.	109
Εικόνα 6-13: Μεταφόρτωση χωρικών δεδομένων.....	110
Εικόνα 6-14: Οπτικοποίηση θεματικού επιπέδου.	111
Εικόνα 6-15: Μορφοποίηση(1).....	111
Εικόνα 6-16: Μορφοποίηση(2).....	112
Εικόνα 6-17: Δημιουργία νέου χαρακτηριστικού.....	113
Εικόνα 6-18: Επεξεργασία από λογισμικό QuantumGIS(1 ^{ος} τρόπος).....	114
Εικόνα 6-19: Επεξεργασία από λογισμικό QuantumGIS (2 ^{ος} τρόπος).....	115
Εικόνα 6-20: Εφαρμογή αίτησης χαρτών.	116
Εικόνα 6-21: Παράθυρο επικοινωνίας.	116
Εικόνα 6-22: Αίτηση.....	117
Εικόνα 6-23: Ελλιπή και πλήρη στοιχεία αίτησης.	118
Εικόνα 7-1: Περιοχή μελέτης.	121
Εικόνα 7-2: Καθορισμός περιοχής ευθύνης.	123
Εικόνα 7-3: Δημιουργία Workspace.	123
Εικόνα 7-4: Μεταφόρτωση διαθέσιμων χαρτών.....	124
Εικόνα 7-5: Δημιουργία νέου χρήστη.....	125

Εικόνα 7-6: Ζητούμενα αγροτεμάχια.	125
Εικόνα 7-7: Εύρεση αγροτεμαχίου στη διανομή.....	126
Εικόνα 7-8: Συμπλήρωση γεωχωρικών στοιχείων (1).	127
Εικόνα 7-9: Συμπλήρωση γεωχωρικών στοιχείων (2).	127
Εικόνα 7-10: Συμπλήρωση γεωχωρικών στοιχείων (3).	128
Εικόνα 7-11: Συμπλήρωση γεωχωρικών στοιχείων (4).	128
Εικόνα 7-12: Συμπλήρωση γεωχωρικών στοιχείων (5).	129
Εικόνα 7-13: Εύρεση αγροτεμαχίου στον αναδασμό.	129
Εικόνα 7-14: Αίτηση 2 ^{ου} παραδείγματος.....	129
Εικόνα 7-15: Διαχείριση αιτήσεων.	130
Εικόνα 7-16: Επεξεργασία αίτησης.....	130
Εικόνα 7-17: Απόκομμα χάρτη συμπληρωματικής διανομής.	131
Εικόνα 7-18: Ολοκλήρωση επεξεργασίας αίτησης.....	131

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 2-1: Διαφορές μεθόδων GET και POST.	18
Πίνακας 2-2: Μεταφορά μεταβλητών σε PHP σελίδες.	20
Πίνακας 2-3: Extensions της php.	20
Πίνακας 2-4: Μέθοδοι επιλογής των στοιχείων.	24
Πίνακας 2-5: Κωδικοί EPSG.....	33
Πίνακας 3-1: Γεωμετρικά αντικείμενα GeoJSON.....	51
Πίνακας 3-2: Θεματικά επίπεδα χωρικών δεδομένων οδηγίας INSPIRE.	53
Πίνακας 3-3: Κατηγορίες Κανονισμών Εφαρμογής.	54
Πίνακας 4-1: SRID για διάφορα συστήματα συντεταγμένων.	59
Πίνακας 4-2: Πηγές δεδομένων.....	65
Πίνακας 4-3: Διαθέσιμες υπηρεσίες του MapGuide server.....	80
Πίνακας 5-1: Χαρακτηριστικά αγροτικού κλήρου πριν και μετά τον αναδασμό.....	96
Πίνακας 7-1: Διαθέσιμη χάρτες στην περιοχή μελέτης.	122

ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ

Διάγραμμα 2-1: Παράδειγμα DOM tree.....	23
---	----

ΠΕΡΙΛΗΨΗ

Απώτερος στόχος της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη ενός ολοκληρωμένου διαδικτυακού γεωγραφικού συστήματος πληροφοριών, που επιτυγχάνεται μέσω της χρήσης εργαλείων και λογισμικών ανοικτού κώδικα. Ακόμη, σκοπός της εργασίας είναι η υποκίνηση περαιτέρω συζήτησης και εμπειρικής έρευνας, με κύριο γνώμονα τη στροφή προς την αξιοποίηση ελεύθερων συστημάτων για την επίλυση διαφόρων θεμάτων. Το συγκεκριμένο σύστημα καλύπτει τις ανάγκες τόσο του διαχειριστή όσο και του χρήστη, ενώ η εγκατάστασή του είναι άμεση και εύχρηστη.

Στο πλαίσιο της διπλωματικής εργασίας επιχειρείται η ανάλυση των βασικών συστατικών που χρησιμοποιήθηκαν για τη δημιουργία του συστήματος, τα οποία αποτελούνται από τη βάση δεδομένων PostgreSQL/PostGIS, τον εξυπηρετητή χαρτών GeoServer και τον εξυπηρετητή Διαδικτύου Apache. Για την ανάπτυξη και τη διασύνδεση των ανωτέρω είναι αναγκαία η χρήση διαφόρων γλωσσών προγραμματισμού, όπως είναι η PHP και η JavaScript. Παράλληλα, θα γίνει μνεία των διαφόρων διαθέσιμων λύσεων για την ανάπτυξη WebGIS εφαρμογών.

Τέλος, το συγκεκριμένο διαδικτυακό γεωγραφικό σύστημα πληροφοριών εφαρμόστηκε με τέτοια μεθοδολογία, ώστε να ανταποκριθεί με τον καλύτερο δυνατό τρόπο στις ανάγκες οπτικοποίησης και αίτησης χαρτών του αγροτικού χώρου στην περιοχή του Έβρου. Οι τελευταίοι, απεικονίζουν στοιχεία ιδιοκτησίας και τηρούνται από το τμήμα Τοπογραφίας, Εποικισμού και Αναδασμού του συγκεκριμένου Νομού.

Λέξεις Κλειδιά: PostgreSQL/PostGIS, GeoServer, Apache, PHP, JavaScript, WebGIS ανοικτού κώδικα, ελεύθερων συστημάτων.

Web development of geospatial content management system through the use of free and open source technologies

Example in rural areas

ABSTRACT

The ultimate aim of this thesis is to develop an integrated web-GIS, through the use of tools and open source software. Moreover, the aim of this work is to stimulate further debate and empirical research, primarily driven shift towards employing free systems to solve various issues. This system covers needs both for the administrator and the users, while the installation is swift and easy.

In the context of this thesis an analysis of the main components used to create the system is attempted, which consist of the database PostgreSQL / PostGIS, the map server GeoServer and web server Apache. For the development and interconnection of the above, the use of various programming languages such as PHP and JavaScript is required. Furthermore, several solutions available for developing WebGIS applications are also presented.

Finally, this web geographical information system's methodology was developed so as to optimally respond to visualization needs and support application requirements concerning maps of rural areas of the region of Evros. The latter, depict land ownership data and are maintained by the Department of Surveying, Settlements and Agrarian Reform of the county.

Keywords: PostgreSQL/PostGIS, GeoServer, Apache, PHP, JavaScript, WebGIS, open source, free systems.

1. ΕΙΣΑΓΩΓΗ

Η ανάπτυξη των γεωγραφικών συστημάτων πληροφοριών προέκυψε από την ανάγκη της χωρικής απεικόνισης συνόλου προβλημάτων, αποτελεσμάτων και πληροφοριών. Η ραγδαία εξέλιξη της τεχνολογίας και του διαδικτύου οδήγησαν στη δημιουργία διαδικτυακών πληροφοριακών συστημάτων, με τη βοήθεια των οποίων η επικοινωνία και η διάθεση χωρικών δεδομένων γίνεται πιο εύκολη, άμεση και αποτελεσματική. Επιπροσθέτως, πρέπει να λάβουμε υπόψη τα σύγχρονα οικονομικά δεδομένα που απαιτούν τη χρεία εύρεσης οικονομικότερων λύσεων. Έτσι, στην παρούσα διπλωματική εργασία επιδιώκεται να δοθεί λύση σε ένα πρόβλημα που αφορά στον αγροτικό χώρο. Ειδικότερα, αυτή θα επιτευχθεί μέσω της υλοποίησης διαδικτυακής εφαρμογής, στηριζόμενης στη χρήση ελεύθερων λογισμικών και τεχνολογιών ανοικτού κώδικα, τα οποία δεν προϋποθέτουν κάποιο κόστος αγοράς. Η εφαρμογή αυτή αξιοποιεί τη βάση δεδομένων PostgreSQL/PostGIS, τον εξυπηρετητή χαρτών GeoServer και τον εξυπηρετητή διαδικτύου Apache.

Αναλυτικότερα, στο δεύτερο κεφάλαιο εξετάζεται το Διαδίκτυο και τα Γεωγραφικά Συστήματα Πληροφοριών. Συγκεκριμένα, αναλύεται ο τρόπος ανάπτυξης δυναμικών ιστοσελίδων με τη χρήση κατάλληλων γλωσσών προγραμματισμού. Επιπλέον, παρουσιάζεται η τεχνολογία Ajax, με την οποία καθίσταται δυνατή η αποστολή δεδομένων στον server, χωρίς να υφίσταται ανάγκη ανανέωσης της ιστοσελίδας. Στη συνέχεια, αναφέρονται οι πηγές χωρικών δεδομένων, οι δομές δεδομένων και τα γεωδαιτικά συστήματα αναφοράς, τα οποία αποτελούν το σκελετό των γεωγραφικών συστημάτων πληροφοριών.

Στο τρίτο κεφάλαιο περιγράφεται η αρχιτεκτονική των WebGIS εφαρμογών, τα πρότυπα γεωχωρικών υπηρεσιών διαδικτύου, οι μορφές διανυσματικών δεδομένων διαδικτύου και η οδηγία Inspire για το κομμάτι των μεταδεδομένων.

Στο τέταρτο κεφάλαιο εξετάζονται τα διάφορα γεωχωρικά εργαλεία, όπως και τα ελεύθερα λογισμικά που χρησιμοποιούνται για την ανάπτυξη των WebGIS εφαρμογών. Επίσης, αναλύονται διεξοδικά οι εξυπηρετητές χαρτών ανοικτού κώδικα και αναφέρονται μερικά από τα υπάρχοντα συστήματα διαχείρισης περιεχομένου.

Στο πέμπτο κεφάλαιο αναπτύσσεται η ιστορική αναδρομή των μεταβολών που υπέστη ο αγροτικός χώρος κατά τη διάρκεια του 19^{ου} και του 20^{ου} αιώνα. Συγχρόνως, παρουσιάζονται τα μέτρα που έλαβε κατά καιρούς η πολιτεία όσον αφορά στον αγροτικό χώρο. Μερικά από αυτά είναι η διανομή της γης και ο αναδασμός, όπου και αναλύονται τα κύρια χαρακτηριστικά τους. Τέλος, γίνεται μνεία στο πρόγραμμα του Εθνικού Κτηματολογίου και την Εταιρεία Κτηματολογίου και Χαρτογράφησης (ΕΚΧΑ).

Το έκτο κεφάλαιο περιλαμβάνει τη σχεδίαση της διαδικτυακής εφαρμογής, η οποία υλοποιήθηκε στο πλαίσιο της παρούσας διπλωματικής εργασίας. Πιο αναλυτικά, παρουσιάζεται η αρχιτεκτονική της εφαρμογής, ο τρόπος εγκατάστασής της, η δυνατότητά της όσον αφορά τη διαχείριση περιεχομένου, ενώ δεν παραλείπονται και οι αιτήσεις των χαρτών.

Στο έβδομο κεφάλαιο παρατίθεται το παράδειγμα της εφαρμογής στην περιοχή του Έβρου. Το συγκεκριμένο θέμα διαρθρώνεται σε δύο ενότητες. Η πρώτη περιλαμβάνει την περιοχή μελέτης και τα διαθέσιμα δεδομένα ενώ στην τρίτη παρουσιάζονται τα παραδείγματα της εφαρμογής.

Στο όγδοο κεφάλαιο παρουσιάζονται τα συμπεράσματα που προκύπτουν από τη μελέτη της διπλωματικής εργασίας. Υπάρχουν αξιολογικοί χαρακτηρισμοί και συγκεκριμένες προτάσεις για τους προβληματισμούς που ανακύπτουν.

Τέλος, στο παράρτημα (“Σκελετός του Συστήματος”) παρουσιάζεται η βασική δομή της εφαρμογής που αναπτύχθηκε. Παράλληλα, στο παράρτημα (“Κώδικας”) της παρούσας εργασίας αναγράφεται ο κώδικας που αναπτύχθηκε κατά τη διάρκεια της εκπόνησής της.

2. ΔΙΑΔΙΚΤΥΟ ΚΑΙ ΓΕΩΓΡΑΦΙΚΑ ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΩΝ

Το Διαδίκτυο δημιουργήθηκε στα τέλη της δεκαετίας του '60 με στόχο την εξυπηρέτηση των στρατιωτικών αναγκών των Ηνωμένων Πολιτειών Αμερικής. Μεγάλη ώθηση στην εξέλιξή του έδωσε η σχεδίαση του παγκόσμιου ιστού (World Wide Web) και η ανάπτυξη της γλώσσας HTML από τον Tim Berners Lee το 1990 στο ερευνητικό κέντρο φυσικής CERN. Αφενός η ανάγκη της δυναμικής απεικόνισης πληροφοριών στο Διαδίκτυο και αφετέρου η δημιουργία περισσότερο διαδραστικών ιστοσελίδων, οδήγησε στην ανάπτυξη διαφόρων διαδικτυακών γλωσσών προγραμματισμού, οι οποίες επιλέγονται βάσει του σκοπού και των αναγκών της εφαρμογής.

Η χωρική πληροφορία αποτέλεσε τον κορμό για την ανάπτυξη των γεωγραφικών συστημάτων πληροφοριών (GIS). Τα συστήματα αυτά απεικονίζουν και δημιουργούν δεδομένα σε καθορισμένο σύστημα αναφοράς, αντλούν στοιχεία βάσει διαφόρων χωρικών εξαρτήσεων των δεδομένων και, τέλος, εξάγουν τα αποτελέσματα σε καθορισμένη μορφή.

Ο συνδυασμός των παραπάνω τεχνολογιών οδήγησε στη δημιουργία των διαδικτυακών πληροφοριακών συστημάτων (WebGIS). Αυτά, καθιστούν δυνατή την απεικόνιση και την αναζήτηση δεδομένων, αναλόγως της χωρικής τους πληροφορίας στο διαδίκτυο.

2.1. Ανάπτυξη Δυναμικής Ιστοσελίδας

Αρχικά, η δημιουργία των ιστοσελίδων στο Διαδίκτυο γινόταν μόνο με τη χρήση της γλώσσας απεικόνισης HTML (HyperText Markup Language). Η τελευταία, δεν αποτελεί γλώσσα προγραμματισμού αλλά μία γλώσσα περιγραφική. Ο browser «αντιλαμβάνεται» και απεικονίζει την ιστοσελίδα μέσω βασικών κανόνων (tags) που περιέχει αυτή η γλώσσα. Οι κανόνες καθορίζουν την αρχή και το τέλος μιας εντολής. Κάτωθι παρατίθεται παράδειγμα της χρήσης των αντίστοιχων κανόνων.

```
<html><body><font color=red> hello</font></body></html>
```

Οι συγκεκριμένοι κανόνες δημιουργούν μια ιστοσελίδα, η οποία περιέχει το κείμενο «hello» σε κόκκινο χρώμα.

Με τη μεμονωμένη χρήση της HTML δημιουργούνται στατικές ιστοσελίδες χωρίς δυνατότητα άντλησης και απεικόνισης πληροφορίας σε πραγματικό χρόνο. Η ανάπτυξη διαφόρων πληροφοριακών συστημάτων, όπως τα GIS, οδήγησε στην ανάγκη ενσωμάτωσης γλωσσών προγραμματισμού, καθιστώντας έτσι τις ιστοσελίδες εξ ολοκλήρου δυναμικές. Επιπροσθέτως, δημιουργήθηκε και η γλώσσα

απεικόνισης CSS προκειμένου να είναι πιο εύκολη και αποτελεσματική η μορφοποίηση της ιστοσελίδας.

2.1.1. HTTPs request methods

Το πρωτόκολλο HTTP (Hypertext Transfer Protocol) σχεδιάστηκε με τέτοιο τρόπο, ώστε να γίνει εφικτή η επικοινωνία μεταξύ του πελάτη και του server. Οι μέθοδοι που εφαρμόζονται για να αποσταλεί ένα αίτημα HTTP με συγκεκριμένα δεδομένα στον server, είναι η GET και η POST. Σημειώνεται ότι η υποβολή και ο καθορισμός της μεθόδου υποβολής των δεδομένων γίνεται μέσω της φόρμας της HTML.

➤ Μέθοδος GET

Με τη μέθοδο GET οι πληροφορίες που υποβάλλονται από τον χρήστη μέσω της φόρμας, επισυνάπτονται με την προς αίτηση σελίδα. Η τελευταία, χωρίζεται από τις πληροφορίες με τον χαρακτήρα «?».

`http://www.test.com/index.htm?name1=value1&name2=value2`

Η συγκεκριμένη μέθοδος είναι περιορισμένη και στέλνει μέχρι 2028 χαρακτήρες. Επιπλέον, δεν χρησιμοποιείται για τη μεταφορά ευαίσθητων δεδομένων (password, username κ.α.) στο server ούτε μπορεί να στείλει δυαδικά δεδομένα, όπως είναι μια εικόνα ή ένα έγγραφο Word.

➤ Μέθοδος POST

Με τη μέθοδο POST τα στοιχεία προς επεξεργασία εισάγονται μόνο στην ιστοσελίδα που έχει καθοριστεί στη φόρμα υποβολής. Επιπροσθέτως, δεν υπάρχει κάποιος περιορισμός στο μέγεθος και τον τύπο των δεδομένων που υποβάλλονται. Τέλος, υπάρχει μεγαλύτερη ασφάλεια στη μεταφορά των δεδομένων απ' ό,τι στην προηγούμενη μέθοδο.

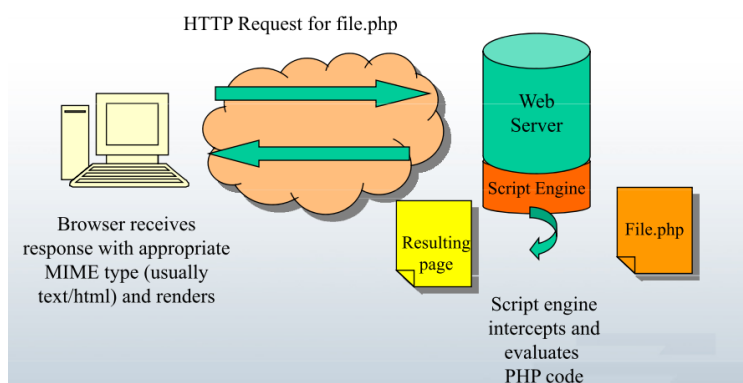
Πίνακας 2-1: Διαφορές μεθόδων GET και POST.

	Get	Post
Ανανέωση/Πίσω	Αβλαβής ενέργεια.	Τα δεδομένα θα υποβληθούν ξανά με μήνυμα ειδοποίησης στον χρήστη.
Ασφάλεια	Δεν υπάρχει ασφάλεια. Τα δεδομένα αποστέλλονται ως μέρος του URL.	Εξαρτάται από το πρωτόκολλο HTTP.
Ορατότητα	Τα δεδομένα φαίνονται ως μέρος του URL.	Δεν εμφανίζονται τα δεδομένα.
Μέγεθος	Περιορίζεται από το μήκος	Δεν υπάρχει περιορισμός.

	περιορισμού του URL (2046 χαρακτήρες).	
Τύπος Δεδομένων	Δεν αποστέλλονται δυαδικά δεδομένα.	Δεν υπάρχει περιορισμός.

2.1.2. PHP (HyperText Preprocessor)

Η PHP είναι μια δημοφιλής, ανοικτού κώδικα, γλώσσα scripting, η οποία εκτελείται στην πλευρά του εξυπηρετητή (server side). Είναι συμβατή με ένα ευρύ φάσμα λειτουργικών συστημάτων και εξυπηρετητών, ενώ η εκμάθησή της είναι αρκετά εύκολη, καθώς περιλαμβάνει πολυάριθμες μεθόδους και πακέτα. Τα αρχεία PHP συλλέγονται και αξιολογούνται από μια scripting engine, η οποία παράγει τη ζητούμενη σελίδα.



Εικόνα 2-1: Διαδικασία PHP.

Οι ιστοσελίδες που δημιουργούνται με ενσωματωμένες δυνατότητες PHP προγραμματισμού, έχουν επέκταση «.php», ενώ ο κώδικας γράφεται εντός συγκεκριμένων κανόνων:

<code><?php</code>	ΚΩΔΙΚΑΣ	<code>?></code>
-----------------------	----------------	--------------------

Μια χρήσιμη δυνατότητα της PHP είναι η ένταξη κοινά χρησιμοποιούμενων scripts σε ένα αρχείο, όπως header, footer κλπ. Οι τρόποι για να εισαχθούν τα αρχεία στην PHP είναι κυρίως η **include** και η **require**.

<pre>require 'connect.php'; include 'connect.php';</pre>
--

Με τη χρήση της οδηγίας `include`, ακόμα και αν δεν βρεθεί το αρχείο, η επεξεργασία της σελίδας θα συνεχιστεί. Αντιθέτως, με την οδηγία `require` θα σταματήσει η εκτέλεση λόγω σφάλματος.

Για να είναι δυνατή η απεικόνιση, ο έλεγχος και η περαιτέρω επεξεργασία των δεδομένων που λαμβάνονται από τον server, η PHP διαθέτει τους «superglobal» πίνακες. Οι τελευταίοι, είναι διαθέσιμοι σε κάθε script.

Πίνακας 2-2:Μεταφορά μεταβλητών σε PHP σελίδες.

Superglobal πίνακες	Περιγραφή
<code>\$_GET</code>	Δεδομένα που περάστηκαν με τη μέθοδο GET.
<code>\$_POST</code>	Δεδομένα που περάστηκαν με τη μέθοδο POST.
<code>\$_COOKIE</code>	Δεδομένα που έχουν αποθηκευτεί σε Cookies.
<code>\$_REQUEST</code>	Δεδομένα που περάστηκαν με μια από τις παραπάνω μεθόδους.
<code>\$_SERVER</code>	Στοιχεία για το τρέχον request και το περιβάλλον του εξυπηρετητή.
<code>\$_FILE</code>	Αρχεία που ανέβασε ο χρήστης.

Η χρήση των extensions ενδυναμώνει τη συγκεκριμένη γλώσσα ακόμα περισσότερο και την κάνει υψηλότερου επιπέδου. Μερικά από τα extensions περιγράφονται στον πίνακα 2.3

Πίνακας 2-3:Extensions της php.

Extension	Περιγραφή
Php_mysql,php_pdo_mysql	Δυνατότητα σύνδεσης με τη βάση δεδομένων MySQL.
Php_pgsql,php_pdo_pgsql	Δυνατότητα σύνδεσης με τη βάση δεδομένων PostgreSQL.
Php_curl	Δυνατότητα σύνδεσης με τον GeoServer.

Βασικό μειονέκτημα της συγκεκριμένης γλώσσας είναι ότι εκτελείται μόνο με την ανανέωση της ιστοσελίδας.

(Πηγή: <https://php.net/manual/en/index.php>)

➤ Άδεια χρήσης της PHP

Η άδεια χρήσης της γλώσσας προγραμματισμού PHP για τη δημιουργία δυναμικών ιστοσελίδων είναι η «Άδεια PHP (version 3.01)». Η τελευταία, αποτελεί άδεια χρήσης ελεύθερου λογισμικού που μεγιστοποιεί την ελευθερία του αρχικού δικαιούχου, σύμφωνα με το Ίδρυμα Ελεύθερου Λογισμικού (Free Software Foundation). Όπως αναφέρει το OSI¹, είναι άδεια ανοικτού κώδικα που δεν είναι όμως συμβατή με την GPL, λόγω των περιορισμών που έχουν τεθεί για τη χρήση του όρου PHP. Η αναδιανομή και χρήση της PHP σε δυαδική μορφή ή σε μορφή πηγαίου κώδικα, με ή χωρίς τροποποιήσεις, επιτρέπεται μόνο υπό την τήρηση των παρακάτω όρων:

- Οι αναδιανομές του πηγαίου κώδικα θα πρέπει να διατηρούν την ειδοποίηση για τα πνευματικά δικαιώματα, την παρούσα λίστα προϋποθέσεων και τη δήλωση αποποίησης των ευθυνών.
- Οι αναδιανομές σε δυαδική μορφή θα πρέπει να διατηρούν την ειδοποίηση για τα πνευματικά δικαιώματα, την παρούσα λίστα προϋποθέσεων, τη δήλωση αποποίησης των ευθυνών και άλλα υλικά που παρέχονται με τη διανομή.
- Το όνομα «PHP» δεν πρέπει να χρησιμοποιηθεί για την υπογραφή ή την προώθηση των προϊόντων που προέρχονται από αυτό το λογισμικό, χωρίς προηγούμενη γραπτή άδεια. Για την έγγραφη άδεια, οι ενδιαφερόμενοι μπορούν να επικοινωνήσουν με e-mail με το group@php.net.
- Τα προϊόντα που προέρχονται από αυτό το λογισμικό δεν μπορούν να ονομάζονται «PHP», ούτε μπορούν να εμφανίζουν τη λέξη «PHP» στο όνομά τους, χωρίς προηγούμενη γραπτή άδεια από το group@php.net. Μπορεί όμως ο δικαιούχος να υποδείξει πως το λογισμικό του λειτουργεί σε συνδυασμό με την PHP, λέγοντας για παράδειγμα «Εφαρμογή για την PHP» αντί να ονομαστεί «PHP Εφαρμογή» ή «ρηρεφαρμογή».
- Η ομάδα της PHP μπορεί να δημοσιεύει αναθεωρημένες και νέες εκδόσεις της άδειας. Σε κάθε έκδοση θα δοθεί ένας διακριτικός αριθμός έκδοσης. Μόλις δημοσιευτεί ο κώδικας για κάποια συγκεκριμένη έκδοση της άδειας, ο δικαιούχος μπορεί να συνεχίσει να τη χρησιμοποιεί υπό τους όρους της έκδοσης αυτής. Κανείς εκτός από την ομάδα της PHP δεν έχει το δικαίωμα να τροποποιήσει τους όρους που ισχύουν για τον κώδικα που δημιουργήθηκε στο πλαίσιο της παρούσας άδειας.

1 OSI: η OSI (Open Source Initiative- Η Πρωτοβουλία Ανοικτού Κώδικα) είναι μια κοινωφελής εταιρία της Καλιφόρνιας, μη κερδοσκοπικού χαρακτήρα, που ιδρύθηκε το 1998. Επιπλέον, υπάρχουν διαχειριστές, οι οποίοι διαχειρίζονται το Open Source Definition (OSD) και το αναγνωρισμένο εκ της κοινότητας όργανο, για την εξέταση και την έγκριση των αδειών βάσει συγκεκριμένων κριτηρίων.

- Οι αναδιανομές οποιασδήποτε μορφής πρέπει να διατηρούν το ακόλουθο αναγνωριστικό: «Αυτό το προϊόν περιλαμβάνει PHP, ελεύθερα διαθέσιμο από το <http://www.php.net/>.

(Πηγή: http://php.net/license/3_01.txt)

2.1.3. JavaScript

Η JavaScript δημιουργήθηκε από τον Brendan Eich που εργαζόταν στην Netscape. Αυτή η γλώσσα προγραμματισμού υποστηρίχτηκε για πρώτη φορά το Δεκέμβριο του 1995 από το πρόγραμμα περιήγησης Netscape 2.0. Η εταιρεία Microsoft, αντιγράφοντας την Netscape, ονόμασε την εφαρμογή της JScript για να αποφύγει θέματα εμπορικών σημάτων. Η τελευταία, περιλήφθηκε στο πρόγραμμα Internet Explorer 3.0, το οποίο κυκλοφόρησε τον Αύγουστο του 1996. Η εποχή 1996-1999 αναφέρεται ως «browser wars», γιατί κάθε πλοηγτής άρχισε να διαφοροποιείται, προσθέτοντας νέες, ασύμβατες μεταξύ τους, δυνατότητες. Η κατάσταση βελτιώθηκε με τη δημιουργία κινήματος υπέρ των web standards και, τελικώς, την καθιέρωση της JavaScript ως πρότυπο από την ECMA (πρότυπο ECMAScript).

Η JavaScript δεν πρέπει να συγχέεται με την Java παρά τις επιφανειακές ομοιότητές τους. Αρχικά, ονομαζόταν Mocha και έπειτα LiveScript, ενώ απέκτησε το τελικό όνομά της για λόγους marketing. Η Java την περίοδο εκείνη ήταν ιδιαίτερα δημοφιλής και η Netscape θεώρησε πως, αν η νέα γλώσσα είχε παρόμοιο όνομα και συντακτικό, θα εξασφάλιζε μεγαλύτερη επιτυχία.

Η JavaScript είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού, που επιτρέπει την προγραμματιστική πρόσβαση σε διάφορα αντικείμενα του φιλοξενούντος περιβάλλοντος (host environment). Σημειώνεται, ότι η JavaScript δεν διαθέτει classes αλλά μόνο αντικείμενα. Συνήθως, υλοποιείται από προγράμματα περιήγησης, έτσι ώστε τα διάφορα σενάρια (scripts) του πελάτη (client) να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να μεταβάλουν δυναμικά το περιεχόμενο τις ιστοσελίδας. Η επεξεργασία του κώδικα JavaScript και η τελική παραγωγή της ιστοσελίδας δεν πραγματοποιείται στον εξυπηρετητή Διαδικτύου, αλλά στο πρόγραμμα περιήγησης. Απόρροια του προηγούμενου είναι η μη ύπαρξη δυνατότητας στη JavaScript να επικοινωνήσει με τον αντίστοιχο εξυπηρετητή Διαδικτύου και κατ' επέκταση με τη βάση δεδομένων. Ο κώδικας της JavaScript περικλείεται από ετικέτες της HTML και μπορεί να μπει σε οποιοδήποτε σημείο της ιστοσελίδας και όχι απαραίτητα στο <head>.

Κώδικας ενσωματωμένος στην HTML

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

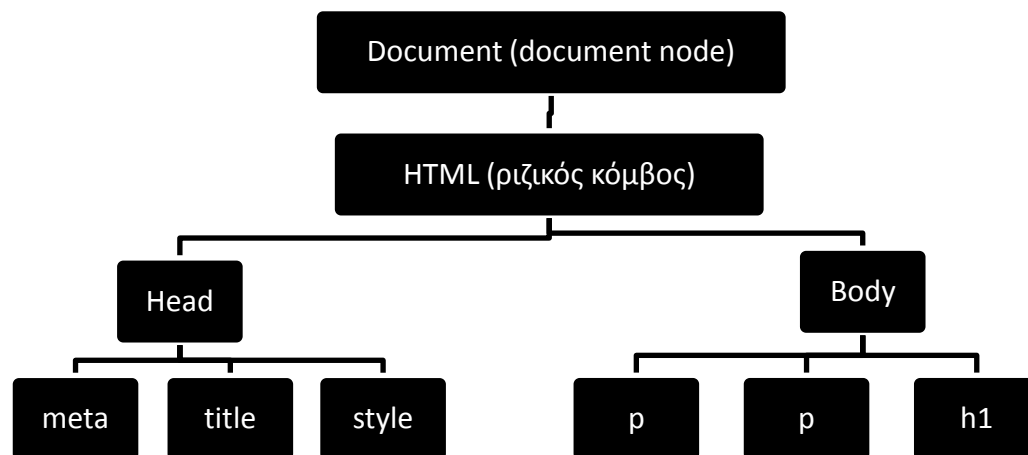
Κώδικας σε ξεχωριστό αρχείο (έστω tutorial.js)

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

(Πηγή: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>)

➤ DOM (Document Object Model)

Το DOM είναι μια ανεξάρτητη πλατφόρμα και γλώσσα σύμβασης, όπου παρέχει μια δομημένη απεικόνιση του XML και (X)HTML εγγράφου και ορίζει τον τρόπο που μπορεί αυτό να προσπελαστεί από κάποια γλώσσα προγραμματισμού, έτσι ώστε να είναι δυνατή η αλλαγή της δομής, του περιεχομένου και του στυλ του εγγράφου. Η δομή του εγγράφου αναπαρίσταται ως ένα δέντρο αποτελούμενο από στοιχεία (element nodes), κείμενο (Text node), ιδιότητες (Attribute node) και σχόλια (comment nodes). Η ρίζα του δέντρου είναι ένα document node, ενώ οι κόμβοι διέπονται από σχέσεις ιεραρχίας. Το πρώτο element node του δέντρου καλείται ριζικός κόμβος (root node) και κάθε στοιχείο εκτός από το ριζικό έχει ακριβώς έναν γονέα (parents). Τα στοιχεία που έχουν τον ίδιο γονέα είναι αδέρφια και απόγονοι του στοιχείου που αποτελεί τον γονέα τους. Στο παρακάτω παράδειγμα ιεραρχίας τα στοιχεία Head και Body είναι αδέρφια μεταξύ τους, απόγονοι της HTML, ενώ τα παιδιά τους (child) είναι τα στοιχεία με τα οποία είναι ενωμένα.



Διάγραμμα 2-1: Παράδειγμα DOM tree.

Με βάση την ιεραρχία του DOM tree, τα διάφορα script επιλέγουν τα στοιχεία πάνω στα οποία θέλουν να ενεργήσουν. Οι τέσσερις βασικοί τρόποι που γίνεται η επιλογή των στοιχείων είναι οι παρακάτω:

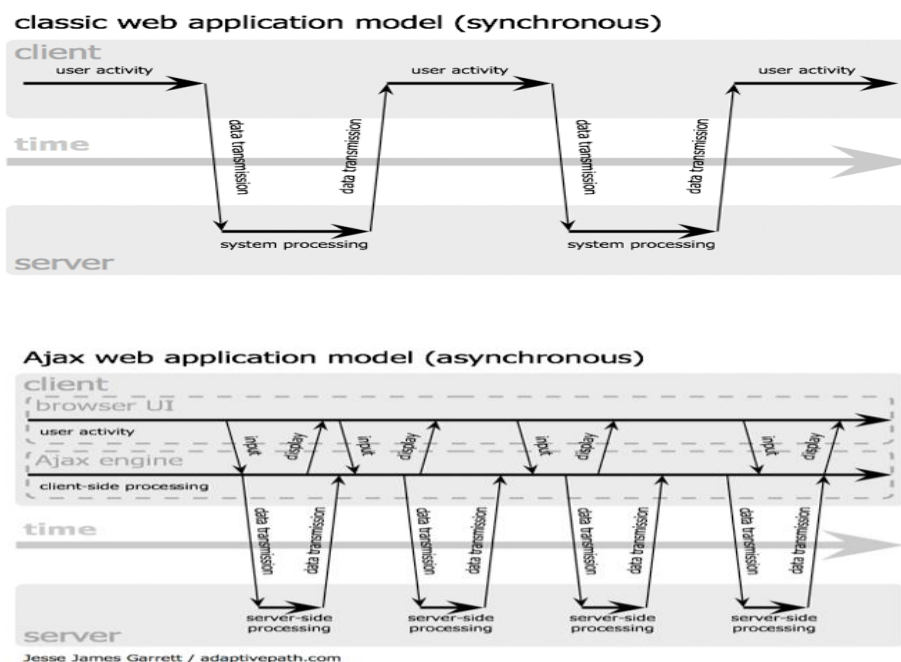
Πίνακας 2-4: Μέθοδοι επιλογής των στοιχείων.

Τρόποι επιλογής	Μέθοδος επιλογής
Id	document.getElementById('contents')
Tag	document.getElementsByTagName('p')
Name	document.getElementsByName(Name)
Class	document.getElementsByClassName(Classname)

Τέλος, μια βασική δυνατότητα που επιτυγχάνεται μέσω του DOM και συγκεκριμένα του document node είναι η δημιουργία και διαγραφή στοιχείων από την ιστοσελίδα, καθώς και η τροποποίηση των χαρακτηριστικών των υπαρχόντων στοιχείων με τη χρήση της γλώσσας JavaScript.

➤ AJAX (Asynchronous JavaScript And XML)

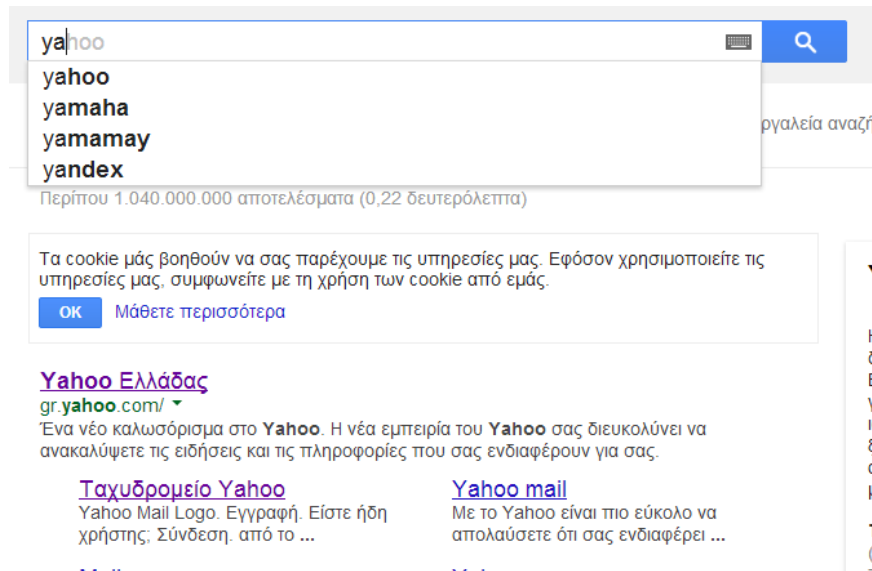
Ο όρος Ajax επινοήθηκε από τον Jesse James Garret και χρησιμοποιείται για την περιγραφή μοντέρνων τεχνολογιών, τεχνικών και μεθόδων, που δεν σχετίζονται απαραίτητα με τα συστατικά μέρη του Ajax (Asynchronous JavaScript And XML). Η βασική ιδέα του είναι να μην χρειάζεται πλέον να ανανεωθεί ολόκληρη η σελίδα προκειμένου να σταλούν δεδομένα στο server. Συγκεκριμένα, θα μπορεί να γίνει όλη η διαδικασία ασύγχρονα στο παρασκήνιο και να ανανεώνεται κάθε φορά μόνο το τμήμα της σελίδας που χρειάζεται ανανέωση. Η λειτουργία του Ajax βασίζεται στο αντικείμενο της JavaScript **XMLHttpRequest**.



Εικόνα 2-2: Ροή παραδοσιακού και Ajax web application.

Πηγή: <http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications/>

Οι τεχνικές του Ajax έγιναν πολύ δημοφιλείς, όταν χρησιμοποιήθηκαν από την Google στο Gmail (2004) και στο Google maps (2005). Το 2006 το W3C δημοσίευσε το πρώτο πρότυπο για το XMLHttpRequest object.



Εικόνα 2-3: Παράδειγμα χρήσης τεχνολογίας Ajax.

Χαρακτηριστικό παράδειγμα χρήσης της συγκεκριμένης τεχνολογίας φαίνεται στην εικόνα 2-3. Παρατηρούμε πως πραγματοποιήθηκε επικοινωνία με τον εξυπηρετητή Διαδικτύου και κατ' επέκταση με τη βάση δεδομένων χωρίς την ανανέωση της ιστοσελίδας. Έτσι, τροποποιήθηκε η τελική ιστοσελίδα με τα αποτελέσματα της αναζήτησης.

Βασικό πλεονέκτημα της χρήσης του Ajax είναι η εξοικονόμηση του εύρους, καθώς αποστέλλονται και λαμβάνονται μόνο τα απαραίτητα. Ακόμη, ελαττώνεται ο φόρτος του server, μιας και δεν ανανεώνεται ολόκληρη η σελίδα με κάθε αίτημα αλλά μόνο ένα κομμάτι της. Τέλος, ένα επιπλέον προτέρημα της χρήσης του αποτελεί η γρηγορότερη απόκριση της διεπαφής, αλλά και η παροχή δυνατότητας για εφαρμογές σε πραγματικό χρόνο.

Η ανταλλαγή δεδομένων μέσω Ajax πραγματοποιείται με τη χρήση μορφοτύπων. Τα πιο συνηθισμένα αναφέρονται συνοπτικά παρακάτω:

✓ HTML

Αποτελεί την πιο κοινή μορφή ανταλλαγής δεδομένων μέσω Ajax. Επιπροσθέτως, καθιστά εύκολη και απλή την ενημέρωση τμήματος της σελίδας, απλώς αναθέτοντας τα ληφθέντα δεδομένα στην ιδιότητα innerHTML κάποιου στοιχείου της ιστοσελίδας.

✓ XML

Συνιστά την κύρια μορφή ανταλλαγής δεδομένων για την οποία επινοήθηκαν οι τεχνικές της Ajax.

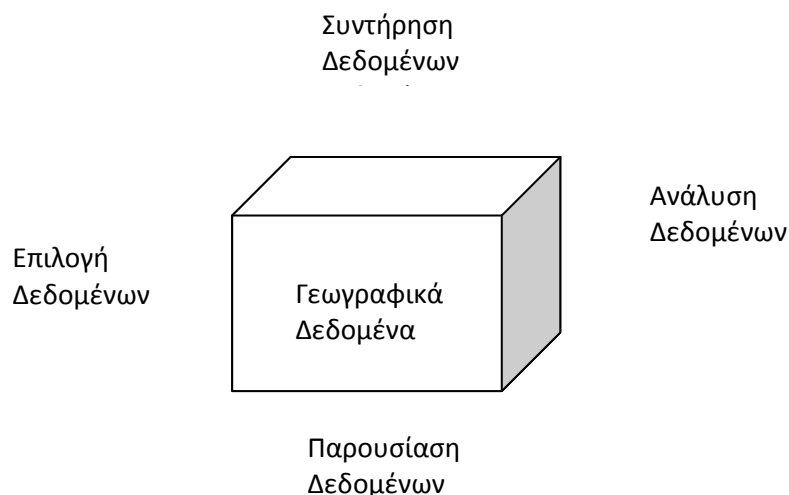
✓ JSON (JavaScript Object Notation)

Η συγκεκριμένη μορφή άρχισε να κερδίζει έδαφος τα τελευταία χρόνια. Συγκριτικά με την XML έχει μικρότερο μέγεθος για τα ίδια δεδομένα. Επιπλέον, είναι πιο εύκολη στην ανάγνωση και στην προσπέλαση των δεδομένων.

Τέλος, χρησιμοποιούνται και άλλες μορφές μεταφοράς δεδομένων, όπως είναι οι JavaScript και CSV (comma separated value), όπου όμως δεν ενδείκνυνται για χρήση.

2.2. Γεωγραφικά Συστήματα Πληροφοριών

Τα γεωγραφικά συστήματα πληροφοριών (GIS) αποτελούν συστήματα πληροφοριών που βασίζονται στη διαχείριση δεδομένων με χωρικές και περιγραφικές πληροφορίες. Είναι σχεδιασμένα για να υποστηρίζουν τη συλλογή, διαχείριση, επεξεργασία, ανάλυση, μοντελοποίηση και απεικόνιση δεδομένων που αναφέρονται στο χώρο και μεταβάλλονται με το χρόνο. Ο βασικός ρόλος των συστημάτων είναι να προσφέρουν στους χρήστες ισχυρά εργαλεία για την επίλυση σύνθετων χωρικών προβλημάτων (Στεφανάκης, 2003). Στην εικόνα 2-3 φαίνονται οι βασικές λειτουργίες που πρέπει να υποστηρίζει ένα γεωγραφικό σύστημα πληροφοριών.



Εικόνα 2-4: Βασικές λειτουργίες ενός ΓΣΠ.

Πηγή: Στεφανάκης, 2003

➤ Συντήρηση Δεδομένων

Με την συντήρηση δεδομένων εννοούμε τις δυνατότητες εισαγωγής, διαγραφής και ενημέρωσης των δεδομένων του συστήματος.

➤ Επιλογή Δεδομένων

Επιλέγονται γεωγραφικές οντότητες αναλόγως με τις τιμές των γνωρισμάτων τους. Οι λειτουργίες επιλογής δεδομένων, ανάλογα με τις διαστάσεις που εμπλέκονται, υποδιαιρούνται στις κάτωθι κατηγορίες: χωρικές επιλογές, θεματικές επιλογές, χρονικές επιλογές και μικτές επιλογές.

➤ Ανάλυση Δεδομένων

Οι λειτουργίες ανάλυσης είναι πολυάριθμες και εξαρτώνται από το πεδίο εφαρμογών. Επεξεργάζονται τα γνωρίσματα των γεωγραφικών οντοτήτων και οδηγούν, είτε στη δημιουργία νέων οντοτήτων, στην τροποποίηση των τιμών των γνωρισμάτων αυτών ή στην εξαγωγή γενικότερων συμπερασμάτων, μέτρων και πληροφοριών για τα γεωγραφικά δεδομένα.

➤ Παρουσίαση Δεδομένων

Η παρουσίαση των αποτελεσμάτων των λειτουργιών επιλογής ή ανάλυσης των δεδομένων μπορεί να είναι σε μορφή χάρτη, γραφήματος, κειμένου, πίνακα, αντικειμένων πολυμέσων, και δίνεται στο χρήστη μέσα από κατάλληλα σχεδιασμένες διεπαφές.

(Στεφανάκης, 2003)

2.2.1. Πηγές χωρικών δεδομένων

Ένα ΓΣΠ ενσωματώνει δεδομένα που προέρχονται από πολλές και διαφορετικές πηγές. Η ικανότητα ενσωμάτωσης δεδομένων αποτελεί ένα από τα σημαντικότερα πλεονεκτήματα της τεχνολογίας των ΓΣΠ. Παρακάτω αναφέρονται οι διάφορες πρωτογενείς και δευτερογενείς πηγές απ' όπου προέρχονται τα δεδομένα:

➤ Τοπογραφικές μετρήσεις

Τα δεδομένα που προκύπτουν με τοπογραφικές μετρήσεις περιγράφουν άμεσα ή έμμεσα τη θέση, είτε με συντεταγμένες είτε με ενδιάμεσες μετρήσεις, όπως κατακόρυφη γωνία, οριζόντια γωνία και οριζόντια απόσταση. Οι τοπογραφικές μετρήσεις προκύπτουν είτε με τη χρήση γεωδαιτικών σταθμών (total station), είτε με τη χρήση GPS.

➤ Υπάρχοντες χάρτες

Οι μετρητικές πληροφορίες από τους υπάρχοντες χάρτες πραγματοποιείται με τη γεωαναφορά του χάρτη και την ψηφιοποίηση των διαφόρων χωρικών πληροφοριών που περιέχει. Σε περίπτωση που οι χάρτες είναι σε αναλογική μορφή, χρησιμοποιούνται κατάλληλοι σαρωτές για τη μετατροπή τους σε ψηφιακή κανονικοποιημένη μορφή.

➤ Φωτογραμμετρία

Η φωτογραμμετρία χρησιμοποιείται κυρίως για την αποτύπωση μεγάλων εκτάσεων και βασίζεται στη λήψη ζευγών αεροφωτογραφιών με σχετική επικάλυψη. Η επικάλυψη αυτή επιτρέπει την αντίληψη του τρισδιάστατου χώρου και τη συλλογή υψομετρικών και οριζοντιογραφικών δεδομένων.

➤ Φωτοερμηνεία-Τηλεπισκόπηση

Με την τηλεπισκόπηση επιδιώκεται η συλλογή και ανάλυση πληροφοριών σχετικών με τον τύπο και τις ιδιότητες των αντικειμένων. Η λήψη των αεροφωτογραφιών και δορυφορικών εικόνων γίνεται με τη χρήση κατάλληλων δεκτών που καταγράφουν των ανακλώμενη από τη φυσική γήινη επιφάνεια ακτινοβολία σε ένα ή περισσότερα κανάλια του ηλεκτρομαγνητικού φάσματος. Ο κατάλληλος συνδυασμός των φασματικών καναλιών δίνουν και την ανάλογη πληροφορία.

2.2.2. Δομές δεδομένων

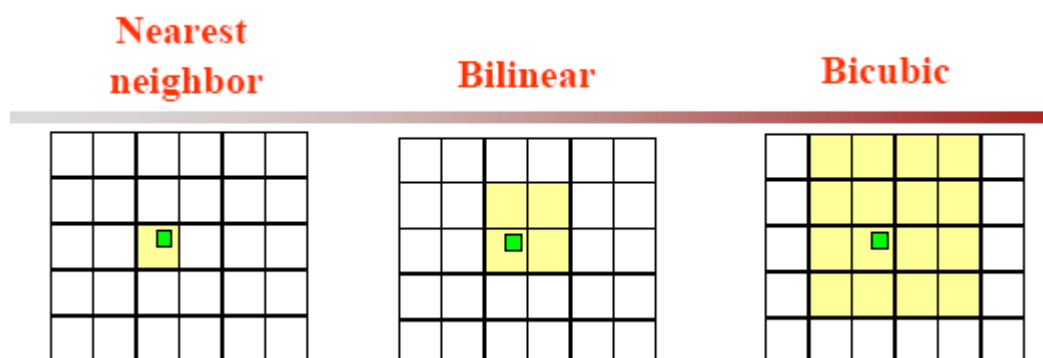
Τα ΓΣΠ αναπτύχθηκαν για την αποτελεσματική αποθήκευση, ανάλυση και επεξεργασία δεδομένων που αφορούν τη γήινη επιφάνεια. Τα δεδομένα αυτά περιλαμβάνουν θεματικές και χωρικές πληροφορίες. Τα μοντέλα που χρησιμοποιούνται για την περιγραφή των γεωγραφικών δεδομένων και τη σύνδεση των θεματικών και γεωμετρικών πληροφοριών είναι το μοντέλο πεδίων και το μοντέλο αντικειμένων. Στο μοντέλο πεδίων η γήινη επιφάνεια θεωρείται ως ένα χωρικό συνεχές και ομογενές μέσο, ενώ στο μοντέλο αντικειμένων ο γεωγραφικός χώρος αποτελείται από στοιχεία (features) ή αντικείμενα (objects) που έχουν γεωμετρική θέση, σχήμα και διάφορα θεματικά χαρακτηριστικά. Για την υλοποίηση αυτών των μοντέλων δημιουργήθηκαν δύο γεωμετρικές δομές, η διανυσματική (vector) και η κανονικοποιημένη (raster) (Κάβουρας, 2004).

➤ Κανονικοποιημένη δομή δεδομένων (raster)

Η κανονικοποιημένη δομή δεδομένων αναπαριστά τον πραγματικό κόσμο ως κάρναβο, ο οποίος αποτελείται από ένα σύνολο κελιών-φατνιών (pixels) με συγκεκριμένο μέγεθος. Το φατνίο αποτελεί τη μικρότερη γεωγραφική μονάδα, που μπορεί να αντιπροσωπεύεται από μια μορφή raster και το μέγεθός του καθορίζει και την τελική ανάλυση της εικόνας. Συνεπώς, όσο περισσότερο πυκνός είναι ο

κάναβος (υψηλή ανάλυση), τόσο πιο μικρό το φατνίο και, έτσι, τόσο πιο λεπτομερής είναι η απεικόνιση της πληροφορίας. Σημειώνεται ότι η κατηγορία των κανονικοποιημένων χωρικών δομών περιέχει όλες τις δομές που περιλαμβάνουν επαναλαμβανόμενο πρότυπο ενός κανονικού πολυγώνου ή πολυέδρου. Επομένως, οι δομές δεδομένων μπορεί να βασίζονται σε τετραγωνικό, τριγωνικό ή εξαγωνικό πλέγμα. Από τα τρία είδη κανάβων ευρέως χρησιμοποιούμενος είναι ο κανονικός τετραγωνικός κάναβος, διότι είναι συμβατός με τις περισσότερες συσκευές που χρησιμοποιούνται για τη συλλογή και την απόδοση δεδομένων.

Μια σημαντική διαδικασία που ακολουθείται, προκειμένου μια δομή raster να περιέχει μετρητικές πληροφορίες στο σύστημα αναφοράς που μας ενδιαφέρει, είναι ο γεωμετρικός μετασχηματισμός και η επανασύστασή της. Ο γεωμετρικός μετασχηματισμός αφορά την μετατροπή των συντεταγμένων των pixel της δομής raster από το ένα σύστημα στο άλλο, μέσω κατάλληλων συναρτήσεων. Η επιλογή των τελευταίων, εξαρτάται από το είδος του μετασχηματισμού που θα χρησιμοποιήσουμε και κατ' επέκταση του συνόλου των ελάχιστων κοινών σημείων που πρέπει να έχουμε στα δύο συστήματα. Οι πιο συνήθεις γεωμετρικοί μετασχηματισμοί είναι ο ομοιότητας, ο αφινικός, ο προβολικός και ο πολυωνιμικός. Η επανασύσταση της εικόνας αναφέρεται στη διαδικασία με την οποία κάθε pixel της τελικής εικόνας παίρνει την τιμή του χρώματος από το αντίστοιχο της αρχικής. Η διαδικασία αυτή περιλαμβάνει τον αντίστροφο μετασχηματισμό από την τελική εικόνα στην αρχική και τη διενέργεια ραδιομετρικής παρεμβολής στα pixel της αρχικής. Οι βασικές τεχνικές παρεμβολής είναι του πλησιέστερου γείτονα, η διγραμμική και η δικυβική.

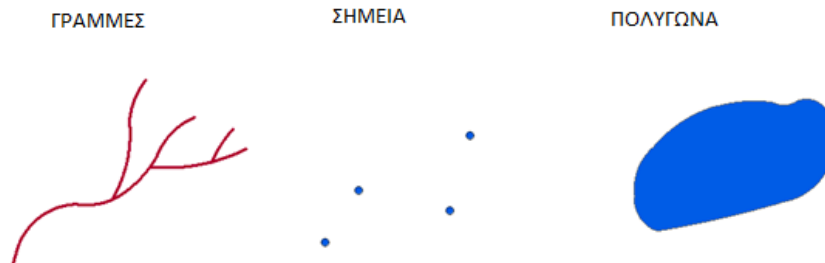


Εικόνα 2-5: Μέθοδοι ραδιομετρικής παρεμβολής εικόνας.

Οι τρεις μέθοδοι που φαίνονται στην εικόνα 2-5 διαφέρουν ως προς τον αριθμό των pixels της αρχικής εικόνας, βάσει των οποίων καθορίζεται η τιμή του χρώματος της τελικής εικόνας.

➤ Διανυσματικές δομές δεδομένων (vector)

Τα γεωμετρικά στοιχεία που χρησιμοποιούνται για την περιγραφή των διανυσματικών δεδομένων είναι τα σημεία, οι γραμμές και τα πολύγωνα.



Εικόνα 2-6: Μορφές διανυσματικών δεδομένων.

Η επιλογή των στοιχείων αυτών εξαρτάται από την κλίμακα παρατήρησης ή το επίπεδο της γενίκευσης. Σε μικρή κλίμακα οι πόλεις για παράδειγμα αναπαρίστανται από σημεία, ενώ οι ποταμοί και οι δρόμοι από γραμμές. Καθώς αυξάνεται η κλίμακα παρατήρησης, πρέπει να λαμβάνεται υπόψη και η επιφανειακή διάσταση των φαινομένων. Συνεπώς, με την αύξηση της κλίμακας οι δρόμοι, οι ποταμοί και οι πόλεις αναπαρίστανται με πολύγωνα. Τα σημεία αποτελούν στοιχεία μηδενικής διάστασης και αναπαριστούν γεωγραφικές οντότητες με πολύ μικρό μέγεθος σε σχέση με την κλίμακα αναπαράστασης. Στον δισδιάστατο χώρο προσδιορίζονται μέσω ενός ζεύγους συντεταγμένων. Οι γραμμές είναι στοιχεία μιας διάστασης και αποτελούνται από ένα σύνολο διαδοχικών τόξων, όπου κάθε ένα από αυτά προσδιορίζεται από τις συντεταγμένες του σημείου αρχής και του σημείου τέλους. Στην περίπτωση που το σημείο αρχής με το σημείο τέλος ταυτίζονται, τότε δημιουργούνται τα πολύγωνα, τα οποία είναι στοιχεία δύο διαστάσεων (Κάβουρας, 2004).

2.2.3. Ελληνικά Γεωδαιτικά Συστήματα Αναφοράς

Το γεωδαιτικό σύστημα αναφοράς που χρησιμοποιείται σήμερα στην Ελλάδα επίσημα είναι το ΕΓΣΑ 87 (Ελληνικό Γεωδαιτικό Σύστημα Αναφοράς). Επιπλέον, λόγω ύπαρξης παλαιότερων δεδομένων, όπως διάφοροι χάρτες διανομών και αναδασμών του Υπουργείου Γεωργίας, χρησιμοποιείται και το Παλιό Ελληνικό Σύστημα Αναφοράς (ΠΕΣΑ). Επιπροσθέτως, το ED50 (Ευρωπαϊκό Datum του 1950) χρησιμοποιείται στους ναυτικούς χάρτες και για στρατιωτικούς σκοπούς.

➤ ΕΓΣΑ 87

Το ΕΓΣΑ 87 χρησιμοποιεί το ελλειψοειδές αναφοράς GRS80 (global reference system), το οποίο είναι μετατοπισμένο, ώστε να προσαρμόζεται καλύτερα στο γεωειδές της περιοχής της Ελλάδας. Συγκεκριμένα, σε σχέση με το WGS84 υπάρχει μια παράλληλη μετατόπιση κατά $\Delta X=+199,73\text{m}$, $\Delta Y=-74,030\text{m}$ και $\Delta Z=-246,018\text{m}$. Στο ΕΓΣΑ 87 έχει επιλεγεί για προβολικό σύστημα η Εγκάρσια Μερκατορική προβολή με μία ζώνη εφαρμογής για όλο τον ελλαδικό χώρο. Η προβολή είναι σύμμορφη και οι παραμορφώσεις αυξάνονται με την απόσταση από τον κεντρικό μεσημβρινό ($\lambda_0=24^\circ$). Για να μην υπάρχουν αρνητικές συντεταγμένες στα x , δόθηκε στον κεντρικό μεσημβρινό η τιμή 500.000m , ενώ για τα y η αφετηρία είναι η τομή του κεντρικού μεσημβρινού με τον ισημερινό ($\phi=0^\circ$). Οι παραμορφώσεις είναι σχετικά μικρές στην ηπειρωτική Ελλάδα αλλά σημαντικές στα άκρα της ζώνης (Κέρκυρα, Ρόδος). Αυτό σημαίνει ότι είναι αναγκαίο να γίνονται οι κατάλληλες αναγωγές για σωστούς υπολογισμούς στο προβολικό επίπεδο.

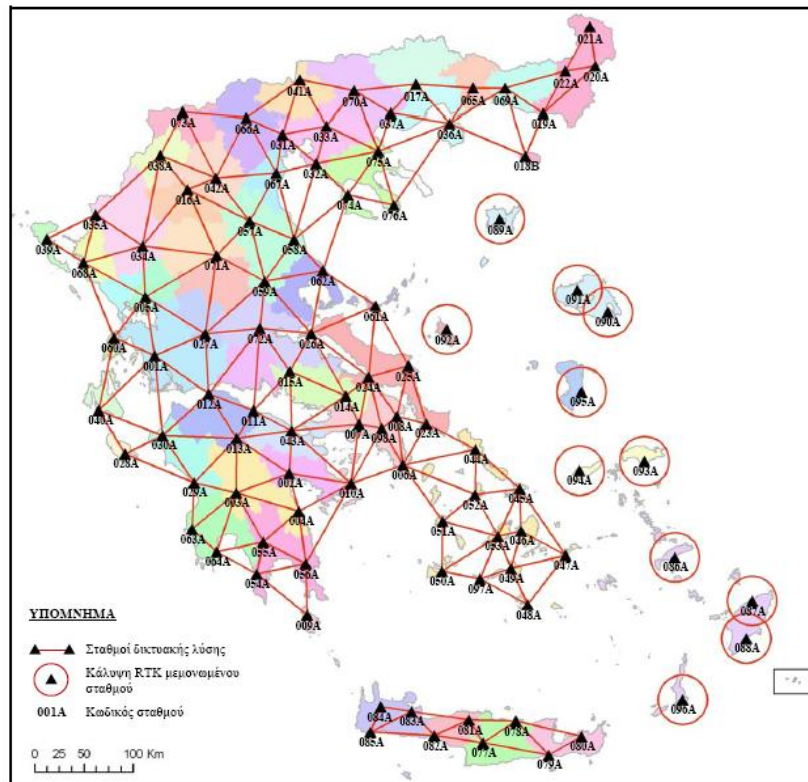
➤ ΠΕΣΑ

Το ΠΕΣΑ χρησιμοποιεί το ελλειψοειδές αναφοράς Bessel και το προβολικό σύστημα Hatt. Το τελευταίο, αποτελεί μια πλάγια αζιμουθιακή ισαπέχουσα απεικόνιση που διατηρεί αναλλοίωτα τα αζιμούθια και τις αποστάσεις από το κέντρο προβολής, ενώ οι παραμορφώσεις αυξάνονται με την απόσταση από το κέντρο της προβολής. Προκειμένου να είναι μικρές οι παραμορφώσεις και να μην χρειάζεται να γίνονται αναγωγές των μετρημένων μεγεθών, η Ελλάδα χωρίστηκε σε τραπεζοειδή τραπέζια διαστάσεων $30'\times 30'$, τα οποία αντιστοιχούν σε χάρτες κλίμακας 1:100.000 της Γ.Υ.Σ (Γεωγραφικής Υπηρεσίας Στρατού). Κάθε χάρτης 1:100.000 υποδιαιρείται σε τέσσερις χάρτες 1:50.000. Η μετατροπή των συντεταγμένων από Hatt σε ΕΓΣΑ 87 γίνεται με βάση τους πολυωνυμικούς συντελεστές της Γ.Υ.Σ που υπάρχουν για κάθε φύλλο χάρτη Hatt 1:50.000.

(Μήλας, Μπιλλήρης, 2006)

2.2.4. HEllenic POsitioning System (HEPOS)

Το Ελληνικό Σύστημα Εντοπισμού HEPOS είναι ένα σύγχρονο σύστημα εντοπισμού, το οποίο επιτρέπει τον προσδιορισμό θέσης με υψηλή ακρίβεια, αξιοποιώντας το υφιστάμενο Παγκόσμιο Δορυφορικό Σύστημα Εντοπισμού GPS. Αποτελείται από ένα σύνολο 98 μόνιμων Σταθμών Αναφοράς GPS, κατανεμημένων σε ολόκληρη τη χώρα, ένα Κέντρο Ελέγχου και ένα τηλεπικοινωνιακό δίκτυο, το οποίο αναλαμβάνει την αμφίδρομη διασύνδεση των Σταθμών Αναφοράς με το Κέντρο Ελέγχου.



Εικόνα 2-7: Οι 98 Σταθμοί Αναφοράς του HEPOS.

Πηγή: <http://www.hepos.gr/>

Οι δέκτες GPS βρίσκονται σε συνεχή λειτουργία και αποστέλλουν ανά δευτερόλεπτο τις μετρήσεις τους μέσω του τηλεπικοινωνιακού δικτύου στο Κέντρο Ελέγχου, που βρίσκεται στις εγκαταστάσεις της ΕΚΧΑ. Στο Κέντρο Ελέγχου γίνεται η συλλογή των δεδομένων, η επεξεργασία τους σε πραγματικό χρόνο και η αρχειοθέτηση των πρωτογενών και παράγωγων δεδομένων. Επιπροσθέτως, το Κέντρο Ελέγχου παραλαμβάνει τα αιτήματα των χρηστών, ετοιμάζει τα δεδομένα που ζητήθηκαν και τα αποστέλλει στους χρήστες.

Το σύστημα αναφοράς του HEPOS είναι το HTRS07² (Hellenic Terrestrial System 2007), το οποίο αποτελεί υλοποίηση του ETRS89³ (European Terrestrial System 1989) στην Ελλάδα.

Για την υλοποίηση του ΕΓΣΑ87, μέσω του HEPOS, η ΕΚΧΑ Α.Ε ανέπτυξε σε συνεργασία με το ΤΑΤΜ ΑΠΘ ένα μοντέλο μετασχηματισμού μεταξύ HTRS07 και

² HTRS07: το HTRS07 αποτελεί την επίσημη υλοποίηση του ETRS89 στην Ελλάδα, η οποία εγκρίθηκε επίσημα τον Ιούνιο του 2010 από την αρμόδια επιτροπή της EUREF (Ευρωπαϊκή επιτροπή για τα συστήματα αναφοράς).

³ ETRS89: το ETRS89 αποτελεί το Ευρωπαϊκό σύστημα αναφοράς, το οποίο ορίστηκε προκειμένου να αποφευχθεί ο δυναμικός χαρακτήρας του Συστήματος Αναφοράς ITRS (ισχύον παγκόσμιο σύστημα αναφοράς για γεωδαιτικές εργασίες).

ΕΓΣΑ87, το οποίο υλοποιήθηκε, χρησιμοποιώντας μετρήσεις 2470 τριγωνομετρικών σημείων της ΓΥΣ (Γεωγραφική Υπηρεσία Στρατού).

Το HEPOS παρέχει δύο ειδών υπηρεσίες προσδιορισμού θέσης: τις υπηρεσίες πραγματικού χρόνου (RTK εφαρμογές με ακρίβεια λίγων cm) και τις υπηρεσίες μετεπεξεργασίας (post-processing). Σημειώνεται ότι και στις δύο μεθόδους μπορεί να χρησιμοποιηθεί ως βασικός σταθμός τόσο ένας εικονικός όσο και ένας απομακρυσμένος μόνιμος.

(ΚΤΗΜΑΤΟΛΟΓΙΟ Α.Ε, HEPOS, 2009)

2.2.5. European Petroleum Survey Group (EPSG)

Το EPSG ιδρύθηκε το 1985 και υπήρξε επιστημονικός οργανισμός με αντικείμενο την εφαρμοσμένη γεωδαισία και χαρτογραφία. Επίτευγμα του συγκεκριμένου οργανισμού είναι η δημιουργία της μεγαλύτερης βάσης δεδομένων ελλειψοειδών, προβολών και συστημάτων αναφοράς. Το 2005 παρέδωσε τα σκήπτρα στην International Association of Oil and Gas Producers (OGP) η οποία και συνεχίζει το έργο του.

Στον πίνακα 2-5 αναφέρονται κάποιοι χρήσιμοι EPSG κωδικοί που αφορούν τα ελληνικά συστήματα αναφοράς αλλά και το παγκόσμιο.

Πίνακας 2-5: Κωδικοί EPSG.

Κωδικοί EPSG	Συστήματα αναφοράς
EPSG 4326	WGS 84
EPSG 4321	ΕΓΣΑ 87
EPSG 23034/5	ED 50 στα όρια του Ελλαδικού χώρου.
EPSG 2100	Προβολή Transverse Mercator στα όρια του ελλαδικού χώρου.
EPSG 900913	Προβολή Spherical Mercator που χρησιμοποιεί η Google στις εφαρμογές της.

3. WEBGIS

Τα WebGIS αποτελούν τα γεωγραφικά συστήματα πληροφοριών που λειτουργούν στο Διαδίκτυο. Μέσω των WebGIS παρέχεται η δυνατότητα διανομής και ανάλυσης των χωρικών δεδομένων, χωρίς την ύπαρξη κάποιου πακέτου GIS στον υπολογιστή του κάθε χρήστη. Τα πλεονεκτήματα που παρέχει ένα γεωγραφικό σύστημα πληροφοριών που λειτουργεί στο Διαδίκτυο είναι τα παρακάτω:

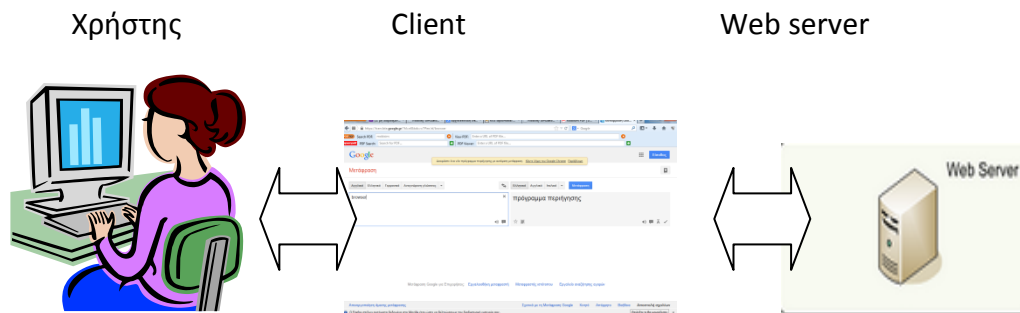
- Γρήγορη και εύκολη διανομή και πρόσβαση σε χωρικά δεδομένα.
- Δυνατότητα ανάλυσης και υποβολής χωρικών ερωτημάτων.
- Άμεση και απλοποιημένη χρήση.
- Χαμηλό κόστος.

3.1. Αρχιτεκτονική των WebGIS

Λέγοντας αρχιτεκτονική εννοούμε το σύνολο των μηχανισμών, βιβλιοθηκών και λογισμικών που πρέπει να χρησιμοποιηθούν προκειμένου ένα γεωγραφικό σύστημα πληροφοριών να είναι πλήρες. Ο βασικός σκελετός ενός WebGIS αποτελείται από τρία βασικά επίπεδα. Το πρώτο επίπεδο είναι ο πελάτης (client), ενώ το δεύτερο αποτελείται από μία ομάδα εξυπηρετητών και λογισμικών, που είτε λειτουργούν στον ίδιο εξυπηρετητή είτε σε διαφορετικούς. Τα λογισμικά ονομάζονται server-side λογισμικά και ουσιαστικά ενεργοποιούν τη δυνατότητα χρήσης των server-side διαδικτυακών γλωσσών προγραμματισμού (Java, PHP, Python κ.α). Ομάδα εξυπηρετητών αποκαλούμε τον εξυπηρετητή Διαδικτύου (web server) και τον εξυπηρετητή χαρτών (map server). Τέλος, το τρίτο επίπεδο αποτελείται από τον εξυπηρετητή δεδομένων, στον οποίο υπάρχει και η βάση δεδομένων.

3.1.1. Client

Ο πελάτης (client) αποτελεί το πρόγραμμα περιήγησης που έχει ο κάθε χρήστης στον υπολογιστή του και είναι απαραίτητος για την περιήγηση στο Διαδίκτυο. Αποστολή του client είναι να επικοινωνεί με τον χρήστη (user) και τον web server. Η επικοινωνία με τον χρήστη στο σύνολό της ουσιαστικά σημαίνει απόδοση της ιστοσελίδας και ασύγχρονη ανταλλαγή δεδομένων μέσω των γλωσσών client side (JavaScript). Όταν ένας χρήστης πληκτρολογεί μία διεύθυνση, ο client λαμβάνει το αίτημα και μέσω του πρωτοκόλλου HTTP το στέλνει στον αντίστοιχο εξυπηρετητή Διαδικτύου (Web server) που βρίσκεται αποθηκευμένη η συγκεκριμένη ιστοσελίδα. Ο web server απαντά στο αίτημα δίνοντάς του την ιστοσελίδα που ζητάει.



Εικόνα 3-1: Λειτουργία client.

Η ασύγχρονη ανταλλαγή δεδομένων μεταξύ του client και του χρήστη πραγματοποιείται με τη χρήση σεναρίων (scripts), τα οποία συντάσσονται με τις client side γλώσσες προγραμματισμού. Τα σενάρια, τις περισσότερες φορές, αποτελούν συναρτήσεις της JavaScript, τα οποία εκτελούνται και τροποποιούν αντίστοιχα την ιστοσελίδα με την εφαρμογή ενός γεγονότος (event). Χαρακτηριστικό παράδειγμα γεγονότος αποτελεί το πάτημα ενός κουμπιού (onclick), του οποίου η σύνταξη φαίνεται παρακάτω.

```
<input name="example" type="button" onClick="function()  
" value="event" />
```

Ειδικότερα, σε γλώσσα HTML δημιουργήθηκε ένα κουμπί, που όταν το πατήσουμε, εκτελείται η συνάρτηση function(). Η τελευταία, αποτελεί το σενάριο και έχει δημιουργηθεί με JavaScript.

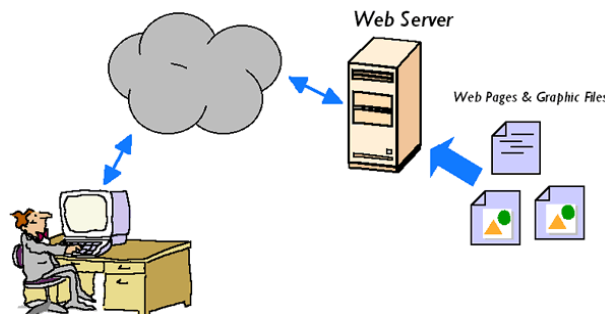
3.1.2. Web Server και Application server

Ο εξυπηρετητής Διαδικτύου αποτελεί το λογισμικό που απαντάει στα αιτήματα του πελάτη μέσω του πρωτοκόλλου HTTP. Έχει σχεδιαστεί, ώστε να απαντάει αποτελεσματικά στα αιτήματα ενός μεγάλου αριθμού πελατών «σερβίροντάς» τους στατικά αρχεία. Το αίτημα υποβάλλεται από έναν χρήστη με τη μορφή URL π.χ.:

```
http://example.com/some/path/page.html  
http://example.com/some/path/image.jpg  
http://example.com/some/path/archive.zip  
http://example.com/some/path/data.xml
```

Όταν δημιουργηθεί το αίτημα, στέλνεται μέσω του HTTP στον web server, ο οποίος αναζητά στα αρχεία του σκληρού δίσκου του server, αν υπάρχει το εν λόγω αρχείο. Ένα αρχείο δεν θα βρεθεί στον server είτε όταν δεν γραφτεί σωστά είτε όταν το αίτημα δεν δοθεί με το σωστό path. Σε περίπτωση που το πρόγραμμα περιήγησης

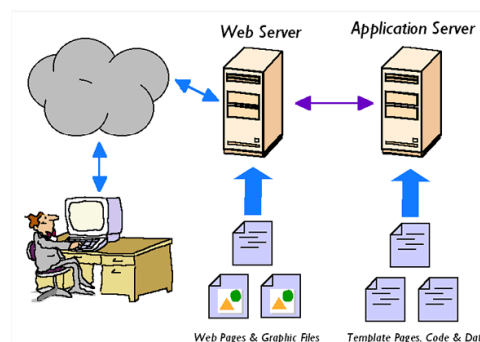
δεν είναι ρυθμισμένο να εμφανίζει κάποιο συγκεκριμένο τύπο αρχείου, τότε ζητείται από τον χρήστη να αποθηκεύσει το αρχείο στον τοπικό του δίσκο.



Εικόνα 3-2:Λειτουργία Web server

Οι πιο δημοφιλείς εξυπηρετητές Διαδικτύου που χρησιμοποιούνται σήμερα είναι ο Apache HTTP server και ο Internet Information Services (IIS). Ο Apache λειτουργεί σε όλα τα σύγχρονα λειτουργικά συστήματα, όπως Windows, Linux., Mac OS X και Unix. Κυκλοφόρησε υπό την άδεια του λογισμικού Apache και είναι λογισμικό ανοικτού κώδικα⁴. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation).

Την αδυναμία του web server να διαχειρίζεται και να επιστρέφει μόνο στατικά έγγραφα έρχεται να καλύψει ο εξυπηρετητής λογισμικών (application server). Ο τελευταίος, διαχειρίζεται τόσο το δυναμικό περιεχόμενο των ιστοσελίδων μέσω τον υποστηριζόμενων script engine, όσο και τις ταυτόχρονες αιτήσεις των χρηστών. Συνεπώς, ο web server περιμένει από τον application server να του επιστρέψει το αποτέλεσμα του δυναμικού περιεχομένου και εκείνος με την σειρά του να κάνει την δουλειά του επιστρέφοντας το τελικό στατικό αρχείο. Μέσω του application server γίνεται δυνατή η επικοινωνία αφενός με τη βάση δεδομένων και αφετέρου με άλλους εξυπηρετητές, όπως είναι ο εξυπηρετητής χαρτών.

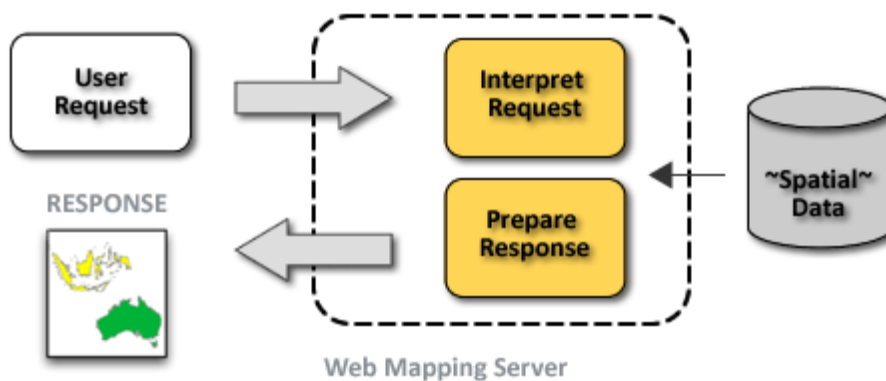


Εικόνα 3-3:Λειτουργία Web server με application server.

⁴ Ανοικτού κώδικα: ο πηγαίος κώδικας διατίθεται με κάποιον τρόπο ελεύθερα σε όσους ζητούν να τον εξετάσουν, ακόμα και να τον τροποποιήσουν ή να τον αξιοποιήσουν σε άλλες εφαρμογές.

3.1.3. Map server

Ο εξυπηρετητής χαρτών αποτελεί ένα είδος εξυπηρετητή εφαρμογών με δυνατότητα διαχείρισης, επεξεργασίας και οπτικοποίησης των χωρικών δεδομένων. Το κύριο χαρακτηριστικό των map servers είναι η λήψη των χωρικών δεδομένων από μία χωρική βάση δεδομένων και η διάχυσή τους στο Διαδίκτυο με χρήση κατάλληλων γεωχωρικών προτύπων και υπηρεσιών. Ένας map server μπορεί είτε από μόνος του να είναι web server, οπότε και ονομάζεται web-mapping server είτε εγκαθίσταται σε έναν web server, που έχει τις κατάλληλες server side γλώσσες προγραμματισμού για να τον υποστηρίξουν.



Εικόνα 3-4: Web-mapping server.

Πηγή: OpenGeo,2012

Οι πιο δημοφιλείς web-mapping servers που χρησιμοποιούνται είναι ο GeoServer, ο MapServer και ο ArcGIS server. Από αυτούς, οι δύο πρώτοι αποτελούν ανοικτού κώδικα web-mapping server, ενώ ο τρίτος είναι εμπορικός.

3.1.4. Data server

Ο εξυπηρετητής δεδομένων διανέμει τα δεδομένα σε μια βάση δεδομένων. Συνήθως, όταν υπάρχουν χωρικά δεδομένα, πραγματοποιείται σύνδεση της βάσης δεδομένων με τον map server και τα ερωτήματα στην βάση πραγματοποιούνται μέσω αυτού. Για μη χωρικά δεδομένα, η λήψη των δεδομένων από τη βάση δεδομένων γίνεται μέσω του εξυπηρετητή εφαρμογών και συγκεκριμένα μέσω των διαθέσιμων server side γλωσσών. Τα ερωτήματα στη βάση δεδομένων είτε είναι χωρικά είτε όχι υποβάλλονται με χρήση της γλώσσας SQL⁵ (Structured Query Language). Οι πιο δημοφιλείς βάσεις δεδομένων που χρησιμοποιούνται για

⁵ SQL: η SQL (Structured Query Language) είναι μία γλώσσα υπολογιστών στις βάσεις δεδομένων, που σχεδιάστηκε για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Database Management System, RDBMS) και η οποία, αρχικά, βασίστηκε στη σχεσιακή άλγεβρα. Η γλώσσα περιλαμβάνει δυνατότητες ανάκτησης και ενημέρωσης δεδομένων, δημιουργίας και τροποποίησης σχημάτων και σχεσιακών πινάκων, αλλά και ελέγχου πρόσβασης στα δεδομένα.

διαδικτυακές εφαρμογές είναι η PostgreSQL, η MySQL και η Oracle. Οι δύο πρώτες αποτελούν βάσεις δεδομένων ανοικτού κώδικα, ενώ η Oracle είναι εμπορικό προϊόν. Επιπροσθέτως, η MySQL δεν ενδείκνυται για χωρικά δεδομένα, λόγω των φτωχών δυνατοτήτων της στο τομέα αυτό. Αντιθέτως, η PostgreSQL με την ενσωμάτωση της χωρικής βάσης δεδομένων PostGIS είναι η πλέον κατάλληλη.

3.2. Πρότυπα Γεωχωρικών Υπηρεσιών Διαδικτύου

Οι γεωχωρικές υπηρεσίες Διαδικτύου αποτελούν υπηρεσίες, μέσω των οποίων γίνεται δυνατή η πρόσβαση και η απεικόνιση των γεωγραφικών πληροφοριών που είναι αποθηκευμένες σε διάφορες βάσεις δεδομένων στο Διαδίκτυο. Επιπροσθέτως, εκτελούνται απλοί και σύνθετοι υπολογισμοί που αφορούν την γεωμετρία ενός συνόλου γεωγραφικών αντικειμένων. Τέλος, επιτυγχάνεται η επιστροφή μηνυμάτων που περιέχουν γεωγραφικές πληροφορίες και μπορούν να παραδοθούν ως κείμενο, αριθμητικά δεδομένα ή γεωγραφικά χαρακτηριστικά γνωρίσματα.

Η OGC (Open Geospatial Consortium) αποτελεί μια διεθνή κοινοπραξία που αποτελείται από 473 εταιρείες, κρατικούς φορείς και πανεπιστήμια που συμμετέχουν στη δημιουργία κατάλληλων προτύπων και υπηρεσιών, προκειμένου να είναι δυνατή η επεξεργασία και η ανταλλαγή γεωχωρικών δεδομένων. Οι πιο σημαντικές γεωχωρικές υπηρεσίες Διαδικτύου που δημιουργήθηκαν από την εν λόγω κοινοπραξία είναι η WMS, WFS, WPS, WCS και WFS-T.

3.2.1. Web Map Service (WMS)

Μέσω του προτύπου WMS διατίθενται στο Διαδίκτυο εικόνες με γεωαναφορά. Ο πελάτης στέλνει ένα αίτημα στον εξυπηρετητή χαρτών και εκείνος, με βάση τις παραμέτρους του αιτήματος, παράγει την τελική εικόνα και απαντά στο αίτημα. Η εικόνα που παράγεται από το WMS αποτελεί τελικό προϊόν αιτήματος που μπορεί να προέρχεται είτε από διανυσματικά είτε από κανονικοποιημένα δεδομένα είτε από συνδυασμό αυτών. Όπως έχει προαναφερθεί στη παράγραφο 3.1, ο εξυπηρετητής χαρτών είναι συνδεδεμένος με τη χωρική βάση δεδομένων. Επιπλέον, όταν δημιουργείται το αίτημα από τον πελάτη, ο εξυπηρετητής χαρτών μετατρέπει τα χωρικά δεδομένα μέσω του WMS σε μια τελική εικόνα με γεωαναφορά και ακολούθως απαντάει στον πελάτη στέλνοντας την εν λόγω εικόνα.

➤ Βασικά αιτήματα WMS

Το αίτημα σε ένα πρότυπο WMS πρέπει να καθορίζεται και να είναι είτε της μορφής GetCapabilities είτε GetMap. Το πρώτο χρησιμοποιείται για την απόδοση των μεταδεδομένων των χωρικών δεδομένων σε μορφή XML εγγράφου. Οι παράμετροι που καθορίζονται σε ένα αίτημα της μορφής GetCapabilities φαίνονται στην εικόνα 3.5.

Request parameter	Mandatory/optional	Description
VERSION=version	O	Request version
SERVICE=WMS	M	Service type
REQUEST=GetCapabilities	M	Request name
FORMAT=MIME_type	O	Output format of service metadata
UPDATESEQUENCE=string	O	Sequence number or string for cache control

Εικόνα 3-5: Παράμετροι αιτήματος GetCapabilities.

Πηγή: Jeff de la Beaujardiere , 2006

Από τις παραπάνω παραμέτρους πρέπει υποχρεωτικά να ορίζεται η υπηρεσία (service) που καθορίζει το πρότυπο που χρησιμοποιεί ο εξυπηρετητής χαρτών καθώς και το είδος του αιτήματος (Request).

Το αποτέλεσμα ενός αιτήματος με τη μορφή GetMap είναι μια εικόνα με γεωαναφορά. Προκειμένου να είναι πλήρες ένα τέτοιο αίτημα, θα πρέπει να περιλαμβάνει στοιχεία που αφορούν το αίτημα, το όνομα του θεματικού επιπέδου όπως έχει καταχωρηθεί στον εξυπηρετητή χαρτών, το στυλ του θεματικού επιπέδου, το σύστημα αναφοράς, τα όρια αναζήτησης και τελικώς το μέγεθος και το format της τελικής εικόνας που δημιουργείται.

(Jeff de la Beaujardiere , 2006)

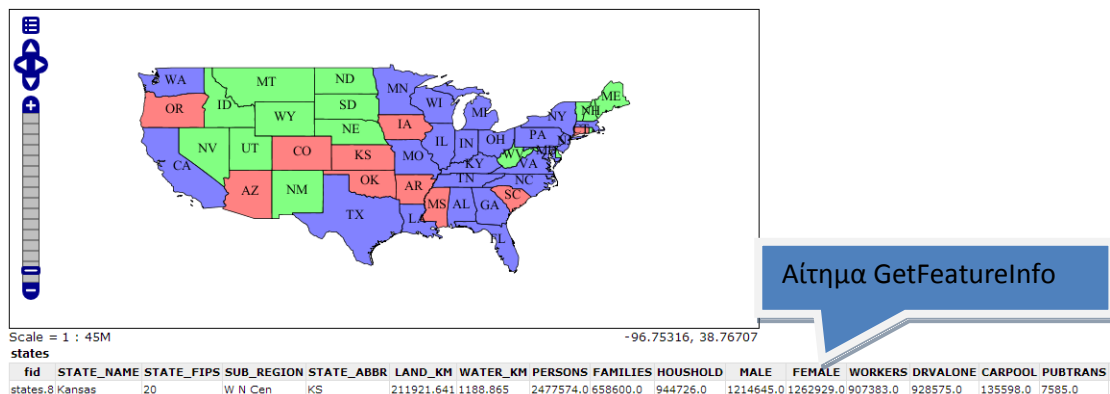
Request parameter	Mandatory/optional	Description
VERSION=1.3.0	M	Request version.
REQUEST=GetMap	M	Request name.
LAYERS=layer_list	M	Comma-separated list of one or more map layers.
STYLES=style_list	M	Comma-separated list of one rendering style per requested layer.
CRS=namespace:identifier	M	Coordinate reference system.
BBOX=minx,miny,maxx,maxy	M	Bounding box corners (lower left, upper right) in CRS units.
WIDTH=output_width	M	Width in pixels of map picture.
HEIGHT=output_height	M	Height in pixels of map picture.
FORMAT=output_format	M	Output format of map.
TRANSPARENT=TRUE FALSE	O	Background transparency of map (default=FALSE).
BGCOLOR=color_value	O	Hexadecimal red-green-blue colour value for the background color (default=0xFFFFFF).
EXCEPTIONS=exception_format	O	The format in which exceptions are to be reported by the WMS (default=XML).
TIME=time	O	Time value of layer desired.
ELEVATION=elevation	O	Elevation of layer desired.
Other sample dimension(s)	O	Value of other dimensions as appropriate.

Εικόνα 3-6: Παράμετροι αιτήματος GetMap.

Πηγή: Jeff de la Beaujardiere, 2006

➤ Προαιρετικό αίτημα WMS

Το αίτημα GetFeatureInfo αποτελεί προαιρετικό αίτημα για την αναζήτηση πληροφοριών των στοιχείων ενός χάρτη που παράγεται από την υπηρεσία WMS με βάση την τιμή του pixel (i,j) της εικόνας.



Εικόνα 3-7: Χρήση αιτήματος GetFeatureInfo.

3.2.2. Web Feature Service (WFS)

Η υπηρεσία WFS χρησιμοποιείται για την άμεση χρησιμοποίηση και εκμετάλλευση των διανυσματικών δεδομένων, επιστρέφοντας την πραγματική γεωμετρία και τα χαρακτηριστικά αυτών. Η επίτευξη της περιγραφής των χωρικών δεδομένων γίνεται με τη χρήση της γλώσσας GML (Geographic Markup Language), η οποία είναι επέκταση της XML.

➤ Αιτήματα WFS

Τα βασικά αιτήματα που περιέχει μια υπηρεσία WFS είναι της μορφής GetCapabilities, DescribeFeatureType, GetFeature, LockFeature και Transaction. Μέσω της GetCapabilities λαμβάνει ο χρήστης τα μεταδεδομένα των χωρικών δεδομένων σε μορφή XML εγγράφου.

```
http://example.com/geoserver/wfs?
service=wfs&
version=1.1.0&
request=GetCapabilities
```

Με το DescribeFeatureType εμφανίζονται, σε μορφή GML, πληροφορίες είτε για ένα θεματικό επίπεδο είτε για ένα συγκεκριμένο στοιχείο του θεματικού επιπέδου.

```
http://example.com/geoserver/wfs?
service=wfs&
```

```
version=2.0.0&
request=DescribeFeatureType&
typeName=namespace:featuretype
```

Με το αίτημα GetFeature επιστρέφονται στον χρήστη τα πραγματικά διανυσματικά δεδομένα με την γεωμετρία και τα περιγραφικά τους χαρακτηριστικά.

```
http://example.com/geoserver/wfs?
service=wfs&
version=2.0.0&
request=GetFeature&
typeName=namespace:featuretype&
featureID=feature
```

Στο παραπάνω παράδειγμα γίνεται αίτημα μέσω HTTP για το στοιχείο feature ενός συγκεκριμένου θεματικού επιπέδου Featuretype, όπως έχει δημιουργηθεί και καταχωρηθεί στον εξυπηρετητή χαρτών GeoServer.

Το αίτημα Transaction δίνει την δυνατότητα δημιουργίας, τροποποίησης και διαγραφής στοιχείων που έχουν δημοσιευθεί μέσω της υπηρεσίας WFS. Η υπηρεσία WFS που υποστηρίζει το συγκεκριμένο αίτημα ονομάζεται WFS-T (Web Feature Service-Transaction).

```
<wfs:Transaction service="WFS" version="1.0.0"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:topp="http://www.openplans.org/topp"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs http://schemas.opengis.net/wfs/
1.0.0/WFS-transaction.xsd http://www.openplans.org/topp
http://localhost:8080/geoserver/wfs/DescribeFeatureType?typename=topp:tasmania_roads">
  <wfs:Insert>
    <topp:tasmania_roads>
      <topp:the_geom>
        <gml:MultiLineString srsName="http://www.opengis.net/gml/srs/epsg.xml#4326">
          <gml:lineStringMember>
            <gml:LineString>
              <gml:coordinates decimal="." cs="," ts=" ">
2494475.71056415,5433016.8189323 494982.70115662,5435041.95096618
              </gml:coordinates>
            </gml:LineString>
          </gml:lineStringMember>
        </gml:MultiLineString>
      </topp:the_geom>
      <topp:TYPE>alley</topp:TYPE>
    </topp:tasmania_roads>
  </wfs:Insert>
</wfs:Transaction>
```

Εικόνα 3-8: Παράδειγμα δημιουργίας στοιχείου μέσω αιτήματος Transaction.

Τέλος, το LockFeature χρησιμοποιείται για την προστασία των στοιχείων από τις δυνατότητες του αιτήματος Transaction.

(Πηγή: <http://docs.geoserver.org/stable/en/user/>)

3.2.3. Web Coverage Service (WCS)

Η WCS αποτελεί την WFS υπηρεσία για τα κανονικοποιημένα δεδομένα (raster). Αποτελεί πρότυπο μέσω του οποίου γίνεται δυνατή η πρόσβαση σε δεδομένα κανάβου (grid coverage's) σε μια διαδικτυακή υπηρεσία. Τα δεδομένα κανάβου είναι δεδομένα που αφορούν τηλεσκοπικές απεικονίσεις, δορυφορικές εικόνες, ψηφιακές αεροφωτογραφίες, ψηφιακά μοντέλα εδάφους καθώς και φαινόμενα που μπορούν να αναπαρασταθούν από τιμές σε κάθε σημείο μέτρησης. Οι μορφές αιτημάτων που υποστηρίζει η υπηρεσία WCS είναι η GetCapabilities, η DescribeCoverage και η GetCoverage. Το πρώτο αίτημα επιστρέφει τα μεταδεδομένα του εν λόγω χωρικού δεδομένου. Η DescribeCoverage επιστρέφει σε μορφή XML εγγράφου την πλήρη περιγραφή του αιτήματος κάλυψης. Τέλος, η GetCoverage επιστρέφει το τελικό αίτημα σε μορφή εικόνας.

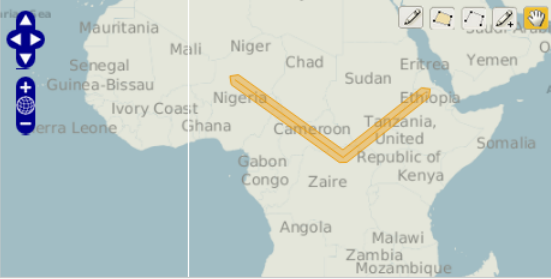
Σημειώνεται ότι δεν πρέπει να συγχέονται τα δύο πρότυπα WMS και WCS. Μέσω του προτύπου WMS πραγματοποιείται οπτικοποίηση των χωρικών δεδομένων σε raster μορφή. Το τελικό αποτέλεσμα αποτελεί μια εικόνα, στην οποία δεν υπάρχει δυνατότητα περαιτέρω ανάλυσης και επεξεργασίας. Αντίθετα, με το πρότυπο WCS η εικόνα αποτελεί έναν κানাβο δεδομένων, στα οποία ο χρήστης μπορεί να έχει πρόσβαση μέσω των αιτημάτων που υποστηρίζει η εν λόγω υπηρεσία.

(Πηγή: <http://docs.geoserver.org/stable/en/user/>)

3.2.4. Web Processing Service (WPS)

Η υπηρεσία WPS παρέχει λειτουργίες ΓΣΠ σε πελάτες ενός δικτύου, συμπεριλαμβανομένης της πρόσβασης σε προγραμματισμένους εκ των προτέρων υπολογισμούς και υπολογιστικά μοντέλα, τα οποία σχετίζονται με χωρικά δεδομένα (Schut, Whiteside, 2005). Μπορεί να παρέχει απλούς υπολογισμούς, όπως τον ορισμό ενός buffer zone ή πολύπλοκους, όπως ένα μοντέλο παγκόσμιων κλιματικών αλλαγών. Η WPS στοχεύει στην επεξεργασία τόσο διανυσματικών όσο και raster δεδομένων. Σημειώνεται ότι με τη συγκεκριμένη υπηρεσία υπάρχει δυνατότητα εφαρμογής διαδοχικών αλγορίθμων και διαδικασιών πάνω σε κάποια δεδομένα. Επομένως, τα αποτελέσματα θα βοηθήσουν τη λήψη αποφάσεων σε πολύπλοκα προβλήματα. Τα αιτήματα που μπορεί να δημιουργηθούν μέσω της υπηρεσίας WPS είναι η GetCapabilities, η DescribeProcess και η Execute. Με την πρώτη μορφή αιτήματος λαμβάνονται πληροφορίες σχετικά με την υπηρεσία WPS σε μορφή XML εγγράφου. Με την DescribeProcess δίνονται πληροφορίες σχετικά με τη διαδικασία που έχει επιλεγεί να εκτελεστεί. Συγκεκριμένα, σε μορφή XML περιέχονται πληροφορίες που αφορούν το όνομα της διαδικασίας, τα χωρικά δεδομένα εισόδου, τις παραμέτρους και τέλος τα format των δεδομένων εξόδου. Τέλος, μέσω της Execute εφαρμόζεται η διαδικασία στην πράξη, δίνοντας το τελικό αποτέλεσμα της επεξεργασίας των χωρικών δεδομένων εισόδου (Schut, Whiteside, 2005).

Οπτικοποίηση
δεδομένων εισόδου
και εξόδου



Διαδικασία Buffer

JTS:buffer
 Returns a polygonal geometry representing the input geometry enlarged by a given distance around its exterior.
 Input:
 geom (select feature, then click field):
 LINESTRING(7.6025390625 15.765380859375,25.1806640625 3.109130859375,38.5400390625 13.656005859375)
 * 1.0 * 1 Round
 Execute

Δεδομένα εισόδου

Output:
 The result should also be visible on the map.
 POLYGON ((25.15294454587931 4.361322655401845, 37.92039477392098 14.440888624908427, 39.324921828033425 14.275650147954021, 39.15968335107902 12.871123093841573, 25.80030835107902 2.324248093841574, 24.596359336654924 2.2975965179235054, 7.018234336654924 14.953846517923505, 6.791004721048505 16.349685585220076, 8.186843788345076 16.576915200826495, 25.15294454587931 4.361322655401845))

Παράμετροι

Δεδομένα εξόδου

Εικόνα 3-9: Παράδειγμα υπηρεσίας WPS.

Στο παράδειγμα της εικόνας 3-9 εφαρμόζεται η διαδικασία buffer που έχει δημιουργηθεί εκ των προτέρων και ονομάζεται JTS:Buffer. Το JTS δηλώνει τη γλώσσα προγραμματισμού που χρησιμοποιήθηκε για να υλοποιηθεί η διαδικασία. Ως δεδομένα εισόδου δέχεται μια γραμμή, ενώ οι παράμετροι που πρέπει να καθοριστούν είναι το πάχος του Buffer και το τελικό του σχήμα. Τέλος, το αποτέλεσμα της διαδικασίας δίνει ένα πολύγωνο, το οποίο μπορεί να χρησιμοποιηθεί ως δεδομένο εισόδου σε άλλες διαδικασίες.

3.3. Μορφές Διανυσματικών Δεδομένων Διαδικτύου

Οι πιο σημαντικές μορφές διανυσματικών δεδομένων που χρησιμοποιούνται σήμερα είναι η GML, η KML και η GeoJSON. Η ύπαρξή τους είναι αρκετά σημαντική προκειμένου να είναι εφικτή η αποθήκευση, μεταφορά και οπτικοποίηση των γεωγραφικών πληροφοριών μέσω WebGis συστημάτων. Σημειώνεται ότι από τις τρεις παραπάνω μορφές, οι δύο πρώτες αποτελούν πλέον διεθνή πρότυπα κωδικοποίησης των γεωγραφικών πληροφοριών.

3.3.1. Geography Markup Language (GML)

Η GML αποτελεί μία επέκταση της γλώσσας XML και έχει οριστεί προκειμένου να εκφράζει και να μεταφέρει γεωγραφικά χαρακτηριστικά στο Διαδίκτυο. Αποτελεί διεθνές πρότυπο και έχει τη δυνατότητα να ενσωματώνει όχι μόνο διανυσματικά χαρακτηριστικά αλλά επιπλέον εικονιστικά δεδομένα και δεδομένα από

αισθητήρες. Η δυνατότητα ενσωμάτωσης των τελευταίων, αποτελεί κλειδί για τη χρησιμότητα της και υλοποιήθηκε με τη δημιουργία της έκδοσης GML 3.0.

➤ Αντικείμενα(objects) της GML

Η GML κωδικοποιεί τα γεωμετρικά αντικείμενα ως στοιχεία σε ένα έγγραφο. Οι τύποι των γεωμετρικών αντικειμένων που υποστηρίζουν οι εκδόσεις GML 1.0 και GML 2.0 είναι σημεία, γραμμές και πολύγωνα. Με την έκδοση GML 3.0 ενσωματώθηκαν νέες δομές, οι οποίες υποστηρίζουν πληροφορίες κάλυψης, κανονικοποιημένες δομές και δεδομένα αισθητήρων.

➤ Χαρακτηριστικά (features) της GML

Τα features ενός GML αντιπροσωπεύουν φυσικές οντότητες, οι οποίες μπορεί να περιέχουν γεωγραφικά ή/και περιγραφικά δεδομένα. Παράδειγμα χαρακτηριστικών αποτελούν ποτάμια, γέφυρες, κτίρια κ.α. Σημειώνεται, ότι δεν πρέπει να συγχέονται με τα γεωμετρικά αντικείμενα, δεδομένου ότι τα τελευταία ορίζουν μια θέση ή περιοχή, ενώ τα feature περιγράφουν τη φυσική οντότητα του στοιχείου. Συνεπώς, ένα feature μπορεί να αποτελείται από ένα ή περισσότερα ή και καθόλου γεωμετρικά αντικείμενα. Παράδειγμα μιας φυσικής οντότητας που μπορεί να περιέχει παραπάνω από δύο γεωμετρικά αντικείμενα είναι ένα κτίριο που περιγράφεται από τη θέση και την έκτασή του.

➤ Συντεταγμένες

Το γεωμετρικά αντικείμενα προσδιορίζονται με συντεταγμένες, οι οποίες καταχωρούνται εντός των παρακάτω κανόνων:

```
<gml:coordinates> 30.45, 54.21 </gml:coordinates>
<gml:pos> συντεταγμένες </gml:pos>
<gml:posList> συντεταγμένες </gml:posList>
```

Με τη χρήση της ετικέτας <gml:coordinates>, όπως φαίνεται στο παραπάνω παράδειγμα, δεν είναι εφικτή η πρόσβαση των συντεταγμένων ξεχωριστά (πχ 30.45), δεδομένου ότι το περιεχόμενο της εν λόγω ετικέτας αποτελεί συμβολοσειρά (String). Από την έκδοση GML 3.0 και μετά χρησιμοποιούνται για την καταχώρηση συντεταγμένων οι ετικέτες <gml:pos> και <gml:posList> προκειμένου να είναι δυνατή η προσβασιμότητα των ατομικών συντεταγμένων μέσω του XML DOM.

➤ Σύστημα αναφοράς συντεταγμένων (CRS)

Το CRS προσδιορίζει το καθορισμένο σύστημα αναφοράς των συντεταγμένων των γεωμετρικών αντικειμένων ενός GML εγγράφου.

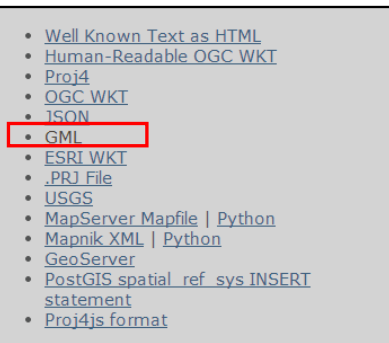
```
<gml:LineString gml:id="p21"srsName="/../EPSG/4326">
<gml:posList>45.67 88.56 55.56 89.44</gml:posList>
</gml:LineString >
```

Σύμφωνα με τον παραπάνω κώδικα δημιουργήθηκε ένα γεωμετρικό αντικείμενο «γραμμή», της οποίας το σύστημα αναφοράς προσδιορίζεται μέσω της ιδιότητας srsName. Η τιμή της τελευταίας είναι ένα URL ενός XML εγγράφου του συγκεκριμένου συστήματος αναφοράς βάσει των κωδικών του EPSG. Συγκεκριμένα, ο κωδικός 4326 προσδιορίζει το WGS84, ενώ η λήψη του αντίστοιχου XML εγγράφου γίνεται μέσω του ιστοτόπου <http://spatialreference.org>.

EPSG:4326

WGS 84 ([Google it](#))

- **WGS84 Bounds:** -180.0000, -90.0000, 180.0000, 90.0000
- **Projected Bounds:** -180.0000, -90.0000, 180.0000, 90.0000
- **Scope:** Horizontal component of 3D system. Used by the GPS satellite navigation system and for NATO military geodetic surveying.
- **Last Revised:** Aug. 27, 2007
- **Area:** World



Εικόνα 3-10: Ιστοτόπος spatialreference.org για το σύστημα αναφοράς WGS 84.

Στην εικόνα 3-10 παρατηρούνται τα διάφορα διαθέσιμα format που μπορεί ο χρήστης να χρησιμοποιήσει, αναλόγως του σκοπού, για τον προσδιορισμό του συστήματος αναφοράς των συντεταγμένων. Το format που επιλέγεται για την καταχώρηση της ιδιότητας srsName είναι το GML.

(Clemens ,2007)

3.3.2. KML Encoding Standard (KML)

Η γλώσσα Keyhole Markup Language (Kml) είναι ένας XML συμβολισμός που επικεντρώνεται στην οπτικοποίηση της γεωγραφικής πληροφορίας, συμπεριλαμβανομένων σημειώσεων πάνω στους χάρτες και τις εικόνες. Η γεωγραφική οπτικοποίηση περιλαμβάνει όχι μόνο την παρουσίαση χωρικών δεδομένων πάνω στην υδρόγειο, αλλά επίσης και τον έλεγχο της πλοήγησης του

χρήστη. Αναπτύχθηκε για χρήση με την εφαρμογή Keyhole Earth Viewer, της εταιρείας Keyhole (εταιρία χρηματοδοτούμενη από την CIA), η οποία το 2004 εξαγοράστηκε από την Google. Η εφαρμογή αναπτύχθηκε περισσότερο και ονομάστηκε Google Earth και σήμερα είναι εξαιρετικά δημοφιλής, με εκατομμύρια χρήστες σε όλο τον κόσμο. Δεδομένης της δημοτικότητας της εφαρμογής, η KML έχει γνωρίσει ευρεία διάδοση και έχει πλέον ενσωματωθεί σε πολλά συστήματα διαχείρισης της γεωγραφικής πληροφορίας, ως τρόπος διασύνδεσης με το Google Earth. Παράλληλα, το 2008 καθιερώθηκε σαν ένα διεθνές πρότυπο από τον OGC. Η KML είναι συμπληρωματική στα περισσότερα πρότυπα του OGC συμπεριλαμβανομένης της GML (Geography Markup Language), του WFS (Web Feature Service) και του WMS (Web Map Service). Σήμερα, η KML (v2.2) ενσωματώνει στοιχεία γεωμετρίας της GML (έκδοση 2.1.2). Αυτά τα στοιχεία περιλαμβάνουν τα σημεία, τις σειρές γραμμών, τους γραμμικούς δακτυλίους και τα πολύγωνα.

➤ Βασική δομή

Η βασική δομή ενός KML βασίζεται στις γενικές αρχές της γλώσσας XML (Extensible Markup Language), η οποία χρησιμοποιεί χαρακτήρες σήμανσης (tags) προκειμένου να καθορίσει την έναρξη και τη λήξη των διαφόρων ιδιοτήτων που περικλείει το αρχείο.

```
<kml xmlns="http://www.opengis.net/kml/2.2">
<Document>
ΚΑΘΟΡΙΣΜΟΣ ΓΕΝΙΚΩΝ ΙΔΙΟΤΗΤΩΝ
<Placemark>
ΚΑΘΟΡΙΣΜΟΣ ΧΩΡΙΚΩΝ ΚΑΙ ΠΕΡΙΓΡΑΦΙΚΩΝ ΔΕΔΟΜΕΝΩΝ
</Placemark>
</Document>
</kml>
```

Η ετικέτα: <kml xmlns="http://www.opengis.net/kml/2.2"> τοποθετείται στην αρχή του KML και δηλώνει ότι το αρχείο θα ακολουθήσει τους κανόνες του προτύπου KML (v2.2) του OGC. Το σύνολο των δεδομένων, μαζί με τον καθορισμό των γενικών ιδιοτήτων, καταχωρούνται εντός της ετικέτας <Document>.

Εντός της ετικέτας <Placemark> τοποθετούνται τα περιγραφικά και χωρικά δεδομένα ενός γεωμετρικού αντικειμένου. Το τελευταίο, μπορεί να αποτελεί ένα σημείο, μία γραμμή ή ένα πολύγωνο.

```
<Placemark>
<Point><coordinates>.....</coordinates></Point>
</Placemark>
```

Σημείο

```

<Placemark>
<LineString><coordinates>.....</coordinates></LineString>
</Placemark>
<Placemark>
<Polygon><outerBoundaryIs><LinearRing><coordinates>.....</coordinates></L
inearRing></outerBoundaryIs></Polygon>
</Placemark>

```

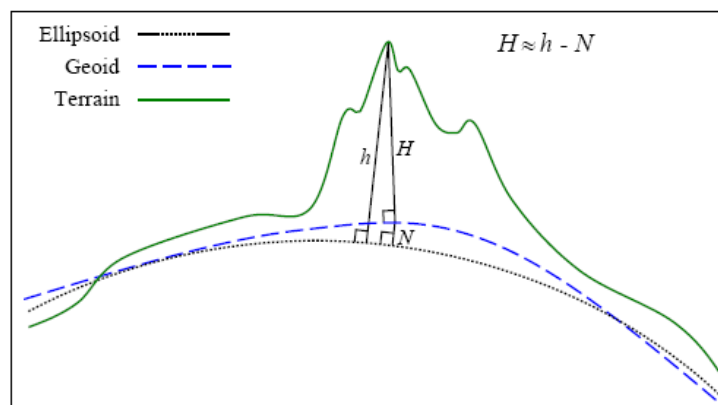
Γραμμή

Πολύγωνο

Σημειώνεται ότι οι συντεταγμένες κάθε γεωμετρικού αντικειμένου καθορίζονται εντός της ετικέτας <coordinates>.

➤ Σύστημα αναφοράς

Η ιδιαιτερότητα που έχει ένα KML αρχείο είναι ότι το σύστημα αναφοράς των γεωμετρικών αντικειμένων είναι προκαθορισμένο στο WGS84. Συνεπώς, οι συντεταγμένες τους πρέπει να δίνονται με γεωδαιτικό μήκος, γεωδαιτικό πλάτος και υψόμετρο. Το τελευταίο αποτελεί ορθομετρικό υψόμετρο και υπολογίζεται με βάση το μοντέλο γεωειδούς EGM 96.

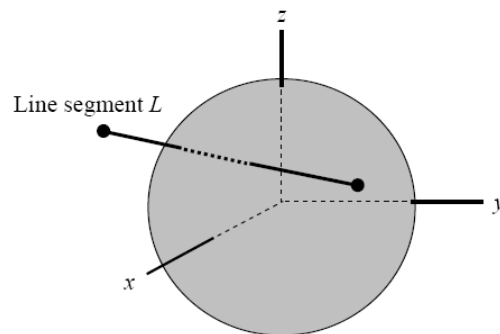


Εικόνα 3-11: Σχέση γεωμετρικού και ορθομετρικού υψομέτρου.

Πηγή: Wilson ,2008.

Οι ετικέτες που καθορίζουν τον τρόπο απεικόνισης των γεωμετρικών αντικειμένων (πολυγώνων και γραμμών) στο χώρο είναι τα <altitudeMode> και <tessellate>. Οι τιμές που παίρνει το πρώτο είναι clampToGround ή absolute, ενώ το δεύτερο true ή false. Ο συνδυασμός των δύο ετικετών επιφέρουν διαφορετικά αποτελέσματα στην οπτικοποίηση των γεωμετρικών αντικειμένων στο χώρο. Προκειμένου να γίνει αντιληπτή η σημαντικότητα των δύο ετικετών, θα γίνει χρήση ενός παραδείγματος για μία γραμμή. Η τελευταία, θα αποτελείται από ένα σημείο αρχής και ένα τέλους.

- 1^η περίπτωση : `<altitubeMode>absolute</ altitubeMode>`

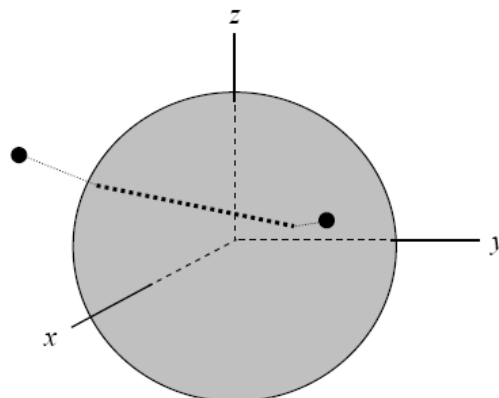


Εικόνα 3-12: Οπτικοποίηση γραμμής στο χώρο (1).
Πηγή: Wilson ,2008.

Τα σημεία αρχής και τέλους της γραμμής απεικονίζονται με βάση τις γεωδαιτικές συντεταγμένες στο χώρο, ενώ η γραμμή που τα ενώνει αποτελεί ευθύγραμμο τμήμα αυτών. Ουσιαστικά, αποτελεί την κεκλιμένη απόσταση μεταξύ δύο σημείων χωρίς να ακολουθεί το ανάγλυφο του εδάφους.

- 2^η Περίπτωση: `<altitubeMode> clampToGround </ altitubeMode>`

`<tessellate>false</tessellate>`

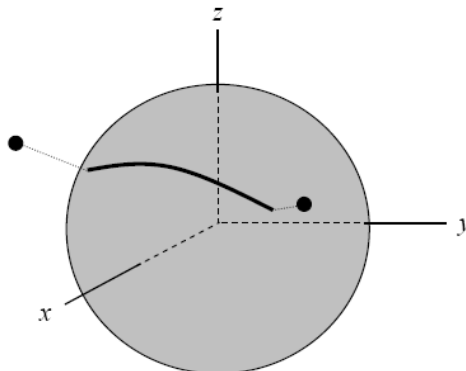


Εικόνα 3-13: Οπτικοποίηση γραμμής στο χώρο (2).
Πηγή: Wilson ,2008.

Σε αυτή την περίπτωση γίνεται προβολή των σημείων στο έδαφος και η γραμμή αποτελεί ευθύγραμμο τμήμα αυτών χωρίς να ακολουθεί το ανάγλυφο του εδάφους.

- 3^η Περίπτωση: `<altitudeMode> clampToGround </ altitudeMode>`

`<tessellate>true</tessellate>`



Εικόνα 3-14: Οπτικοποίηση γραμμής στο χώρο(3).

Πηγή: Wilson ,2008.

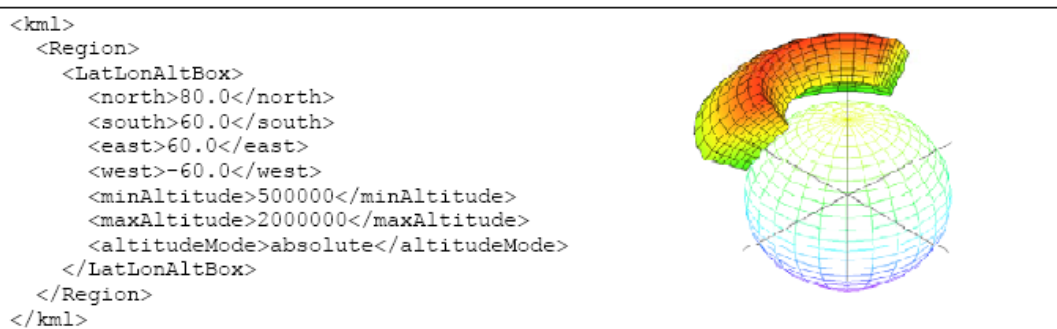
Στο τελευταίο παράδειγμα, τα δύο σημεία της γραμμής προβάλλονται στο έδαφος, ενώ η γραμμή που τα ενώνει ακολουθεί το ανάγλυφο του εδάφους.

Σημειώνεται ότι με την ίδια λογική γίνεται και η οπτικοποίηση των πολυγώνων στο χώρο.

➤ Ετικέτα <region>

Η ετικέτα region χρησιμοποιείται για την οπτικοποίηση τρισδιάστατων κουτιών, όπου οι γεωδαιτικές συντεταγμένες μπαίνουν σε ετικέτες με βάση τον προσανατολισμό (east, west, north, south), ενώ οι γραμμές που τις ενώνουν ακολουθούν τους αντίστοιχους μεσημβρινούς και παράλληλους της γης.

(Wilson ,2008)



Εικόνα 3-15: Παράδειγμα χρήσης της ετικέτας region.

Πηγή: Wilson ,2008.

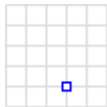
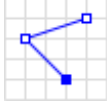
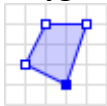
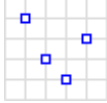
3.3.3. GeoJSON

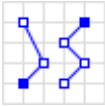
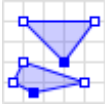
Το GeoJSON αποτελεί μορφή κωδικοποίησης της γεωγραφικής πληροφορίας, βασισμένο στο πρότυπο μεταφοράς δεδομένων JSON (JavaScript Object Notation). Σε σύγκριση με τις δύο προηγούμενες μορφές, η GeoJSON δεν αποτελεί διεθνές πρότυπο βάσει του OGC αλλά είναι αποτέλεσμα μίας ομάδας προγραμματιστών του διαδικτύου. Ένα GeoJSON μπορεί να αντιπροσωπεύει μία γεωμετρία, ένα χαρακτηριστικό ή μία συλλογή από χαρακτηριστικά. Σε ένα GeoJSON ένα αντικείμενο αποτελείται από ένα σύνολο μελών, τα οποία περιγράφονται από ένα όνομα (name) και μία τιμή (value). Το όνομα είναι πάντοτε μια συμβολοσειρά (String), ενώ η τιμή μπορεί να είναι συμβολοσειρά, νούμερο, αντικείμενο, πίνακας, λογική τιμή (true/false) ή μηδενική (null).

➤ Γεωμετρικά αντικείμενα

Τα γεωμετρικά αντικείμενα που υποστηρίζει ένα GeoJSON είναι σημεία (Point), γραμμές (LineString), πολύγωνα (Polygon), πολλαπλά σημεία (MultiPoint), πολλαπλές γραμμές (MultiLineString), πολλαπλά πολύγωνα (MultiPolygon) και συλλογή γεωμετρικών αντικειμένων (GEOMETRYCOLLECTION).

Πίνακας 3-1: Γεωμετρικά αντικείμενα GeoJSON.

Γεωμετρικό Αντικείμενο	Παράδειγμα	Παρατήρηση
Point 	<pre>{ "type": "Point", "coordinates": [30, 10] }</pre>	Οι συντεταγμένες αποτελούν μία θέση.
LineString 	<pre>{ "type": "LineString", "coordinates": [[30, 10], [10, 30], [40, 40]] }</pre>	Οι συντεταγμένες αποτελούν έναν πίνακα δύο ή περισσότερων θέσεων.
Polygon 	<pre>{ "type": "Polygon", "coordinates": [[[30, 10], [40, 40], [20, 40], [10, 20], [30, 10]]] }</pre>	Οι συντεταγμένες αποτελούν έναν πίνακα που περιλαμβάνει τον πίνακα συντεταμένων LineString.
MultiPoint 	<pre>{ "type": "MultiPoint", "coordinates": [[10, 40], [40, 30], [20, 20], [30, 10]] }</pre>	Οι συντεταγμένες αποτελούν έναν πίνακα πολλαπλών θέσεων.

MultiLineString 	<pre>{ "type": "MultiLineString", "coordinates": [[[10, 10], [20, 20], [10, 40]], [[40, 40], [30, 30], [40, 20], [30, 10]]] }</pre>	<p>Οι συντεταγμένες αποτελούν έναν πίνακα που αποτελείται από δύο ή περισσότερους πίνακες συντεταγμένων LineString.</p>
MultiPolygon 	<pre>{ "type": "MultiPolygon", "coordinates": [[[[30, 20], [45, 40], [10, 40], [30, 20]]], [[[15, 5], [40, 10], [10, 20], [5, 10], [15, 5]]]] }</pre>	<p>Οι συντεταγμένες αποτελούν έναν πίνακα που αποτελείται από δύο ή περισσότερους πίνακες συντεταγμένων Polygon.</p>
GeometryCollection	<pre>{ "type": "GeometryCollection", "geometries": [{ "type": "Point", "coordinates": [100.0, 0.0] }, { "type": "LineString", "coordinates": [[101.0, 0.0], [102.0, 1.0]] }] }</pre>	<p>Το όνομα που καταχωρείται στο συγκεκριμένο αντικείμενο είναι geometries, το οποίο είναι ένας πίνακας που αποτελείται από δύο ή περισσότερα γεωμετρικά αντικείμενα.</p>

➤ Χαρακτηριστικά αντικείμενα

Το feature - χαρακτηριστικό σε ένα GeoJSON περιέχει ένα γεωμετρικό αντικείμενο μαζί με τις ιδιότητές του. Συγκεκριμένα, ένα χαρακτηριστικό περιγράφεται από δύο ζεύγη ονομάτων και τιμών. Το πρώτο είναι το geometry, το οποίο παίρνει τα στοιχεία ενός γεωμετρικού αντικειμένου. Το δεύτερο είναι το properties, στο οποίο καταχωρούνται οι ιδιότητες του χαρακτηριστικού.

```
{ "type": "Feature",
  "geometry": { "type": "Point", "coordinates": [102.0, 0.5] },
  "properties": { "prop0": "value0" } }
```

➤ Σύστημα αναφοράς

Το σύστημα αναφοράς σε ένα GeoJSON καθορίζεται μέσω του αντικειμένου “crs”, το οποίο περιγράφεται από δύο ονόματα. Το πρώτο είναι το type και παίρνει τιμή EPSG, ενώ το δεύτερο είναι το properties που αποτελεί αντικείμενο και παίρνει διάφορα ζεύγη ονομάτων και τιμών. Το πιο σημαντικό είναι το code, το οποίο παίρνει ως τιμή τον αντίστοιχο κωδικό EPSG του συστήματος αναφοράς που μας ενδιαφέρει. Σε περίπτωση που δεν καθοριστεί το σύστημα αναφοράς, τότε θεωρείται σαν προεπιλογή το WGS 84. Σημειώνεται ότι την όλη περιγραφή του αντικειμένου crs μπορεί κάποιος να την βρει αναλόγως του συστήματος αναφοράς μέσω του ιστότοπου <http://spatialreference.org/>.

```
"crs": {
  "type": "EPSG",
  "properties": {
    "code": 4326
  }
}
```

(Πηγή: <http://geojson.org/geojson-spec.html>)

3.4. Οδηγία Inspire

Στις 14 Μαρτίου 2007 ψηφίστηκε από το Ευρωπαϊκό Κοινοβούλιο η Οδηγία 2007/2/EK INSPIRE (Infrastructure for Spatial Information in Europe), η οποία τέθηκε σε ισχύ στις 15 Μαΐου 2007. Η εφαρμογή της προβλέπεται να γίνει σταδιακά, με πλήρη εφαρμογή το 2019. Η Οδηγία δημιουργεί το νομικό πλαίσιο για την ίδρυση και λειτουργία της Υποδομής για τη γεωχωρική πληροφορία στην Ευρώπη, με σκοπό τη διαμόρφωση, εφαρμογή, διαχείριση και εκτίμηση των πολιτικών της Ευρωπαϊκής Ένωσης σε όλα τα επίπεδα αλλά και για την παροχή πληροφοριών προς το κοινό. Για την υλοποίηση της Οδηγίας, η Ευρωπαϊκή Επιτροπή είναι ο Συντονιστής σε Ευρωπαϊκό επίπεδο και τα κράτη μέλη είναι αρμόδια για την υλοποίηση της Οδηγίας σε εθνικό επίπεδο. Η Οδηγία 2007/2/EK INSPIRE περιλαμβάνει γεωγραφικές και περιβαλλοντικές πληροφορίες, συνολικά σε 34 θεματικές ενότητες, ομαδοποιημένες σε τρία Παραρτήματα του κειμένου της Οδηγίας, που αντιστοιχούν περίπου και στην προτεραιότητα ένταξής τους στο πληροφοριακό σύστημα. (<http://www.inspire.okxe.gr>).

Πίνακας 3-2:Θεματικά επίπεδα χωρικών δεδομένων οδηγίας INSPIRE.

ΠΑΡΑΡΤΗΜΑ 1			
Συστήματα αναφοράς	Διοικητικές μονάδες	Δίκτυα μεταφορών	
Σύστημα γεωγραφικού πλέγματος	Διευθύνσεις	Υδρογραφία	
Γεωγραφικά ονόματα	Κτηματικά τεμάχια	Προστατευόμενες περιοχές	

ΠΑΡΑΡΤΗΜΑ 2			
Υψόμετρα	Κάλυψης Γης	Ορθό-εικόνες	Γεωλογία

ΠΑΡΑΡΤΗΜΑ 3		
Στατιστικές μονάδες	Βιομηχανικές και παραγωγικές εγκαταστάσεις	Ωκεανογραφικά γεωγραφικά στοιχεία
Κτίρια	Αγροτικές και υδρολογικές εγκαταστάσεις	Θαλάσσιες περιοχές
Έδαφος	Κατανομή πληθυσμού και δημογραφία	Βίο-γεωγραφικές περιοχές
Χρήσης γης	Ζώνες διαχείρισης περιοχών/περιορισμών/κανονισμών και μονάδες εκθέσεων	Φυσικό περιβάλλον και βιότοποι
Ανθρώπινη υγεία και ασφάλεια	Ζώνες φυσικής επικινδυνότητας	Κατανομή βιολογικών οργανισμών
Υπηρεσίες κοινής ωφέλειας και κυβερνητικές υπηρεσίες	Ατμοσφαιρικές συνθήκες	Ενεργειακοί πόροι
Εγκαταστάσεις περιβαλλοντικού ελέγχου	Μετεωρολογικά γεωγραφικά στοιχεία	Ορυκτοί πόροι

Πηγή: περί ΚΤΗΜΑΤΩΝ ΛΟΓΟΣ και ΚΤΗΜΑΤΟΛΟΓΙΟ (ΤΟΜΟΣ Β')

Για την εφαρμογή της οδηγίας INSPIRE έπρεπε να συνταχθούν Κανονισμοί Εφαρμογής, οι οποίοι είναι ευρωπαϊκά νομοθετήματα και είναι υποχρεωτικά και απευθείας εκτελεστά από τα κράτη μέλη. Για την σύνταξη των εν λόγω Κανονισμών, που προβλέπονται από την οδηγία, η Ευρωπαϊκή Επιτροπή συγκρότησε ομάδες σύνταξης αποτελούμενες από εξειδικευμένα στελέχη που προτάθηκαν από αρμόδιους ανά χώρα φορείς. Οι Κανονισμοί Εφαρμογής αφορούν τις παρακάτω κατηγορίες:

Πίνακας 3-3: Κατηγορίες Κανονισμών Εφαρμογής.

Κατηγορίες Κανονισμών Εφαρμογής
Μεταδεδομένα
Δικτυακές Υπηρεσίες
Προδιαγραφές διαλειτουργικότητας δεδομένων
Κοινοχρησία συνόλων χωρικών δεδομένων και υπηρεσιών για αυτά
Παρακολούθηση και Υποβολή Εκθέσεων

Πηγή: Σαριφίδης, 2011

3.4.1. Ευρωπαϊκός Κανονισμός 1205/2008 για τα Μεταδεδομένα

Τα μεταδεδομένα είναι πληροφορίες για τα δεδομένα, οι οποίες περιγράφουν σύνολα και υπηρεσίες γεωχωρικών δεδομένων και καθιστούν δυνατή την εξεύρεση,

την καταγραφή και τη χρήση τους. Επιπροσθέτως, τα μεταδεδομένα για ένα γεωχωρικό σύνολο δίνουν πληροφορίες για το πότε δημιουργήθηκε, ποιος είναι ο δημιουργός του, ποιες προδιαγραφές πιθανόν ακολουθήθηκαν για τη δημιουργία του και άλλες πληροφορίες. (<http://www.inspire.okxe.gr>).

Ο Κανονισμός (ΕΚ) 1205/2008 αποτελεί Ευρωπαϊκό Κανονισμό για την εφαρμογή της οδηγίας 2007/2/ΕΚ INSPIRE του Ευρωπαϊκού Κοινοβουλίου και του Συμβουλίου για τα μεταδεδομένα. Συνεπώς, στον εν λόγω κανονισμό καθορίζονται οι απαιτήσεις για τη δημιουργία και τη τήρηση μεταδεδομένων σχετικά με σύνολα χωρικών δεδομένων⁶, σειρές συνόλων χωρικών δεδομένων⁷ και υπηρεσίες χωρικών δεδομένων, σύμφωνα με τα θέματα που απαριθμούνται στα παραρτήματα Ι, ΙΙ και ΙΙΙ στην οδηγία 2007/2/ΕΚ. (πίνακας 3-2). (ΕΚ 1205/2008-άρθρο 1)

Ο ΕΚ 1205/2008 αποτελείται από 4 θεματικές ενότητες. Στην πρώτη ενότητα (μέρος Α) δίνονται κάποιοι ορισμοί που θα χρησιμοποιηθούν στις επόμενες ενότητες. Στη δεύτερη ενότητα (μέρος Β) περιγράφονται τα στοιχεία που πρέπει να αποτελούν τα μεταδεδομένα των χωρικών δεδομένων. Στην τρίτη ενότητα (μέρος Γ) δίνονται υποδείξεις σχετικά με την πολλαπλότητα και τους όρους των στοιχείων μεταδεδομένων⁸. Τέλος, στην τελευταία ενότητα (μέρος Δ) δίνεται η ταξινόμηση των θεματικών επιπέδων των χωρικών δεδομένων της οδηγίας 2007/2/ΕΚ (πίνακας 3-2) σύμφωνα με το πρότυπο ISO 19115. Επιπροσθέτως, αναγράφεται η κατηγοριοποίηση των υπηρεσιών των χωρικών δεδομένων, καθώς και το πεδίο τιμών που πρέπει να επιστρέφει κάθε υπηρεσία.

3.4.2. Νόμος 3882/2010

Ο Νόμος 3882/2010 (ΦΕΚ166/Α'/22-09-2010) «Εθνική Υποδομή Γεωχωρικών Πληροφοριών (ΕΥΓΕΠ) – Εναρμόνιση με την οδηγία 2007/2/ΕΚ του Ευρωπαϊκού Κοινοβουλίου και του Συμβουλίου της 14ης Μαρτίου 2007 και άλλες διατάξεις» έχει στόχο την εξασφάλιση της ισότιμης πρόσβασης στα γεωχωρικά δεδομένα και τις υπηρεσίες για όλους τους πολίτες και τη Δημόσια Διοίκηση.

Σκοπός του Νόμου είναι αφενός η ανάπτυξη και η λειτουργία της Εθνικής Υποδομής Γεωχωρικών Πληροφοριών και αφετέρου η θέσπιση γενικών αρχών, κανόνων, μέτρων και διαδικασιών σε διοικητικό, νομικό και τεχνολογικό επίπεδο για την οργάνωση ενιαίων πρακτικών διαχείρισης, διάθεσης και κοινοχρησίας γεωχωρικών

⁶ Σύνολα χωρικών δεδομένων: μια αναγνωρίσιμη συλλογή από γεωγραφικά δεδομένα (π.χ το σύνολο των γεωγραφικών δεδομένων ενός φύλλου χάρτη 1:5000 της ΓΥΣ).

⁷ Σειρές συνόλων χωρικών δεδομένων: αποτελούνται από περισσότερα από ένα σύνολα δεδομένων που μοιράζονται την ίδια προδιαγραφή προϊόντος (π.χ η σειρά φύλλων 1:5000 της ΓΥΣ).

⁸ Στοιχεία μεταδεδομένων: διακριτή μονάδα μεταδεδομένων, σύμφωνα με το πρότυπο EN ISO 19115.

πληροφοριών. Για την υλοποίηση των παραπάνω, ο νόμος προβλέπει πως κάθε δημόσια αρχή ή τρίτος που εντάσσεται στο ΕΥΓΕΠ, οφείλει, μέσω του ΚΟΣΕ (Κομβικό Σημείο Επαφής) που έχει ορίσει, να φροντίσει για τη συστηματική καταγραφή, μεταξύ άλλων, των μεταδεδομένων των γεωχωρικών δεδομένων και υπηρεσιών που διαθέτει.

3.4.3. Πρότυπο μεταδεδομένων ISO 19115

Για την πλήρη τεκμηρίωση των γεωχωρικών δεδομένων μπορούν να καταγραφούν πλήθος στοιχείων που το αφορούν. Ένα μεγάλο μέρος αυτών των στοιχείων μεταδεδομένων γεωχωρικής πληροφορίας έχει προτυποποιηθεί από τον Διεθνή Οργανισμό Προτυποποίησης (International Organization for Standardization – ISO). Η σειρά ISO 19100 είναι μια σειρά προτύπων που αφορούν γεωγραφική πληροφορία. Το πρότυπο ISO 19115 είναι ένα μέρος αυτής της σειράς, το οποίο ορίζει πάνω από 300 στοιχεία μεταδεδομένων (metadata elements) για την περιγραφή της γεωγραφικής πληροφορίας. Το ISO 19115 ανήκει στην οικογένεια των προτύπων, τα οποία συντάσσονται και προωθούνται από την τεχνική επιτροπή (Technical Committee) ISO TC 211 και περιλαμβάνει ένα πλήρες λεξικό δεδομένων για μεταδεδομένα, ενώ συγχρόνως καθορίζει τα απαραίτητα στοιχεία μεταδεδομένων που πρέπει να καταγράφονται για όλες τις ομάδες δεδομένων. Περιγράφει, επίσης, τη μεθοδολογία για την επέκταση του λεξικού δεδομένων για τα μεταδεδομένα, καθώς και τις ειδικές απαιτήσεις που υπάρχουν για τη διαχείρισή τους.

4. ΣΥΣΤΗΜΑΤΑ ΑΝΑΠΤΥΞΗΣ WEBGIS

Ένα σύστημα ανάπτυξης WebGIS αποτελεί το συνολικό πακέτο εργαλείων, λογισμικών, εξυπηρετητών Διαδικτύου και διαδικτυακών συστημάτων διαχείρισης περιεχομένου, τα οποία χρησιμοποιούνται για την υλοποίηση, επεξεργασία και ενημέρωση ενός διαδικτυακού γεωχωρικού συστήματος πληροφοριών.

4.1. Γεωχωρικά Εργαλεία και Ελεύθερα Λογισμικά

Τα γεωχωρικά εργαλεία και τα λογισμικά που υπάρχουν για την ανάπτυξη WebGIS είναι αρκετά, ωστόσο για τις ανάγκες της διπλωματικής εργασίας θα γίνει μνεία σε μερικά από αυτά.

4.1.1. ΣΔΒΔ και PostgreSQL/PostGIS

Οι γεωγραφικές βάσεις δεδομένων βασίζονται στην ύπαρξη ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ) που θα έχει δυνατότητες υποστήριξης χωρικών τύπων (γεωμετριών), όπως σημεία, γραμμές και επιφάνειες. Ένα ΣΔΒΔ μπορεί να υποστηρίζει χωρικούς τύπους αυτόνομα ή να βασίζεται σε μία χωρική επέκταση για αυτό το σκοπό. Οι γεωγραφικές βάσεις δεδομένων μπορεί να περιέχουν διανυσματικά (vector) ή ψηφιδωτά (raster) δεδομένα.

Η υποστήριξη χωρικών τύπων από τις βάσεις δεδομένων βασίζεται στην ύπαρξη του τύπου δεδομένων geometry, ο οποίος χρησιμοποιείται για να αναπαραστήσει μια ποικιλία χωρικών δεδομένων. Η ύπαρξη χωρικής υποστήριξης σημαίνει ότι το ΣΔΒΔ πρέπει να υποστηρίζει διάφορα συστήματα προβολής και μετασχηματισμούς μεταξύ τους, χωρικούς τελεστές (π.χ. το αντικείμενο Α επικαλύπτεται με το αντικείμενο Β), συναρτήσεις (π.χ. υπολογισμός του εμβαδού μίας επιφανειακής οντότητας) και χωρικά ευρετήρια. Όταν δημιουργείται μία Χωρική Βάση Δεδομένων, κάθε οντότητα με χωρική υπόσταση (π.χ. ένα σημείο ενδιαφέροντος) είναι μία εγγραφή σε έναν πίνακα που περιέχει όλα τα αντίστοιχα σημεία, ενώ μεταξύ των πεδίων του πίνακα υφίσταται και ένα ξεχωριστό πεδίο για τη γεωμετρία του αντικειμένου (π.χ. τη θέση του σημείου ενδιαφέροντος, η οποία δίνεται από τις συντεταγμένες του).

➤ PostgreSQL

Η PostgreSQL είναι ένα αντικείμενο-σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων (objectrelational database management system - ORDBMS). Είναι ελεύθερο και ανοικτού κώδικα λογισμικό και δεν αναπτύσσεται από μία μόνο εταιρεία, αλλά από μία παγκόσμια κοινότητα χρηστών, εταιρειών και ιδρυμάτων. Η διαχείριση γίνεται κυρίως μέσω του PgAdmin III, αλλά και άλλες εφαρμογές τρίτων: (PgAccess, PhpPgAdmin, WinSQL).

Η άδεια χρήσης της PostgreSQL είναι η «PostgreSQL license». Αυτή αποτελεί “ελεύθερη” ανοικτού λογισμικού (open source) άδεια, αναγνωρισμένη από την OSI. Κύριο χαρακτηριστικό της άδειας αυτής είναι ότι επιτρέπει στους χρήστες να κάνουν ό,τι θέλουν με τον κώδικα, αλλά και να μεταπωλούν το εκτελέσιμο προϊόν χωρίς τον πηγαίο κώδικα (source code). Αυτή η πιστοποιημένη κατά OSI άδεια χρήσης αξιολογείται ευρέως ως ευέλικτη και φιλική προς τις επιχειρήσεις, καθώς δεν περιορίζει τη χρήση της PostgreSQL σε εμπορικές και αποκλειστικές εφαρμογές. Μαζί με την υποστήριξη πολλών εταιριών και τη δημόσια ιδιοκτησία του κώδικα, η άδεια καθιστά την PostgreSQL πολύ δημοφιλή στους προμηθευτές, οι οποίοι επιθυμούν να ενσωματώσουν μια βάση δεδομένων στο δικό τους προϊόν, χωρίς να φοβούνται τις χρεώσεις ή τις αλλαγές στους όρους της Άδειας Χρήσης. Εκατοντάδες εταιρείες και οργανισμοί χρησιμοποιούν την PostgreSQL για να κατασκευάσουν προϊόντα, ιστοσελίδες και εργαλεία σε διάφορους τομείς, όπως στο ηλεκτρονικό εμπόριο, την υγεία, τα μέσα ενημέρωσης, την εκπαίδευση, τη βιομηχανία, την τεχνολογία και τις τηλεπικοινωνίες. Ο μοναδικός περιορισμός της είναι ότι δεν μπορεί να καταστεί νομικά υπεύθυνη για προβλήματα που αφορούν το λογισμικό.

➤ PostGIS

Η PostgreSQL υποστηρίζει τη συνεργασία επιπρόσθετων λογισμικών (addons), κάθε ένα από τα οποία επιτρέπει την εκτέλεση συγκεκριμένων και εξειδικευμένων λειτουργιών. Το ευρέως διαδεδομένο από αυτά τα λογισμικά είναι η επέκταση γεωγραφικών λειτουργιών PostGIS. Η τελευταία, δίνει τη δυνατότητα στην PostgreSQL να υποστηρίζει χωρικά δεδομένα, σύμφωνα με το πρότυπο του OGC. 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 Quantum GIS.

Ο τύπος Geometry του PostGIS περιέχει όλη τη γεωμετρία (τις συντεταγμένες) του κάθε αντικειμένου. Ο τύπος αυτός είναι γενικός και μπορεί να περιέχει χωρικά αντικείμενα των τριών συνηθέστερων κατηγοριών (σημείο, γραμμή, πολύγωνο), καθώς και συλλογές αυτών των αντικειμένων (δηλαδή πολλά πολύγωνα ή πολλά σημεία μαζί με κάποιες γραμμές κ.ο.κ.). Η αναπαράσταση του τύπου Geometry σε μορφή πίνακα δεν μπορεί να γίνει εύκολα αντιληπτή. Στην πραγματικότητα, κάτι τέτοιο είναι σχεδόν αδύνατο να συμβεί, από τη στιγμή που οι γεωμετρίες αναπαρίστανται από δεκαεξαδικές μορφές, όπως η παρακάτω:

```
"01060000203408000001000000010300000001000000070000008BC3E53F90901C416C
1C7573A103504109A69A955B901C412032DD7CB70350418B00E19E04911C419FDD48A
3B90350419B060D012B911C4166D1451AAB035041FC5482022B911C41178E9219AB035
04173B28BD23E911C41AA125FB4A30350418BC3E53F90901C416C1C7573A1035041"
```

Για αυτό το λόγο, το Open GIS Consortium προτείνει τη χρήση του μορφότυπου WELL-KNOWN TEXT (WKT), ο οποίος υποστηρίζεται από το PostGIS. Με το WKT μπορούν να αναπαρασταθούν με έναν εύκολα αντιληπτό τρόπο σημεία, γραμμές, πολύγωνα καθώς και πολλαπλές γεωμετρίες (π.χ. πολυ-πολύγωνα) και συλλογές γεωμετριών. Το PostGIS παρέχει συναρτήσεις που μετατρέπουν το WKT (που μπορεί να διαβαστεί και να γίνει κατανοητό σχετικά εύκολα) σε γεωμετρίες της παραπάνω μορφής (που δε μπορούν να είναι απευθείας κατανοητές από τους ανθρώπους) και αντίστροφα. Μάλιστα, γεωμετρίες που έχουν συναχθεί σύμφωνα με το WKT format μπορούν να δημιουργηθούν / αναπαρασταθούν σε οποιοδήποτε σύστημα είναι συμβατό με τις κατευθύνσεις που δίνονται από το OGC (π.χ. Oracle, DB2 κ.α.). Παρακάτω παρατίθενται μερικά παραδείγματα του WKT format:

Σημείο: 'POINT(X Y)'

Γραμμή: 'LINESTRING(X1 Y1, X2 Y2, ...)'

Πολύγωνο: 'POLYGON((X1 Y1, X2 Y2, ..., X1 Y1), (Xn,Yn, Xn+1 Yn+1,..., Xn,Yn))'

Εκτός από τους παραπάνω απλούς τύπους, υπάρχουν και οι πολλαπλοί τύποι MULTIPOINT, MULTILINESTRING, MULTIPOLYGON, που είναι απλές γενικεύσεις των προηγούμενων απλών τύπων (π.χ. ο MULTIPOINT περιέχει πολλά σημεία).

Το PostGIS υποστηρίζει πλήθος Γεωγραφικών και Προβολικών συστημάτων συντεταγμένων, κάθε ένα από τα οποία περιγράφεται από ένα SRID (Spatial Reference Identifier). Το SRID είναι ένας ακέραιος αριθμός (η ταυτότητα) που αντιστοιχεί στο σύστημα συντεταγμένων, στο οποίο δίνονται οι συντεταγμένες του συγκεκριμένου αντικειμένου. Πιο αναλυτικά, τα συστήματα συντεταγμένων που υποστηρίζονται από το PostGIS μπορούν να βρεθούν μέσα στον πίνακα spatial_ref_sys μίας οποιαδήποτε βάσης δεδομένων του PostGIS.

Πίνακας 4-1: SRID για διάφορα συστήματα συντεταγμένων.

	Σύστημα Συντεταγμένων	Τύπος	Μονάδες μέτρησης	SRID
1	Ελληνικό Γεωδαιτικό Σύστημα Αναφοράς 1987 (ΕΓΣΑ 87)	Προβολικό	Μέτρα	2100
2	UTM Ζώνη 34 επί του WGS84	Προβολικό	Μέτρα	32634
3	UTM Ζώνη 35 επί του WGS84	Προβολικό	Μέτρα	32635
4	WGS84	Γεωγραφικό	Μοίρες	4326
5	Google (Mercator Spheric)	Γεωγραφικό	Μέτρα	900913

Πηγή: Φρέντζος, 2009-2010

Πρέπει να επισημανθεί ότι στις τελευταίες εκδόσεις του PostGIS έχουν ενσωματωθεί συναρτήσεις για δυνατότητες εισαγωγής και χωρικής επεξεργασίας κανονικοποιημένων δεδομένων (raster).

Η άδεια χρήσης της PostGIS είναι η «GNU General Public License version 2 (GPLv2)», η οποία δημιουργήθηκε το 1991 και αποτελεί τροποποίηση της άδειας GPL. Η τελευταία, δημοσιεύθηκε το 1987 και δίνει στους κατόχους ενός προγράμματος τα ακόλουθα τέσσερα δικαιώματα, που στην κοινότητα του ελεύθερου λογισμικού είναι γνωστά και ως Τέσσερις Ελευθερίες:

- να μπορούν να εκτελούν ένα πρόγραμμα για οποιονδήποτε λόγο.
- να μπορούν να μελετούν τη λειτουργία ενός προγράμματος και να το τροποποιούν κατά την κρίση τους.
- να μπορούν να διανείμουν αντίγραφα του προγράμματος.
- να μπορούν να βελτιώσουν το πρόγραμμα και να προσφέρουν τις προσθήκες τους στην κοινότητα, έτσι ώστε να ωφεληθεί ολόκληρη η κοινότητα.

Πρέπει να σημειωθεί ότι παράγωγο και προϊόν ενός GPL προγράμματος υποχρεούται να κυκλοφορεί κι αυτό υπό την ίδια άδεια. Η κύρια τροποποίηση της άδειας GPLv2 σε σχέση με την GPL είναι η ενότητα 7. Η ενότητα αυτή προβλέπει ότι, εάν έχουν επιβληθεί σε κάποιον περιορισμοί που τον αποτρέπουν από τη διανομή ενός λογισμικού, το οποίο είναι κατοχυρωμένο με άδεια GPL, τότε αυτός δεν μπορεί να το διανέμει καθόλου.

(Φρέντζος, 2009-2010)

4.1.2. OpenLayers

Η OpenLayers είναι ουσιαστικά μια αντικειμενοστραφής βιβλιοθήκη Javascript ή ένα JavaScript/Ajax framework για την απεικόνιση χαρτογραφικών δεδομένων στους σύγχρονους φυλλομετρητές. Δημιουργήθηκε το 2005 από την MetaCarta και το 2007 υιοθετήθηκε από την OSGeo. Λειτουργεί ανεξάρτητα από τον εξυπηρετητή και επιτρέπει στον προγραμματιστή να μεταχειρίζεται έτοιμες λύσεις για τη σύνθεση λογισμικού Web GIS.

Ως framework αποσκοπεί στο διαχωρισμό των χαρτογραφικών εργαλείων από τα δεδομένα, έτσι ώστε όλα τα εργαλεία να λειτουργούν επί του συνόλου των δεδομένων σύμφωνα με σαφώς ορισμένα πρότυπα (όπως π.χ. τα WMS, WFS και WCS). Μολονότι μοιράζεται ομοιότητες με τις πλατφόρμες των Google και Microsoft, έχει το πλεονέκτημα πως αναπτύχθηκε από την κοινότητα των χρηστών και είναι ανοικτό και ελεύθερο⁹.

⁹ Ελεύθερο: ελεύθερο θεωρείται ένα λογισμικό, του οποίου η σχετική άδεια εξασφαλίζει τις παρακάτω προϋποθέσεις:
- ελευθερία ως προς τη χρήση του προγράμματος για οποιοδήποτε σκοπό.



Εικόνα 4-1:OpenLayers.

➤ Βασικά αντικείμενα της OpenLayers

Τα δύο βασικά αντικείμενα της συγκεκριμένης βιβλιοθήκης είναι ο χάρτης (map) και το θεματικό επίπεδο (Layer). Το αντικείμενο map δημιουργείται μία φορά και ενσωματώνεται σε στοιχείο της HTML με βάση τον κωδικό του (ID). Σε ένα στοιχείο “map” αποθηκεύονται οι βασικές πληροφορίες που χαρακτηρίζουν ένα χάρτη, όπως σύστημα αναφοράς, όρια περιοχής, μονάδες κ.α. Μέσα στο χάρτη, τα δεδομένα προβάλλονται μέσω θεματικών επιπέδων, η δημιουργία των οποίων έγκειται στο αντικείμενο Layer της Openlayers, το οποίο καθορίζει και το αντίστοιχο πρότυπο γεωχωρικών υπηρεσιών Διαδικτύου.

```
var map = new OpenLayers.Map('map');
var kthmatologio=new OpenLayers.Layer.WMS(
"ΚΤΗΜΑΤΟΛΟΓΙΟ",
"http://gis.ktimanet.gr/wms/wmsopen/wmsserver.aspx",
{layers: "KTBASEMAP", format: "image/png", transparent:
true},{isBaseLayer:true,numZoomLevels: 20,projection:new
OpenLayers.Projection("EPSG:900913")})
map.addLayer(kthmatologio);
```

Τα θεματικά επίπεδα χωρίζονται σε βασικά και μη βασικά. Τα βασικά θεματικά επίπεδα (Baselayer) είναι αμοιβαίως αποκλειόμενα layers και, συνεπώς, μόνο ένα μπορεί να είναι ενεργό κάθε φορά. Επιπροσθέτως, το ενεργοποιημένο Baselayer καθορίζει το σύστημα συντεταγμένων και τα επίπεδα zoom του χάρτη. Συνήθως, ως βασικά θεματικά επίπεδα χρησιμοποιούνται τα εμπορικά θεματικά επίπεδα Google, Bing και Yahoo. Τα μη βασικά θεματικά επίπεδα (overlays) μπορούν να

- ελευθερία ως προς τη μελέτη του τρόπου λειτουργίας του προγράμματος και την προσαρμογή του στις απαιτήσεις του χρήστη.
- η ελευθερία ως προς την αναπαραγωγή και τη διανομή.
- η ελευθερία ως προς τη δυνατότητα βελτίωσης του προγράμματος και τη διάθεση των βελτιώσεων στο κοινό προς όφελος όλων.

ενεργοποιούνται ταυτόχρονα πολλά μαζί. Επιπλέον, υπάρχει δυνατότητα ρύθμισής τους, ώστε να ενεργοποιούνται ή να απενεργοποιούνται ανάλογα με το zoom.

➤ Πηγές

Οι πηγές πληροφοριών που δύναται να φορτωθούν σε ένα θεματικό επίπεδο είναι αρκετές και αναγράφονται στην εικόνα 4-2

• OGC WMS	• Google Maps
• OGC WFS	• MSN Virtual Earth
• GeoRSS	• Yahoo! Maps
• CSV	• Multimap
• ka-Map	• GML
• Nasa WorldWind	• KML
• Canvas	

Εικόνα 4-2: Διαθέσιμες πηγές πληροφοριών της Openlayers

➤ Αντικείμενο control

Ένα ακόμα σημαντικό αντικείμενο που διαθέτει η Openlayers είναι το control, μέσω του οποίου γίνεται δυνατή η επικοινωνία του χρήστη με το χάρτη. Η επικοινωνία αυτή μπορεί να πραγματοποιηθεί είτε με έτοιμα χειριστήρια (controls) που διαθέτει η Openlayers είτε με χειριστήρια που δημιουργούνται από τον προγραμματιστή για τις ανάγκες της εφαρμογής.

➤ Άδεια χρήσης της OpenLayers

Η άδεια χρήσης της OpenLayers είναι η «Freebsb» ή «2-clause BSD License», η οποία αποτελεί την πιο απλοποιημένη έκδοση της άδειας BSD. Το Ίδρυμα Ελεύθερου Λογισμικού - Free Software Foundation (FSF) FreeBSD τη χαρακτήρισε ως συμβατή με την άδεια GNU GPL.

(Κοντόπουλος, 2010)

4.1.3. ExtJS/GeoExt/GXP

➤ ExtJS

Το Extjs είναι ένα JavaScript Framework που εκτελείται στην πλευρά του πελάτη. Υποστηρίζει την τεχνολογία Ajax και δημιουργήθηκε για την ανάπτυξη διαδικτυακών εφαρμογών. Για να είναι δυνατή η εκτέλεση ενός σεναρίου γραμμένου σε ExtJS, αρκεί αφενός να διαθέτει το πρόγραμμα περιήγησης ερμηνευτή JavaScript και αφετέρου να ενσωματωθούν στην ιστοσελίδα οι παρακάτω βιβλιοθήκες JavaScript.

```
<script type="text/JavaScript" src="../../../ext-base.js"></script>
<script type="text/JavaScript" src="../../../ext-all.js"></script>
```

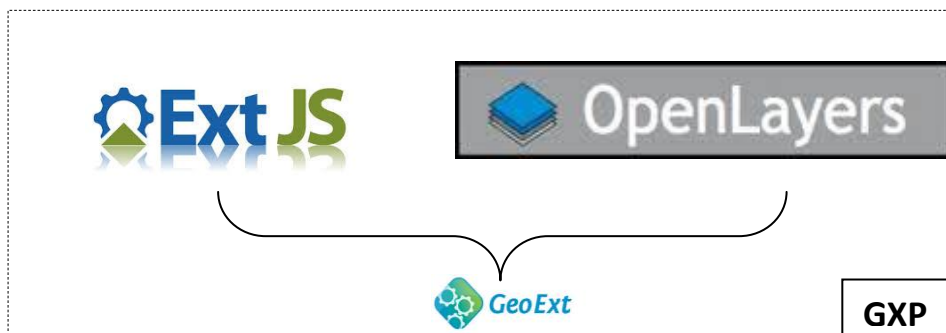
Το Extjs μέσω της τεχνολογίας Ajax μπορεί να στείλει POST αιτήματα σε οποιοδήποτε server και, επομένως, να πραγματοποιείται επικαιροποίηση του περιβάλλοντος χωρίς την ανάγκη ανανέωσης της ιστοσελίδας.

- Άδεια χρήσης ExtJS

Το ExtJS παρέχεται με δύο ειδών άδειες. Η μία είναι η εμπορική και είναι κατάλληλη αν ο χρήστης επιθυμεί να δημιουργήσει εμπορική, εφαρμογή όπου θέλει να κρατήσει τον κώδικα (ιδιόκτητη εφαρμογή). Η δεύτερη, είναι η ανοικτού κώδικα άδεια και είναι συμβατή με την GNU GPL license v3. Η δεύτερη άδεια είναι κατάλληλη για δημιουργία εφαρμογών ανοικτού κώδικα.

➤ GeoEXT

Το GeoExt αποτελεί μια ταχέως αναπτυσσόμενη βιβλιοθήκη JavaScript, η οποία βασίζεται στο Framework Extjs και τη βιβλιοθήκη OpenLayers. Μέσω του GeoExt γίνεται δυνατή η δημιουργία εφαρμογών WebGIS σε εντυπωσιακό διαδραστικό περιβάλλον και η ενσωμάτωση όλων των δυνατοτήτων που υποστηρίζει τόσο η OpenLayers όσο και το ExtJS. Η άδεια χρήσης του GeoEXT είναι η BSD license.



Εικόνα 4-3: WebGIS εφαρμογή με χρήση βιβλιοθήκης GeoExt.

➤ GXP

Το GXP είναι το επόμενο βήμα στη δημιουργία εφαρμογών WebGIS, το οποίο συνδυάζει όλες τις παραπάνω τεχνολογίες (ExtJS, GeoEXT, OpenLayers). Το GXP αποτελεί client side βιβλιοθήκη JavaScript που δημιουργήθηκε με σκοπό την υλοποίηση, με λίγες γραμμές κώδικα (υψηλού επιπέδου προγραμματισμός), μιας ολοκληρωμένης εφαρμογής διαδικτυακής χαρτογράφησης. Μερικές από τις δυνατότητες που περιέχει είναι τα έτοιμα εργαλεία προσθήκης / αφαίρεσης

θεματικών επιπέδων, η μέτρηση εμβαδών / αποστάσεων, η επεξεργασία θεματικών επιπέδων, τα χωρικά ερωτήματα σε θεματικά επίπεδα κ.α. Η άδεια χρήσης του GXP είναι η GPLv3, η οποία δημοσιεύθηκε το 2007.

4.1.4. QuantumGIS

Το QGIS είναι μία ελεύθερη GIS εφαρμογή ανοικτού κώδικα που χρησιμοποιείται για τη δημιουργία, τροποποίηση, ανάλυση και παρουσίαση των γεωγραφικών δεδομένων. Η ανάπτυξη του λογισμικού ξεκίνησε από τον Gary Sherman το 2002 και σύντομα, γύρω από αυτό, αναπτύχθηκε το Open Source Geospatial Foundation. Το πρόγραμμα συντηρείται από μία ενεργή κοινότητα χρηστών και προγραμματιστών, που σε τακτά διαστήματα κυκλοφορούν νέες εκδόσεις, επεκτείνοντας τις δυνατότητες των παλιών και συνάμα διορθώνοντας σφάλματα. Στο περιβάλλον λειτουργίας του QGIS υπάρχει μια περιοχή εμφάνισης των χωρικών δεδομένων (map view) και μια περιοχή διαχείρισης του υπομνήματος, στην οποία παρατίθενται τα ονόματα και τα σύμβολα των δεδομένων, περίπου με τον ίδιο τρόπο που αυτά παρουσιάζονται στο ArcGIS. Επίσης, περιέχονται διάφορα εργαλεία και μενού επιλογών. Το βασικό προτέρημα του QGIS είναι ότι μπορεί να εισάγει, οπτικοποιήσει, διαχειριστεί, τροποποιήσει – δημιουργήσει δεδομένα που περιέχονται σε μια βάση δεδομένων PostgreSQL, σύμφωνα με το πρότυπο της επέκτασης PostGIS.

Η πρώτη έκδοση του προγράμματος ανέπτυξε κυρίως λειτουργίες απεικόνισης και εισαγωγής δεδομένων στη χωρική βάση δεδομένων PostgreSQL/PostGIS. Σταδιακά ενσωματώθηκαν λειτουργίες υποστήριξης αρχείων shapefiles, όσο και άλλων διανυσματικών format. Στη σημερινή του μορφή το πρόγραμμα προσφέρει αυξημένες δυνατότητες οπτικοποίησης, διαχειρίζεται raster αρχεία, εκτελεί λειτουργίες ανάλυσης (π.χ buffer), ενώ διακρίνεται για την αξιοπιστία και τη σταθερότητά του. Είναι συμβατό με τους περισσότερους εμπορικούς τύπους αρχείων, τη διαδικτυακή εφαρμογή χωρικών δεδομένων MapServer, καθώς και τα πρότυπα WMS και WFS του Open Geospatial Consortium. Σαν λογισμικό ανοικτού κώδικα μπορεί να τροποποιηθεί, έτσι ώστε να εκτελεί διαφορετικές ή εξειδικευμένες λειτουργίες. Η άδεια χρήσης του QGIS είναι η GNU GPL.

(Κολιός, 2009)

4.2. Εξυπηρετητές Χαρτών Ανοικτού Κώδικα

Όπως προαναφέρθηκε και στο κεφάλαιο της αρχιτεκτονικής μιας WebGIS εφαρμογής, ο εξυπηρετητής χαρτών αποτελεί έναν διαμεσολαβητή μεταξύ της χωρικής βάσης δεδομένων και του πελάτη, με σκοπό την απάντηση των αιτημάτων του τελευταίου και την οπτικοποίηση των χωρικών δεδομένων στο Διαδίκτυο, με τη χρήση κατάλληλων γεωχωρικών προτύπων και υπηρεσιών. Από τους διαθέσιμους

εξυπηρετητές χαρτών που υπάρχουν, θα γίνει μνεία στους GeoServer και MapServer, δεδομένου του σκοπού της παρούσας διπλωματικής εργασίας.

4.2.1. GeoServer

Ο GeoServer αποτελεί έναν ανοικτού κώδικα εξυπηρετητή χαρτών, ο οποίος γράφτηκε σε γλώσσα προγραμματισμού Java και επιτρέπει στους χρήστες να μοιράζονται και να τροποποιούν εύκολα χωρικά δεδομένα. Ο GeoServer είναι διαθέσιμος υπό την GPLv2 άδεια. Βασικό πλεονέκτημά του αποτελεί η σχεδίασή του για την υποστήριξη της διαλειτουργικότητας και γι' αυτό δύναται να δημοσιεύει στοιχεία από πληθώρα πηγών χωρικών δεδομένων, με τη χρήση ανοικτών προτύπων.

➤ Πηγές Δεδομένων

Οι πηγές δεδομένων μπορεί να είναι είτε εξωτερικές βάσεις δεδομένων είτε αρχεία από τον τοπικό δίσκο. Κάτωθι θα αναφερθούν μερικές από τις πιο σημαντικές πηγές που μπορεί να διαβάσει ο GeoServer:

Πίνακας 4-2: Πηγές δεδομένων.

ΑΡΧΕΙΑ	ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ
ShapeFile	PostGIS
GeoTIFF	ArcSDE
ArcGrid	Oracle Spatial
JPEG2000	DB2
GDAL formats	SQL Server
JPG και PNG (με το συνοδευτικό world file)	MySql

➤ Υποστηριζόμενα OGC πρωτόκολλα

Με τον GeoServer τα δεδομένα δημοσιεύονται στους χρήστες μέσω του πρωτοκόλλου WMS με ασφαλή τρόπο και με μοναδικό σκοπό τη διαδικτυακή απεικόνιση των δεδομένων σε μορφή εικόνας. Επιπροσθέτως, μέσω του WFS ένας χρήστης μπορεί να κατεβάσει τα διανυσματικά δεδομένα και να τα χρησιμοποιήσει για χαρτοσύνθεση, χωρική ανάλυση και άλλες λειτουργίες. Επίσης, εφόσον ο χρήστης είναι πιστοποιημένος, μπορεί να αλλάξει τα δεδομένα και να τα στείλει πίσω στον εξυπηρετητή (GeoServer) για αποθήκευση, μέσω της υπηρεσίας WFS-T. Τα δεδομένα μπορούν να αποσταλούν μέσω του προτύπου GML (συμπιεσμένα), αλλά και μέσω άλλων δημοφιλών προτύπων, όπως είναι τα shapefile και json. Όσον αφορά τις εικόνες, μέσω του πρωτοκόλλου WCS, ένα λογισμικό GIS μπορεί να ζητήσει από τον GeoServer τα πραγματικά εικονιστικά δεδομένα για τη χρήση σε χωρική ανάλυση. Αυτό επιτρέπει στον χρήστη να δημιουργήσει εφαρμογές, οι

οποίες μοντελοποιούν διαδικασίες που περιγράφουν τα δεδομένα. Τέλος, υποστηρίζει την υπηρεσία WPS με σκοπό την αλυσιδωτή εκτέλεση σύνθετων επεξεργασιών και, τελικώς, την οπτικοποίηση του αποτελέσματος μέσω της υπηρεσίας WMS, ενώ διαθέτει τη δυνατότητα άμεσης αλλαγής συστήματος αναφοράς.

➤ Οργάνωση των δεδομένων

Η αποθήκευση των δεδομένων στον GeoServer γίνεται σε θεματικά επίπεδα (Layers) ή ομάδα θεματικών επιπέδων (Layer Group), τα οποία ανήκουν σε κάποια αποθήκη (store) με βάση την πηγή τους. Τα stores ομαδοποιούνται αναλόγως του σκοπού σε ομάδες εργασίες (workspaces).

- Workspaces

Μια ομάδα εργασίας (Workspace) αποτελεί ένα «θεωρητικό δοχείο» μέσα στο οποίο ομαδοποιούνται χωρικά δεδομένα διαφορετικών πηγών, αλλά με το ίδιο θεματικό περιεχόμενο. Παράδειγμα ομάδων εργασίας αποτελούν οι διάφορες πράξεις εφαρμογής που αφορούν τον αγροτικό χώρο (διανομές, αναδασμοί, απαλλοτριώσεις κ.α). Σημειώνεται, ότι δύο θεματικά επίπεδα (layers), αν ανήκουν σε διαφορετικές ομάδες εργασίας, γίνεται να έχουν το ίδιο όνομα.

- Stores

Η αποθήκη (Store) αναφέρεται σε μια συγκεκριμένη πηγή δεδομένων, η οποία μπορεί να είναι ένα Shapefile, μία βάση δεδομένων ή οποιαδήποτε πηγή υποστηρίζει η έκδοση του GeoServer (πίνακας 4-2). Επιπροσθέτως, στην αποθήκη καθορίζονται στοιχεία που αφορούν την προβολή, το στυλ, το πλαίσιο οριοθέτησης κ.α του θεματικού επιπέδου που συνδέεται. Τέλος, ένα θεματικό επίπεδο πρέπει να συνδέεται μόνο με μία αποθήκη.

- Layers και Layer Group

Ένα θεματικό επίπεδο αποτελεί μια συλλογή γεωχωρικών χαρακτηριστικών ή καλύψεων, το οποίο συνήθως περιέχει ένα γεωμετρικό τύπο κάθε φορά (σημείο, γραμμές, πολύγωνα και εικόνες), καθώς και έναν τύπο περιεχομένου (δρόμοι, σπίτια, σύνορα κ.α). Το θεματικό επίπεδο δημιουργείται βάσει των πηγών δεδομένων που υποστηρίζει ο GeoServer και επιλέγεται μέσω της αποθήκης (store) με την οποία συνδέεται. Το Layer Group αποτελεί μια δυνατότητα του GeoServer, με την οποία διοχετεύει ομάδες θεματικών επιπέδων μέσω μιας υπηρεσίας WMS. Ουσιαστικά, γίνεται ομαδοποίηση πολλών θεματικών επιπέδων σε μια κατηγορία, η οποία πλέον αποτελεί ένα θεματικό επίπεδο. Σημειώνεται, ότι τα Layer Group αφορούν μόνο τα αιτήματα WMS. (OpenGeo, 2012)

Layers











Manage the layers being published by GeoServer

 Add a new resource

 Remove selected resources

<< < 1 2 > >> Results 1 to 10 (out of 18 items)

 Search

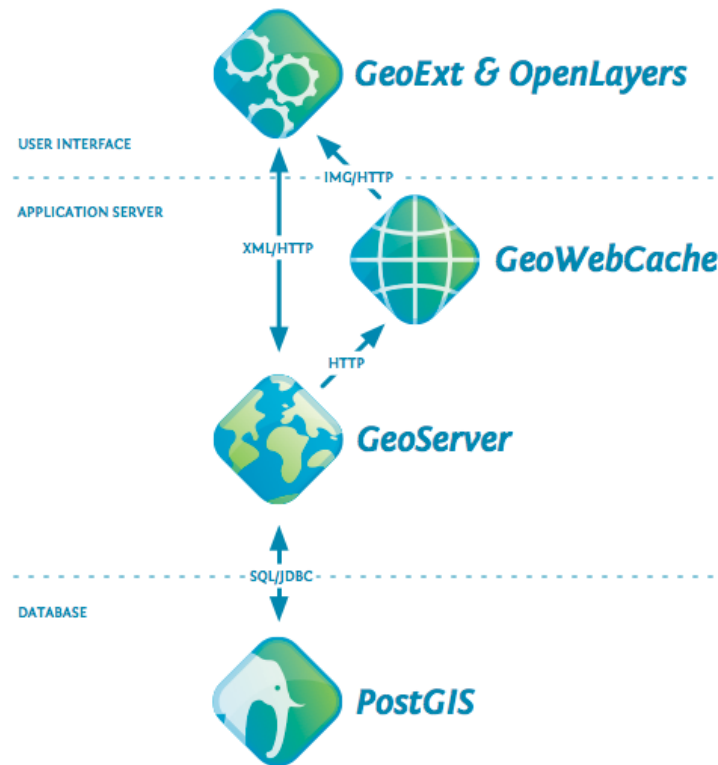
<input type="checkbox"/>	Type	Workspace	Store	Layer Name	Enabled?	Native SRS
<input type="checkbox"/>		nurc	img_sample2	Pk50095	✓	EPSG:32633
<input type="checkbox"/>		nurc	mosaic	mosaic	✓	EPSG:4326
<input checked="" type="checkbox"/>		nurc	worldImageSample	Img_Sample	✓	EPSG:4326
<input type="checkbox"/>		sf	sf	archsites	✓	EPSG:26713
<input type="checkbox"/>		sf	sf	bugsites	✓	EPSG:26713
<input checked="" type="checkbox"/>		sf	sf	restricted	⚠	EPSG:26713
<input type="checkbox"/>		sf	sf	roads	✓	EPSG:26713
<input checked="" type="checkbox"/>		sf	sf	streams	✓	EPSG:26713
<input type="checkbox"/>		sf	sfdem	sfdem	✓	EPSG:26713
<input type="checkbox"/>		tiger	nyc	giant_polygon	✓	EPSG:4326

<< < 1 2 > >> Results 1 to 10 (out of 18 items)

Εικόνα 4-4:Οργάνωση θεματικών επιπέδων στον GeoServer.

➤ GeoWebCache

Ο GeoWebCache είναι ένα εργαλείο, το οποίο βρίσκεται μεταξύ του εξυπηρετητή wms και του πελάτη και έχει ως αποστολή τον έλεγχο των διαφόρων αιτημάτων του τελευταίου. Σε περίπτωση που το αίτημα γίνεται πρώτη φορά, τότε ο GeoServer το διαχειρίζεται και ο GeoWebCache απλώς αποθηκεύει την εικόνα του αιτήματος. Αντίθετα, όταν το αίτημα έχει υποβληθεί ξανά, τότε το τελευταίο δεν καταλήγει στον GeoServer αλλά η απάντηση αποστέλλεται απευθείας από τον GeoWebCache. Το πλεονέκτημα του GeoWebCache είναι η ταχύτητα της απάντησης των αιτημάτων του πελάτη, χωρίς να απαιτείται η άσκοπη δημιουργία χαρτών, οι οποίοι είχαν ήδη δημιουργηθεί από τον GeoServer λόγω παλαιότερων αιτημάτων και είχαν αποθηκευτεί στον GeoWebCache. Σημειώνεται ότι ο GeoWebCache διατίθεται υπό την άδεια GPLv2.



Εικόνα 4-5: Ενσωμάτωση του GeoWebCache στην αρχιτεκτονική ενός WebGIS.

Πηγή: <http://boundlessgeo.com/whitepaper/opengeo-architecture/>

4.2.2. MapServer

Ο MapServer είναι μια ανοικτού κώδικα πλατφόρμα, η οποία δημοσιεύει τα χωρικά δεδομένα στο Διαδίκτυο. Γραμμένος σε C, η ανάπτυξή του ξεκίνησε στα μέσα της δεκαετίας του 1990 στο Πανεπιστήμιο της Μινεσότα (UMN), σε συνεργασία με τη NASA και το γραφείο φυσικών πόρων της Μινεσότα. Λειτουργεί σε όλες τις κύριες πλατφόρμες (Windows, Linux, Mac OS/X), ενώ τώρα είναι ένα έργο της OSGeo. Η διαχείριση, υποστήριξη και εξέλιξή του γίνεται από έναν αυξανόμενο αριθμό προγραμματιστών (σχεδόν 20) από όλο τον κόσμο και καθοδηγείται από την MapServer PSC (Project Steering Committee) υπό την εποπτεία της OSGeo. Η άδεια χρήσης του MapServer είναι η «MIT license». Η τελευταία, είναι μία από τις πιο φιλελεύθερες άδειες που υπάρχουν. Δημιουργήθηκε από το Τεχνολογικό Ινστιτούτο της Μασαχουσέτης (MIT) και είναι συμβατή με το GNU GPL. Με τη χρήση της, μπορεί κάποιος να αντιγράψει, τροποποιήσει, συγχωνεύσει, εκδώσει, διανείμει, εκμισθώσει και να πουλήσει μέρος του λογισμικού, χωρίς να φέρει οποιαδήποτε ευθύνη για οτιδήποτε.

➤ Format δεδομένων

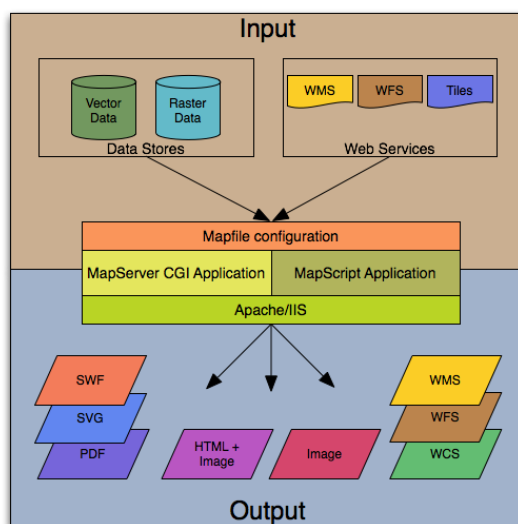
Τα δεδομένα που υποστηρίζει ο MapServer βασίζονται στις βιβλιοθήκες GDAL¹⁰ και OGR. Η πρώτη, χρησιμοποιείται για τα κανονικοποιημένα δεδομένα (raster), ενώ η

¹⁰ GDAL: η GDAL (Geospatial Data Abstraction Library) αποτελεί ανοικτού κώδικα βιβλιοθήκη μέσω της οποίας δύναται να διαβαστούν και να τροποποιηθούν ποικίλες μορφές κανονικοποιημένων

δεύτερη αποτελεί κομμάτι της πρώτης και χρησιμοποιείται για τα διανυσματικά δεδομένα. Συνεπώς, οι μορφές των δεδομένων που υποστηρίζει ο MapServer εξαρτώνται από τις ανωτέρω βιβλιοθήκες.

➤ Οργάνωση και λειτουργία του MapServer

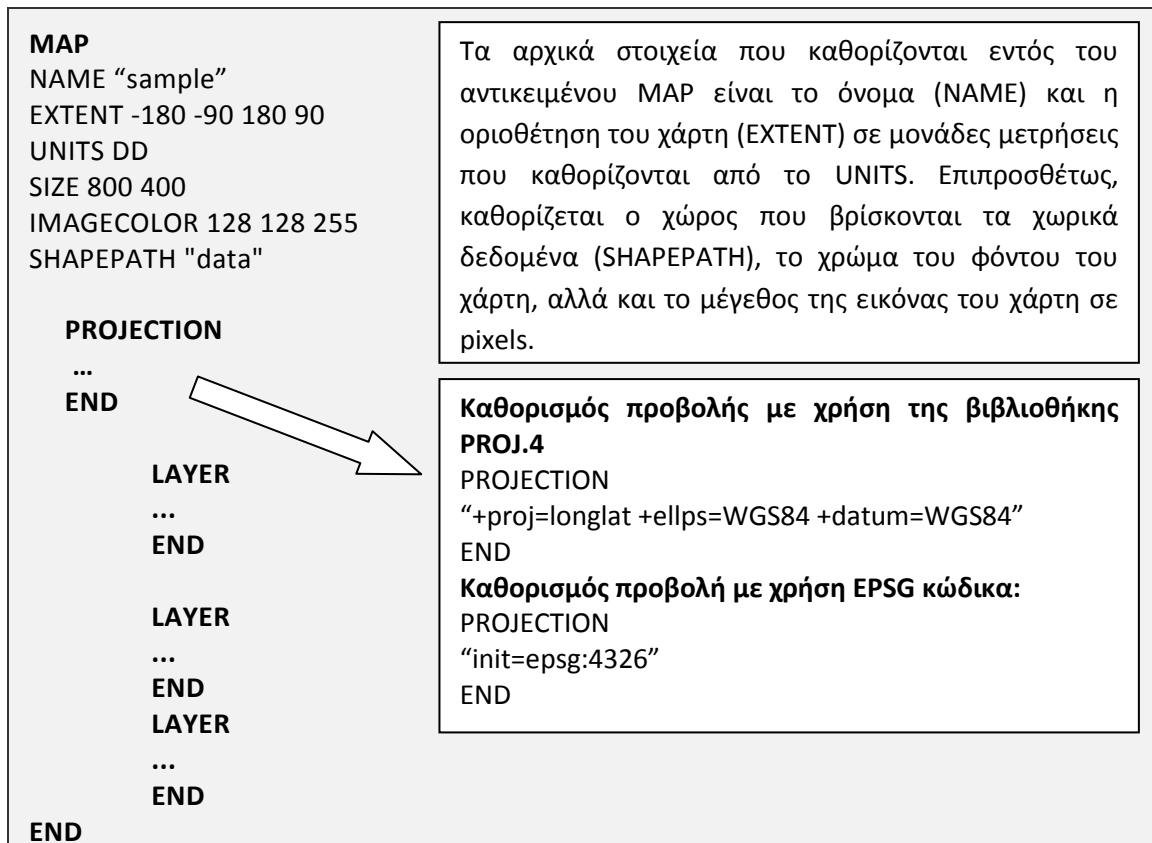
Το βασικό αρχείο στο οποίο βασίζεται η λειτουργία του MapServer είναι το mapfile και έχει προέκταση map. Το mapfile είναι ASCII αρχείο κειμένου που αποτελείται από ένα σύνολο αντικειμένων, τα οποία, με βάση κάποιες παραμέτρους, καθορίζουν το χαρτογραφικό υπόβαθρο. Συγκεκριμένα, μέσω των αντικειμένων προσδιορίζονται τα όρια του χάρτη, η περιοχή εύρεσης των δεδομένων, το μέρος όπου βρίσκεται η εξαγόμενη τελική εικόνα, τα θεματικά επίπεδα, καθώς και στοιχεία που αφορούν τον προβολή και το συμβολισμό των τελευταίων.



Εικόνα 4-6: Δομή λειτουργίας του MapServer.

Σημειώνεται πως όλα τα επιμέρους αντικείμενα που αναφέρθηκαν παραπάνω καθορίζονται εντός του βασικού αντικειμένου του mapfile, που είναι το MAP. Το εν λόγω αντικείμενο ξεκινάει με το όρισμα MAP και τελειώνει με το END.

δεδομένων (raster). Η συγκεκριμένη βιβλιοθήκη διατηρείται από τον Frank Warmerdam και μιας ομάδας προγραμματιστών της κοινότητας Open Source Geospatial Foundation. Τέλος, πρέπει να σημειωθεί ότι χρησιμοποιείται από πολλά άλλα έργα, όπως το Google Earth, το ArcGIS 9, το FME, το GRASS το Quantum GIS κ.α.



Κάθε αντικείμενο LAYER περιλαμβάνει τα χωρικά δεδομένα συγκεκριμένου γεωμετρικού τύπου (raster, polygon, line, point) που εισάγονται στο συγκεκριμένο χάρτη. Από την έκδοση MapServer 5.0 και μετά υπάρχει δυνατότητα καταχώρισης απεριόριστου αριθμού θεματικών επιπέδων σε ένα αρχείο mapfile.

- Αντικείμενο Layer τύπου Raster

Προκειμένου να δημιουργηθεί ένα θεματικό επίπεδο τύπου Raster, πρέπει στο όρισμα TYPE να γράψουμε τη λέξη RASTER. Επιπλέον, στο όρισμα DATA γράφεται το όνομα της εικόνας που θέλουμε να εισαχθεί στον χάρτη. Σημειώνεται, ότι το SHAPEPATH, όπως έχει καθορισθεί παραπάνω, καθώς και το DATA είναι άρρηκτα συνδεδεμένα μεταξύ τους. Συνεπώς, η εικόνα, προκειμένου να εμφανισθεί στον χάρτη, θα πρέπει να βρίσκεται στο συγκεκριμένο path: **data/ bath_mapserver.tif**

```

LAYER
NAME όνομα εικονας
TYPE RASTER
STATUS DEFAULT
DATA bath_mapserver.tif
END

```

- Αντικείμενο Layer τύπου Vector

Ένα θεματικό επίπεδο τύπου Vector μπορεί να είναι γραμμή, σημείο ή πολύγωνο. Στο παρακάτω παράδειγμα δημιουργείται ένα αντικείμενο LAYER τύπου Vector και συγκεκριμένα πολύγωνο (TYPE POLYGON), το οποίο είναι shapefile και βρίσκεται στο path: **data/ shapefile/countries_area.shp**.

```
LAYER
NAME "world_poly"
DATA 'shapefile/countries_area.shp'
STATUS ON
TYPE POLYGON
CLASS
NAME 'The World'
STYLE
OUTLINECOLOR 0 0 0
END
END
END # layer
```

Στο συγκεκριμένο παράδειγμα διαμορφώνεται το στυλ του πολυγώνου έτσι ώστε να εμφανίζεται μόνο το περίγραμμα του σε χρώμα: (rgb:0 0 0)

- Αντικείμενο Layer από PostgreSQL/PostGIS.

Μια βασική δυνατότητα του MapServer είναι να δέχεται ως πηγή χωρικών δεδομένων μια χωρική βάση δεδομένων, όπως είναι η PostgreSQL/PostGIS.

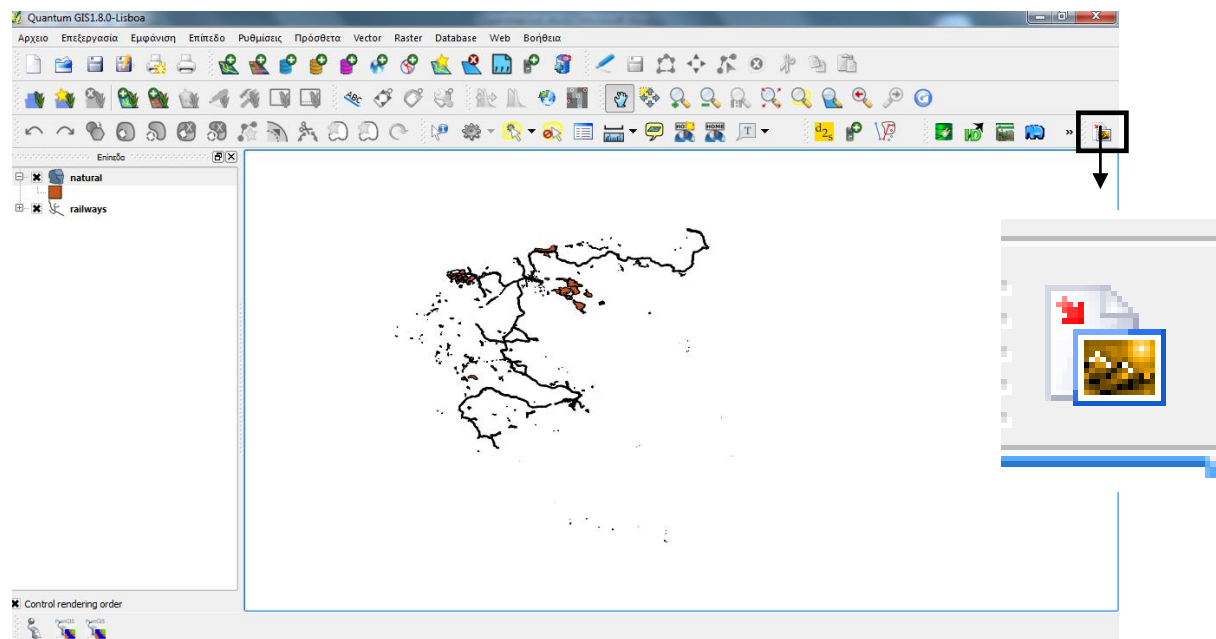
```
LAYER
NAME "nomoi"
TYPE POLYGON
CONNECTIONTYPE POSTGIS
CONNECTION "host=127.0.0.1 dbname=topogeomap user=postgres password=*****
port=5432"
DATA "the_geom FROM public.nomoi using SRID=4321"
STATUS DEFAULT
CLASS
NAME "Countries"
COLOR 255 0 0
END
END
```

Τα βασικά ορίσματα που πρέπει να καθοριστούν προκειμένου να επιτευχθεί η λήψη του χωρικού δεδομένου από τη βάση δεδομένων είναι το CONNECTIONTYPE, το CONNECTION, το port και το DATA. Συγκεκριμένα, στο CONNECTIONTYPE καθορίζεται ο τύπος της χωρικής βάσης δεδομένων, στην οποία θέλουμε να συνδεθούμε. Στο όρισμα CONNECTION καθορίζονται τα στοιχεία σύνδεσης με τη βάση δεδομένων. Τα τελευταία, απαρτίζονται από το host που αποτελεί τη διεύθυνση στην οποία εγκατασταθηκε η βάση δεδομένων (στο παράδειγμα είναι σε τοπικό σύστημα), από το όνομα της βάσης δεδομένων (topogeomap) και από

το username και τον κωδικό εισαγωγής (password). Ένα επιπλέον στοιχείο σύνδεσης αποτελεί η θύρα στην οποία εγκαταστάθηκε η βάση δεδομένων (το 5432 αποτελεί προκαθορισμένη θύρα εγκατάστασης). Τέλος, στο όρισμα DATA καθορίζεται το μέρος από όπου θα ληφθούν τα γεωμετρικά στοιχεία από τη βάση δεδομένων. Στο συγκεκριμένο παράδειγμα επιλέγεται η στήλη the_geom του πίνακα ποιοί χρησιμοποιώντας ως σύστημα προβολής το ΕΓΣΑ 87 (SRID:4321). Πρέπει να σημειωθεί ότι η στήλη the_geom περιέχει όλη τη γεωμετρία (τις συντεταγμένες) του κάθε αντικειμένου.

- Αυτόματη δημιουργία αρχείου Mapfile μέσω του QGIS

Το QGIS διαθέτει ένα σημαντικό εργαλείο, με το οποίο η δημιουργία του αρχείου Mapfile γίνεται αυτόματη. Η μόνη ενέργεια που απαιτείται από τον χρήστη είναι η σύνθεση και η διαμόρφωση του χάρτη μέσω QGIS.



Εικόνα 4-7: Σύνθεση χάρτη σε QGIS.

Στην εικόνα 4-7 παρατίθεται παράδειγμα σύνθεσης ενός χάρτη σε QGIS. Συγκεκριμένα, η εικόνα περιλαμβάνει το σιδηροδρομικό δίκτυο και τις προστατευόμενες περιοχές σε μια περιοχή. Επιλέγοντας το εργαλείο που φαίνεται παραπάνω, πραγματοποιείται η επιθυμητή μετατροπή, διαμορφώνοντας το αρχείο Mapfile κατά τέτοιο τρόπο, ώστε το αποτέλεσμα να είναι ακριβώς όπως φαίνεται παραπάνω.

Απόσπασμα του αρχείου Mapfile για το θεματικό επίπεδο natural (προστατευόμενες περιοχές), όπως δημιουργήθηκε αυτόματα από το QGIS:

```
LAYER
NAME 'natural'
TYPE POLYGON
DUMP true
TEMPLATE fooOnlyForWMSGetFeatureInfo
EXTENT 16.639223 33.541511 35.466617 43.441801
DATA '../ natura/natural.shp'
METADATA
  'ows_title' 'natural'
END
STATUS OFF
TRANSPARENCY 100
PROJECTION
  'proj=longlat'
  'datum=WGS84'
  'no_defs'
END
CLASS
  NAME 'natural'
  STYLE
    WIDTH 0.91
    OUTLINECOLOR 0 0 0
    COLOR 199 86 29
  END
END
END
```

(Jeff McKenna ,2008)

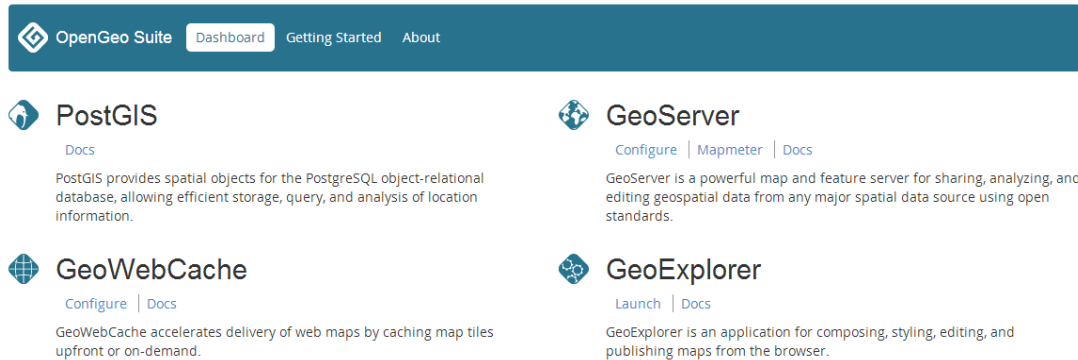
4.3. Συστήματα Διαχείρισης Περιεχομένου

Τα συστήματα διαχείρισης περιεχομένου (ΣΔΠ, Content Management Systems, CMS) είναι διαδικτυακές εφαρμογές που επιτρέπουν την online τροποποίηση του περιεχομένου ενός δικτυακού τόπου. Οι διαχειριστές, μέσω του Διαδικτύου, ενημερώνουν το περιεχόμενο στο ΣΔΠ, το οποίο είναι εγκατεστημένο σε ένα διακομιστή. Οι αλλαγές αυτές γίνονται αυτόματα διαθέσιμες πάλι μέσω του Διαδικτύου σε όλους τους επισκέπτες και χρήστες του δικτυακού τόπου.

4.3.1. OpenGeo Suite

Το OpenGeo Suite είναι ένα ολοκληρωμένο πακέτο λογισμικών, το οποίο χρησιμοποιείται για την υλοποίηση γεωχωρικών εφαρμογών Διαδικτύου και διατίθεται υπό την άδεια GPLv2. Συγκεκριμένα, με την εγκατάστασή του εγκαθίσταται η χωρική βάση δεδομένων PostGIS, ο εξυπηρετητής χαρτών

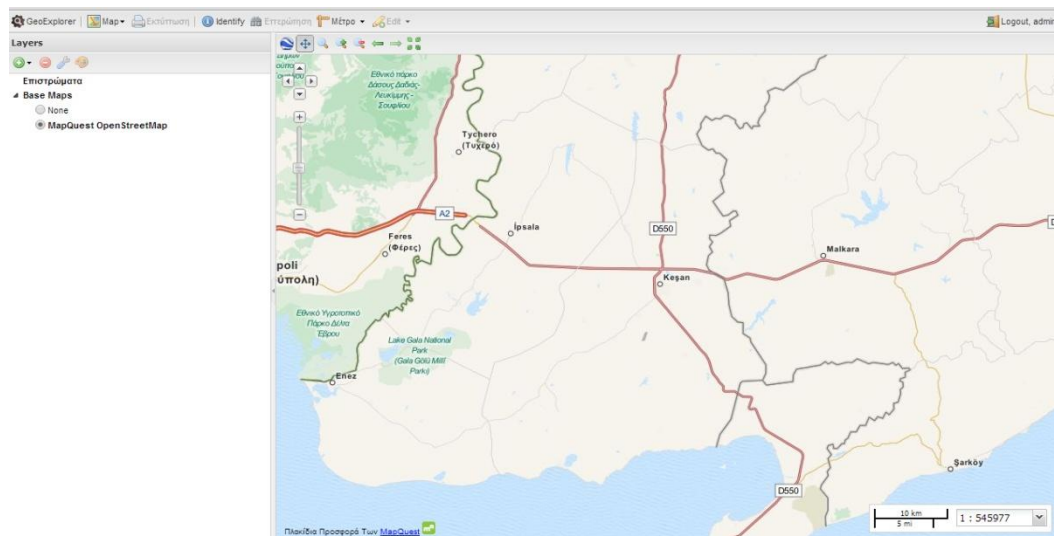
GeoServer, το εργαλείο GeoWebCache και η διαδικτυακή εφαρμογή GeoExplorer. Σημειώνεται, ότι τα ανωτέρω λογισμικά είναι ανοικτού κώδικα και αποτελούν το σκελετό της αρχιτεκτονικής μιας WebGIS εφαρμογής.



Εικόνα 4-8: Πακέτο λογισμικών OpenGeo Suite.

➤ GeoExplorer

Ο GeoExplorer είναι μια διαδικτυακή εφαρμογής που βασίζεται στο Framework GeoExt και χρησιμοποιείται για την εύκολη σύνθεση, διαμόρφωση και δημοσίευση χαρτών στο Διαδίκτυο. Μέσω του GeoExplorer γίνεται εύκολη η συγκέντρωση χαρτών από πληθώρα εξυπηρετητών χαρτών αλλά και OGC WMS.

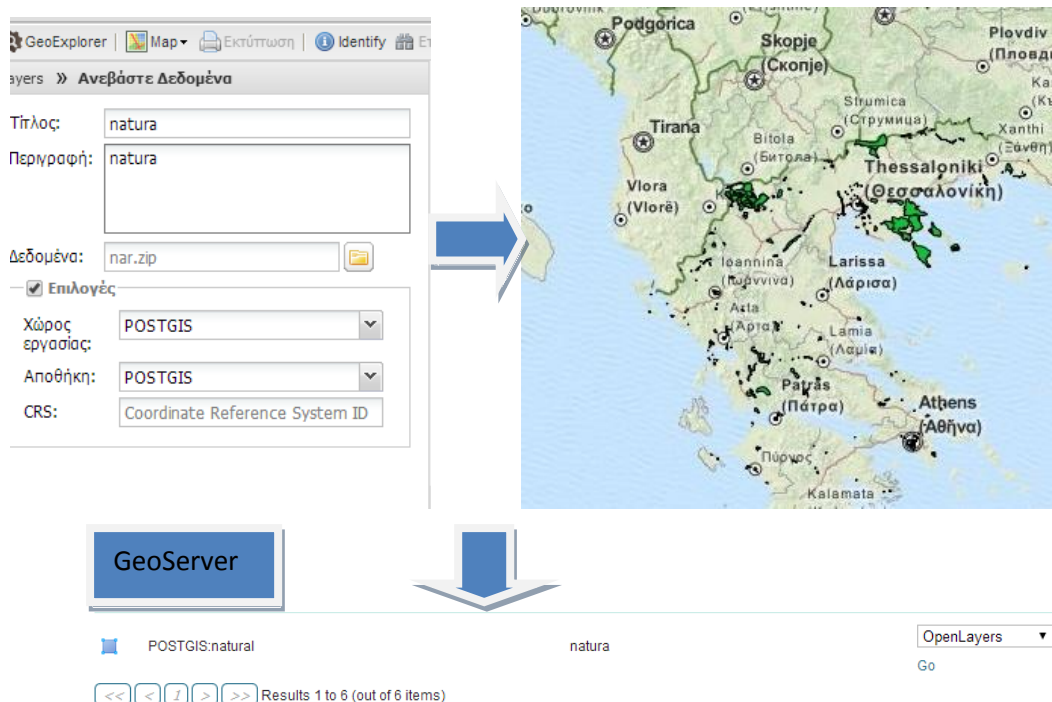


Εικόνα 4-9: GeoExplorer.

Ο διαχειριστής μπορεί με τέσσερα βήματα να επιτύχει τη σύνθεση και μορφοποίηση ενός χάρτη και, τελικώς, τη δημοσίευσή του στο Διαδίκτυο, μέσω μιας διαδραστικής WebGIS εφαρμογής.

- **Βήμα 1^ο** : Μεταφόρτωση των Δεδομένων

Με το GeoExplorer ο διαχειριστής μεταφορτώνει διανυσματικά δεδομένα στη βάση δεδομένων PostGIS, ενώ στη συνέχεια αυτομάτως δημοσιεύονται τα δεδομένα στον εξυπηρετητή χαρτών GeoServer. Βασική προϋπόθεση είναι να έχει συνδεθεί η PostGIS με τον Geoserver, δηλαδή να έχει δημιουργηθεί αποθήκη (Store) με πηγή δεδομένων την PostGIS.

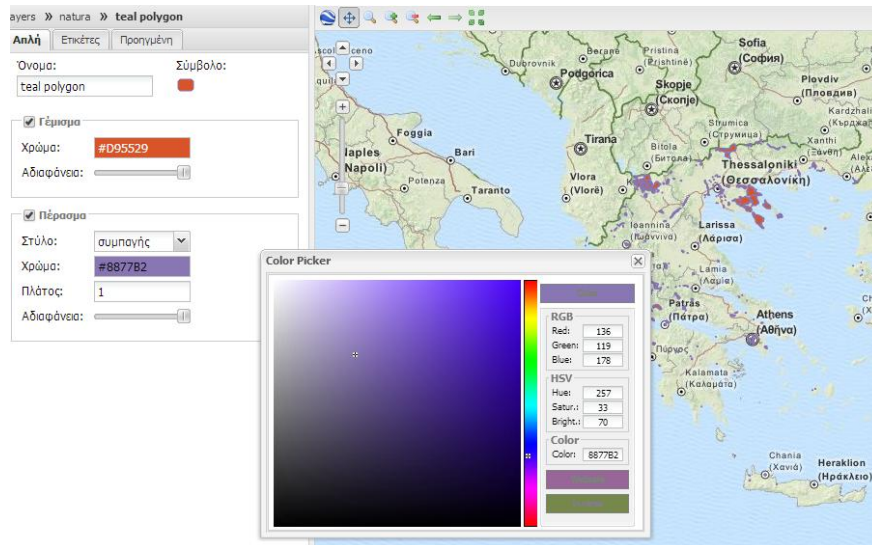


Εικόνα 4-10: Μεταφόρτωση διανυσματικών δεδομένων.

Με την ίδια διαδικασία γίνεται μεταφόρτωση raster δεδομένων, χωρίς όμως να αποθηκεύονται στην PostGIS.

- **Βήμα 2^ο** : Μορφοποίηση του στυλ του θεματικού επιπέδου

Το GeoExplorer διαθέτει ένα εργαλείο της GXP (gxp.plugins.styler), μέσω του οποίου η μορφοποίηση και η αποθήκευση του στυλ του θεματικού επιπέδου γίνεται εύχρηστη, διαθέτοντας πληθώρα δυνατοτήτων μορφοποίησης.



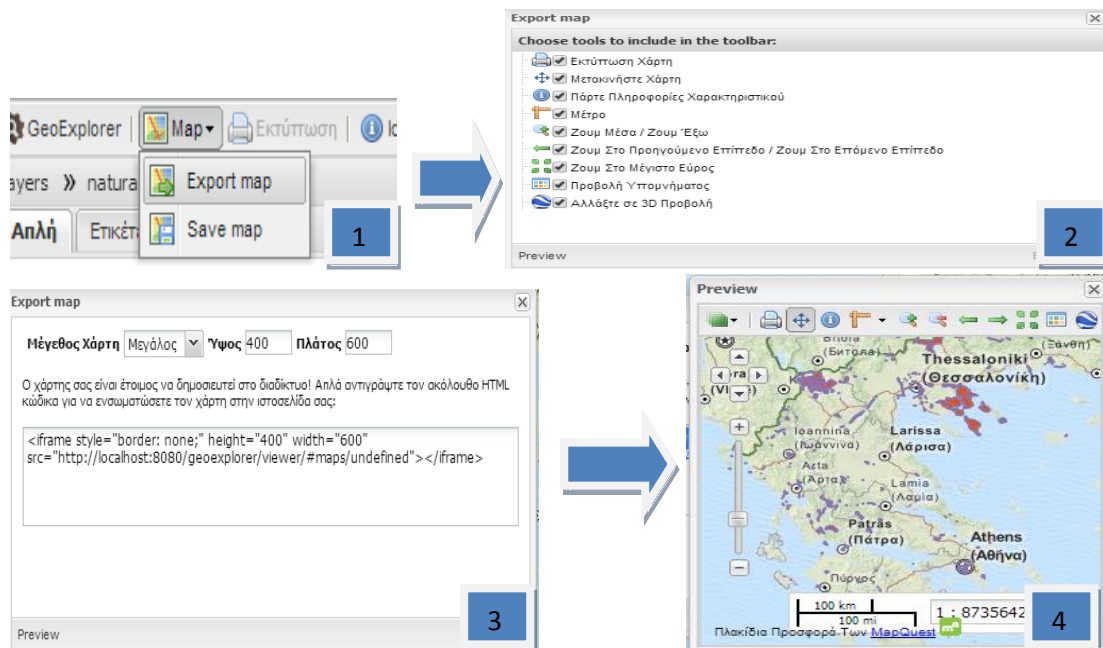
Εικόνα 4-11: Μορφοποίηση διανυσματικών δεδομένων.

- **Βήμα 3^ο** : Δυνατότητα τροποποίησης του θεματικού επιπέδου

Με το GeoExplorer υπάρχει δυνατότητα τροποποίησης και δημιουργίας στοιχείων του θεματικού επιπέδου με άμεση ενημέρωση τόσο της PostGIS όσο και του GeoServer. Το εργαλείο που χρησιμοποιείται είναι της GXP (`gxp.plugins.featureEditor`), ενώ η τροποποίηση και η άμεση ενημέρωση γίνεται μέσω της υπηρεσίας WFS-T (Web Feature Service-Transaction).

- **Βήμα 4^ο** : Αποθήκευση και δημοσίευση του χάρτη

Αφού ολοκληρωθούν τα παραπάνω τρία βήματα και ο διαχειριστής πετύχει το επιθυμητό αποτέλεσμα στο χάρτη, θα ακολουθήσει η αποθήκευση και η δημοσίευσή του. Συγκεκριμένα, ο διαχειριστής μέσω του GeoExplorer, εξάγει το χάρτη, επιλέγοντας ταυτόχρονα και τα εργαλεία που επιθυμεί να είναι διαθέσιμα στην WebGIS εφαρμογή.



Εικόνα 4-12: Διαδικασία εξαγωγής του χάρτη.

Πρέπει να σημειωθεί ότι η τελική WebGIS εφαρμογή που δημιουργείται, μπορεί να ενσωματωθεί σε οποιαδήποτε ιστοσελίδα, βάζοντας απλώς τον HTML κώδικα που παρουσιάζεται στην εικόνα 4-12 (3).

(OpenGeo, 2012)

4.3.2. MapGuide OpenSource

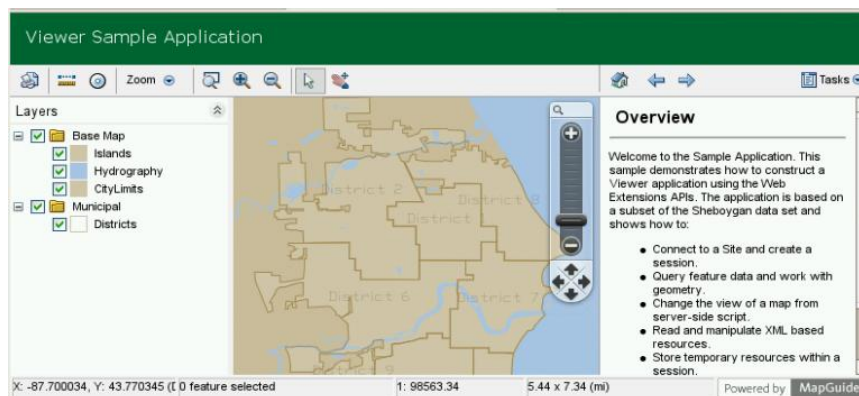
Το MapGuide OpenSource είναι ένα διαδικτυακό λογισμικό που επιτρέπει στους χρήστες να αναπτύξουν και να παρουσιάσουν εφαρμογές χαρτογραφίας μέσω Διαδικτύου και διαδικτυακών υπηρεσιών. Το MapGuide ενσωματώνει μια διαδραστική εφαρμογή οπτικοποίησης χαρτών, η οποία υποστηρίζει την επιλογή χαρακτηριστικών, την παρατήρηση περιγραφικών δεδομένων, την εμφάνιση πληροφοριών στο χάρτη αλλά και διαδικασίες, όπως είναι η δημιουργία ζώνης επιρροής (buffer), οι χωρικές επιλογές και οι μετρήσεις. Το MapGuide περιλαμβάνει μια βάση δεδομένων XML για τη διαχείριση περιεχομένου. Το λογισμικό Maestro επιτρέπει στους χρήστες να επεξεργάζονται τη βάση XML και να χτίζουν μια διαδικτυακή εφαρμογή χωρίς να γράψουν ούτε μια γραμμή κώδικα. Το MapGuide, επίσης, υποστηρίζει τα πιο δημοφιλή πρότυπα χωρικών δεδομένων, τις πιο διαδεδομένες βάσεις δεδομένων και τα ανοικτά πρότυπα. Το MapGuide μπορεί να εγκατασταθεί σε περιβάλλον Linux ή Windows, υποστηρίζει τους εξυπηρετητές Apache και IIS, ενώ συνάμα παρέχει εκτεταμένη υποστήριξη για τις γλώσσες PHP, NET, Java, και JavaScript για την ανάπτυξη εφαρμογών. Το MapGuide Open Source διατίθεται υπό την άδεια LGPL (Lesser General Public Licence), η οποία αποτελεί μια πιο ελαστική έκδοση της GPL. Μια LGPL άδεια επιτρέπει την ανάγνωση,

τροποποίηση και αναδιανομή του λογισμικού, μαζί με τον πηγαίο κώδικά του, ακόμα και αν μια βιβλιοθήκη LGPL συνδεθεί με ένα πρόγραμμα που χρησιμοποιεί άλλη άδεια και δεν αποτελεί ελεύθερο λογισμικό.

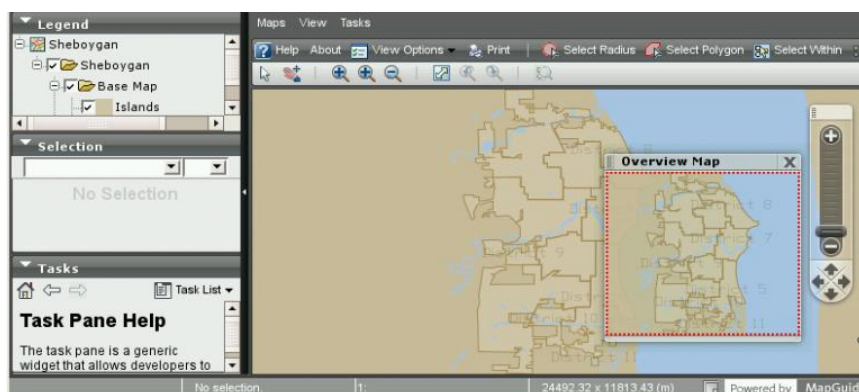
(Πηγή: http://live.osgeo.org/el/overview/mapguide_overview.html)

➤ Βασικά χαρακτηριστικά

Ένα βασικό πλεονέκτημα του λογισμικού MapGuide OpenSource είναι ότι προσφέρει μια διαδραστική οπτικοποίηση χαρτών, όπου το χαρτογραφικό αποτέλεσμα είναι αρκετά ποιοτικό. Συγκεκριμένα, η θέαση των χαρτών γίνεται μέσω δύο viewers: του AJAX viewer και του Fusion viewer. Ο πρώτος, αποτελεί μέσο θέασης χαρτών που εκμεταλλεύεται την τεχνολογία AJAX προσφέροντας ομαλή πλοήγηση, ενώ ο δεύτερος αποτελεί framework για την ανάπτυξη εφαρμογών χαρτογραφίας στο Διαδίκτυο δίνοντας έμφαση στο ευέλικτο στυλ και τη λειτουργικότητα. Επιπροσθέτως, διατίθενται εργαλεία για μετρήσεις αποστάσεων και εμβαδών, για επιλογή χαρακτηριστικών βάσει σημείου, πολυγώνου και κύκλου, για εκτύπωση του χάρτη και για δημιουργία, επεξεργασία και ανάλυση της γεωμετρίας χαρακτηριστικών μέσω ερωτημάτων (queries) και buffering.



Εικόνα 4-13: Παράδειγμα AJAX viewer.

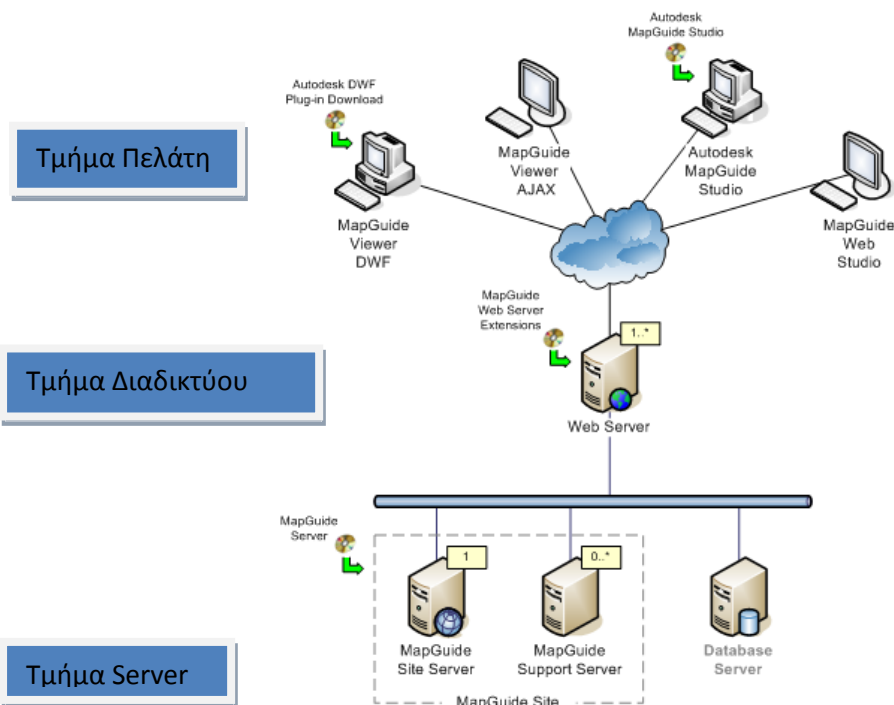


Εικόνα 4-14: Παράδειγμα fusion viewer.

Ένα ακόμα χαρακτηριστικό που διαθέτει είναι η ομοιογενής πρόσβαση στα δεδομένα μέσω χρήσης της βιβλιοθήκης FDO (Feature Data Objects). Η τελευταία, παρέχει έναν προκαθορισμένο τρόπο για πρόσβαση σε πληθώρα πηγών δεδομένων, περιλαμβανομένων των: SHP, SDF, Oracle, MySQL, και PostGIS. Τέλος, όσον αφορά τα θεματικά επίπεδα, υποστηρίζει λειτουργίες για προσθήκη, αφαίρεση και ομαδοποίηση θεματικών επιπέδων, καθώς και λειτουργίες δημιουργίας υπομνήματος.

➤ Αρχιτεκτονική του MapGuide Open Source

Η αρχιτεκτονική του MapGuide Open Source στηρίζεται στην τυπική αρχιτεκτονική τριών επιπέδων. Κάθε συστατικό μέρος του λογισμικού δραστηριοποιείται σε ένα από τα τρία επίπεδα. Αυτά τα συστατικά μέρη (components) μπορούν να λειτουργούν σε έναν υπολογιστή ή μπορεί να είναι διανεμημένα σε διάφορους υπολογιστές.



Εικόνα 4-15: Αρχιτεκτονική MapGuide Open Source.
Πηγή: <http://mapguide.osgeo.org/1.2/gettingstarted.html>

- MapGuide Server

Ο εξυπηρετητής του MapGuide (MapGuide Server) είναι αυτός που διατηρεί τις υπηρεσίες του MapGuide Open Source και ανταποκρίνεται στα ερωτήματα που θέτει ο εκάστοτε χρήστης μέσω του πρωτοκόλλου TCP/IP. Κάθε υπηρεσία είναι υπεύθυνη να παρέχει ένα συγκεκριμένο σύνολο λειτουργιών, ανάλογα με την

εφαρμογή που έχει δομηθεί. Οι υπηρεσίες που παρέχονται είναι επτά και αναγράφονται στον πίνακα 4-3:

Πίνακας 4-3: Διαθέσιμες υπηρεσίες του MapGuide server.

ΔΙΑΘΕΣΙΜΕΣ ΥΠΗΡΕΣΙΕΣ	ΠΑΡΑΤΗΡΗΣΕΙΣ
Site Service	παρέχει λειτουργίες για τη δημιουργία μίας ιστοσελίδας
Resource Service	διαχειρίζεται τα δεδομένα στη βάση
Drawing Service	παρέχει λειτουργίες σχεδίασης ενός χάρτη
Feature Service	διαχειρίζεται τα περιγραφικά χαρακτηριστικά των δεδομένων
Mapping Service	παρέχει λειτουργίες οπτικοποίησης και εκτύπωσης ενός χάρτη
Rendering Service	παρέχει λειτουργίες φωτοσκίασης στον χάρτη
Tile Service	

Στην περίπτωση που χρησιμοποιείται ένας και μοναδικός εξυπηρετητής, τότε όλες αυτές οι υπηρεσίες είναι αποθηκευμένες και λειτουργούν σε αυτόν. Διαφορετικά, όταν χρησιμοποιούνται περισσότεροι από έναν server, τότε ο site server λειτουργεί και διαχειρίζεται πάντα τις τέσσερις πρώτες υπηρεσίες, οι οποίες έχουν να κάνουν με την πρόσβαση και τη διαχείριση των πόρων της ιστοσελίδας. Οι υπόλοιπες υπηρεσίες (Mapping, Rendering και Tile) μπορούν να διαχωριστούν και να τοποθετηθούν σε διαφορετικούς εξυπηρετητές.

- MapGuide Web Server Extensions

Οι επεκτάσεις του εξυπηρετητή MapGuide είναι εσωτερικά συστατικά μέρη του όλου συστήματος, όπου η πρόσβαση σε αυτά γίνεται μέσω του MapGuide Web API (Application Programming Interface). Οι επεκτάσεις αυτές περιέχουν διαδικασίες που μετατρέπουν τις υπηρεσίες που προσφέρονται από τον MapGuide server (MapGuide server services) σε εφαρμογές του διαδικτύου τόσο σε τοπικό επίπεδο όσο και στον παγκόσμιο ιστό, χρησιμοποιώντας το HTTP πρωτόκολλο. Το interface του MapGuide (MapGuide Web API) υποστηρίζει τρεις διαφορετικές γλώσσες – περιβάλλοντα ανάπτυξης μίας εφαρμογής, τα ASP.NET, Java/JSP και PHP.

- MapGuide Viewer

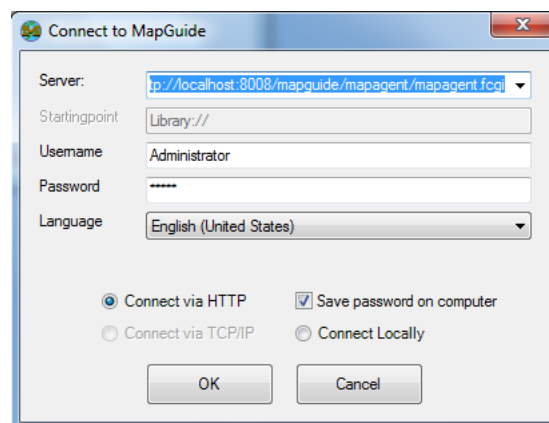
Πρόκειται για ένα από τα συστατικά μέρη που λειτουργούν στο τρίτο τμήμα (τμήμα πελάτη), μέσω του οποίου γίνεται η θέαση του χάρτη που δημιουργήθηκε στο Διαδίκτυο.

(Πηγή: <http://mapguide.osgeo.org/1.2/gettingstarted.html>)

➤ Παράδειγμα WebGIS

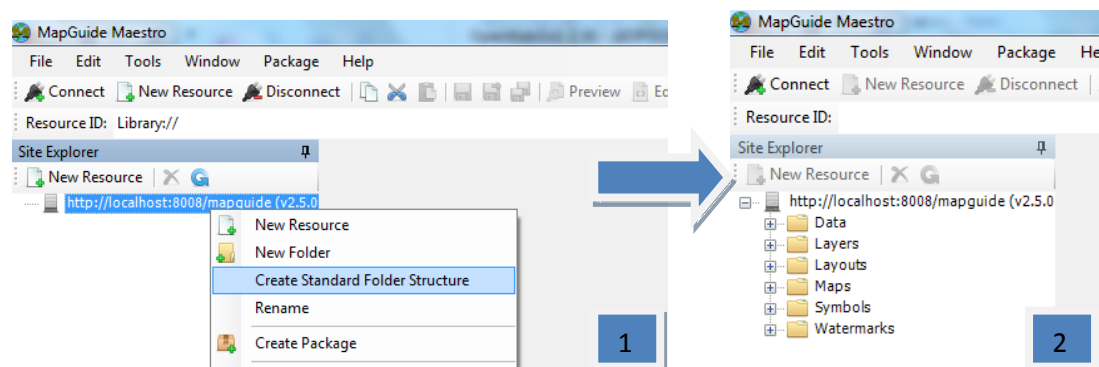
Προκειμένου να γίνει εμφανής η ευκολία της δημιουργίας διαδικτυακής εφαρμογής μέσω του λογισμικού Maestro θα ακολουθήσει η παράθεση παραδείγματος. Επιδιώκεται, επομένως, να δημιουργηθεί μια διαδικτυακή εφαρμογή μέσω της οποίας θα απεικονίζεται το γενικό πολεοδομικό σχέδιο του Δήμου Αργυρούπολης. Τα εν λόγω δεδομένα ελήφθησαν μέσω της πύλης **geodata.gov.gr** σε μορφή Shapefile (SHP). Η λειτουργία της πύλης **geodata.gov.gr** είναι μια σημαντική καινοτομία στο χώρο της διάθεσης της κυβερνητικής πληροφορίας, καθώς αποτέλεσε το πρώτο βήμα για την πρακτική εφαρμογή της ελεύθερης διάθεσης γεωχωρικών δεδομένων δίχως θεσμικούς ή τεχνικούς φραγμούς.

Αρχικά, με την εκκίνηση του Maestro, ζητείται η σύνδεση με τον εξυπηρετητή που έχει εγκατασταθεί το MapGuide OpenSource.



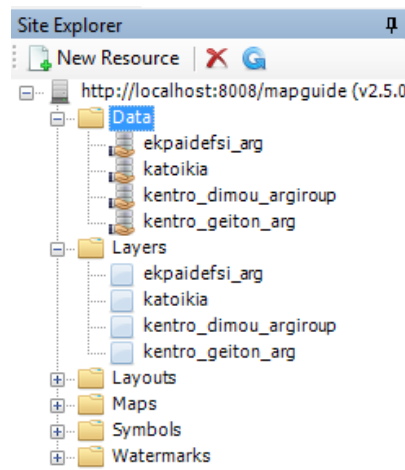
Εικόνα 4-16: Σύνδεση Maestro με MapGuide.

Αφού επιτευχθεί η σύνδεση, τότε μέσω της διαδικασίας που φαίνεται στην εικόνα 4-17, δημιουργούνται οι απαραίτητοι φάκελοι που αποτελούν τη δομή της διαδικτυακής εφαρμογής. Συγκεκριμένα, δημιουργούνται οι φάκελοι Data, Layers, Layouts, Maps, Symbols και Watermarks



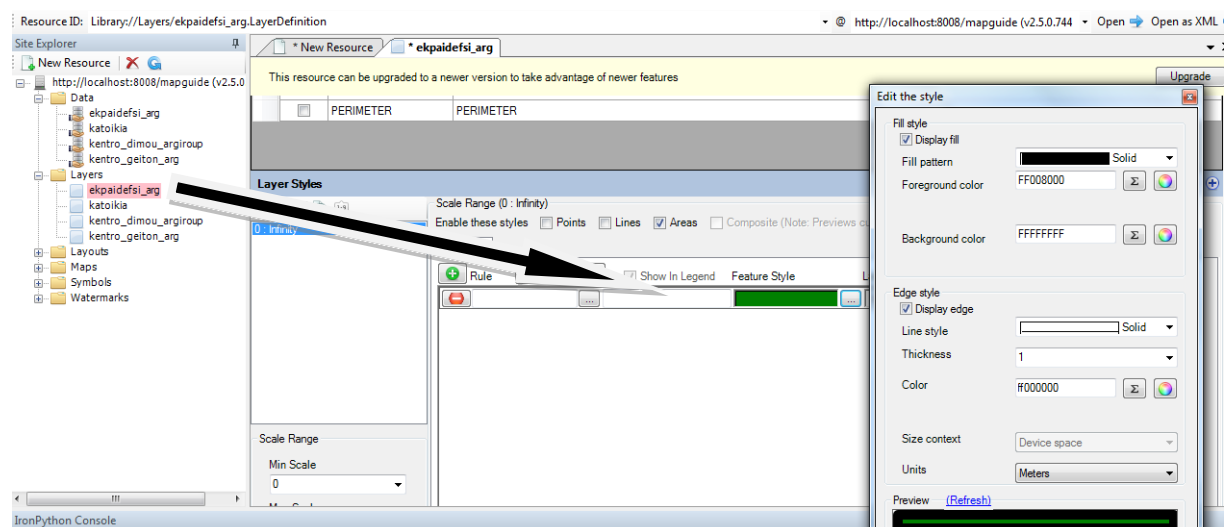
Εικόνα 4-17: Δημιουργία Βασικών φακέλων.

Μέσα στο φάκελο Data καθορίζεται η πηγή και η θέση των διανυσματικών δεδομένων. Σημειώνεται ότι η πρόσβαση των δεδομένων γίνεται μέσω της βιβλιοθήκης FDO. Επιπροσθέτως, προσδιορίζεται το σύστημα αναφοράς και ο φάκελος που θα αποθηκευτούν τα θεματικά επίπεδα. Το σύστημα αναφοράς είναι το ΕΓΣΑ 87, ενώ ο φάκελος είναι ο Layers. Με την ολοκλήρωση της διαδικασίας εμφανίζεται το αποτέλεσμα που φαίνεται στην εικόνα 4-18:



Εικόνα 4-18: Προσδιορισμός πηγής δεδομένων.

Στη συνέχεια, μέσω του φακέλου Layers, υπάρχει η δυνατότητα μορφοποίησης των διαθέσιμων θεματικών επιπέδων τόσο στην εμφάνιση όσο και στην περαιτέρω επεξεργασία των περιγραφικών δεδομένων τους. Η επεξεργασία αυτή περιλαμβάνει αφενός τη δημιουργία επιπρόσθετων πληροφοριών βασιζόμενων στο πρωταρχικά δεδομένα με χρήση διαθέσιμων συναρτήσεων και αφετέρου την επιλογή των πληροφοριών που επιθυμεί ο διαχειριστής να είναι εφικτή προς θέαση.

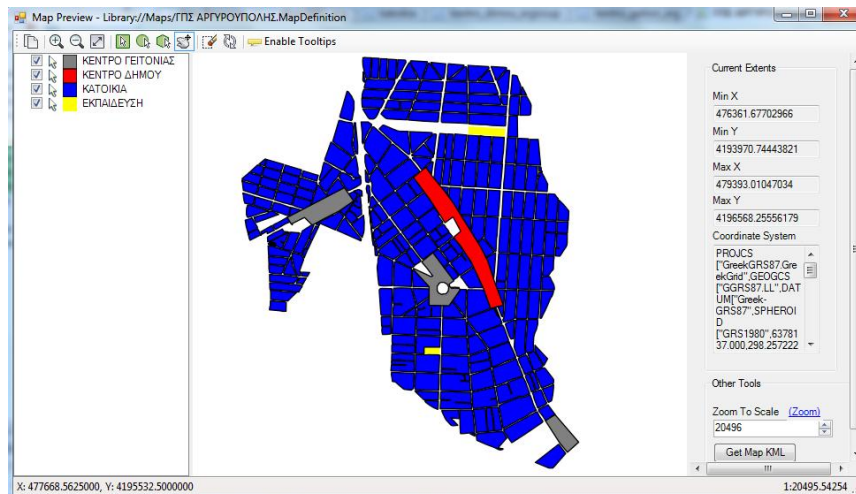


Εικόνα 4-19: Μορφοποίηση του στυλ των θεματικών επιπέδων.

Στο επόμενο βήμα, μόλις ολοκληρωθεί η επιθυμητή τροποποίηση των θεματικών επιπέδων, σειρά έχει η δημιουργία του χάρτη. Η εντολή που εκτελείται είναι:

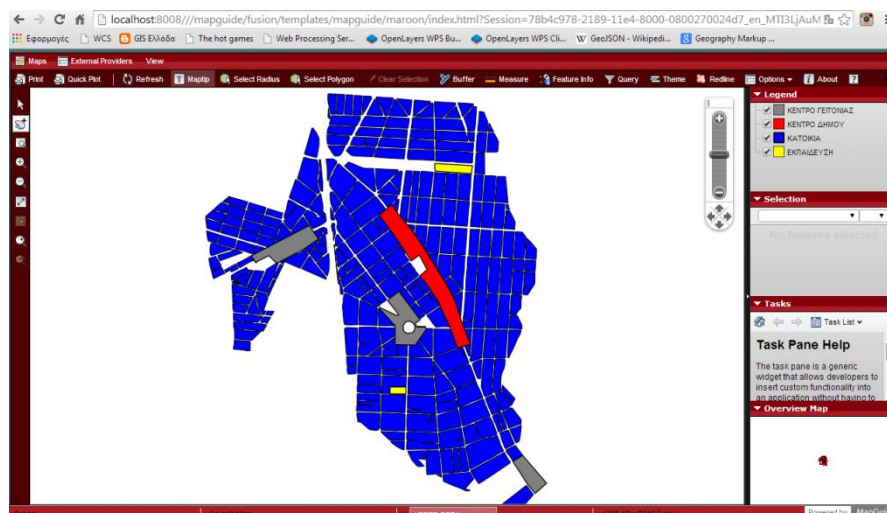
Maps→New Resource→ Map Definition

Στον χάρτη που δημιουργείται, καθορίζεται το όνομά του, το σύστημα αναφοράς, τα όρια θέασης και τα θεματικά επίπεδα που θα περιέχει. Στην εικόνα 4-20 φαίνεται σε προεπισκόπηση ο χάρτης, όπως διαμορφώθηκε μέχρι στιγμής.



Εικόνα 4-20: Προεπισκόπηση του χάρτη.

Τελευταίο στάδιο της διαδικασίας είναι η τελική εξαγωγή του χάρτη στο Διαδίκτυο, καθορίζοντας ταυτόχρονα τη γραμμή εργαλείων, την εμφάνιση του WebGIS και τα βασικά υπόβαθρα. Ως βασικά υπόβαθρα μπορεί να επιλέξει ο διαχειριστής μεταξύ των υποβάθρων της Google, του Bing και του Open Street Map. Στο συγκεκριμένο παράδειγμα ως μέσο οπτικοποίησης του χάρτη χρησιμοποιήθηκε ο fusion viewer.



Εικόνα 4-21: Τελική εφαρμογή.

4.3.3. Cartaro

Το λογισμικό Cartaro παρέχει γεωχωρικές λειτουργίες και διαδικτυακές υπηρεσίες στο Σύστημα Διαχείρισης Περιεχομένου (CMS) Drupal και διατίθεται υπό την GPLv2 άδεια. Μέσω του Cartaro υποστηρίζονται διαδικτυακές λειτουργίες OGC, καθώς και η προβολή και η χρήση χαρτών μέσα σε ιστοσελίδες. Επιπροσθέτως, είναι δομημένο με χρήση Ανοικτών Λογισμικών όπως την PostGIS, το GeoServer, το GeoWebCache και την OpenLayers. Σημειώνεται, ότι το Cartaro είναι σχεδιασμένο για χρήση τόσο σε ιστοχώρους CMS που χειρίζονται γεωχωρικά δεδομένα όσο και σε οργανισμούς που επιθυμούν μια Υποδομή Χωρικών Δεδομένων (SDI) με ελάχιστη διαμόρφωση και προγραμματισμό.

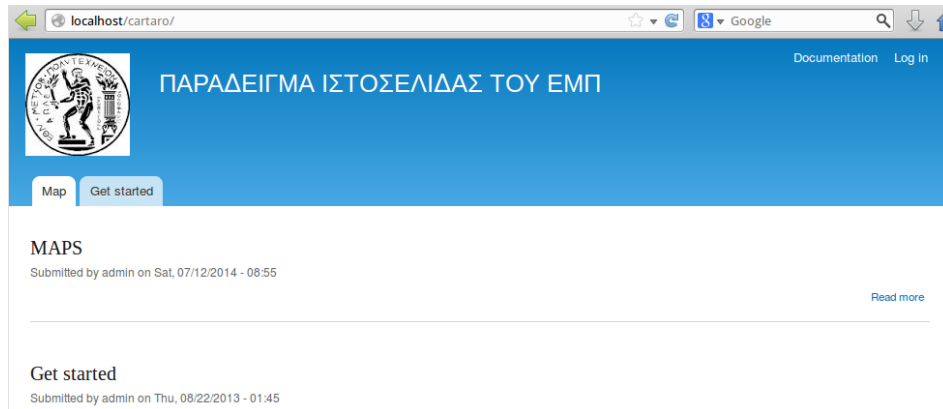


Εικόνα 4-22: Cartaro.

Είναι αξιοσημείωτο ότι οι επιπρόσθετες δυνατότητες του Drupal ενσωματώνονται μέσω module. Τα τελευταία, αποτελούν μικρά προγραμματιστικά πακέτα, όπου αναλόγως των αναγκών, προσθέτουν κάποια λειτουργία στο Drupal. Μερικά από τα module που χρησιμοποιούνται για την ενσωμάτωση γεωχωρικών δυνατοτήτων στο Drupal είναι το PostGIS module, το Openlayers module, το GeoServer module, το Feeds module και το GDAL module. Τα εν λόγω module έχουν ενσωματωθεί στο λογισμικό Cartaro και έτσι δεν χρειάζεται να ενσωματωθούν αυτοτελώς.

Παρακάτω θα παρατεθούν τα βήματα με τα οποία ο διαχειριστής μπορεί να δημιουργήσει εύκολα έναν χάρτη με χωρικές πληροφορίες που ο ίδιος επιθυμεί. Οι τελευταίες, αυτόματα θα εισαχθούν στη βάση δεδομένων PostGIS και θα δημοσιευθούν μέσω του εξυπηρετητή χαρτών GeoServer.

Αφού ολοκληρωθεί επιτυχώς η εγκατάσταση του λογισμικού Cartaro, εμφανίζεται η παρακάτω ιστοσελίδα, η οποία διαμορφώθηκε κατάλληλα, ώστε να αποτελεί παράδειγμα ιστοσελίδας του Ε.Μ.Π.

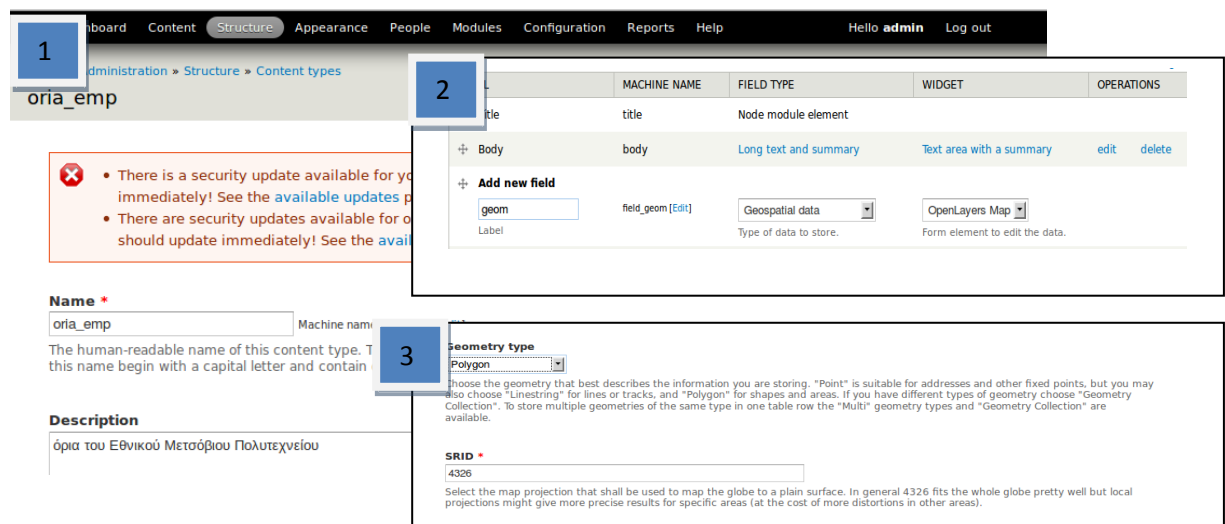


Εικόνα 4-23: Παράδειγμα ιστοσελίδα του ΕΜΠ.

Με τη σύνδεση του χρήστη ως “διαχειριστή” εμφανίζεται μια κατάλληλη γραμμή εργαλείων, την οποία ο διαχειριστής μπορεί να διαμορφώσει ανάλογα με την ιστοσελίδα του.

➤ Δημιουργία θεματικού επιπέδου (Layer)

Το πρώτο βήμα για τη σύνθεση ενός χάρτη είναι η δημιουργία των κατάλληλων θεματικών επιπέδων. Στο συγκεκριμένο παράδειγμα θα αναπτυχθεί ένα θεματικό επίπεδο τύπου polygon (πολύγωνο), το οποίο θα ορίζει τα όρια του Εθνικού Μετσόβιου Πολυτεχνείου. Αρχικά, δημιουργείται ένα νέο content type με όνομα oria_emp, στο οποίο προστίθεται ένα νέο πεδίο με γεωμετρικές ιδιότητες τύπου πολύγωνο. Επιπροσθέτως, καθορίζεται και το προβολικό σύστημα σε κώδικα SRID.



Εικόνα 4-24: Νέο content type.

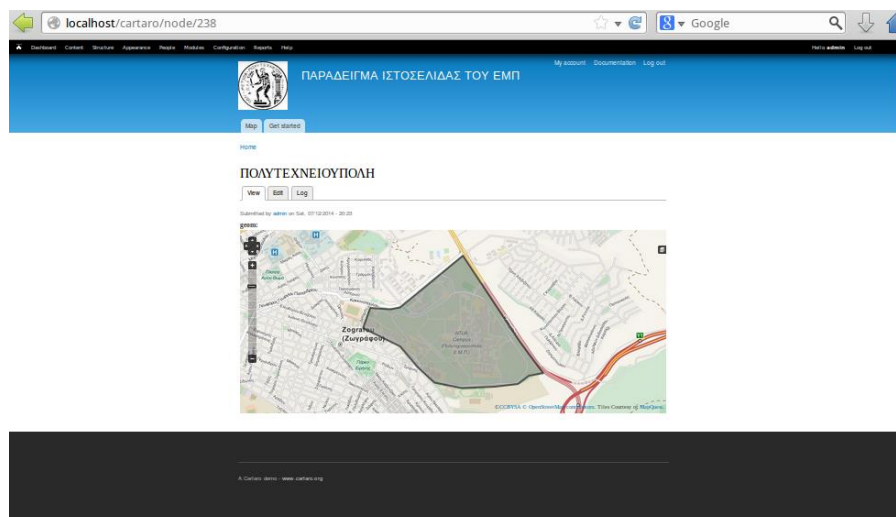
Στην συνέχεια, ακολουθείται η διαδικασία δημιουργίας και αντιστοίχισης του θεματικού επιπέδου στον GeoServer. Συγκεκριμένα, με την ολοκλήρωση της

διαδικασίας θα αποθηκευτεί το θεματικό επίπεδο στη χωρική βάση δεδομένων PostGIS και θα δημοσιευθεί στο Διαδίκτυο μέσω του GeoServer.

Εικόνα 4-25: Δημιουργία θεματικού επιπέδου στον GeoServer.

➤ Ψηφιοποίηση του πολυγώνου και δημιουργία του χάρτη

Στο επόμενο βήμα δημιουργείται ένα νέο content τύπου oria_emp, στο οποίο πραγματοποιείται η ψηφιοποίηση του πολυγώνου με τη χρήση των κατάλληλων εργαλείων. Τέλος, με την αποθήκευση της ψηφιοποίησης, ολοκληρώνεται η διαδικασία και δημιουργείται ο χάρτης με την επιθυμητή πληροφορία.



Εικόνα 4-26: Τελική ιστοσελίδα με τον επιθυμητό χάρτη.

Μερικές επιπρόσθετες δυνατότητες που παρέχει το λογισμικό Cartaro είναι η μεταφόρτωση διαφόρων format διανυσματικών επιπέδων, όπως GeoJson και kml, καθώς επίσης και η δημιουργία σημείων από αρχείο CSV. Τέλος, υπάρχει δυνατότητα μορφοποίησης του στυλ των διαφόρων διανυσματικών δεδομένων, που δημοσιεύονται μέσω του GeoServer, μέσω του πρότυπου SLD¹¹.

(Πηγή: <http://cartaro.org/documentation>)

4.3.4. GeoNode

Το GeoNode αποτελεί ένα ολοκληρωμένο και ελεύθερο σύστημα διαχείρισης περιεχομένου με πολλαπλές δυνατότητες. Διατίθεται υπό την άδεια GPLv2 και στον πυρήνα του είναι δομημένο στα λογισμικά GeoServer, Pycsw¹², Django¹³ και GeoExt.

Πάνω από τον πυρήνα το έργο περιλαμβάνει ένα εργαλείο σύνθεσης χαρτών, καθώς και εργαλεία για ανάλυση και δημιουργία αναφορών.

➤ Μεταφόρτωση δεδομένων και δημιουργία μεταδεδομένων

Αφού ολοκληρωθεί η εγγραφή του χρήστη, του δίνεται η δυνατότητα να ανεβάσει στην εφαρμογή διανυσματικά και εικονιστικά δεδομένα στην αρχική τους μορφή και προβολή, μέσω μιας διαδικτυακής φόρμας. Τα διανυσματικά δεδομένα ανεβαίνουν στη μορφή ESRI Shapefile και τα εικονιστικά δεδομένα σε μορφή GeoTIFF. Επίσης, με τη μεταφόρτωση των δεδομένων, καθορίζει σε ποιους θα επιτρέψει αντίστοιχα να κατεβάσουν, να μορφοποιήσουν ή να δουν τα συγκεκριμένα δεδομένα.

¹¹ SLD: το πρότυπο Styled Layer Descriptor (SLD) του OGC® παρέχει δομές για συμβολισμούς και χρώματα γεωγραφικών χαρακτηριστικών και εικόνων. Χρησιμοποιείται ευρύτατα για εφαρμογή στυλ σε διαδικτυακές υπηρεσίες WMS ή σε αρχεία GML που προέρχονται από υπηρεσίες WFS.

¹² Pycsw: ο pycsw είναι ένας εξυπηρετητής (server) που επιτρέπει στους χρήστες να δημοσιεύσουν καταλόγους με πληροφορίες για σύνολα δεδομένων (μεταδεδομένα) στο Διαδίκτυο, με τη χρήση καθιερωμένων και ανοικτών προτύπων και διεπαφών, με τρόπο, ώστε να μπορούν να απαντούν σε ερωτήματα αλλά και να επικαιροποιούνται από λογισμικά πελάτες (client) του καταλόγου. Ο pycsw είναι μια υλοποίηση του Προτύπου Υπηρεσίας Καταλόγου του OGC (Catalogue Services for the Web ή CSW) σε γλώσσα προγραμματισμού Python. Το λογισμικό είναι πιστοποιημένο από το OGC και είναι Υλοποίηση Αναφοράς OGC.

¹³ Django: το Django είναι ένα ελεύθερο και ανοικτού κώδικα framework, γραμμένο σε python, το οποίο υποστηρίζει την εύκολη και γρήγορη ανάπτυξη δυναμικών ιστοσελίδων που στηρίζονται σε βάσεις δεδομένων. Ακολουθεί το model-view-controller (MVC) αρχιτεκτονικό πρότυπο και συντηρείται από το ίδρυμα DSF (Django Software Foundation).

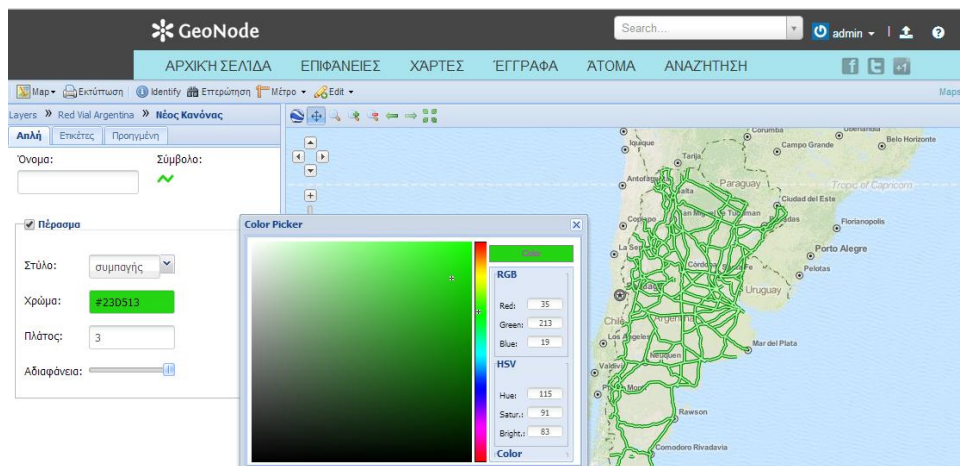
Εικόνα 4-27: Μεταφόρτωση δεδομένων και καθορισμός προσβασιμότητας.

Ειδική έμφαση έχει δοθεί στα πρότυπα μεταδεδομένων, όπως είναι το ISO 19139:2007. Μετά το ανέβασμα των δεδομένων, ο χρήστης συμπληρώνει μια φόρμα με τα μεταδεδομένα, τα οποία στη συνέχεια είναι διαθέσιμα μέσω της διεπαφής καταλόγου CSW. Οι χρήστες μπορούν να ανεβάσουν ακόμα και αρχεία μεταδεδομένων σε μορφή XML (τύπου ISO, FGDC ή Dublin Core), ώστε να συμπληρώσουν τα μεταδεδομένα αυτόματα.

Εικόνα 4-28: Δημιουργία μεταδεδομένων για τα δεδομένα.

➤ Μορφοποίηση δεδομένων

Η εφαρμογή παρέχει ένα εργαλείο διαχείρισης στυλ, το οποίο επιτρέπει στον χρήστη αφενός να διαμορφώσει τον τρόπο που θα παρουσιάζονται τα δεδομένα και αφετέρου να προβεί σε προεπισκόπηση των αποτελεσμάτων σε πραγματικό χρόνο. Η συγκεκριμένη δυνατότητα παρέχεται μέσω της εφαρμογής GeoExplorer, η οποία είναι ανοικτού κώδικα και ενσωματώθηκε στο Geonode.



Εικόνα 4-29: Μορφοποίηση του στυλ των δεδομένων.

➤ Πολλαπλοί τρόποι αναζήτησης δεδομένων

Μόλις ανεβούν τα δεδομένα στον εξυπηρετητή, το GeoNode επιτρέπει στον χρήστη να πραγματοποιήσει χωρική αναζήτηση ή αναζήτηση μέσω λέξεων - κλειδιών και να δημιουργήσει νέους χάρτες με τα επιθυμητά χωρικά δεδομένα. Όταν οι χάρτες αποθηκευτούν, είναι δυνατό να ενσωματωθούν σε οποιαδήποτε ιστοσελίδα ή να εκτυπωθούν σε PDF.

Όλα τα χαρτογραφικά επίπεδα μετατρέπονται αυτόματα σε Μερκατορική προβολή για τη θέασή τους στο Διαδίκτυο, ώστε να συνδυαστούν με άλλα δεδομένα, όπως τα Open Street Map, Google Satellite ή Bing.

(Πηγή: http://live.osgeo.org/el/quickstart/geonode_quickstart.html)

5. ΑΓΡΟΤΙΚΟΣ ΧΩΡΟΣ

Ως αγροτικός χώρος ορίζεται θετικά ο χώρος που προσδιορίζεται για αγροτική εκμετάλλευση της γης, ενώ αρνητικά ορίζεται εκείνος που δεν είναι ούτε αστικός ούτε δασικός (Ζεντέλης, 2011). Συνεπώς, ο αγροτικός χώρος αποτελείται από το αμιγές και το μη αμιγές τμήμα. Στο μη αμιγές τμήμα περιλαμβάνονται είτε οι περιοχές που καλύπτονται από κάθε μορφής οριοθετημένες και θεσμοθετημένες ζώνες είτε οι περιοχές όπου το εγκεκριμένο Γ.Π.Σ επιτρέπει τη δόμησή τους. Επιπροσθέτως, εντάσσονται και οι περιοχές εκτός σχεδίου, οι οποίες αναπτύσσονται νόμιμα σύμφωνα με τις ισχύουσες προϋποθέσεις. Στο αμιγές τμήμα περιέχονται οι περιοχές του μη αστικοποιημένου χώρου, οι οποίες χρησιμοποιούνται προς αγροτική εκμετάλλευση (Ζεντέλης, 2011). Οι τελευταίες, αποτελούνται από τμήματα γης που ονομάζονται αγροτεμάχια, ενώ μέρος αυτών είναι τα κληροτεμάχια, τα οποία αποτελούν το γεωργικό κλήρο που παραχωρήθηκε από το ελληνικό δημόσιο σε ακτήμονες καλλιεργητές.

5.1. Ιστορική αναδρομή

Στην ιστορική του πορεία το ελληνικό κράτος βρέθηκε πολλές φορές αντιμέτωπο με το ζήτημα της αξιοποίησης του αγροτικού χώρου. Το θέμα της ιδιοκτησίας της γης απασχόλησε ιδιαίτερα τους πολιτικούς και τις κυβερνήσεις τους κατά τη διάρκεια του 19^{ου} και του 20^{ου} αιώνα.

5.1.1. Ο αγροτικός χώρος κατά τον 19^ο αιώνα

Ένα από τα πολλά ζητήματα που κληροδότησε η οθωμανική κυριαρχία στο νέο ελληνικό κράτος υπήρξε το θέμα των <<εθνικών γαιών>>. Εθνικές γαίες ήταν οι ακίνητες κτηματικές ιδιοκτησίες των Οθωμανών στις περιοχές που περιήλθαν στον έλεγχο του ελληνικού κράτους. Υπολογίζεται ότι η έκταση των εθνικών κτημάτων ανερχόταν σε 10.000.000 στρέμματα και αποτελούσαν το 35% της γεωργικής γης. Η διανομή των εθνικών γαιών, αν και αποτελούσε γενική επιθυμία, συναντούσε πολλά προβλήματα στην πράξη. Το 1836 εκδόθηκε νόμος, με τον οποίο θα συντασσόταν κτηματολόγιο που θα στηριζόταν κατά ένα μεγάλο μέρος στις δηλώσεις των ιδιοκτητών επί των τίτλων ιδιοκτησίας τους, βασιζόμενο σε υποθετικούς υπολογισμούς. Ωστόσο, ο νόμος αυτός δεν εφαρμόστηκε ποτέ. Εν τέλει, η οριστική αντιμετώπιση του προβλήματος επήλθε με νομοθετικές ρυθμίσεις την περίοδο 1870-1871. Στόχος των νομοθετημάτων ήταν να εξασφαλιστούν κατά προτεραιότητα οι ακτήμονες χωρικοί, με την παροχή γης, απαραίτητης για την επιβίωσή τους. Έτσι, η κυβέρνηση Κουμουνδούρου ήταν αυτή που προχώρησε στη διανομή των εθνικών γαιών στους καλλιεργητές το 1871 κι έτσι προέκυψε η Πρώτη Αγροτική Μεταρρύθμιση. Σύμφωνα με τις σχετικές νομοθετικές ρυθμίσεις, οι δικαιούχοι αγρότες μπορούσαν να αγοράσουν όση γη ήθελαν, με ανώτατο όριο τα 80 στρέμματα για ξηρικά εδάφη και τα 40 στρέμματα για αρδευόμενα. Από το 1870

ως το 1911 διανεμήθηκαν 2.650.000 στρέμματα σε 350.000 κλήρους, μέσης έκτασης 7,5 στρεμμάτων, πράγμα που δείχνει ότι ο πολυτεμαχισμός της γης υπήρξε ήδη μεγάλος. Επρόκειτο για σημαντική διανομή καλλιεργήσιμων γαιών, ιδιαίτερα αν συγκριθεί με τα 600.000 στρέμματα εθνικών γαιών που είχαν διανεμηθεί τα προηγούμενα χρόνια, από το 1833 μέχρι το 1870.

(Δημοπούλου, 2014), (Καλλιβρετάκης, 1990).

5.1.2. Ο αγροτικός χώρος κατά τον 20^ο αιώνα

Με τη Βιομηχανική Επανάσταση η κυριαρχία του αγροτικού χώρου στον παραγωγικό και κοινωνικό τομέα άρχισε προοδευτικά να υποχωρεί. Η προοδευτική διανομή των εθνικών γαιών που προέκυψε από τον επαναστατικό αγώνα του 1821-1828 δημιούργησε πλήθος αγροτών με μικρές ή μεσαίες ιδιοκτησίες. Αργότερα όμως, η διεύρυνση του ελληνικού κράτους (προσάρτηση της Θεσσαλίας, της Άρτας και αργότερα το 1912 της Μακεδονίας) έφερε στο προσκήνιο το ζήτημα της μεγάλης ιδιοκτησίας. Έτσι, άνοιξαν οι δρόμοι για την Αγροτική Μεταρρύθμιση, δηλαδή την κατάργηση των μεγάλων ιδιοκτησιών και την κατάτμηση των αξιοποιήσιμων εδαφών σε μικρές παραγωγικές μονάδες, οικογενειακού χαρακτήρα, που ανταποκρίνονταν καλύτερα στις νέες παραγωγικές και κοινωνικές συνθήκες. Από το 1880 ως το 1922 υπήρξε έξαρση του αγροτικού ζητήματος. Τα «τσιφλίκια» της Θεσσαλίας αγοράστηκαν από πλούσιους Έλληνες του εξωτερικού, οι οποίοι άσκησαν πολιτικές και κοινωνικές πιέσεις για να κερδοσκοπήσουν. Οι πρακτικές αυτές δημιούργησαν εντάσεις και οδήγησαν στην ψήφιση νόμων το 1907, οι οποίοι επέτρεπαν στην εκάστοτε ελληνική κυβέρνηση να απαλλοτριώνει μεγάλες ιδιοκτησίες, ώστε να μπορεί να τις διανέμει σε ακτήμονες. Το αποφασιστικό βήμα προς την ολοκλήρωση της Αγροτικής Μεταρρύθμισης έγινε το 1917 από την κυβέρνηση του Ελευθερίου Βενιζέλου, όπου κατέστη δυνατή η απαλλοτρίωση των μεγάλων αγροτικών ιδιοκτησιών, που προέκυψε ως ανάγκη για την αποκατάσταση των προσφύγων από την Ανατολική Ρωμυλία. Η αναδιανομή που έγινε έφτασε στο 85% των καλλιεργήσιμων εκτάσεων στη Μακεδονία και στο 68% στη Θεσσαλία. Στο σύνολο της καλλιεργήσιμης γης της χώρας το ποσοστό αυτό ανήλθε σε 40%. Από το 1922 ως το 1950 προχώρησε η Δεύτερη Αγροτική Μεταρρύθμιση λόγω της Συνθήκης της Λωζάννης με την υποχρεωτική ανταλλαγή πληθυσμών. Μετά το 1922 παρατηρείται η απαλλοτρίωση των τσιφλικιών. Οι κυβερνήσεις των ετών 1922-1926 προχώρησαν σε μια σειρά από τροποποιήσεις του νομοθετικού πλαισίου, προς την κατεύθυνση της άμεσης διάλυσης των μεγάλων και μεσαίων κτημάτων και τη διανομή τους σε ακτήμονες και πρόσφυγες καλλιεργητές. Το 1923 η κυβέρνηση Πλαστήρα προχώρησε στη διανομή 18.000.000 στρεμμάτων (50% της γεωργικής γης) σε 300.000 ακτήμονες. Οι διαδικασίες της Αγροτικής Μεταρρύθμισης μπήκαν στην τελική ευθεία το 1927, όταν στο νέο Σύνταγμα μπήκε ειδική διάταξη σχετικά με την απαλλοτρίωση (άρθρο 119). Στο τέλος του 1930 το 40% των γεωργών ήταν

νέοι ιδιοκτήτες. Μέχρι το 1931, 1.623 μεγάλες ιδιοκτησίες είχαν παραχωρηθεί σε 115.000 οικογένειες γηγενών ακτημόνων/μικροκαλλιεργητών και 145.000 οικογένειες προσφύγων αποκαταστάθηκαν αγροτικά. Η αγροτική Ελλάδα είχε μετατραπεί σε έθνος μικροϊδιοκτητών. Κατά τη διάρκεια του Δευτέρου Παγκοσμίου Πολέμου υπήρξε ανακοπή της πορείας που αφορούσε στην εξέλιξη της ιδιοκτησίας της γης. Στη συνέχεια, η Μεταπολίτευση σηματοδοτεί ευρύτατες ανακατατάξεις και στον αγροτικό τομέα. Στη δεκαετία του '80 εφαρμόστηκε μια «φιλοαγροτική» πολιτική ενίσχυσης των εισοδημάτων των αγροτών μέσω των συνεταιρισμών. Ωστόσο, κατά τη δεκαετία του '90 παρουσιάζεται στροφή σε πιο ρεαλιστικούς στόχους και σήμερα ο αγροτικός χώρος φαίνεται να έχει επηρεαστεί από τις ραγδαίες οικονομικές εξελίξεις που ταλανίζουν τη χώρα.

(Μανσόλα, 1867), (Μαραβέγιας, 1992)

5.2. Μέτρα πολιτικής γης στον αγροτικό χώρο

Μερικά από τα πιο σημαντικά μέτρα πολιτικής γης που εφαρμόστηκαν στον αγροτικό χώρο από τα τέλη του 19^{ου} αιώνα μέχρι και σήμερα είναι αρχικά η διανομή αγροτικών εκτάσεων και μετέπειτα ο αναδασμός.

5.2.1. Διανομή

Παλαιότερα, με τη διανομή γης γινόταν η παραχώρηση σε ακτήμονες τίτλων κυριότητας επί κρατικής ή απαλλοτριωμένης γης. Στην Ελλάδα οι διανομές των εκτάσεων ξεκίνησαν το 1871. Κύρια αιτία όμως της σύνταξης διανομών υπήρξε η αποκατάσταση των προσφύγων της Μικρασιατικής καταστροφής του 1922. Οι διανομές διακρίνονται σε διανομές οικοπέδων σε συνοικισμούς (κλίμακα 1:1000) και σε διανομές κληροτεμαχίων σε αγροτικές εκτάσεις (κλίμακα 1:5000). Η διανομή δημιουργεί γεωτεμάχια, τα οποία διαφέρουν μεταξύ τους ως προς τα όρια, το σχήμα, την επιφάνεια, την παραγωγικότητα και τη δυνατότητα διοικητικού ελέγχου. Η διαμόρφωση των αγροτεμαχίων γίνεται είτε με συστηματική είτε με φυσική διανομή. Στην πρώτη περίπτωση, η διανομή βασίζεται σε κανονικοποιημένη και σχετικά ομοιόμορφη κατάτμηση της γης σε συνήθως μεγάλες επιφάνειες, με απώτερο στόχο την ευκολότερη, καλύτερη και παραγωγικότερη εκμετάλλευση του αγροτικού χώρου, αλλά και του αποτελεσματικότερου διοικητικού ελέγχου. Στη δεύτερη περίπτωση, έχουμε τη φυσική διανομή της αγροτικής γης, η οποία προέκυψε ιστορικά από αλληλοδιάδοχες κατατμήσεις, απ' την οποία τα όρια των γεωτεμαχίων έχουν προκύψει αυθαίρετα ή αντιπροσωπεύουν φυσικά χαρακτηριστικά, όπως ρέματα ή κορυφογραμμές (Ζεντέλης, 2011). Στον ελλαδικό χώρο υπερισχύει η δεύτερη κατηγορία διανομής λόγω πολιτικών, κοινωνικών και οικονομικών αιτιών. Τέλος, οι διανομές διακρίνονται σε οριστικές, προσωρινές και συμπληρωματικές, ενώ η κάθε μία χαρακτηρίζεται από την εποχή που έχει γίνει, για παράδειγμα ΠΟΡΟΣ (ΕΒΡΟΥ) ΣΥΜΠΛΗΡΩΜΑΤΙΚΗ ΔΙΑΝΟΜΗ ΕΤΟΥΣ 1953.

➤ Σύνταξη και εκτέλεση διανομής

Η επιλογή μιας περιοχής για την πραγματοποίηση μιας διανομής βασίζεται σε ένα σύνολο κριτηρίων. Αρχικά, θα πρέπει να βρίσκεται κοντά σε κάποιο αστικό κέντρο και σε μια οδική αρτηρία, δεύτερον να είναι πλούσια σε φυσικό κάλλος και σε φυσικά διαθέσιμα και, τέλος, να διαθέτει εύφορα εδάφη. Αφού επιλεγθεί η περιοχή, ιδρύεται ένα τριγωνομετρικό δίκτυο, που τις περισσότερες φορές είναι εξαρτημένο από το κρατικό. Σημειώνεται, ότι την περίοδο που γινόνταν οι περισσότερες διανομές, το Υπουργείο Γεωργίας (Τοπογραφική Υπηρεσία) χρησιμοποιούσε τα λεγόμενα μικρά φύλλα HATT, τα οποία έχουν διαστάσεις 6'x6' και προκύπτουν από την υποδιαίρεση ενός μεγάλου φύλλου HATT (κλίμακα 1:100.000). Στο επόμενο βήμα απαιτείται η ίδρυση πολυγωνομετρικού δικτύου και η αποτύπωση της υπάρχουσας κατάστασης. Η τελευταία, επιτυγχάνεται με την εγκατάσταση πρωτευουσών και δευτερευουσών οδεύσεων, ενώ πρέπει να είναι όσο το δυνατόν πιο λεπτομερής, τόσο οριζοντιογραφικά όσο και υψομετρικά. Συγκεκριμένα, είναι αναγκαίο να γίνει αποτύπωση των ορίων ιδιοκτησιών, των καλλιεργειών, των τεχνικών έργων, των υπάρχοντων δρόμων, των ρεμάτων καθώς και κάθε άλλου χρήσιμου στοιχείου. Το κύριο στάδιο στη μελέτη σύνταξης μιας διανομής είναι η διανομή των αγροτεμαχίων στο χάρτη. Προκειμένου αυτή να επιτευχθεί, γίνεται χάραξη στην περιοχή μιας γραμμής Βορρά-Νότου. Με βάση αυτήν προσδιορίζονται οι άξονες της διανομής και οι άξονες ενός συστήματος δρόμων, παράλληλων και κάθετων προς τη γραμμή αυτή, και είναι συνήθως ισοπλατείς. Απόρροια των παραπάνω είναι η δημιουργία ζωνών διανομής ίσου πλάτους, μέσα στα οποία προσδιοριζόταν τα αγροτεμάχια από τα αναλυτικά γεωμετρικά τους στοιχεία (αποστάσεις). Το τελευταίο στάδιο στην ολοκλήρωση της διανομής είναι η εφαρμογή του χάρτη στο έδαφος. Βάσει των στοιχείων των τριγωνισμών και της πολυγωνομετρίας γίνεται αρχικά η χάραξη των δρόμων και των αξόνων διανομής με κλασικές τοπογραφικές μεθόδους. Στη συνέχεια, σημαίνονται στο έδαφος με ορόσημα (κολωνάκια από σκυρόδεμα), στα οποία δίνονται αναλυτικές τιμές, ώστε να σχεδιαστούν στο διάγραμμα της διανομής. Οι κορυφές των κληροτεμαχίων δεν υπολογίζονται αναλυτικά αλλά με βάση τα γεωμετρικά τους στοιχεία, ενώ σε περίπτωση ακανόνιστων οριογραμμών (π.χ ρέματα) τα όρια των ιδιοκτησιών ακολουθούν την ακανόνιστη οριογραμμή, όπως σχεδιάστηκε στο χάρτη της διανομής.



Εικόνα 5-1: Απόσπασμα οριστικής διανομής έτους 1932 στον Πόρο.

Στο παραπάνω απόσπασμα οριστικής διανομής φαίνονται τα γεωμετρικά στοιχεία των αγροτεμαχίων, καθώς και τα ορόσημα (μαύρο τετραγωνάκι), τα οποία συνήθως τοποθετούνται στις άκρες των αξόνων και των δρόμων διανομής, σε ενδιάμεσα σημεία πυκνώσεως και, τέλος, στις τομές τους με άλλους δρόμους μη παράλληλους προς τους βασικούς.

Τέλος, πρέπει να σημειωθεί ότι οι διαφορές από διανομή σε διανομή είναι έντονες σε σχέση με την ακρίβεια των αναλυτικών στοιχείων που παρέχονται, η οποία είναι συσχετισμένη με τη χρονική εποχή που πραγματοποιήθηκε κάθε διανομή. Η χρονική εποχή καθορίζει και το σύστημα συντεταγμένων των τριγωνομετρικών σημείων, στο οποίο βασίστηκε η σύνταξη της διανομής του τότε Υπουργείου Γεωργίας, ενώ κάθε επόμενη εργασία πάνω στην υπάρχουσα διανομή (π.χ. συμπληρωματική διανομή) ακολουθεί το αρχικό σύστημα, ώστε να είναι εύκολος ο συσχετισμός των συνοριακών γραμμών και η παράστασή τους σε εξαρτημένα μεταξύ τους διαγράμματα.

(Βέργος, 2010-2011)

5.2.2. Αναδασμός

Αναδασμός ονομάζεται η διαδικασία, κατά την οποία η αρμόδια κρατική αρχή αποσπά όλες τις έγγειες ιδιοκτησίες ενός καθορισμένου τόπου από τους κυρίους τους, χαράσσει νέα όρια στο εσωτερικό τους και τελικά τις επαναδιανέμει μεταξύ των πρότερων ιδιοκτητών, με νέους τίτλους κυριότητας και με τη μορφή ενός ενιαίου τεμαχίου. Αποτελεί μια μέθοδο που εφαρμόζεται κατά κύριο λόγο σε

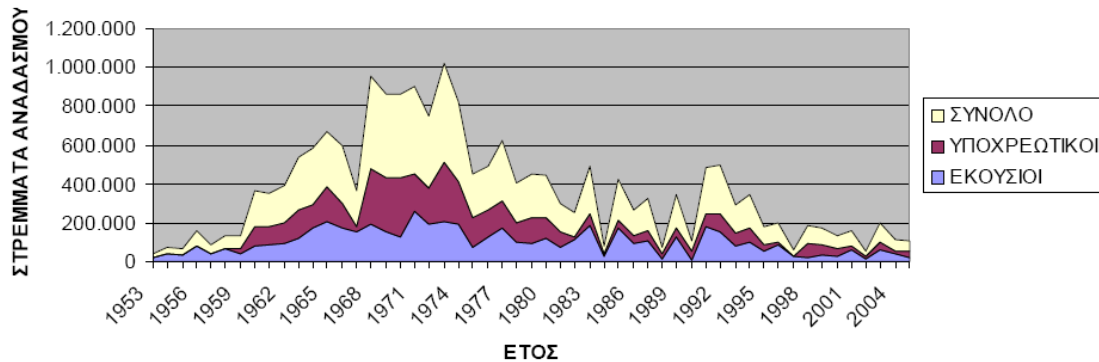
αγροτικές περιοχές, με σκοπό τον εξορθολογισμό και την τόνωση της γεωργικής παραγωγής.

Πίνακας 5-1: Χαρακτηριστικά αγροτικού κλήρου πριν κα μετά τον αναδασμό.

ΠΡΙΝ ΤΟΝ ΑΝΑΔΑΣΜΟ	ΜΕΤΑ ΤΟΝ ΑΝΑΔΑΣΜΟ
<ul style="list-style-type: none"> • Πολλά και σκόρπια τεμάχια γης 	<ul style="list-style-type: none"> • Συγκεντρωμένα τεμάχια γης
<ul style="list-style-type: none"> • Τεμάχια γης με ακανόνιστα σχήματα 	<ul style="list-style-type: none"> • Τεμάχια γης με κανονικά σχήματα
<ul style="list-style-type: none"> • Συνιδιοκτησία και δυαδική/ πολλαπλή μορφή ιδιοκτησίας 	<ul style="list-style-type: none"> • Επίλυση προβλημάτων συνιδιοκτησίας και δυαδικής/ πολλαπλής μορφής ιδιοκτησίας
<ul style="list-style-type: none"> • Έλλειψη αγροτικών δρόμων 	<ul style="list-style-type: none"> • Αγροτικοί δρόμοι σε όλα τα τεμάχια γης

Πηγή: Μάρκου,2009

Ο αναδασμός στην Ελλάδα άρχισε να τίθεται σε εφαρμογή από τις αρχές της δεκαετίας του 1950. Μέσω του αναδασμού επιτυγχάνεται η ποιοτική αναβάθμιση του αγροτικού βίου, αμβλύνεται το φαινόμενο της αστικοποίησης, δημιουργείται ανταγωνιστική γεωργία και ευδοκιμεί η αγροτική γη. Επιπρόσθετα πλεονεκτήματα αυτής της πράξης εφαρμογής είναι η μείωση του κόστους παραγωγής αλλά και η αύξηση τόσο της παραγωγικότητας όσο και του γεωργικού εισοδήματος. Με αυτό τον τρόπο επιλύονται τα χρόνια προβλήματα ιδιοκτησίας που ταλανίζουν τον αγροτικό κλήρο. Επιπλέον, οι γεωργικές μονάδες λειτουργούν εύρυθμα, οι καλλιέργειες αναδιαρθρώνονται και εκμηχανίζονται. Ο αναδασμός πρέπει να γίνεται υπό σωστές προϋποθέσεις και συγκεκριμένα με την εκούσια βούληση των δικαιούχων, ώστε να μη λάβει χαρακτηριστικά απαλλοτρίωσης. Είναι επιτακτική ανάγκη να συμμετέχουν ενεργά τόσο οι δικαιούχοι όσο και οι αγρότες. Ο αναδασμός αποτελείται από δυο είδη: απ' τον εκούσιο-προαιρετικό, κατά τη διάρκεια του οποίου υποβάλλονται αιτήσεις, και απ' τον υποχρεωτικό-αναγκαστικό, ο οποίος επιβάλλεται για την εκτέλεση εγγειοβελτιωτικών έργων και για κτήματα των οποίων η διανομή δεν έχει οριστικοποιηθεί. Επίσης, ο υποχρεωτικός αναδασμός διενεργείται σε παραμεθόριες περιοχές. Για τη διενέργεια αναδασμού εκδίδεται απόφαση του νομάρχη στην περιφέρεια του οποίου ανήκει η αναδιανεμητέα περιοχή, η οποία έπειτα δημοσιεύεται στην Εφημερίδα της Κυβερνήσεως. Ωστόσο, αναδύονται και κάποια μειονεκτήματα αυτού του μέτρου, όπως είναι η χρονοβόρα διαδικασία, η μη τήρηση των κτηματολογικών δεδομένων, η μη βιωσιμότητα του κλήρου αλλά και η απροθυμία των ιδιοκτητών να εγκατασταθούν στα νέα αγροτεμάχια.



Εικόνα 5-2: Περαιτωθέντες αναδασμοί ανά έτος 1953-2005.

Πηγή: Μάρκου, 2009

➤ Διαδικασία υλοποίησης αναδασμού

Η διαδικασία υλοποίησης αναδασμού σε μια περιοχή ξεκινάει με την κτηματογράφηση της περιοχής και την κοινοποίηση των στοιχείων για είκοσι μέρες, προκειμένου να γίνουν ενστάσεις από τους ενδιαφερόμενους. Μόλις οριστικοποιηθούν οι κτηματολογικοί πίνακες και τα διαγράμματα, ακολουθεί η κατηγοριοποίηση των εδαφών με τον καθορισμό των αντίστοιχων συντελεστών αξίας. Συνεπώς, στην υπό αναδασμό περιοχή δημιουργούνται ζώνες αξιών βάσει των οποίων υπολογίζονται τα πλασματικά εμβαδά κάθε ιδιοκτήτη. Τα πλασματικά εμβαδά αποτελούν το γινόμενο του εμβαδού μια ιδιοκτησίας με τον αντίστοιχο συντελεστή αξίας. Στη συνέχεια, αφού υπολογισθεί η συνολική πλασματική επιφάνεια ανά ιδιοκτήτη, υπολογίζεται το συνολικό πλασματικό εμβαδόν προ αναδασμού. Το τελευταίο υπολογίζεται προσθέτοντας στο συνολικό πλασματικό εμβαδό των ιδιοκτητών, το πλασματικό εμβαδό των κοινοχρήστων προ του αναδασμού και αφαιρώντας το πλασματικό εμβαδό του κοινόχρηστου χώρου μετά τον αναδασμό. Έπειτα, υπολογίζεται ο συντελεστής λ σε όλη την περιοχή: $\lambda = \frac{A}{\alpha_{ολ}}$,

όπου A η συνολική πλασματική επιφάνεια προς διανομή, ενώ $\alpha_{ολ}$ η συνολική πλασματική επιφάνεια των ιδιοκτητών προ του αναδασμού. Με βάση τον συντελεστή λ υπολογίζεται η συνολική πλασματική επιφάνεια ανά ιδιοκτήτη μετά τον αναδασμό, πολλαπλασιάζοντας τον με την αντίστοιχη συνολική πλασματική επιφάνεια ανά ιδιοκτήτη προ του αναδασμού: $A_i = \alpha_i \cdot \lambda$. Πρέπει να σημειωθεί ότι ο συντελεστής λ έχει τιμή μικρότερη ή μεγαλύτερη της μονάδας, αναλόγως των κοινόχρηστων χώρων που υπάρχουν μετά τον αναδασμό. Τέλος, η χωροθέτηση των ιδιοκτησιών πραγματοποιείται, αφού γίνει αναγωγή των πλασματικών εμβαδών σε πραγματικά, μέσω των αντίστοιχων συντελεστών αξίας.

(Πότσιου, 2008)

5.3. Εθνικό Κτηματολόγιο και Χαρτογράφηση (ΕΚΧΑ)

Το Εθνικό Κτηματολόγιο αποτελεί, σύμφωνα με το νόμο 2664/1998, σύστημα οργανωμένων σε κτηματοκεντρική βάση νομικών, τεχνικών και άλλων πρόσθετων πληροφοριών για όλα τα ακίνητα της επικράτειας. Στο Κτηματολόγιο καταχωρούνται νομικές και τεχνικές πληροφορίες που αποσκοπούν στον ακριβή καθορισμό των ορίων των ακινήτων και στη δημοσιότητα των εγγραφτέων στα κτηματολογικά βιβλία δικαιωμάτων βαρών, με τρόπο που διασφαλίζει τη δημόσια πίστη, προστατεύοντας κάθε καλόπιστο συναλλασσόμενο που στηρίζεται στις κτηματολογικές εγγραφές.

Το Κτηματολόγιο διέπεται από τις ακόλουθες αρχές:

- Την αρχή της κτηματοκεντρικής οργάνωσης των κτηματολογικών πληροφοριών, η οποία απαιτεί τη σύνταξη, τήρηση και διαρκή ενημέρωση κτηματολογικών διαγραμμάτων.
- Την αρχή του ελέγχου της νομιμότητας των τίτλων και των λοιπών αναγκαίων στοιχείων για την αποδοχή της αίτησης εγγραφής στα κτηματολογικά βιβλία.
- Την αρχή της διασφάλισης της τάξης των κτηματολογικών εγγραφών, ανάλογα με το χρόνο υποβολής της σχετικής αίτησης (αρχή της χρονικής προτεραιότητας).
- Την αρχή της δημοσιότητας των κτηματολογικών βιβλίων.
- Την αρχή της διασφάλισης της δημόσιας πίστης, ώστε να προστατεύεται κάθε καλόπιστος συναλλασσόμενος που στηρίζεται στις κτηματολογικές εγγραφές.
- Την αρχή της καταλληλότητας του Κτηματολογίου ως συστήματος δεκτικού καταχώρισης και πρόσθετων κατηγοριών πληροφοριών σε οποιονδήποτε χρόνο στο μέλλον (αρχή του ανοικτού Κτηματολογίου).

Τα στοιχεία που πρέπει να τηρούνται από τα Κτηματολογικά Γραφεία είναι τα κτηματολογικά διαγράμματα για όλη την κτηματογραφημένη περιοχή που εμπίπτει στην αρμοδιότητα του οικείου Κτηματολογικού Γραφείου, καθώς και το κτηματολογικό βιβλίο. Το τελευταίο, συντίθεται από τα κτηματολογικά φύλλα. Σε ένα κτηματολογικό φύλλο αντιστοιχεί ένα ακίνητο ή αυτοτελές ιδιοκτησιακό αντικείμενο, ενώ οι εγγραφές ενός φύλλου αποτελούν τα στοιχεία που καταγράφονται για το εν λόγω ιδιοκτησιακό αντικείμενο. Τα απαραίτητα στοιχεία και πράξεις που πρέπει να καταχωρούνται στα κτηματολογικά φύλλα αναγράφονται λεπτομερώς στο άρθρο 11 και 12 του νόμου 2664/1998. Η ψηφιακή μορφή του κτηματολογικού βιβλίου είναι η περιγραφική βάση δεδομένων του ΕΚΧΑ, ενώ η

χωρική πληροφορία αποθηκεύεται σε χωρική βάση δεδομένων. Σημειώνεται ότι η έντυπη μορφή των κτηματολογικών διαγραμμάτων γίνεται βάσει της χωρικής βάσης δεδομένων. Η σύνδεση του γεωμετρικού και περιγραφικού μέρους του Κτηματολογίου επιτυγχάνεται μέσω του Κωδικού Αριθμού Εθνικού Κτηματολογίου (ΚΑΕΚ). Ο τελευταίος, είναι ο μοναδικός για κάθε ακίνητο 12ψήφιος αριθμός, με τον οποίο γίνεται εύκολα και γρήγορα η αναζήτηση στη βάση δεδομένων του Κτηματολογίου. Κάθε ένα από τα ψηφία που τον αποτελούν προσδιορίζει συγκεκριμένη πληροφορία, πχ Νομό στον οποίο βρίσκεται το γεωτεμάχιο, Δήμο, Δημοτικό Διαμέρισμα ή Κοινότητα, τον «κτηματολογικό τομέα», την «κτηματολογική ενότητα» και, τέλος, τον αύξοντα αριθμό του γεωτεμαχίου εντός της ενότητας.

5.3.1. Ένταξη Διανομών και Αναδασμών στο ΕΚΧΑ

Οι εργασίες των Αναδασμών και Διανομών περιλαμβάνουν τοπογραφικά διαγράμματα και συνοδευτικούς κτηματολογικούς πίνακες με τα όρια των ιδιοκτητών / κληρούχων. Τα διαγράμματα προσδιορίζουν τη θέση, τα όρια, το μέγεθος και τη χωρική κατανομή των γεωτεμαχίων και λοιπών εκτάσεων των Αγροκτημάτων εντός των οποίων πραγματοποιήθηκαν Αναδασμοί και Διανομές. Οι κτηματολογικοί πίνακες που συνοδεύουν τα διαγράμματα καθορίζουν τα δικαιώματα κυριότητας των απεικονιζόμενων στα διαγράμματα γεωτεμαχίων και λοιπών εκτάσεων και τα στοιχεία των ιδιοκτητών κατά την ημερομηνία κύρωσης της πράξης του Αναδασμού ή της Διανομής. Συνεπώς, οι τίτλοι κυριότητας που αποκτώνται από Αναδασμό ή Διανομή αποτελούν πρωτότυπους τίτλους κτήσης και δεν επιδέχονται αμφισβήτησης.

Η ένταξη μια περιοχής στο κτηματολόγιο συνεπάγεται και την καταχώριση ενός νέου ΚΑΕΚ για κάθε αγροτεμάχιο ή ιδιοκτησία. Επιπροσθέτως, συμπληρώνονται και ενημερώνονται πλήρως τα κτηματολογικά βιβλία. Συνεπώς, ακόμα και στην περίπτωση που έγινε Αναδασμός ή Διανομή στη περιοχή, δίνονται στα αντίστοιχα αγροτεμάχια νέα ΚΑΕΚ, ενώ οι αντίστοιχες πράξεις καταχωρούνται στα αντίστοιχα κτηματολογικά φύλλα.

5.3.2. Περιοχή με Αναδασμό ή Διανομή μη ενταγμένης στο ΕΚΧΑ

Σε περίπτωση που μια περιοχή δεν έχει ενταχθεί στο ΕΚΧΑ και έχει υλοποιηθεί Αναδασμός ή Διανομή, τότε ιδιοκτησιακά στοιχεία σχετικά με τα αγροτεμάχια της περιοχής βρίσκονται στις κατά τόπους Διευθύνσεις Γεωργίας. Οι τελευταίες, τηρούν και ενημερώνουν τους κτηματολογικούς πίνακες των Αναδασμών και Διανομών με τις μεταβολές που επέρχονται με την πάροδο του χρόνου στα ονόματα των κυρίων των γεωτεμαχίων. Σε κάθε περίπτωση πρέπει να ελέγχεται και το συμβόλαιο μεταβίβασης του κάθε ιδιοκτήτη, προκειμένου να βεβαιωθεί η κυριότητα επί του ακινήτου.

(ΚΤΗΜΑΤΟΛΟΓΙΟ

Α.Ε.,

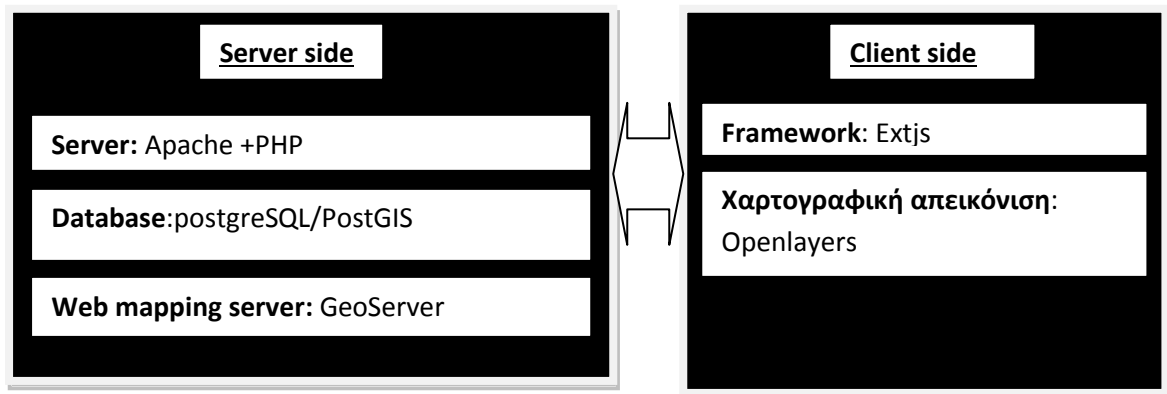
2012)

6. ΣΧΕΔΙΑΣΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ

Στο πλαίσιο της παρούσας διπλωματικής εργασίας δημιουργήθηκε ένα διαδικτυακό σύστημα, το οποίο στηρίχθηκε σε ελεύθερα λογισμικά και τεχνολογίες ανοικτού κώδικα και αποτελείται από τρία επιμέρους υποσυστήματα. Το πρώτο χρησιμοποιείται για την εύκολη και άμεση εγκατάσταση των απαραίτητων στοιχείων του συστήματος. Το δεύτερο αποτελεί ένα σύστημα διαχείρισης περιεχομένου και το τρίτο δημιουργήθηκε για τις ανάγκες της παρούσας εργασίας και συγκεκριμένα αφορά τη διαδικτυακή αίτηση στοιχείων που σχετίζονται είτε με την πράξη του αναδασμού είτε με τη διανομή για ένα συγκεκριμένο αγροτεμάχιο.

6.1. Αρχιτεκτονική Συστήματος

Η ανάπτυξη της εφαρμογής βασίστηκε στο framework Extjs και συνεπώς η βασική γλώσσα υλοποίησής της είναι η JavaScript. Ακόμη, ως βάση δεδομένων χρησιμοποιήθηκε η PostgreSQL/PostGIS, ενώ για τη λήψη δεδομένων από αυτήν χρησιμοποιήθηκε η server-side γλώσσα προγραμματισμού PHP. Για τη δημοσίευση των χωρικών δεδομένων στο Διαδίκτυο χρησιμοποιήθηκε ως Web-mapping server ο GeoServer, ενώ για τη χαρτογραφική απεικόνισή τους η Openlayers. Τέλος, ως εξυπηρετητής Διαδικτύου χρησιμοποιήθηκε ο Apache HTTP server.

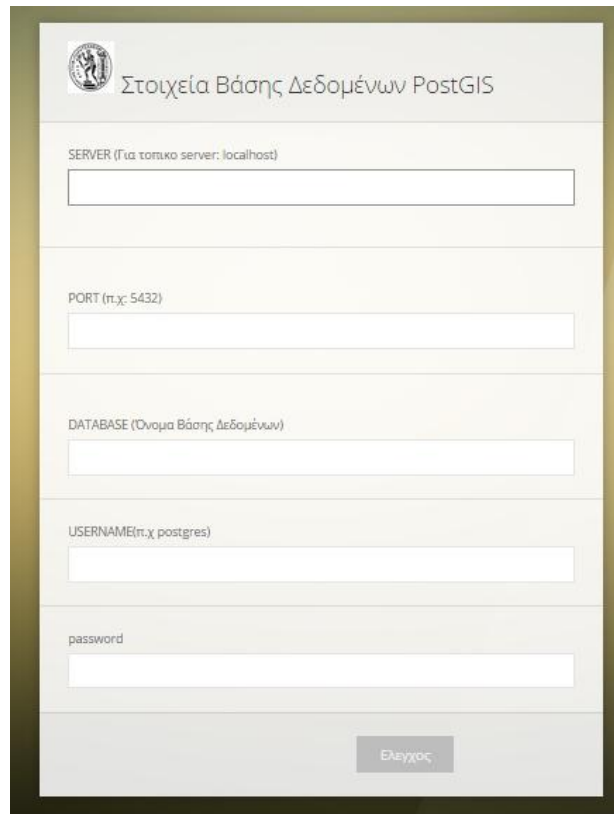


Εικόνα 6-1: Αρχιτεκτονική συστήματος.

6.2. Εγκατάσταση Συστήματος

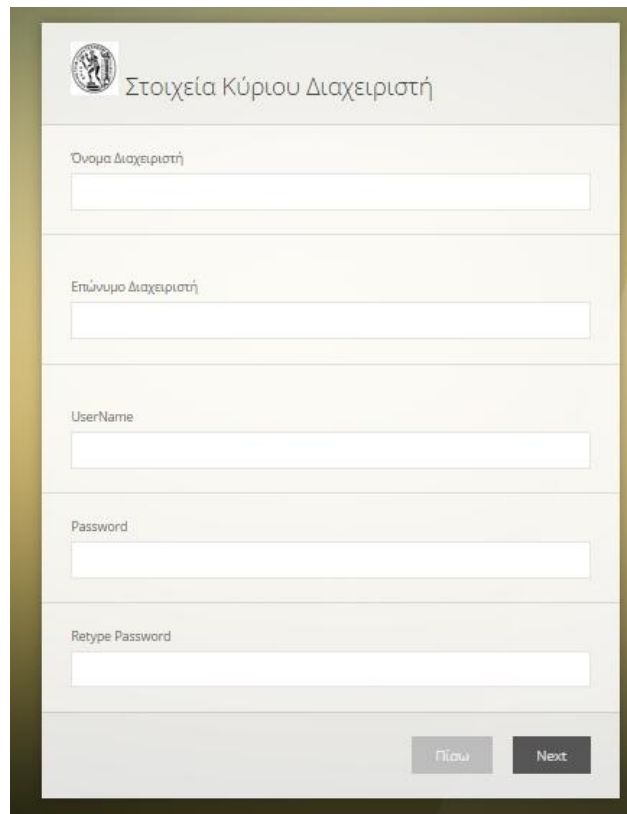
Βασικά προαπαιτούμενα πριν την έναρξη της εγκατάστασης της εφαρμογής είναι η εγκατάσταση της βάσης δεδομένων PostgreSQL μαζί με τις επιπρόσθετες χωρικές δυνατότητες που της αποδίδει η PostGIS. Επιπροσθέτως, θα πρέπει να δημιουργηθεί και μια κατάλληλη βάση δεδομένων με ενσωματωμένες χωρικές λειτουργίες. Τέλος, θα πρέπει να γίνει εγκατάσταση του GeoServer μαζί με την επιπρόσθετη λειτουργία (plugin): **geoserver-xx-SNAPSHOT-printing-plugin**, προκειμένου να είναι δυνατή η εκτύπωση δεδομένων σε pdf.

Μόλις ολοκληρωθούν τα παραπάνω βήματα, τότε ανοίγουμε την εφαρμογή που βρίσκεται στο φάκελο install, ώστε να ξεκινήσει η βασική εγκατάσταση.



Εικόνα 6-2: Καταχώρηση στοιχείων βάσης δεδομένων PostGIS.

Στην πρώτη σελίδα που εμφανίζεται καλείται ο χρήστης να συμπληρώσει τα στοιχεία σύνδεσης με τη βάση δεδομένων PostGIS. Συγκεκριμένα, πρέπει να συμπληρωθούν ο server, η θύρα και το όνομα της βάσης δεδομένων που έχει δημιουργήσει στο προηγούμενο βήμα. Τέλος, πρέπει να συμπληρωθούν το username και το password, τα οποία επιτρέπουν τη σύνδεση με την PostGIS. Στο κάτω μέρος της σελίδας υπάρχει το κουμπί «έλεγχος», με το οποίο ελέγχεται η πληρότητα και η ορθότητα των στοιχείων. Αφού ολοκληρωθεί ο έλεγχος, δημιουργούνται αυτόματα όλοι οι απαραίτητοι πίνακες στη βάση δεδομένων.



Εικόνα 6-3: Καταχώρηση στοιχείων κύριου διαχειριστή.

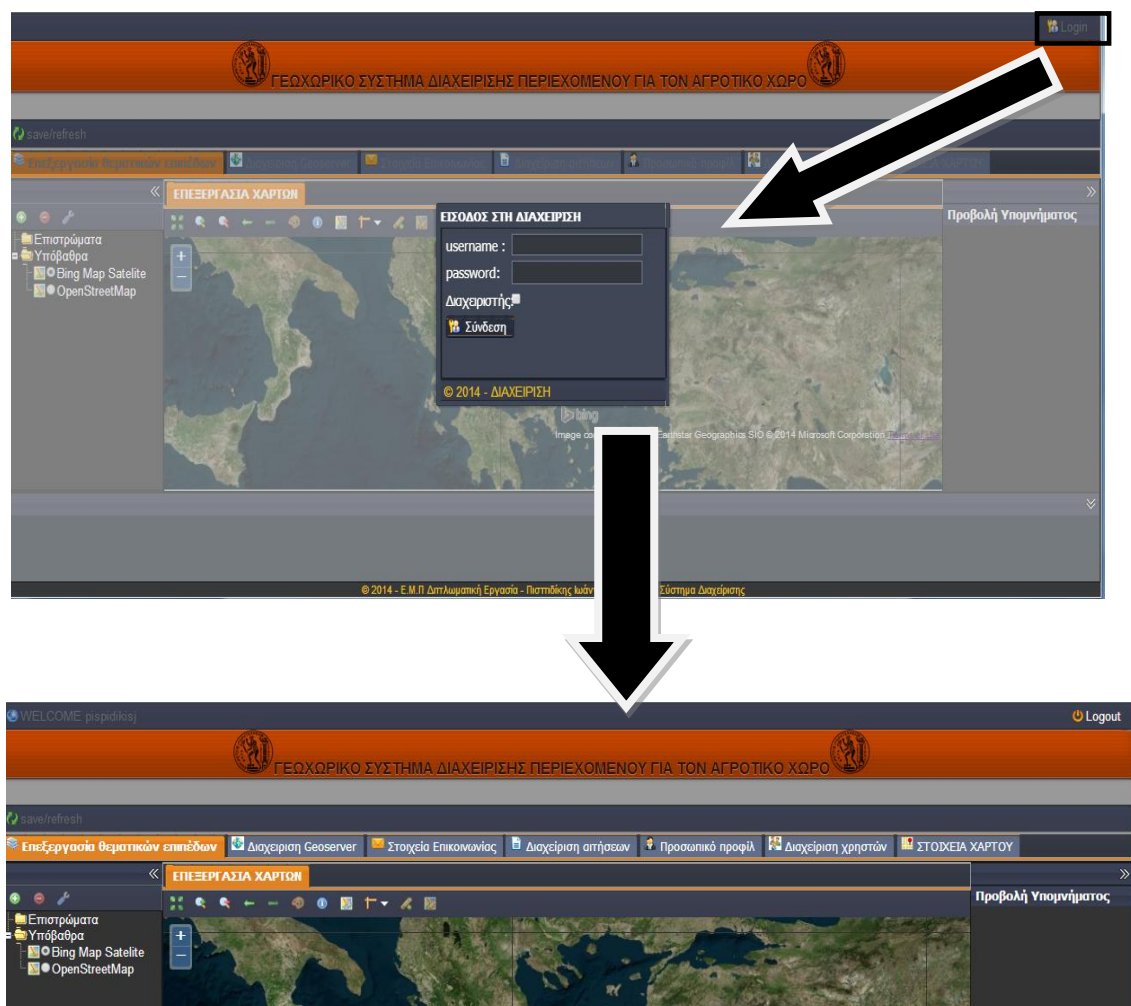
Στη δεύτερη σελίδα συμπληρώνονται τα στοιχεία του κύριου διαχειριστή προκειμένου να έχει πρόσβαση στο σύστημα διαχείρισης περιεχομένου. Τέλος, όπως προαναφέρθηκε και στην αρχιτεκτονική της εφαρμογής, υπάρχουν δύο εξυπηρετητές. Στον έναν, τον Apache, θα είναι εγκατεστημένη όλη η εφαρμογή, ενώ στο δεύτερο θα υπάρχει ο εξυπηρετητής χαρτών GeoServer. Προκειμένου να μην υπάρχουν προβλήματα προσβασιμότητας των δεδομένων του Geoserver, είναι απαραίτητο να γραφτεί ο παρακάτω κώδικας εντός του αρχείου httpd.conf του Apache:

```
<VirtualHost *:80>
ProxyRequests off
ProxyPreserveHost On
<Proxy *>
Order deny,allow
Allow from all
</Proxy>
ProxyPass /geoserver http://localhost:8080/geoserver
ProxyPassReverse /geoserver http://localhost:8080/geoserver
</VirtualHost>
```

Συγκεκριμένα, μέσω του παραπάνω κώδικα δημιουργείται ένας Διακομιστής μεσολάβησης (proxy server) για τον GeoServer, εντός του εξυπηρετητή Διαδικτύου Apache της εφαρμογής.

6.3. Σύστημα Διαχείρισης Περιεχομένου

Ο βασικός κορμός του προγράμματος είναι το σύστημα διαχείρισης περιεχομένου. Το τελευταίο, αποτελείται από επτά συνολικά κύριες ενότητες μέσω των οποίων επιτυγχάνεται η διαχείριση των χρηστών, του εξυπηρετητή χαρτών GeoServer, των αιτήσεων, των στοιχείων επικοινωνίας και την επεξεργασία των θεματικών επιπέδων. Βασική προϋπόθεση είναι η είσοδος στην εφαρμογή, μέσω του username και του password που δημιουργήθηκε κατά τη διαδικασία της εγκατάστασης.

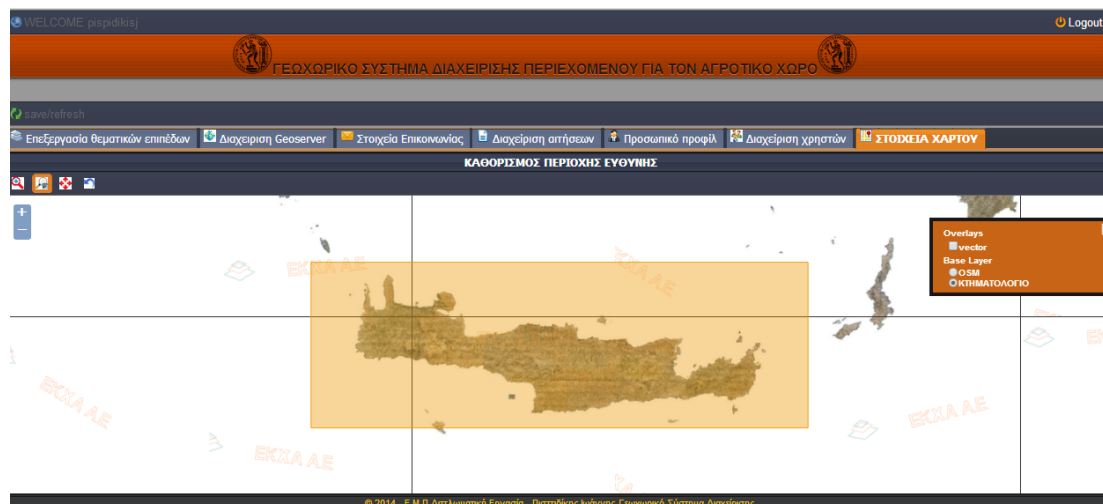


Εικόνα 6-4: Εισαγωγή στο σύστημα διαχείρισης περιεχομένου.

Με την είσοδο στο σύστημα, ανάλογα με την προσβασιμότητα (Διαχειριστής/χρήστης), ενεργοποιούνται και οι αντίστοιχες ενότητες.

6.3.1. Στοιχεία χάρτη

Η συγκεκριμένη ενότητα ενεργοποιείται μόνο σε αυτόν που έχει δικαιώματα διαχειριστή και χρησιμοποιείται, προκειμένου ο τελευταίος να καθορίσει τη ζώνη ορατότητας των δεδομένων επί του χάρτη. Η προεπιλογή του συστήματος είναι τα όρια της Ελλάδος, ενώ ο διαχειριστής με τη βοήθεια των ορθοφωτοχάρτων του ΕΚΧΑ, καθορίζει τα όρια που επιθυμεί.



Εικόνα 6-5: Καθορισμός ζώνης ορατότητας των δεδομένων.

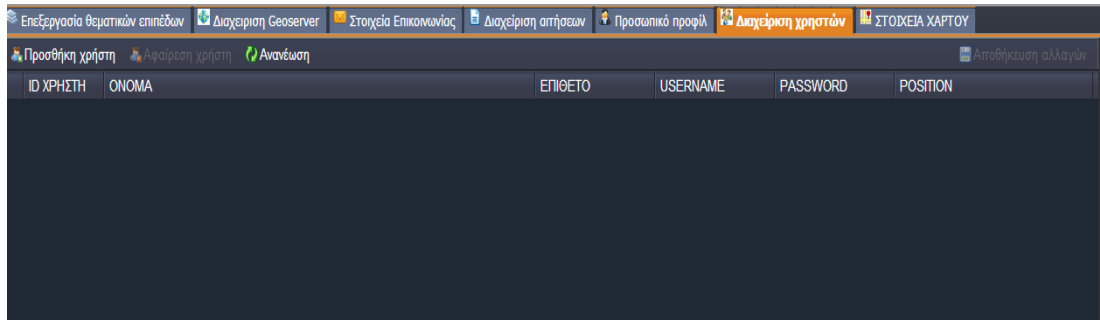
Συγκεκριμένα, η γραμμή εργαλείων αποτελείται από τέσσερα κουμπιά:



- ✓ Το πρώτο χρησιμοποιείται για την εύρεση της περιοχής, βάσει ορίων που καθορίζονται από τις συν/νες των κορυφών του ορθογωνίου, που σχεδιάζεται από τον χρήστη.
- ✓ Το δεύτερο χρησιμοποιείται για τον τελικό καθορισμό της ζώνης ορατότητας.
- ✓ Το τρίτο κουμπί χρησιμοποιείται για να γίνει μεγέθυνση στην προκαθορισμένη ζώνη ορατότητας.
- ✓ Το τελευταίο κουμπί προβαίνει σε αναίρεση της ζώνης καθορίζοντας ως ζώνη τα όρια της Ελλάδας.

6.3.2. Διαχείριση χρηστών

Στη διαχείριση χρηστών έχει πρόσβαση μόνο ο διαχειριστής του συστήματος. Μέσω της ενότητας αυτής, ο τελευταίος, μπορεί να διαγράψει, να τροποποιήσει και να δημιουργήσει νέους χρήστες στο σύστημα. Οι συγκεκριμένοι, θα έχουν δυνατότητα πρόσβασης στην εφαρμογή, χωρίς ωστόσο δικαιώματα διαχειριστή.



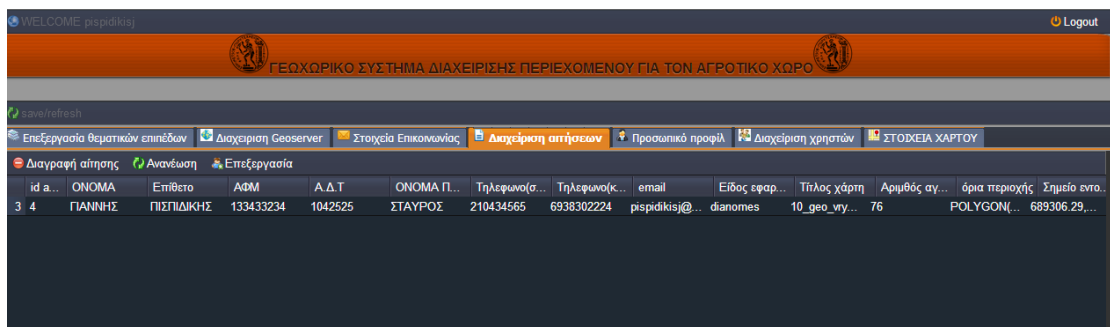
Εικόνα 6-6: Διαχείριση χρηστών.

6.3.3. Προσωπικό προφίλ

Στο προσωπικό προφίλ έχουν πρόσβαση όλοι οι χρήστες, με δυνατότητα τροποποίησης των προσωπικών στοιχείων του προφίλ τους, όπως είναι το όνομα, το επίθετο, το username και το password.

6.3.4. Διαχείριση αιτήσεων

Στη συγκεκριμένη ενότητα υπάρχουν οι αιτήσεις που υποβάλουν οι πολίτες για ένα αγροτεμάχιο.



Εικόνα 6-7: Διαχείριση αιτήσεων.

Επιλέγοντας οι χρήστες αρχικά μία αίτηση και στη συνέχεια το κουμπί «Επεξεργασία», εμφανίζεται ένα νέο παράθυρο. Σε αυτό, μέσω μιας WebGIS εφαρμογής, μπορούν να επεξεργαστούν τις συγκεκριμένες αιτήσεις, έχοντας συγχρόνως τη δυνατότητα δημιουργίας pdf αρχείου, με απόσπασμα του χάρτη που θέλουν να χορηγήσουν. Η WebGIS εφαρμογή αποτελείται από τρία τμήματα. Το αριστερό τμήμα περιλαμβάνει τα στοιχεία που αφορούν το απόσπασμα του χάρτη που θα δημιουργηθεί. Συγκεκριμένα, καθορίζεται το dpi (dots per inches), τα σχόλια και τα όρια του αποσπάσματος (κλίμακα και στροφή). Στο δεξί τμήμα αναγράφονται τα στοιχεία του πελάτη που έκανε την αίτηση, καθώς και στοιχεία που αφορούν την πράξη εφαρμογής και τον αριθμό του αγροτεμαχίου. Τέλος, στο κεντρικό τμήμα βρίσκεται ο χάρτης, όπου ο χρήστης διαδραστικά επιλέγει το τμήμα του αποσπάσματος που θέλει να εκτυπωθεί σε pdf.



Εικόνα 6-8: Επεξεργασία αίτησης.

Με την επιλογή του κουμπιού «Ολοκλήρωση αίτησης», ενημερώνεται το σύστημα, πως το συγκεκριμένο αίτημα το ανέλαβε ο εν λόγω χρήστης. Τέλος, δυνατότητα διαγραφής μιας αίτησης έχει μόνο ο διαχειριστής.

6.3.5. Στοιχεία επικοινωνίας

Τα στοιχεία επικοινωνίας περιλαμβάνουν το email επικοινωνίας της διαχείρισης με του πελάτες, ενώ δικαίωμα τροποποίησης δίνεται μόνο στους χρήστες που έχουν δικαίωμα διαχειριστή.

6.3.6. Διαχείριση του GeoServer

Η διαχείριση του GeoServer αποτελεί τη σημαντικότερη ενότητα της συγκεκριμένης εφαρμογής, δεδομένου ότι πρόκειται για γεωχωρικό σύστημα διαχείρισης περιεχομένου. Πιο αναλυτικά, αποτελείται από πέντε υποενότητες οι οποίες αναφέρονται παρακάτω:

➤ Workspace

Μια ομάδα εργασίας (workspace) αποτελεί ένα «θεωρητικό δοχείο» μέσα στο οποίο ομαδοποιούνται χωρικά δεδομένα διαφορετικών πηγών αλλά με ίδιο θεματικό περιεχόμενο. Μέσω της συγκεκριμένης υποενότητας δημιουργούνται και διαγράφονται οι εν λόγω ομάδες εργασίας στον GeoServer αυτόματα. Βασική προϋπόθεση, προκειμένου να διαγραφεί μια ομάδα εργασίας, είναι να μην περιέχει καμία αποθήκη και κατ' επέκταση κανένα θεματικό επίπεδο.

Εικόνα 6-9: Workspace.

➤ Αποθήκη θεματικών επιπέδων

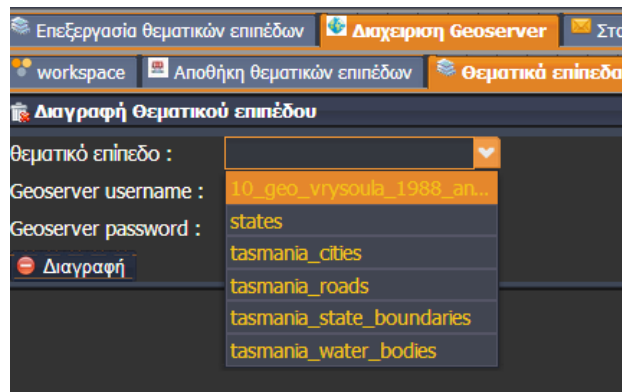
Η αποθήκη (Store) αναφέρεται σε μια συγκεκριμένη πηγή δεδομένων, η οποία μπορεί να είναι ένα Shapefile, μια βάση δεδομένων ή οποιαδήποτε πηγή υποστηρίζει η έκδοση του Geoserver. Ο χρήστης μπορεί να διαγράψει μια αποθήκη είτε αυτή αποτελεί πηγή διανυσματικών είτε κανονικοποιημένων δεδομένων.

Εικόνα 6-10: Store.

Αρχικά, επιλέγεται η ομάδα εργασίας (workspace) και στη συνέχεια η αποθήκη που επιθυμεί ο χρήστης να διαγράψει. Εφόσον αυτή είναι κενή, διαγράφεται αυτόματα.

➤ Θεματικά επίπεδα

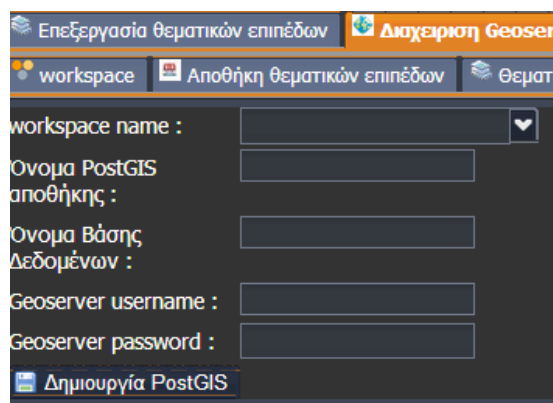
Στην υποενότητα θεματικά επίπεδα υπάρχει διαθέσιμη η λίστα με όλα τα διαθέσιμα θεματικά επίπεδα που έχουν δημοσιευθεί στον GeoServer. Τα τελευταία, ο χρήστης διαθέτει τη δυνατότητα να τα διαγράψει.



Εικόνα 6-11: Διαγραφή θεματικού επιπέδου.

➤ Διαχείριση PostGIS

Μια βασική υποενότητα προκειμένου να αξιοποιηθεί η χωρική βάση δεδομένων PostGIS είναι η Διαχείριση PostGIS. Προκειμένου τα διανυσματικά δεδομένα να μεταφορτωθούν στη βάση δεδομένων και στη συνέχεια να δημοσιευθούν στο Διαδίκτυο μέσω του GeoServer, θα πρέπει να δημιουργηθεί στον τελευταίο μια αποθήκη, η οποία να έχει ως πηγή τη βάση δεδομένων που έχει δημιουργηθεί στην PostGIS.



Εικόνα 6-12: Δημιουργία PostGIS store.

Τα στοιχεία που είναι απαραίτητα να συμπληρωθούν είναι η ομάδα εργασίας, το όνομα της PostGIS αποθήκης, το όνομα της βάσης δεδομένων που θα ανεβούν τα χωρικά δεδομένα και τα στοιχεία σύνδεσης με τον GeoServer. Πρέπει να σημειωθεί ότι τα στοιχεία, όπως είναι η θύρα, ο server και τα στοιχεία σύνδεσης με την PostGIS δεν χρειάζονται, δεδομένου ότι η εφαρμογή τα έχει ήδη αποθηκευμένα κατά την εγκατάστασή της. Επιπροσθέτως, είναι εφικτό να δημιουργηθούν πολλαπλές αποθήκες, οι οποίες θα βασίζονται σε διαφορετικές βάσεις δεδομένων PostGIS.

➤ Μεταφόρτωση δεδομένων

Την ολοκλήρωση στη διαχείριση του GeoServer έρχεται να καλύψει η δυνατότητα μεταφόρτωσης χωρικών δεδομένων.

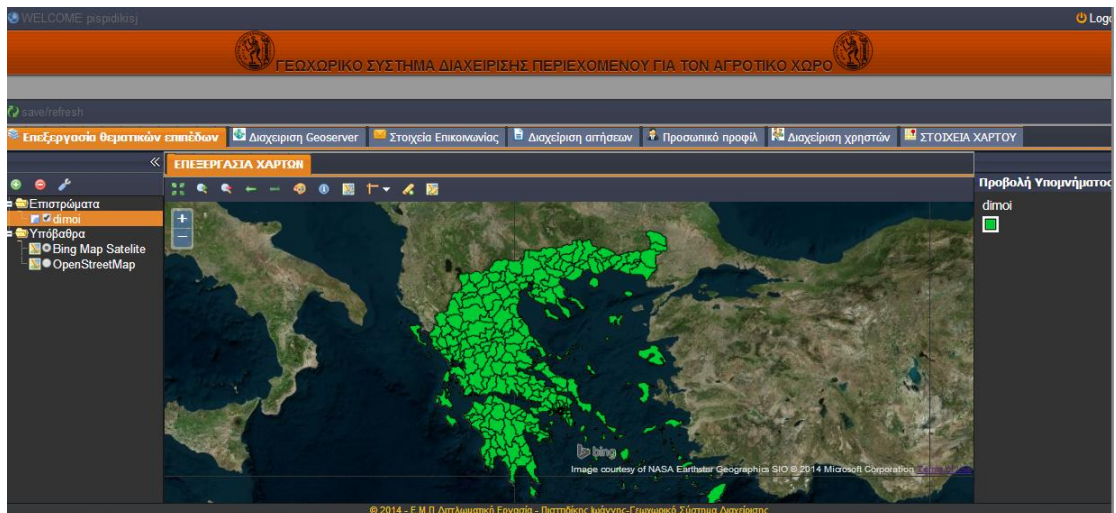
Εικόνα 6-13: Μεταφόρτωση χωρικών δεδομένων.

Ο χρήστης δύναται να μεταφορτώσει είτε κανονικοποιημένα είτε διανυσματικά δεδομένα. Εφόσον μεταφορτώσει κανονικοποιημένα δεδομένα, δεν μπορεί να επιλέξει ως αποθήκη την PostGIS. Η αδυναμία αυτή οφείλεται στον GeoServer, ο οποίος μέχρι τη σημερινή έκδοση, δεν μπορεί να διαβάσει τους πίνακες της PostGIS που περιέχουν τις πληροφορίες της εικόνας. Συνεπώς, για να είναι δυνατή η μεταφόρτωση μιας εικόνας, απαιτείται να είναι είτε geotiff είτε να συνοδεύεται από το αντίστοιχο tfw, το οποίο περιέχει τις πληροφορίες της γεωαναφοράς της.


Σε περίπτωση διανυσματικών δεδομένων, αν επιλεγθεί ως αποθήκη μια PostGIS, τότε με το πάτημα του κουμπιού «Ανέβασμα» και την επιτυχή ολοκλήρωση της μεταφόρτωσης επιτυγχάνεται αφενός η μεταφόρτωση των δεδομένων στη βάση δεδομένων και αφετέρου η δημοσίευσή τους στον GeoServer.

6.3.7. Επεξεργασία θεματικών επιπέδων

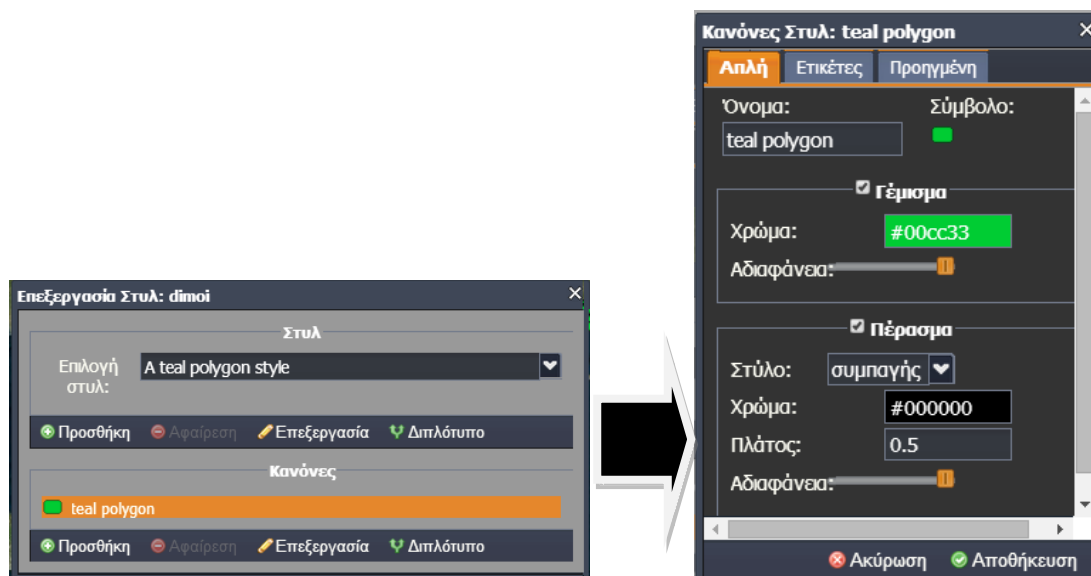
Η επεξεργασία θεματικών επιπέδων αποτελεί μια διαδικτυακή GIS εφαρμογή μέσω της οποίας επιτυγχάνεται η επεξεργασία και η μορφοποίηση κυρίως των διανυσματικών επιπέδων. Αναλυτικότερα, έχει δυνατότητα οπτικοποίησης όλων των θεματικών επιπέδων του GeoServer, ενώ διαθέτει ένα σύνολο εργαλείων μέσω των οποίων η μορφοποίηση, δημιουργία, επεξεργασία και διαγραφή στοιχείων ενός Layer γίνεται εύκολη υπόθεση. Προκειμένου να γίνει η παρουσίαση των ανωτέρω εργαλείων, μεταφορτώθηκε και δημοσιεύθηκε αντίστοιχα ένα shapefile, το οποίο περιέχει τους δήμους της Ελλάδας. Στην εικόνα 6-14 φαίνεται η οπτικοποίηση του θεματικού επιπέδου dimoi στην WebGIS εφαρμογή.



Εικόνα 6-14: Οπτικοποίηση θεματικού επιπέδου.

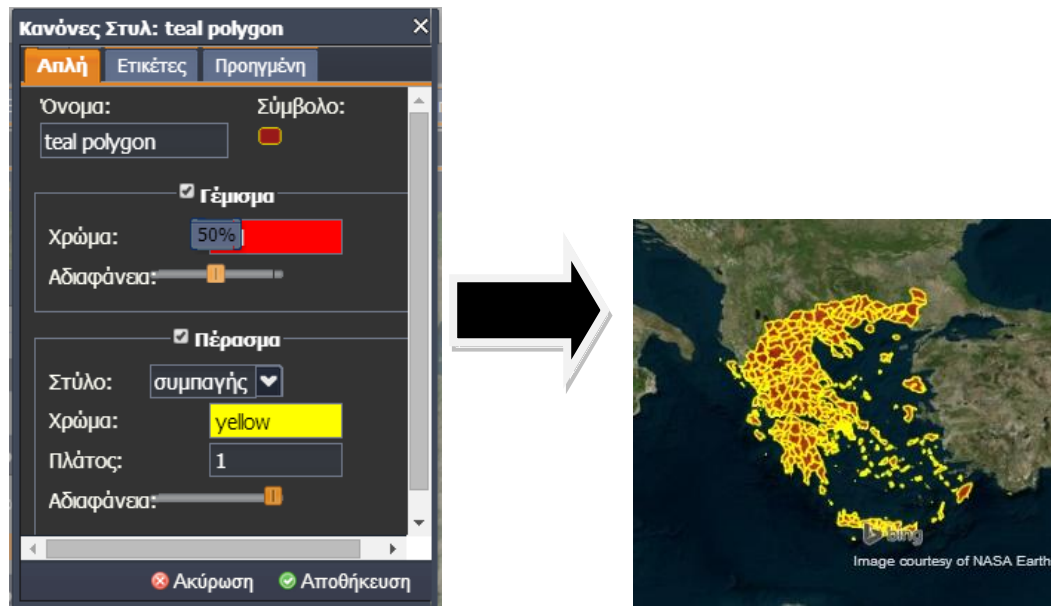
- Μορφοποίηση του στυλ (εργαλείο: )

Με την επιλογή του συγκεκριμένου εργαλείου ανοίγει ένα παράθυρο μέσω του οποίου δίνεται η δυνατότητα μορφοποίησης του Layer σε πραγματικό χρόνο.



Εικόνα 6-15: Μορφοποίηση(1).

Αν, παράδειγμα, ο χρήστης επιθυμούσε οι δήμοι να φαίνονται με κόκκινο χρώμα έχοντας διαφάνεια 50% και το περίγραμμα να είναι σε κίτρινο χρώμα με πάχος «1», τότε θα προχωρούσαμε στις ρυθμίσεις που φαίνονται στην εικόνα 6-16.



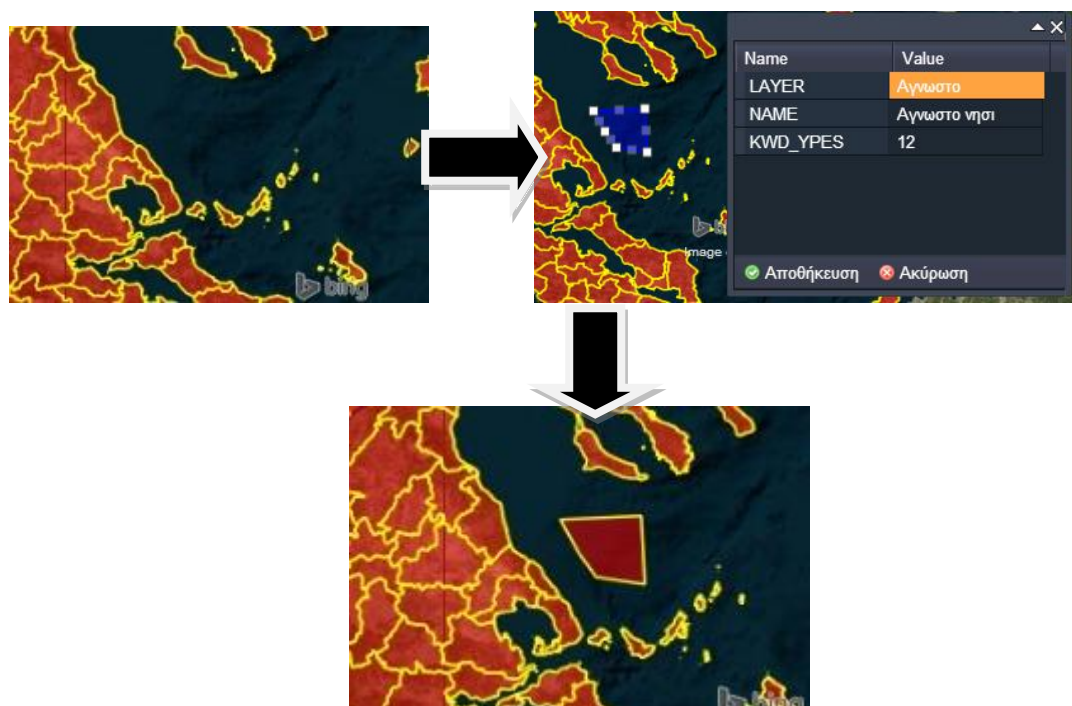
Εικόνα 6-16: Μορφοποίηση(2).

Η ενδιάμεση διαδικασία που γίνεται με το πάτημα του κουμπιού «Αποθήκευση» είναι η αυτόματη δημιουργία του αρχείου SLD, το οποίο στη συνέχεια αποθηκεύεται στον GeoServer και καθορίζεται ως στυλ για το συγκεκριμένο layer. Στον παρακάτω κώδικα φαίνεται το SLD αρχείο που δημιουργήθηκε αυτόματα με την παραπάνω διαδικασία:

```
<?xml version="1.0" encoding="UTF-8"?><sld:StyledLayerDescriptor
xmlns="http://www.opengis.net/sld" xmlns:sld="http://www.opengis.net/sld"
xmlns:ogc="http://www.opengis.net/ogc" xmlns:gml="http://www.opengis.net/gml"
version="1.0.0"><sld:NamedLayer><sld:Name>example_dimoi</sld:Name><sld:UserSt
yle><sld:Name>example_dimoi</sld:Name><sld:Title>A teal polygon
style</sld:Title><sld:IsDefault>1</sld:IsDefault><sld:FeatureTypeStyle><sld:Name>na
me</sld:Name><sld:Rule><sld:Title>teal
polygon</sld:Title><sld:PolygonSymbolizer><sld:Fill><sld:CssParameter
name="fill">#FF0000</sld:CssParameter><sld:CssParameter name="fill-
opacity">0.5</sld:CssParameter></sld:Fill><sld:Stroke><sld:CssParamete
r name="stroke">#FFFF00</sld:CssParameter></sld:Stroke></sld:PolygonSymbolizer></
sld:Rule></sld:FeatureTypeStyle></sld:UserStyle></sld:NamedLayer></sld:StyledLayer
Descriptor>
```


- Δημιουργία νέου χαρακτηριστικού (εργαλείο: )

Με το εν λόγω εργαλείο ο χρήστης μπορεί να δημιουργήσει νέο χαρακτηριστικό στο συγκεκριμένο layer, συμπληρώνοντας ταυτόχρονα και τα επιπρόσθετα περιγραφικά δεδομένα που το συνοδεύουν.



Εικόνα 6-17: Δημιουργία νέου χαρακτηριστικού.

Με το πάτημα του κουμπιού αποθήκευσης στέλνεται ένα αίτημα Transaction προκειμένου να δημιουργηθεί το νέο στοιχείο. Η υπηρεσία WFS που υποστηρίζει το συγκεκριμένο αίτημα ονομάζεται WFS-T(Web Feature Server-Transaction) .

- Τροποποίηση υπάρχοντος χαρακτηριστικού (εργαλείο: )

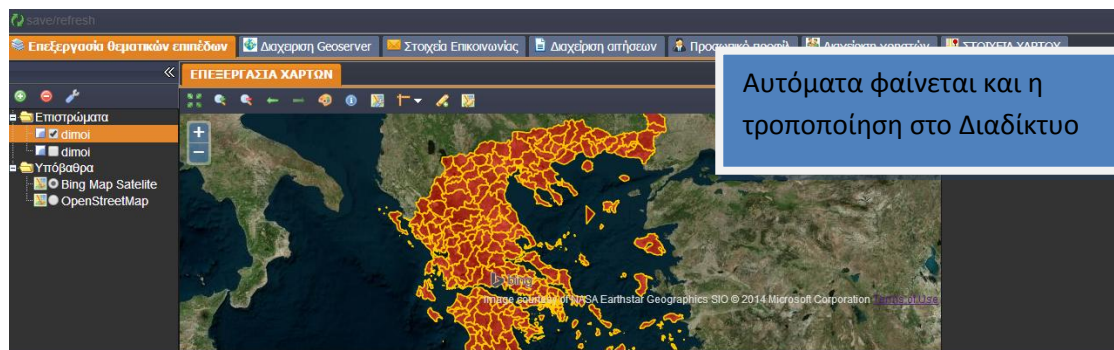
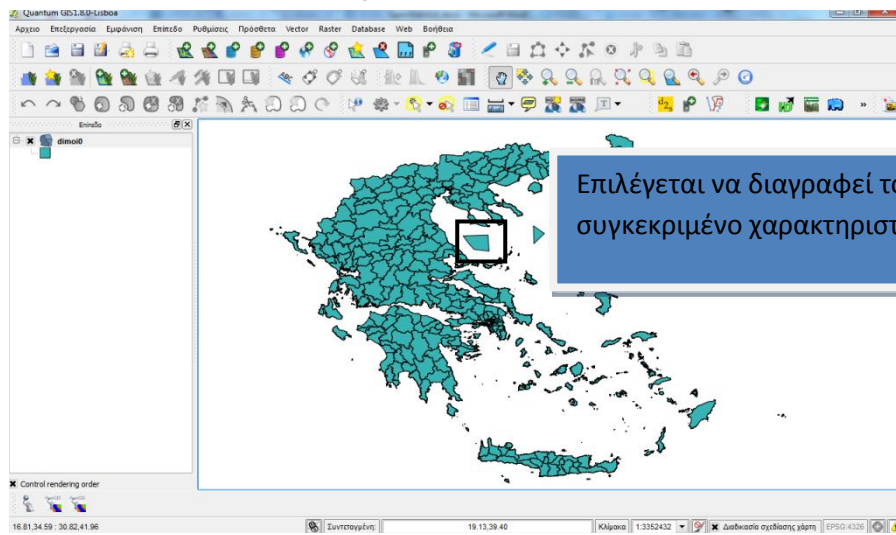
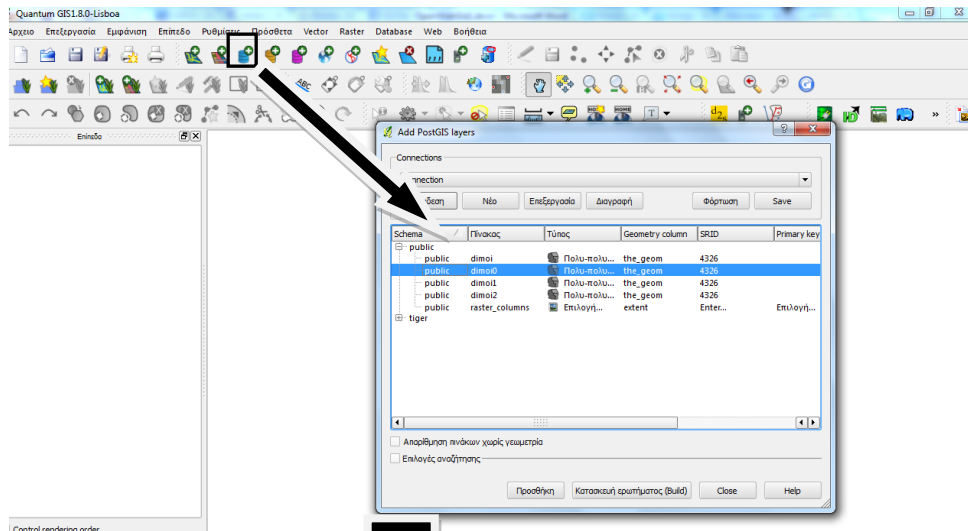
Με βάση την υπηρεσία WFS-T πραγματοποιείται και τροποποίηση ενός υπάρχοντος χαρακτηριστικού. Συγκεκριμένα, με αυτό το εργαλείο ο χρήστης μπορεί να τροποποιήσει και να διαγράψει τα γεωμετρικά και περιγραφικά στοιχεία ενός χαρακτηριστικού.

6.3.8. Αξιοποίηση του QuantumGIS

Ένα βασικό πλεονέκτημα που δημιουργείται με την αποθήκευση των διανυσματικών δεδομένων στη βάση δεδομένων και τη δημοσίευσή τους μέσω του GeoServer είναι ότι μπορούν να φορτωθούν στο λογισμικό QuantumGIS. Έτσι, επιτυγχάνεται η offline επεξεργασία αυτών των δεδομένων, η τροποποίηση των οποίων θα φαίνεται άμεσα στην WebGIS εφαρμογή. Υπάρχουν δύο τρόποι με τον οποίο δύναται να πραγματοποιηθεί αυτό:

- Προσθήκη PostGIS layer

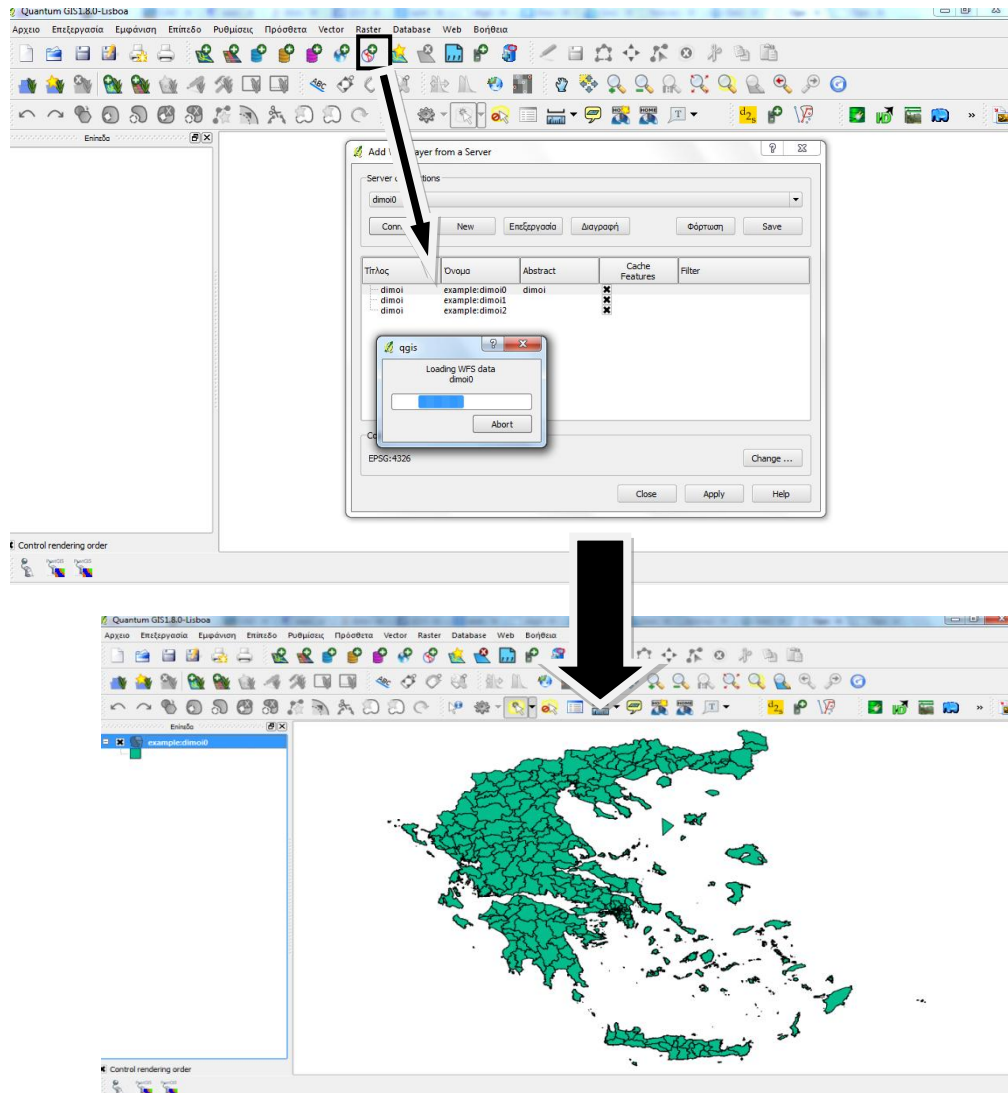
Ο πρώτος τρόπος είναι να γίνει προσθήκη του layer απευθείας από τη βάση δεδομένων PostGIS.



Εικόνα 6-18: Επεξεργασία από λογισμικό QuantumGIS(1^{ος} τρόπος).

➤ Προσθήκη WFS layer

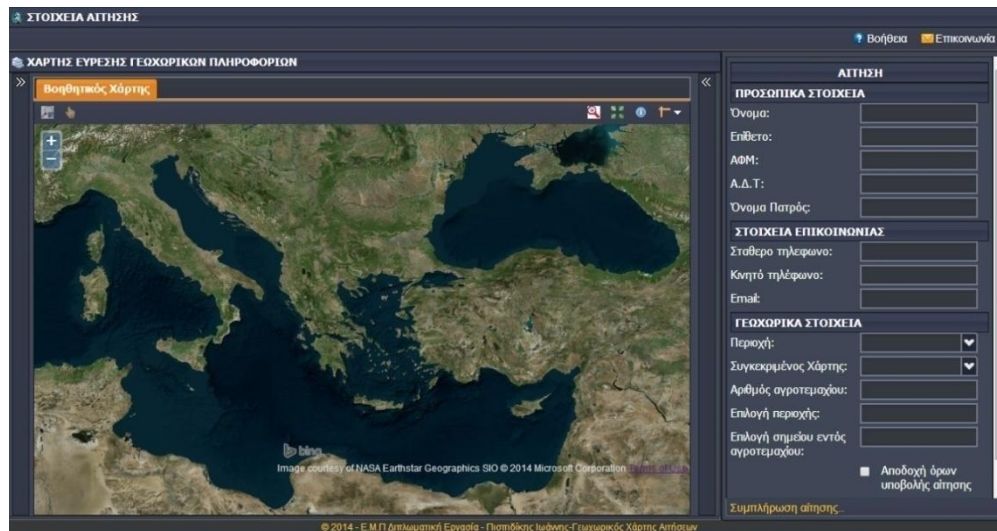
Ο δεύτερος τρόπος είναι η προσθήκη του layer από τον GeoServer μέσω της υπηρεσίας WFS, προκειμένου να είναι εφικτή η τροποποίηση του εν λόγω θεματικού επιπέδου, τόσο γεωμετρικά όσο και περιγραφικά.



Εικόνα 6-19: Επεξεργασία από λογισμικό QuantumGIS (2^{ος} τρόπος).

6.4. Σύστημα Αίτησης Χαρτών

Το συγκεκριμένο κομμάτι του προγράμματος αποτελεί εφαρμογή για τον πελάτη και αφορά την online δυνατότητα αίτησης στοιχείων για ένα αγροτεμάχιο. Με την εν λόγω εφαρμογή επιδιώκεται να μειωθεί η γραφειοκρατία και να δημιουργηθεί ένας πιο αυτοματοποιημένος και εύκολος τρόπος για να αιτηθεί ο πολίτης ή ο τοπογράφος μηχανικός κτηματολογικά στοιχεία, που αφορούν κάποιο αγροτεμάχιο, το οποίο ανήκει σε κάποιο αναδασμό ή διανομή.

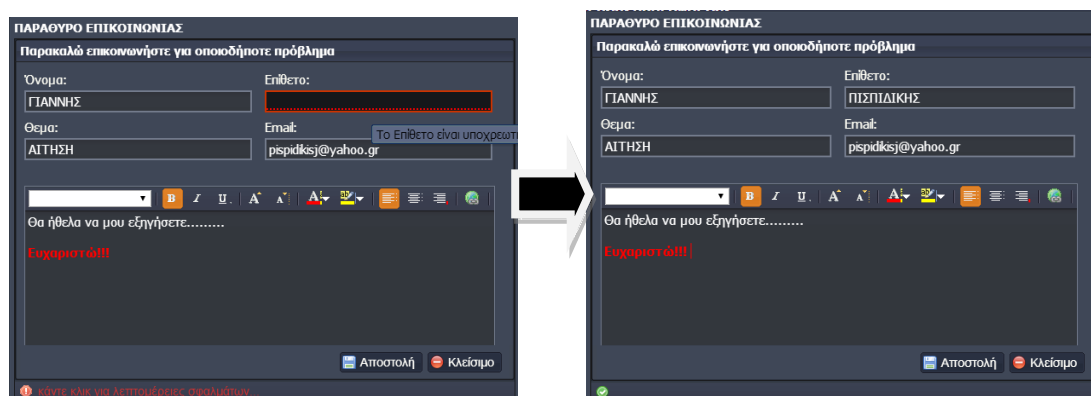


Εικόνα 6-20: Εφαρμογή αίτησης χαρτών.

Όπως φαίνεται στην εικόνα 6-20, η WebGIS εφαρμογή αποτελείται από δύο τμήματα. Στο δεξί τμήμα εμφανίζεται η αίτηση που συμπληρώνει ο πολίτης, ενώ στο αριστερό τμήμα υπάρχει ο γεωχωρικός χάρτης, ο οποίος διαθέτει διάφορα εργαλεία που οδηγούν στη σωστή συμπλήρωση των στοιχείων. Επιπροσθέτως, στο πάνω δεξιά μέρος της εφαρμογής υπάρχουν δύο κουμπιά: η βοήθεια και η επικοινωνία. Πατώντας το κουμπί «βοήθεια» ανοίγει ένα pdf, το οποίο βοηθά τον πελάτη να συμπληρώσει ορθά την αίτηση.

6.4.1. Δυνατότητα επικοινωνίας

Στη συγκεκριμένη διαδικτυακή εφαρμογή υπάρχει η δυνατότητα επικοινωνίας με την αρμόδια υπηρεσία, έτσι ώστε οι πολίτες να θέτουν κάποια προβλήματα όσον αφορά την αίτηση και την εφαρμογή. Επιλέγοντας το κουμπί «Επικοινωνία» ανοίγει το αντίστοιχο παράθυρο επικοινωνίας.



Εικόνα 6-21: Παράθυρο επικοινωνίας.

Στην εικόνα 6-21 φαίνεται ένα παράδειγμα επικοινωνίας ενός πολίτη με την υπηρεσία. Προκειμένου να αποσταλεί το μήνυμα, είναι απαραίτητο να έχουν συμπληρωθεί όλα τα απαραίτητα στοιχεία, διαφορετικά εμφανίζεται αντίστοιχη προειδοποίηση. Με την αποστολή του μηνύματος έρχεται απευθείας ενημερωτική απάντηση από την υπηρεσία στον χρήστη ότι θα υπάρξει σύντομα επικοινωνία μαζί του.

6.4.2. Διαδικασία αίτησης

Κατά τη διαδικασία της αίτησης ο πολίτης καλείται να συμπληρώσει στοιχεία που αφορούν τόσο τα προσωπικά του δεδομένα όσο και δεδομένα για το αγροτεμάχιο που ενδιαφέρεται.

Εικόνα 6-22: Αίτηση.

Επιλέγοντας αποστολή αίτησης ελέγχεται από το σύστημα εάν έχουν συμπληρωθεί όλα τα υποχρεωτικά πεδία. Στη συνέχεια, είτε αποστέλλεται η αίτηση είτε ενημερώνεται αντίστοιχα ο πολίτης.

The image shows two versions of a web form titled 'ΑΙΤΗΣΗ' (Request). Both forms have sections for 'ΠΡΟΣΩΠΙΚΑ ΣΤΟΙΧΕΙΑ' (Personal Data) and 'ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΙΑΣ' (Contact Information). The left form, labeled 'Ελλιπή Στοιχεία' (Incomplete Data), has empty fields for 'ΠΕΓΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ' (Geographic Data) and shows a list of error messages: 'Απαιτείται επιλογή περιοχής', 'Απαιτείται το όνομα του φύλλου χάρτη', 'Απαιτείται αριθμός αγροτεμαχίου', 'Απαιτείται περιοχή επιλογής', and 'Απαιτείται σημείο εντός αγροτεμαχίου'. The right form, labeled 'Πλήρη στοιχεία' (Complete Data), has all fields filled out, including 'ΠΕΓΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ' (Region: dianomes, Map Sheet: 10_geo_vrysoula, Plot Number: 52, Selection Area: POLYGON((689339., 689552.21, 4533231)), Selection Point: 689552.21, 4533231), and a checked box for 'Αποδοχή όρων υποβολής αίτησης' (Acceptance of submission terms). Both forms have buttons for 'Αποστολή αίτησης' (Submit request) and 'Άκυρο' (Cancel).

Ελλιπή Στοιχεία

Πλήρη στοιχεία

Εικόνα 6-23: Ελλιπή και πλήρη στοιχεία αίτησης.

Η αναλυτική περιγραφή του τρόπου υποβολής των αιτήσεων θα περιγραφεί στο επόμενο κεφάλαιο με παρατιθέμενο παράδειγμα.

6.5. Αξιολόγηση Συστήματος

Ο σχεδιασμός και η δημιουργία του συγκεκριμένου συστήματος βασίστηκε σε σύγχρονες τεχνολογίες μέσω των οποίων επιτυγχάνεται η ομαλή και γρήγορη λειτουργία της εφαρμογής σε όλα τα επίπεδα. Συγκεκριμένα, αξιοποιήθηκε η τεχνολογία AJAX, όπου μέσω του μορφότυπου JSON πραγματοποιείται λήψη δεδομένων από τον Server ασύγχρονα στο παρασκήνιο χωρίς την ανάγκη ανανέωσης της εφαρμογής. Επιπροσθέτως, βασίστηκε στο Framework Extjs, το οποίο είναι το πλέον κατάλληλο για την ανάπτυξη εφαρμογών στο Διαδίκτυο. Επιπλέον, αξιοποιήθηκε η βιβλιοθήκη GXP που περιλαμβάνει έναν μεγάλο αριθμό χαρτογραφικών εργαλείων για το Διαδίκτυο. Η αξιοποίηση των δυνατοτήτων του GeoServer αλλά και η χρήση του εργαλείου GeoWebCache καθιστά την εφαρμογή ταχύτερη, δεδομένου ότι η απάντηση στα διάφορα χωρικά αιτήματα των πελατών γίνεται με τον γρηγορότερο και αποτελεσματικότερο τρόπο.

Το σύστημα διαχείρισης περιεχομένου που δημιουργήθηκε παρέχει δυνατότητα διαχείρισης σε θέματα που αφορούν την επεξεργασία χρηστών, την οργάνωση και μεταφόρτωση των χωρικών δεδομένων στο εξυπηρετητή χαρτών GeoServer, καθώς

και την επεξεργασία αυτών σε πραγματικό χρόνο μέσω της υπηρεσίας WFS-T που υποστηρίζει ο GeoServer.

Μερικά στοιχεία που μπορούν να ενσωματωθούν και να κάνουν το σύστημα πιο ολοκληρωμένο είναι τα παρακάτω:

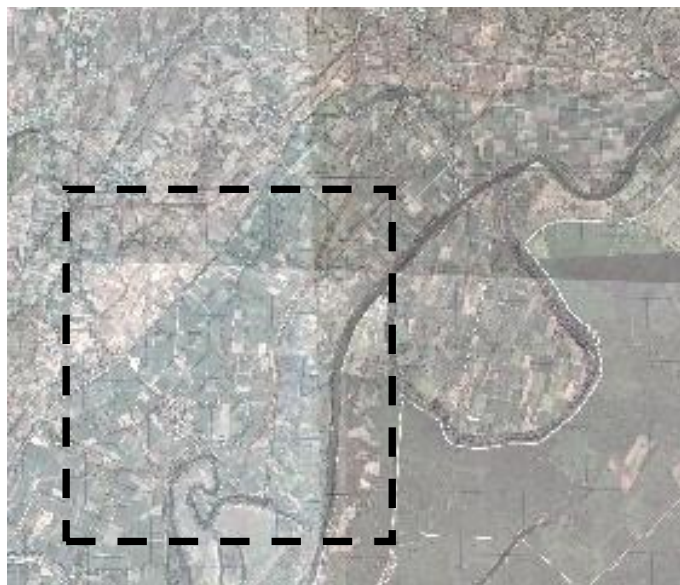
- Χρήση του GeoExplorer: παρέχει επιπρόσθετες δυνατότητες δημοσίευσης WebGIS εφαρμογών αναλόγως των αναγκών.
- Αξιοποίηση των ρόλων που διαθέτει ο GeoServer και βασίζεται η όλη λειτουργία της ασφάλειάς του. Οι ρόλοι ανατίθενται σε χρήστες και ομάδες χρηστών και καθορίζουν ποιές ενέργειες επιτρέπεται να κάνουν στον GeoServer. Παράδειγμα ρόλων είναι η προσβασιμότητα σε διάφορα θεματικά επίπεδα, η πρόσβαση σε υπηρεσίες WFS και η δυνατότητα διαχείρισης του GeoServer.
- Χρήση CSW (Catalog Service-Web) για δυνατότητα αναζήτησης δεδομένων βάσει των μεταδεδομένων τους.

7. ΠΑΡΑΔΕΙΓΜΑ ΣΤΗΝ ΠΕΡΙΟΧΗ ΤΟΥ ΕΒΡΟΥ

Ο Νομός Έβρου είναι ένας νομός πεδινός. Συγκεκριμένα, μόνο το 10% αποτελούν οι ορεινοί όγκοι έναντι των εκτεταμένων πεδιάδων που καταλαμβάνουν το 62% της έκτασής του, εκ των οποίων ένα μεγάλο μέρος καλλιεργείται. Στην καλλιέργεια συμβάλλει η ύπαρξη του ποταμού Έβρου, των παραποτάμων του αλλά και πλήθος αρδευτικών καναλιών. Συνεπώς, στην περιοχή έχουν γίνει πολλαπλές διανομές από τις αρχές του 20^{ου} αιώνα, πολλές από τις οποίες τροποποιήθηκαν λόγω αναδασμών προκειμένου να αντιμετωπιστούν προβλήματα, όπως είναι ο κατακερματισμός των εκτάσεων.

7.1. Περιοχή Μελέτης και Διαθέσιμα Δεδομένα

Ως περιοχή μελέτης επιλέχθηκε η περιοχή του Πόρου του Νομού Έβρου. Ο Πόρος ανήκει στο δήμο Αλεξανδρούπολης της Περιφερειακής Ενότητας Έβρου, που βρίσκεται στην Περιφέρεια Ανατολικής Μακεδονίας - Θράκης, σύμφωνα με τη διοικητική διαίρεση της Ελλάδας, όπως διαμορφώθηκε με το πρόγραμμα Καλλικράτης. Η επιλογή της συγκεκριμένης περιοχής έγινε με βάση τους διαθέσιμους χάρτες διανομών και αναδασμών που υπάρχουν και είναι διαθέσιμοι, για να χρησιμοποιηθούν ως παράδειγμα αίτησης στην εφαρμογή.



Εικόνα 7-1: Περιοχή μελέτης.

Πιο αναλυτικά, οι χάρτες που υπάρχουν στην περιοχή μελέτης αναφέρονται στον πίνακα 7-1:

Πίνακας 7-1: Διαθέσιμη χάρτες στην περιοχή μελέτης.

Χάρτης και ημερομηνία	Τύπος εφαρμογής
Πόρος 1932	Αρχική διανομή (πινακίδες 2)
Πόρος 1953	Συμπληρωματική διανομή (πινακίδες 2)
Πόρος 1961	Αρχική διανομή
Πόρος 1977	Συμπληρωματική διανομή (πινακίδες 2)
Πόρος 1990	Αναδασμός
Πόρος 1993	Συμπληρωματική διανομή

7.2. Παραδείγματα Λειτουργίας

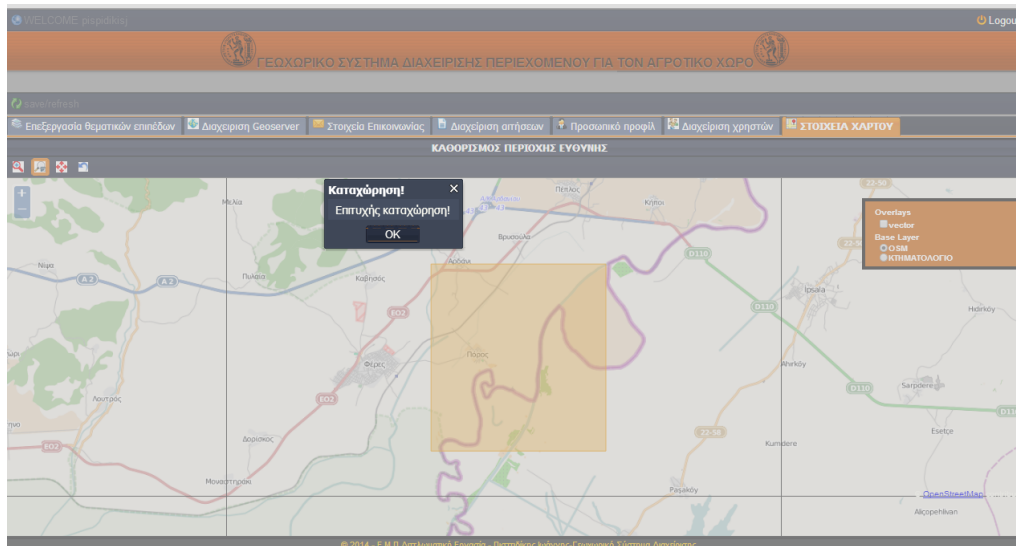
Στο συγκεκριμένο παράδειγμα η αρμόδια υπηρεσία, που είναι το τμήμα Τοπογραφίας, Εποικισμού και Αναδασμού της Διεύθυνσης Αγροτικής Οικονομίας και Κτηνιατρικής της Νομαρχίας Αλεξανδρούπολης, διαθέτει τους προαναφερθέντες χάρτες σε raster μορφή και στο σύστημα αναφοράς ΕΓΣΑ 87. Την εν λόγω σάρωση και μετασχηματισμό των χαρτών την υλοποίησε η ΑΓΡΟΓΗ Α.Ε. Συνεπώς, η όλη διαδικασία αίτησης χαρτών που δημιουργήθηκε είναι βασισμένη σε raster δεδομένα. Παρόλα αυτά, το γεωχωρικό σύστημα διαχείρισης περιεχομένου που δημιουργήθηκε είναι πιο γενικό και καλύπτει και την επεξεργασία διανυσματικών δεδομένων.

7.2.1. Σύστημα Διαχείρισης για προετοιμασία δεδομένων

Έχοντας εγκαταστήσει την εφαρμογή, ο προϊστάμενος της υπηρεσίας οφείλει αρχικά να την ενημερώσει, καθορίζοντας τα όρια ευθύνης, οργανώνοντας τον εξυπηρετητή χαρτών GeoServer και μεταφορτώνοντας τους διαθέσιμους χάρτες προς θέαση. Επιπροσθέτως, μπορεί να δημιουργήσει νέους χρήστες, οι οποίοι θα είναι αρμόδιοι για την επεξεργασία των αιτήσεων των πολιτών.

➤ Καθορισμός ορίων του χάρτη

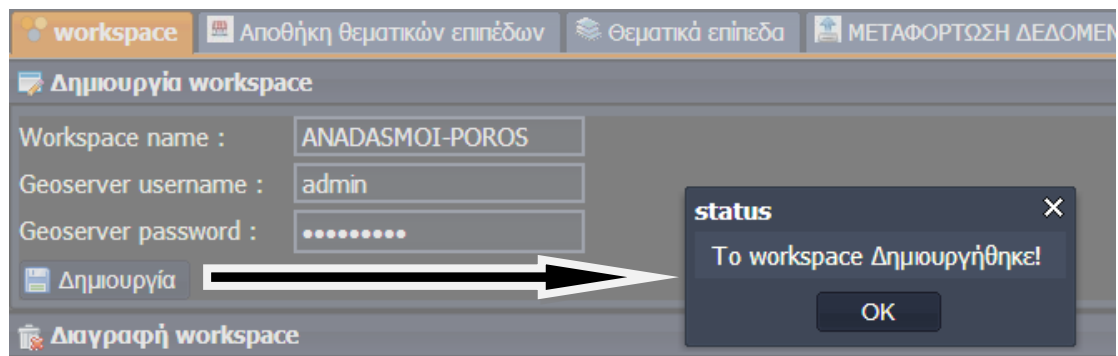
Επιλέγοντας την ενότητα «Στοιχεία χάρτη» με τη βοήθεια των ορθοφωτοχάρτων του ΕΚΧΑ αλλά και του Open Street Map, είναι δυνατή η εύρεση της περιοχής του Πόρου αλλά και ο καθορισμός της περιοχής ευθύνης.



Εικόνα 7-2: Καθορισμός περιοχής ευθύνης.

➤ Οργάνωση του GeoServer

Στο επόμενο βήμα θα πρέπει ο διαχειριστής να καθορίσει τις ομάδες εργασίας (workspaces) και στη συνέχεια να μεταφορτώσει τις εικόνες. Σημειώνεται ότι δεν θα δημιουργηθεί αποθήκη (store), επειδή πρόκειται για raster δεδομένα. Οι ομάδες εργασίας που θα δημιουργηθούν είναι δύο. Η μία θα αφορά τους χάρτες των αναδασμών με όνομα ANADASMOI-POROS, ενώ η άλλη τις διανομές με όνομα DIANOMES-POROS.



Εικόνα 7-3: Δημιουργία Workspace.

Στη συνέχεια, στην υποενότητα «ΜΕΤΑΦΟΡΤΩΣΗ ΔΕΔΟΜΕΝΩΝ» ο διαχειριστής μπορεί να μεταφορτώσει τις διαθέσιμες εικόνες που διαθέτει σε μορφή geotiff. Συγκεκριμένα, θα δημιουργήσει δύο συμπιεσμένα αρχεία, όπου το ένα θα περιέχει τις εικόνες που αφορούν τις Διανομές του Πόρου και το άλλο τον Αναδασμό. Σε περίπτωση που θέλει να μεταφορτώσει μία εικόνα δεν χρειάζεται συμπίεση.

Διαδικασία μεταφόρτωσης των χαρτών

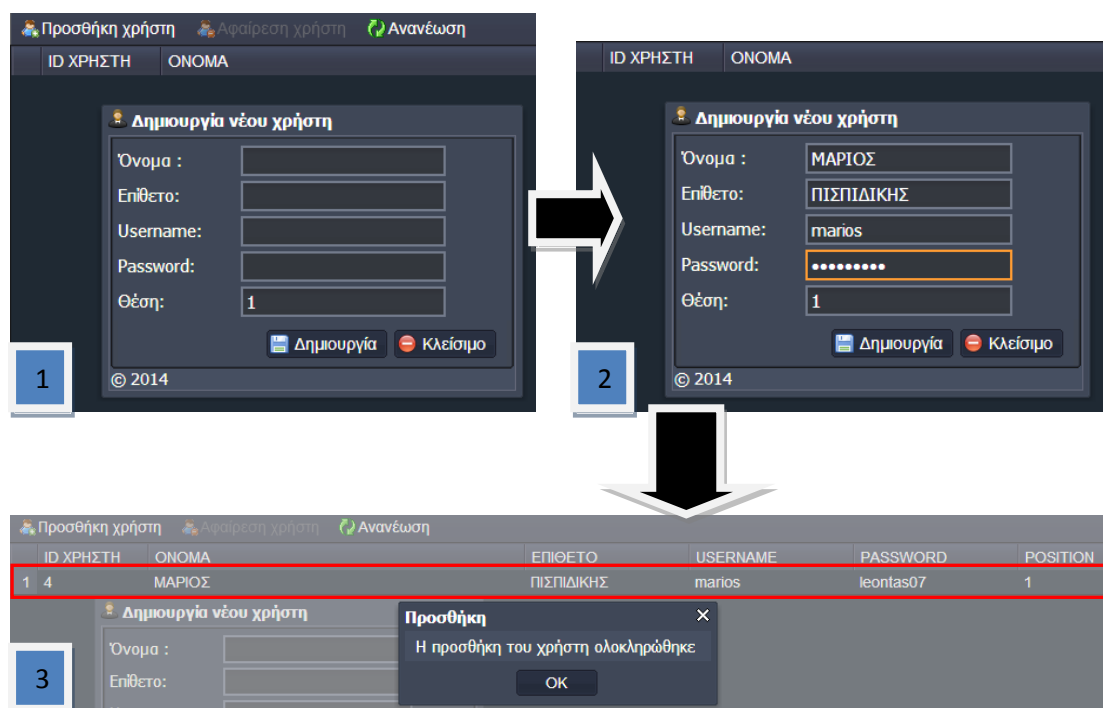
Id	Τίτλος
ANADASMOI-POROS:1990	1990
DIANOMES-POROS:ARXIKI_1932_(1_2)	ARXIKI_1932_(1_2)
DIANOMES-POROS:ARXIKI_1932_(2_2)	ARXIKI_1932_(2_2)
DIANOMES-POROS:ARXIKI_1961_(1_1)	ARXIKI_1961_(1_1)
DIANOMES-POROS:SIMPLIROMATIKI_1953_(1_2)	SIMPLIROMATIKI_1953_(1_2)
DIANOMES-POROS:SIMPLIROMATIKI_1953_(2_2)	SIMPLIROMATIKI_1953_(2_2)
DIANOMES-POROS:SIMPLIROMATIKI_1967_(1_1)	SIMPLIROMATIKI_1967_(1_1)
DIANOMES-POROS:SIMPLIROMATIKI_1977_(1_2)	SIMPLIROMATIKI_1977_(1_2)
DIANOMES-POROS:SIMPLIROMATIKI_1977_2_2	SIMPLIROMATIKI_1977_2_2
DIANOMES-POROS:SIMPLIROMATIKI_1993	SIMPLIROMATIKI_1993

Διαθέσιμοι χάρτες μετά τη διαδικασία μεταφόρτωσης.

Εικόνα 7-4: Μεταφόρτωση διαθέσιμων χαρτών.

➤ Δημιουργία χρηστών

Για τις ανάγκες του παραδείγματος ο διαχειριστής θα δημιουργήσει έναν καινούριο χρήστη, ο οποίος θα έχει πρόσβαση στο σύστημα και θα μπορεί να επεξεργαστεί τις διάφορες αιτήσεις των πελατών. Έτσι, στην υποενότητα «Διαχείριση χρηστών», επιλέγοντας το κουμπί «Προσθήκη χρήστη» ανοίγει το αντίστοιχο παράθυρο, όπου καταχωρούνται τα στοιχεία του νέου χρήστη. Με την ολοκλήρωση της διαδικασίας, εφόσον ελεγχθεί η πληρότητα συμπλήρωσης των στοιχείων, τότε ενημερώνεται αντίστοιχα η βάση δεδομένων, η οποία αποστέλλει τα νέα δεδομένα πίσω στην εφαρμογή μέσω του μορφότυπου JSON. Σημειώνεται ότι η αλληλεπίδραση με τον server γίνεται στο παρασκήνιο, χωρίς τις ανάγκες ανανέωσης της εφαρμογής.



Εικόνα 7-5: Δημιουργία νέου χρήστη.

7.2.2. Εφαρμογή αίτησης χαρτών

Αρχικά, ο πολίτης μπορεί να αναζητήσει το αγροτεμάχιο που τον ενδιαφέρει με τη βοήθεια του χάρτη της εφαρμογής, ο οποίος διαθέτει δορυφορικό υπόβαθρο της υπηρεσίας Bing. Αφού εντοπίσει το αγροτεμάχιο, αναζητά από το αρχείο τους διαθέσιμους χάρτες ξεκινώντας από τον πιο πρόσφατο, δεδομένου ότι η πιο πρόσφατη πράξη αναιρεί την παλαιότερη.



Εικόνα 7-6: Ζητούμενα αγροτεμάχια.

Πρέπει να σημειωθεί ότι, αν πραγματοποιούνταν ψηφιοποίηση των χαρτών, τότε εύκολα ο χρήστης θα μπορούσε να αναζητήσει στοιχεία για το αγροτεμάχιο με βάση προκαθορισμένα ερωτήματα στη βάση δεδομένων. Συγκεκριμένα, επιλέγοντας ένα σημείο εντός του αγροτεμαχίου, θα γινόταν αυτόματα διερεύνηση στη χωρική βάση δεδομένων για τα στοιχεία που ισχύουν για το συγκεκριμένο αγροτεμάχιο και, έτσι, δεν θα χρειαζόταν αναζήτηση από τον ίδιο τον πολίτη.

- 1^ο Παράδειγμα (αγροτεμάχιο 893 συμπληρωματικής διανομής 1993 στον Πόρο)

Στο πρώτο παράδειγμα το ζητούμενο αγροτεμάχιο βρίσκεται σε περιοχή όπου έχει γίνει συμπληρωματική διανομή το 1993. Ο χρήστης με τη χρήση εργαλείων όπως είναι το «transparent» μπορεί να εντοπίσει σχετικά εύκολα τον αριθμό αγροτεμαχίου της συμπληρωματικής διανομής του 1993.



Εικόνα 7-7: Εύρεση αγροτεμαχίου στη διανομή.

Κατόπιν, ακολουθείται η συμπλήρωση της αίτησης του χρήστη. Πρώτα συμπληρώνονται τα προσωπικά στοιχεία και τα στοιχεία επικοινωνίας και έπειτα τα γεωχωρικά.

- Συμπλήρωση γεωχωρικών στοιχείων

Κατ' αρχήν συμπληρώνεται η «περιοχή», στην οποία ο χρήστης επιλέγει αν πρόκειται για διανομή ή αναδασμό. Στο συγκεκριμένο πεδίο οι επιλογές που εμφανίζονται είναι οι διαθέσιμες ομάδες εργασίας (Workspaces) του GeoServer.

ΓΕΩΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ	
Περιοχή:	<input type="text"/>
Συγκεκριμένος Χάρτης:	DIANOMES-POROS
	ANADASMOI-POROS

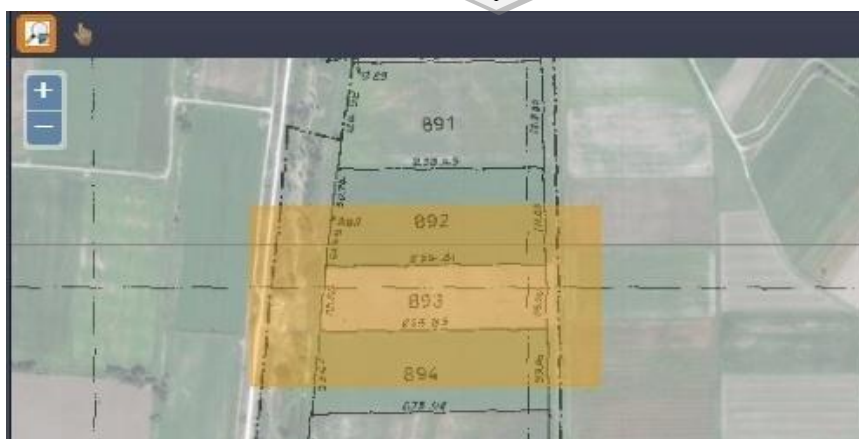
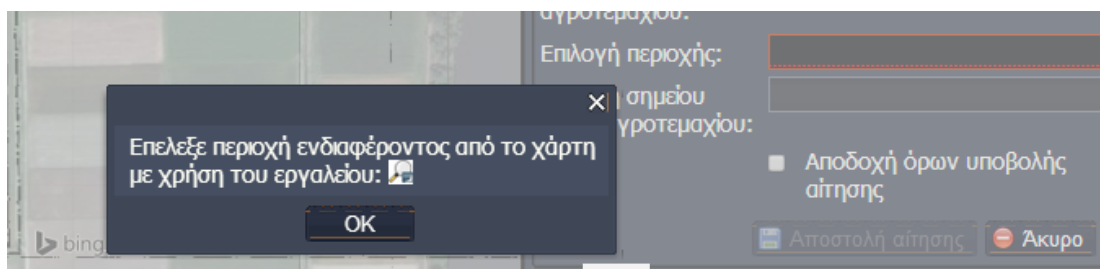
Εικόνα 7-8: Συμπλήρωση γεωχωρικών στοιχείων (1).

Στη συνέχεια επιλέγεται ο συγκεκριμένος χάρτης. Ο χάρτης που επιλέγεται σε αυτό το παράδειγμα είναι συμπληρωματική διανομή του 1993.

A.Δ.Τ:	ARXIKI_1932_(1_2)
Όνομα Πατρός:	ARXIKI_1932_(2_2)
	ARXIKI_1961_(1_1)
ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΝ	
Σταθερο τηλέφωνο:	SIMPLIROMATIKI_1953_(1_2)
Κινητό τηλέφωνο:	SIMPLIROMATIKI_1953_(2_2)
Email:	SIMPLIROMATIKI_1967_(1_1)
	SIMPLIROMATIKI_1977_(1_2)
ΓΕΩΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ	SIMPLIROMATIKI_1977_2_2
Περιοχή:	SIMPLIROMATIKI_1993
Συγκεκριμένος Χάρτης:	<input type="text"/>

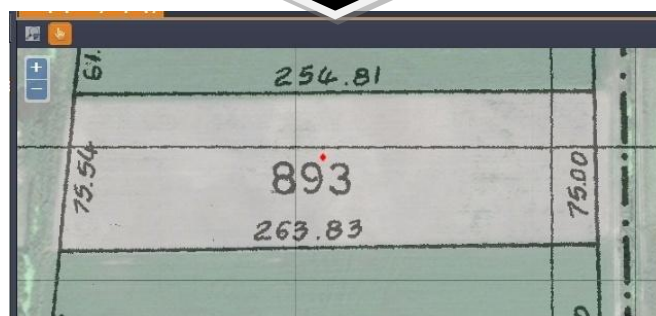
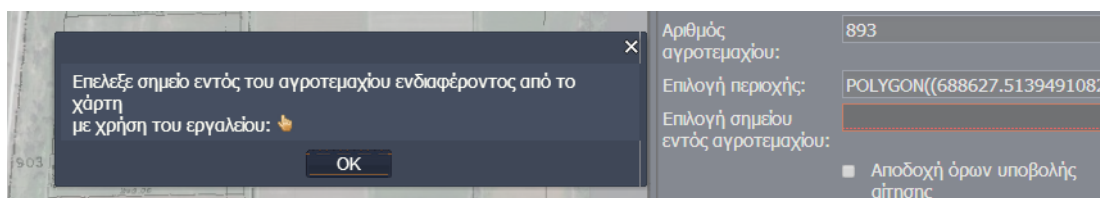
Εικόνα 7-9: Συμπλήρωση γεωχωρικών στοιχείων (2).

Οι διαθέσιμοι χάρτες που εμφανίζονται στη συγκεκριμένη επιλογή είναι μόνο αυτοί που καταχωρήθηκαν στην ομάδα εργασίας «DIANOMES-POROS». Έπειτα, συμπληρώνεται ο αριθμός αγροτεμαχίου. Στη συνέχεια, στο πεδίο «Επιλογή περιοχής», μόλις ο χρήστης επιλέξει εντός του πεδίου, εμφανίζεται αντίστοιχο μήνυμα που ενημερώνει τον χρήστη για το διαθέσιμο εργαλείο του χάρτη, με το οποίο θα επιλέξει ευρύτερη περιοχή του ζητούμενου αγροτεμαχίου.



Εικόνα 7-10: Συμπλήρωση γεωχωρικών στοιχείων (3).

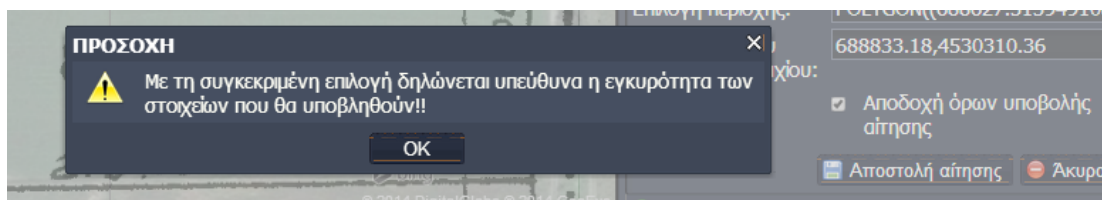
Με την ίδια λογική λειτουργεί και το πεδίο «Επιλογή σημείου εντός αγροτεμαχίου»



Εικόνα 7-11: Συμπλήρωση γεωχωρικών στοιχείων (4).

Ολοκληρώνοντας τη συμπλήρωση της αίτησης, ο χρήστης επιβάλλεται να επιλέξει το πεδίο με το οποίο αποδέχεται τους όρους υποβολής της αίτησης και αφορούν

την εγκυρότητα των στοιχείων που θα συμπληρωθούν. Έπειτα, ενεργοποιείται το κουμπί «Αποστολή αίτησης» και ο χρήστης μπορεί να την υποβάλλει. Σημειώνεται ότι αρχικά ελέγχεται η πληρότητα συμπλήρωσης των απαραίτητων στοιχείων της αίτησης και είτε αποστέλλονται τα στοιχεία στη βάση δεδομένων είτε ενημερώνεται ανάλογα ο χρήστης για τα στοιχεία που πρέπει να συμπληρώσει.



Εικόνα 7-12: Συμπλήρωση γεωχωρικών στοιχείων (5).

- 2^ο Παράδειγμα (αγροτεμάχιο 280 αναδασμού 1990 στον Πόρο)



Εικόνα 7-13: Εύρεση αγροτεμαχίου στον αναδασμό.

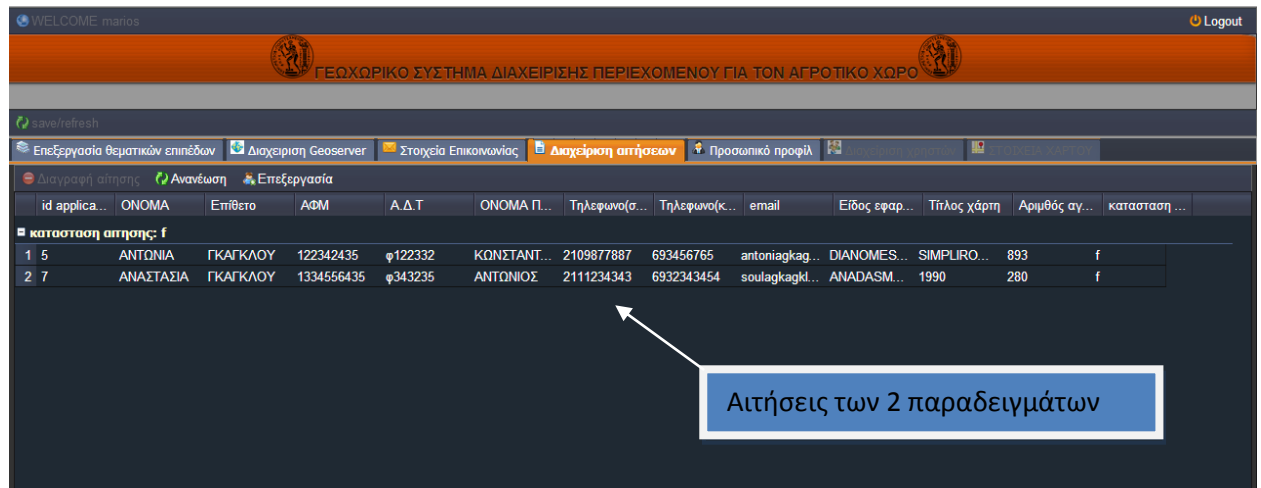
Η συμπλήρωση της αίτησης του εν λόγω παραδείγματος γίνεται με τον ίδιο τρόπο, όπως το αρχικό παράδειγμα. Τα στοιχεία της αίτησης φαίνονται στην εικόνα 7-14.

The image shows a form titled "ΑΙΤΗΣΗ" (Application) with two main sections: "ΠΡΟΣΩΠΙΚΑ ΣΤΟΙΧΕΙΑ" (Personal Data) and "ΓΕΩΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ" (Geographical Data).
ΠΡΟΣΩΠΙΚΑ ΣΤΟΙΧΕΙΑ
 Όνομα: ΑΝΑΣΤΑΣΙΑ
 Επώνυμο: ΓΚΑΓΚΛΟΥ
 ΑΦΜ: 1334556435
 Α.Δ.Τ.: Φ343235
 Όνομα Πατρός: ΑΝΤΩΝΙΟΣ
ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ
 Σταθερό τηλέφωνο: 2111234343
 Κινητό τηλέφωνο: 6932343454
 Email: soulagkaglou@yahoo.gr
ΓΕΩΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ
 Περιοχή: ANADASMOI-POROS
 Συγκεκριμένος Χάρτης: 1990
 Αριθμός αγροτεμαχίου: 280
 Επιλογή περιοχής: POLYGON((688167.2300571240
 Επιλογή σημείου εντός αγροτεμαχίου: 688365.27,4530042.70
 There is a checked checkbox for "Αποδοχή όρων υποβολής αίτησης" and buttons for "Αποστολή αίτησης" and "Άκυρο".

Εικόνα 7-14: Αίτηση 2^{ου} παραδείγματος.

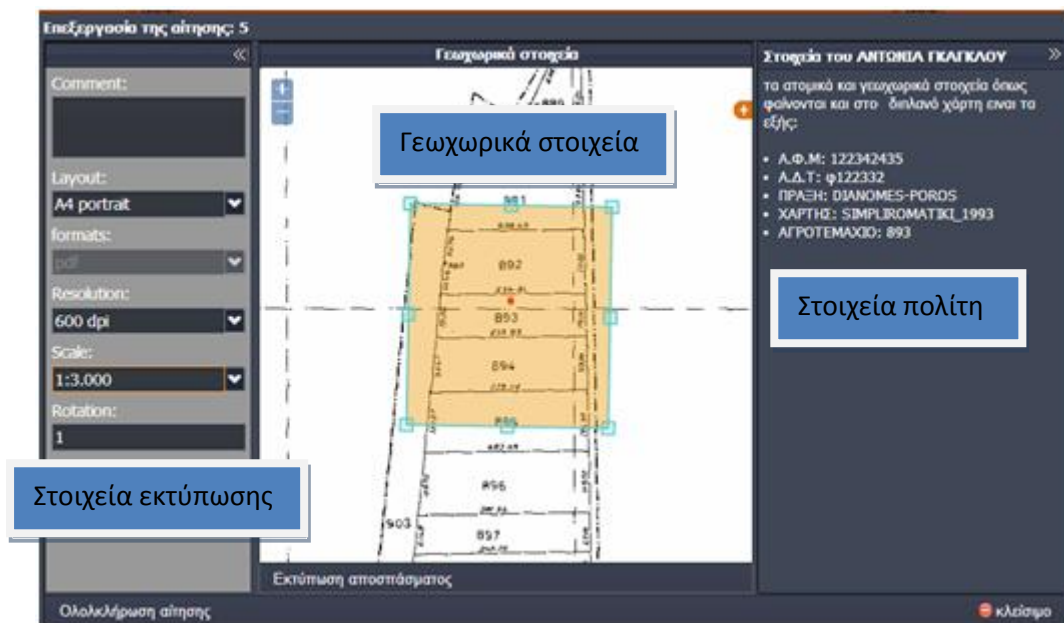
7.2.3. Σύστημα Διαχείρισης για επεξεργασία αιτήσεων

Μόλις ανοίξει ο χρήστης που δημιουργήθηκε από τον διαχειριστή το σύστημα διαχείρισης περιεχόμενου και συγκεκριμένα την ενότητα «Διαχείριση αιτήσεων», αντικρίζει τις αιτήσεις των πολιτών.



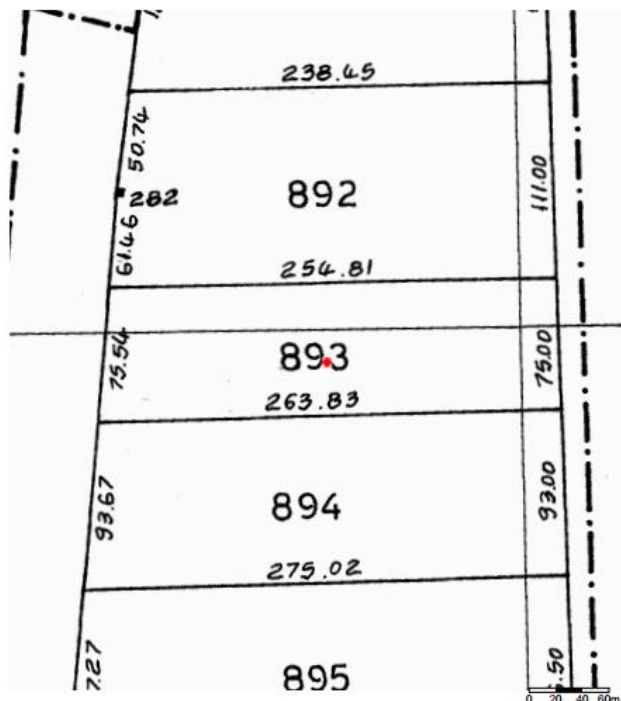
Εικόνα 7-15: Διαχείριση αιτήσεων.

Διαλέγοντας στη συνέχεια μία από τις δύο αιτήσεις και επιλέγοντας το κουμπί «Επεξεργασία» εμφανίζεται η WebGIS εφαρμογή της εικόνας 7-16. Στο συγκεκριμένο παράδειγμα, επιλέγεται η αίτηση υπ' αριθμόν 5 της Αντωνίας Γκάγκλου.



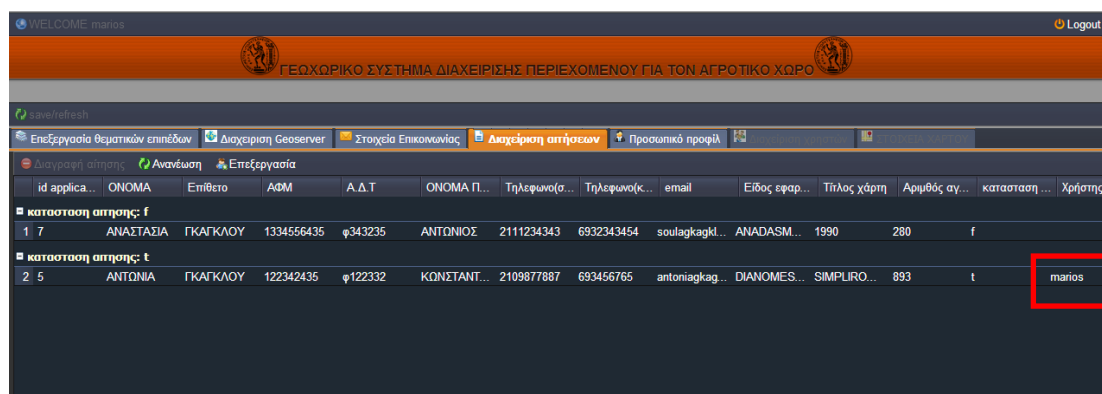
Εικόνα 7-16: Επεξεργασία αίτησης.

Ο χρήστης μπορεί οπτικά να επιλέξει την περιοχή του χάρτη όπου θέλει να αποκοπεί και να επιλέξει την ανάλυσή του σε μονάδες dpi. Επιλέγοντας, επομένως, 600dpi εκτυπώνεται σε μορφή pdf η εικόνα 7-17.



Εικόνα 7-17: Απόκομμα χάρτη συμπληρωματικής διανομής.

Τελειώνοντας, ο χρήστης επιλέγει «Ολοκλήρωση αίτησης», ώστε να κλείσει το αντίστοιχο παράθυρο και να ενημερωθεί η βάση δεδομένων και κατ' επέκταση το σύστημα ότι ο εν λόγω χρήστης ανέλαβε την επεξεργασία.



Εικόνα 7-18: Ολοκλήρωση επεξεργασίας αίτησης.

8. ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΡΟΤΑΣΕΙΣ

Με βάση τη διερεύνηση που έγινε για την εκπόνηση της παρούσας διπλωματικής εργασίας, διαπιστώθηκε ότι τα τελευταία χρόνια υπάρχει ραγδαία εξέλιξη και ενδιαφέρον για την ανάπτυξη ελευθέρων συστημάτων αλλά και τεχνολογιών ανοικτού κώδικα. Συνεπώς, η στροφή προς οικονομικότερες λύσεις είναι πλέον δυνατή και αναγκαία. Επιπροσθέτως, η ανάπτυξη προτύπων τόσο για τα μεταδεδομένα όσο και για τον τρόπο οπτικοποίησης και μεταφοράς των χωρικών δεδομένων στο Διαδίκτυο, οδηγεί στο συμπέρασμα ότι η ύπαρξη κοινής «γλώσσας» για την ανάγνωση και τη διάδοση των διαθέσιμων πληροφοριών είναι από τα πρωταρχικά μελήματα της παγκόσμιας κοινότητας. Λαμβάνοντας υπόψη τα παραπάνω, έγινε προσπάθεια ανάπτυξης ενός γεωχωρικού συστήματος διαχείρισης περιεχομένου, το οποίο θα χρησιμοποιούνταν για την επίλυση ενός προβλήματος (γραφειοκρατικού και χρονοβόρου) που αφορά στην παροχή χαρτών διανομών και αναδασμών στον αγροτικό χώρο. Με δεδομένο ότι η ανάπτυξη του συστήματος ξεκίνησε εκ του μηδενός, δεν βασίστηκε σε έτοιμα συστήματα διαχείρισης περιεχομένου (OpenGeosuite, MapGuide Opensoures, Cartaro, Geonode) και αποτελεί εφαρμογή επιπέδου διπλωματικής εργασίας, το αποτέλεσμα είναι ικανοποιητικό. Ωστόσο, υπάρχουν αρκετά σημεία βελτίωσης σε θέματα που αφορούν στην ασφάλεια, την προσβασιμότητα και τον σχεδιασμό του συστήματος. Επιπλέον, δεν περιέχει δυνατότητες που αφορούν την αναζήτηση δεδομένων βασιζόμενων σε καταλόγους μεταδεδομένων (CSW). Τέλος, δεν αξιοποιήθηκαν πλήρως οι δυνατότητες του GeoServer. Συγκεκριμένα, δεν αξιοποιήθηκαν τα διαθέσιμα plugins που διαθέτει, όπως είναι τα inspire, CSW, WPS και SQL server. Τα πρώτα δύο αφορούν τα μεταδεδομένα. Το WPS αφορά τη δυνατότητα παροχής λειτουργιών ΓΣΠ συμπεριλαμβανομένης της πρόσβασης σε προγραμματισμένους εκ των προτέρων υπολογισμούς και υπολογιστικά μοντέλα, τα οποία σχετίζονται με χωρικά δεδομένα (raster και vector). Το SQL server αφορά τη δυνατότητα δημιουργίας SQL ερωτημάτων με βάση προκαθορισμένες παραμέτρους για τη λήψη συγκεκριμένων χωρικών πληροφοριών.

Μερικές προτάσεις που αφορούν την επέκταση της παρούσας διπλωματικής εργασίας τόσο σε θεωρητικό όσο και σε πρακτικό επίπεδο είναι οι παρακάτω:

- Περαιτέρω διερεύνηση των εμπορικών και ελεύθερων γεωχωρικών συστημάτων διαχείρισης περιεχομένου, καθώς και συγκριτική θεώρηση μεταξύ τους.
- Περαιτέρω διερεύνηση των εμπορικών και ελεύθερων εξυπηρετητών χαρτών με επιπρόσθετη σύγκριση μεταξύ τους.
- Οικονομοτεχνική μελέτη σκοπιμότητας για τη δημιουργία μιας διαδικτυακής εφαρμογής για την επίλυση ενός προβλήματος με εμπορικά και ελεύθερα

συστήματα συμπεριλαμβανομένου του κόστους λογισμικών, χρόνου και εργασίας.

- Επανασχεδιασμός της βάσης δεδομένων του συστήματος βάσει διεθνών προτύπων, όπως είναι το L.A.D.M (Land Administration Domain Model).
- Αξιοποίηση της γνώσης, ώστε να διερευνηθεί και να υλοποιηθεί διαδικτυακή εφαρμογή για το ΕΚΧΑ.

9. BIBΛΙΟΓΡΑΦΙΑ

Clemens Portele. (2007). OpenGIS® Geography Markup Language (GML) Encoding Standard. Open Geospatial Consortium Inc.

Jeff McKenna. (2008). MapServer User's Manual.

Jeff de la Beaujardiere. (2006). OpenGIS® Web Map Server Implementation Specification.

OpenGeo. (2012). Introduction to The OpenGeo Suite.

Schut P., Whiteside A. (2005). *OpenGIS® Web Processing Service*, Open Geospatial Consortium Inc.

Tim Wilson. (2008). *OGC® KML*. Open Geospatial Consortium Inc.

Βέργος Γ.Σ. (2011-2012). *Εφαρμογές Ρυμοτομικών Σχεδίων και Τοπογραφικές Μελέτες*. ΑΠΘ/ΤΑΤΜ.

Βεργόπουλος Κ. (1974). *Το Αγροτικό Ζήτημα στην Ελλάδα*. Εκδόσεις Εξάντας.

Γιαννίου Μ. (2012). *Εξελίξεις στην Εθνική Γεωδαιτική Υποδομή-Ο ρόλος του HEPOS*.

Γρηγοριάδου Ε. (2012). *Διάθεση γεωχωρικών δεδομένων μέσω υπηρεσιών ιστού στις Υποδομές Γεωχωρικών Πληροφοριών*. ΕΜΠ Διατμηματικό Πρόγραμμα Μεταπτυχιακών Σπουδών Γεωπληροφορικής.

Δημοπούλου Ε. (2014). *Βασικές έννοιες-ιστορικό-εξέλιξη κτηματολογίου (Σημειώσεις μαθήματος "ΣΥΣΤΗΜΑΤΑ ΚΤΗΜΑΤΟΛΟΓΙΟΥ ΚΑΙ ΠΟΛΙΤΙΚΗ ΓΗΣ")*.

Ζεντέλης Π. (2011). *Περί ΚΤΗΜΑΤΩΝ ΛΟΓΟΣ και ΚΤΗΜΑΤΟΛΟΓΙΟ Α΄ ΤΟΜΟΣ*. Εκδόσεις Παπασωτηρίου.

Ζεντέλης Π. (2011). *Περί ΚΤΗΜΑΤΩΝ ΛΟΓΟΣ και ΚΤΗΜΑΤΟΛΟΓΙΟ Β΄ ΤΟΜΟΣ*. Εκδόσεις Παπασωτηρίου.

Κάβουρας Μ. (2004). *Αρχές Γεωπληροφορικής και Συστήματα Γεωγραφικών Πληροφοριών*. ΕΜΠ, Σχολή Αγρονόμων και Τοπογράφων Μηχανικών.

Καλλιβρετάκης Λ. (1990). *Η δυναμική του αγροτικού εκσυγχρονισμού στην Ελλάδα του 19ου αιώνα*. Εκδόσεις Μορφωτικό Ινστιτούτο Αγροτικής Τραπεζής.

Καπαγερίδης Ι. (2006). *Εισαγωγή στα Γεωγραφικά Συστήματα Πληροφοριών*.

Κολιός Ν. (2009). *ΕΛ/ΛΑΚ. Χωρική Βάση Δεδομένων PostgreSQL/PostGIS και Σύστημα Γεωγραφικών Πληροφοριών QuantumGIS - Οδηγός Χρήσης*.

Κοντόπουλος Γ. (2010). *Ανάπτυξη διαδικτυακών εφαρμογών GIS με λογισμικό ανοιχτού κώδικα(Geoserver)*. Μεταπτυχιακή εργασία. Πανεπιστήμιο Μακεδονίας.

ΚΤΗΜΑΤΟΛΟΓΙΟ Α.Ε, HEPOS. (2009). *Οδηγίες χρήσης του Ελληνικού Συστήματος Εντοπισμού HEPOS για την εκτέλεση των συμβάσεων Β' φάσης Κτηματογράφησης*.

ΚΤΗΜΑΤΟΛΟΓΙΟ Α.Ε. (2012). *ΟΔΗΓΙΑ ΚΑΤΑΧΩΡΙΣΗΣ ΚΥΡΩΜΕΝΩΝ ΑΝΑΔΑΣΜΩΝ ΣΤΟ ΚΤΗΜΑΤΟΛΟΓΙΟ*.

Λάμπρου Ε., Πανταζής Γ. (2010). *Εφαρμοσμένη Γεωδαισία. 2^η έκδοση διορθωμένη*. Εκδόσεις Ζήτη.

Λουκάς Α., Γκιόλης Σ. (2007). *Εισαγωγή στα Γεωγραφικά Συστήματα Πληροφοριών: βασικές έννοιες, δεδομένα, σχεδιασμός και διεθνής εμπειρία*. Βολος.

Μαστόρης Δ., Μητροπούλου Ελ. (2010). *Το ελληνικό σύστημα εντοπισμού HEPOS-οριζόντιες και δυνατότητες χρήσης του για εφαρμογές GIS*.

Μανσόλα Α. (1867). *Πολιτειογραφικά πληροφορίες περί Ελλάδος*. Εκδόσεις Εθνικών Τυπογραφείων.

Μαραβέγιας Ν. (1992). *Αγροτική Πολιτική και Οικονομική Ανάπτυξη στην Ελλάδα*. Εκδόσεις Νέα Σύνορα.

Μάρκου Ε. (2009). *Οι Αναδασμοί στον Ελληνικό Χώρο-Διερεύνηση καταλληλότητας των Διαγραμμάτων Αναδασμών/ Διανομών για το Εθνικό Κτηματολόγιο*. Διπλωματική εργασία.ΕΜΠ.

Μήλας Π., Μπιλλήρης Χ. (2006). *Μετατροπές Συστημάτων Αναφοράς και Προβολικών Συστημάτων - Υψομετρικά Συστήματα Αναφοράς*. ΕΜΠ. Σχολή Αγρονόμων και Τοπογράφων μηχανικών. Αθήνα

Πότσιου Χ. (2008). *Ο Αναδασμός ως Εργαλείο Διαχείρισης Γης*.

Σαριφίδης Δ. (2011). *Οδηγία INSPIRE, μεταδεδομένα και GIS*. Τεχνικό επιμελητήριο Ελλάδας, τμήμα Κεντρικής Μακεδονίας.

Στεφανάκης Ε. (2003). *Βάσεις Γεωγραφικών Δεδομένων και Συστήματα Γεωγραφικών Πληροφοριών*. Εκδόσεις Παπασωτηρίου.

Σωτηρίου Ι. (2010). *Σύγκριση και σύνθεση λογισμικών WebGIS: Μελέτη περίπτωσης του δήμου Περάματος*. Διπλωματική Εργασία, ΔΠΜΣ Γεωπληροφορικής, Αθήνα.

Τάκου Β. (2008). *Διαδικτυακή εφαρμογή διαχείρισης παρουσιάσεων*. Πανεπιστήμιο Μακεδονίας, τμήμα Εφαρμοσμένης Πληροφορικής.

Τζιαχρής Π. *Γεωχωρικές υπηρεσίες και ηλεκτρονική διακυβέρνηση με την χρήση ανοικτών προτύπων και λογισμικού ανοικτού κώδικα (OSS)*. Θέρμη-Θεσσαλονίκη.

Φρέντζος Η. (2009-2010). *Εισαγωγή στην PostgreSQL-postGIS*. Πανεπιστήμιο Πειραιώς, τμήμα Πληροφορικής.

Χάλαρης Γ. (2004). *Τα Μεταδεδομένα και ο ρόλος τους στην Εθνική Υποδομή Γεωγραφικών Πληροφοριών (Ε.Υ.Γε.Π)*.

<http://www.hepos.gr/>

<http://cartaro.org/documentation>

http://live.osgeo.org/el/quickstart/geonode_quickstart.html

http://users.sch.gr//tsibinos/internet_history/history_1.html

<http://www.php.net/manual/en/>

http://php.net/license/3_01.txt

<https://php.net/manual/en/index.php>

<http://live.osgeo.org/el/overview/overview.html>

<http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications/>

<http://www.opengeospatial.org/standards/gml>.

<http://geojson.org/geojson-spec.html>

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

<http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>

<http://www.sencha.com/legal/sencha-sdk-software-license-agreement>

<http://www.gnu.org/copyleft/gpl.html>

<http://www.postgresql.org/about/licence/>

<http://opensource.org/licenses/BSD-2-Clause>

<http://opensource.org/licenses/MIT>

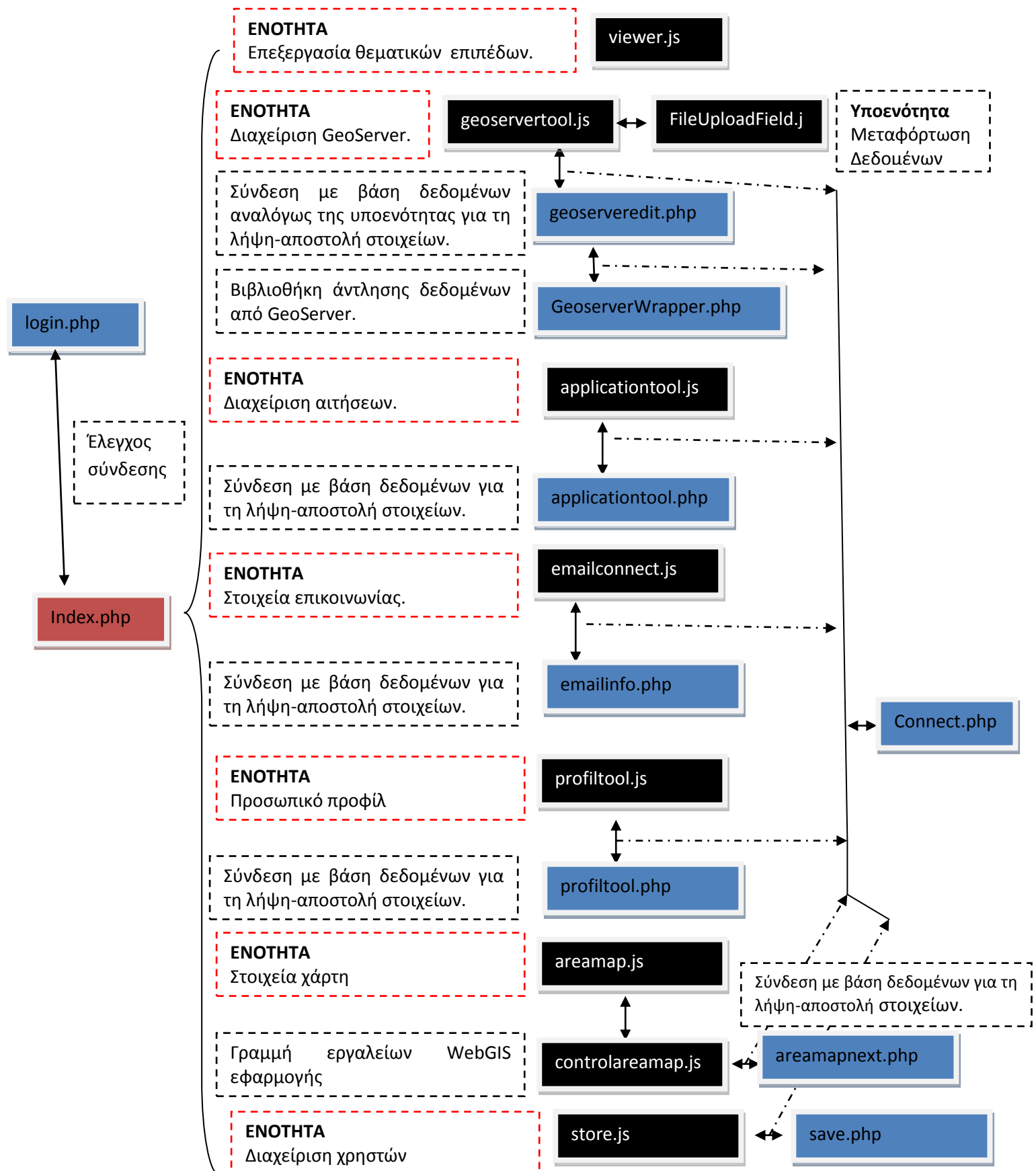
<http://docs.geoserver.org/stable/en/user/>

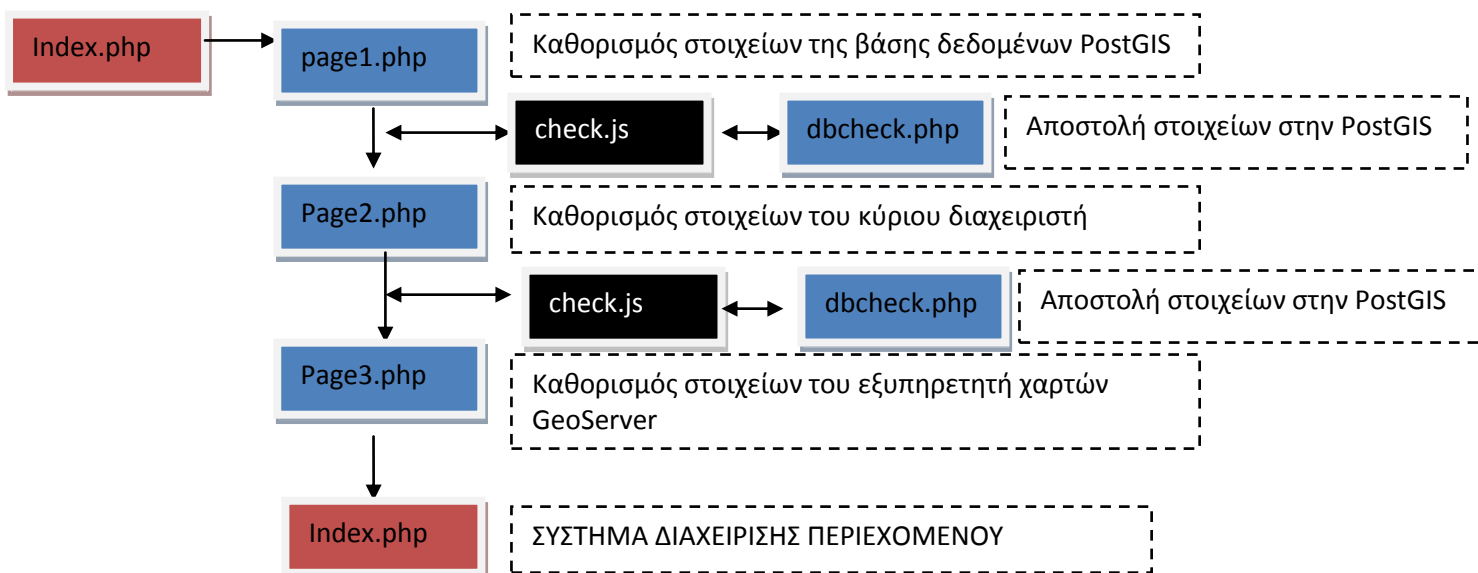
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>

https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model

ΠΑΡΑΡΤΗΜΑ (ΣΚΕΛΕΤΟΣ ΣΥΣΤΗΜΑΤΟΣ)

ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ



ΕΦΑΡΜΟΓΗ ΑΙΤΗΣΗΣ ΧΑΡΤΩΝ**ΕΦΑΡΜΟΓΗ ΓΙΑ ΕΓΚΑΤΑΣΤΑΣΗ ΣΥΣΤΗΜΑΤΟΣ****ΒΙΒΛΙΟΘΗΚΕΣ**

proj4js
ext-3.4.0
GeoExt
OpenLayers
Jquery

ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ

```

1 <html>
2 <head>
3 <title>ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ</title>
4 <link rel="icon" href="images/page.png" type="image/x-icon">
5 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
6 <link rel="stylesheet" type="text/css" href="css/main.css"/>
7 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/resources/css/ext-all.css" />
8 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/grid/grid-examples.css" />
9 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/shared/examples.css" />
10 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/openlayers/theme/default/style.css" />
11 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/openlayers/theme/default/google.css" />
12 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/geoext/resources/css/geoext-all.css" />
13 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/resources/css/xtheme-access2.css" />
14 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/ux/css/RowEditor.css" />
15 <link rel="stylesheet" type="text/css" href="../external/gxp-master/src/theme/all.css"/>
16 <script type="text/javascript" src="lib/cookieslibrary.js"></script>
17 <script type="text/javascript" src="../external/proj4js/proj4js-compressed.js"></script>
18 <script src="../external/gxp-master/externals/openlayers/lib/OpenLayers.js"></script>
19 <script type="text/javascript" src="../external/function.js"></script>
20 <script type="text/javascript" src="/geoserver/pdf/info.json?var=printCapabilities"></script>
21 <?php include "../areamap/area.php"; ?>
22 <script type="text/javascript" >
23 Proj4js.defs ["EPSG:4121"] ="+proj=tmerc +lat_0=0 +lon_0=24 +k=0.9996 +x_0=500000 +y_0=0 +ellps=GRS80 +towgs84=-199.87,74.79,24
24 6.62,0,0,0,0 +units=m +no_defs";
25
26 var items=[];
27 var boundarymap="<?php echo $area; ?>";
28
29 var vectors = new OpenLayers.Layer.Vector("vector", {isBaseLayer: false});
30 var feature2 = new OpenLayers.Feature.Vector(
31     OpenLayers.Geometry.fromWKT (
32         ""+ boundarymap +""
33     )
34 );
35 vectors.addFeatures([feature2]);
36 var ft = vectors.features;
37 for(var i=0; i< ft.length; i++){
38     var boundary=vectors.features[i].geometry.getBounds().toArray();
39 };
40
41 </script>
42
43 <?php
44
45 if (isset($_COOKIE['user1']) && isset($_COOKIE['pass1'])) { //den to elenxo an einai sosto
46 //setcookie("users", "gsdfgsdf", time()-3600);
47
48 ?>
49
50 <script type="text/javascript" >
51
52 var namewelcome=false;
53 var logincheck=true;
54 var logoutcheck=false;
55 var processcheck=false;
56 var users1=getCookie('user1');
57 var password=getCookie('pass1');
58
59 </script>
60
61 <?php
62 if(isset($_COOKIE['mainadmin']))
63 {
64
65 ?>
66 <script type="text/javascript" >
67 var processcheck1=false;
68
69 </script>
70 <?php
71 }
72 else
73 {
74 ?>
75 <script type="text/javascript" >
76 var processcheck1=true;
77

```



```

78     </script>
79     <?php
80
81     }
82
83 }
84 else
85 {
86
87     ?>
88     <script type= "text/javascript" >
89         var namewelcome=true;
90     var logincheck=false;
91     var logoutcheck=true;
92     var processcheck=true;
93     var processcheck1=true;
94     var users1="";
95     </script>
96     <?php
97 }
98     ?>
99
100     <script src= "../external/ext-3.4.0/adaptor/ext/ext-base.js" ></script>
101     <script src= "../external/ext-3.4.0/ext-all.js" ></script>
102     <script src= "../external/gxp-master/externals/geoext/lib/GeoExt.js" ></script>
103     <script src= "../external/gxp-master/externals/geoext/lib/overrides/override-ext-ajax.js" ></script>
104     <script src= "../external/gxp-master/src/script/loader.js" ></script>
105     <script type= "text/javascript" src= "../external/gxp-master/examples/viewer/ux/RowExpander.js" ></script>
106     <script type= "text/javascript" src= "viewer.js" ></script>
107     <link rel= "stylesheet" type= "text/css" href= "../external/gxp-master/src/theme/all.css" />
108     <link rel= "stylesheet" type= "text/css" href= "../external/gxp-master/examples/viewer.css" />
109     <script type= "text/javascript" src= "../external/jquery-1.11.0.min.js" ></script>
110     <link href= "css/style_login.css" rel= "stylesheet" type= "text/css" />
111     <script src= "../external/gxp-master/src/script/locale/el.js" ></script>
112     <script type= "text/javascript" src= "../external/ext-3.4.0/examples/ux/RowEditor.js" ></script>
113     <script type= "text/javascript" src= "lib/profiltool.js" ></script>
114     <script type= "text/javascript" src= "store.js" ></script>
115     <script type= "text/javascript" src= "http://localhost/geoserver/web/resources/org.geoserver.printing.PrintDemoPage/GeoExtPri
nting.js" ></script>
116     <script type= "text/javascript" src= "lib/applicationtool.js" ></script>
117     <script type= "text/javascript" src= "lib/FileUploadField.js" ></script>
118     <script type= "text/javascript" src= "lib/geoservertools.js" ></script>
119     <script type= "text/javascript" src= "lib/controlareamap.js" ></script>
120     <script type= "text/javascript" src= "lib/areamap.js" ></script>
121     <script type= "text/javascript" src= "lib/emailconnect.js" ></script>
122
123 <script type= "text/javascript" >
124
125 Ext.onReady (function () {
126     //////////////////////////////////////////////////energopoihsh grid
127
128     var profilgrid=Ext.getCmp('profilgrid');
129     var gridaction=Ext.getCmp('gridaction');
130
131     var grid = Ext.getCmp('grid');
132     //////////////////////////////////////////////////
133     var loginitems = new Ext.FormPanel({
134         labelWidth : 80,
135         url:'login.php' ,
136         frame:true,
137         id:'loginForm',
138         items:[
139             {
140                 xtype:'textfield' ,
141                 fieldLabel : 'username' ,
142                 name : 'User' ,
143                 autoHeight : true,
144                 id:'Username'
145             },
146             {
147                 xtype:'textfield' ,
148                 fieldLabel : 'password' ,
149                 inputType : 'password' ,
150                 name : 'Pass' ,
151                 id:'Password'
152             },
153             {
154                 xtype:'checkbox' ,

```

```

155     fieldLabel : 'remeberme' ,
156     //disabled:true,
157     hidden : true,
158     id: 'remember',
159     name: 'remember',
160     checked: true
161
162
163 },
164
165 {
166     xtype : 'checkbox' ,
167     fieldLabel : 'Διαχειριστής' ,
168     id : 'commander' ,
169     name : 'commander' ,
170     checked : false
171
172
173
174 }
175 , {
176     xtype : 'button' ,
177     iconCls : 'login' ,
178     name : 'login' ,
179     text : 'Σύνδεση' ,
180     width : 80,
181     formBind : true,
182     handler: function() {
183         loginitems.getForm ().submit({
184             method: 'POST',
185             waitTitle: 'Connecting',
186             waitMsg: 'Sending data...',
187             success: function(form, action){
188                 Ext.Msg.alert('status', 'Επιτυχημένη σύνδεση!', function(btn, text){
189                     if (btn == 'ok'){
190
191                         //////////
192                         obj = Ext.util.JSON.decode (action.response.responseText);
193
194                         var loginwindow = Ext.getCmp('loginwindow');
195                         var logout = Ext.getCmp('logoutButton');//logout se orato
196                         logout.setVisible(true);
197
198                         var login = Ext.getCmp('loginButton');//bazo to login se mi orato
199                         login.setVisible(false)
200
201                         var editusers = Ext.getCmp('editusers');
202                         editusers.setDisabled(obj.noeditable);//eleynxos tis dinatothtas diaxeirishw
203
204
205                         var welcomeusers1 = Ext.getCmp('users1');
206                         welcomeusers1.setVisible(true);
207                         var Us = Ext.getCmp('Username').getValue();
208                         welcomeusers1.setText('WELCOME '+Us);//emafanisi minimatos welcome
209
210                         var mappanels = Ext.getCmp('mappanels');
211                         mappanels.setDisabled(false);//dinatotita diaxeirishw me login
212
213                         var mapclue = Ext.getCmp('mapclue');
214                         mapclue.setDisabled(obj.noeditable);
215
216                         //////////koumpi diagrafis application
217                         var deleteapply = Ext.getCmp('deleteapply');
218                         deleteapply.setDisabled(obj.noeditable);
219
220
221                         var goeserver = Ext.getCmp('goeserver');
222                         goeserver.setDisabled(false);
223                         var connecting = Ext.getCmp('connecting');
224                         connecting.setDisabled(false);
225                         //ginete afairesi tou emailconnect prokeimenou na ksana mpei analoga tin prosbasi
226                         //poy elenxei h sunrtisi emailconnecting(processcheck1)
227                         connecting.remove('emailconnect', true);
228                         processcheck1=obj.noeditable;
229
230                         connecting.add(emailconnecting(processcheck1));
231                         connecting.doLayout();
232

```

```

233     var apply = Ext.getCmp('apply');
234     apply.setDisabled(false);
235
236     var profil = Ext.getCmp('profil');
237     profil.setDisabled(false);
238
239
240
241
242     //deleteapply
243
244     loginwindow.hide();
245
246 } // button ok
247 }); //message
248 }, //success function
249 failure: function(form, action){
250     if(action.failureType == 'server'){
251         obj = Ext.util.JSON.decode(action.response.responseText);
252         Ext.Msg.alert('Login Failed!', obj.errors.reason);
253     } else {
254         Ext.Msg.alert('Warning!', 'Authentication server is unreachable : ' + action.response.responseText);
255     }
256     var loginitems = Ext.getCmp('loginForm').getForm();
257     loginitems.reset();
258 }
259
260 }); //submit
261
262 }
263 }
264 ]
265 });
266 //kathorismos twon stoixeiwn toy tabpanel
267 items=[{xtype:'panel',title:'Επεξεργασία θεματικών επιπέδων',html:'<div id="tab"></div>',disabled:processcheck,id:'mappanels',
268     iconCls: 'editlayer' },
269
270
271 {
272
273     xtype:'panel' ,
274     layout:'border' ,
275     // html:'asdfas' ,
276     title:' Διαχείριση Geoserver' ,
277     iconCls: 'geoserver' ,
278     disabled :processcheck,
279     id:'goeserver' ,
280     items :Geoserver
281
282     },{
283
284     xtype:'panel' ,
285     //id:'emailconnect',
286     // layout:'border',
287     // html:'asdfas' ,
288     title:' Στοιχεία Επικοινωνίας' ,
289     layout: 'border' ,
290     iconCls: 'email' ,
291     width :this.width,
292     height: this.height,
293     disabled:processcheck,
294     id:'connecting',
295     items:emailconnecting(processcheck1)
296
297     },{
298         ////////////////////////////////////////////////////////////////////den einai aitimo
299         xtype:'panel' ,
300         id:'apply' ,
301         iconCls: 'apply' ,
302         disabled :processcheck,
303         title:' Διαχείριση αιτήσεων' ,
304         layout: 'border' ,
305
306         layoutConfig : {
307             columns : 1
308
309         },
310         width:this.width,

```



```

311         height: this.height,
312         items: [gridaction]
313         //items:emailconnecting(processcheck1)
314
315
316         },{
317             xtype : 'panel' ,
318             id : 'profil' ,
319             title : 'Προσωπικό προφίλ' ,
320             iconCls : 'profil' ,
321             disabled : processcheck,
322             layout : 'border' ,
323             width : this.width,
324             height: this.height,
325             items:[profilgrid]
326
327
328
329
330             },{xtype:'panel',title:'Διαχείριση χρηστών',disabled:processcheck1,id:'editusers',iconCls: 'users',
331
332
333             layout: 'border',
334
335             //layoutConfig: {
336                 // columns: 1
337
338                 //
339             },
340             width:this.width,
341             height: this.height,
342             items:[grid]
343
344         },{
345             xtype : 'panel' ,
346             layout : 'border' ,
347             title : 'ΣΤΟΙΧΕΙΑ ΧΑΡΤΟΥ' ,
348
349             iconCls : 'areamap' ,
350             disabled : processcheck1,
351             id : 'mapclue' ,
352             items : areamap
353
354         }
355     ];
356     var viewport=new Ext.Viewport({layout:"border",id:'viewport',items:[{
357         xtype : 'panel' ,
358         region : 'north' ,
359         height : this.height,
360         id: 'headerPanel',
361         html:'<h1 id="ff-proof2" class="ribbonTitle"><font size=4>ΓΕ
ΩΧΩΡΙΚΟ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ ΓΙΑ ΤΟΝ ΑΓΡΟΤΙΚΟ ΧΩΡΟ</font></h1>',
362         //html:'adfasgasdfasdfas',
363         tbar: [{text:'WELCOME '+ users1,iconCls:'usertext',disabled:true,
364             id:'users1',hidden :namewelcome}],->,
365
366         {
367             xtype : 'button' ,
368             id : 'loginButton' ,
369             iconCls : 'login' ,
370             margin : 6,
371             text : 'Login' ,
372             hidden : logincheck,
373             handler : function(){
374                 loginwindow =new Ext.Window({
375                     id:'loginwindow' ,
376                     title : ' ΕΙΣΟΔΟΣ ΣΤΗ ΔΙΑΧΕΙΡΙΣΗ' ,
377                     layout : 'fit' ,
378                     width : 300,
379                     height : 220,
380
381                     closable : false,
382                     draggable: false,
383                     resizable: false,
384
385                     // plain: true,
386                     border: false,

```

```

387 bbar: [{
388     xtype: 'tbtext' ,
389     text: ' © 2014 - ΔΙΑΧΕΙΡΙΣΗ'
390 }],
391 items:loginitems
392
393 });//windows
394 loginwindow.show();
395 Ext.getCmp('loginButton').setDisabled(true);
396
397 //function
398 },
399 {
400     xtype: 'button' ,
401     id: 'logoutButton' ,
402     iconCls: 'logout' ,
403     hidden: logoutcheck,
404     margin: 6,
405     text: 'Logout' ,
406     handler: function(){
407         Ext.Msg.alert('Status', 'Αποσύνδεση!', function(btn, text){
408             if (btn == 'ok'){
409                 // alert(users1);
410
411                 document.cookie = "pass1=ff; expires=Thu, 01 Jan 1970 00:00:00 GMT;" ;
412                 document.cookie = "user1=ff; expires=Thu, 01 Jan 1970 00:00:00 GMT;" ;
413
414                 window.location ="index.php?" ;
415             }
416         })
417     })
418 }
419 }
420 }
421
422 ],bbar:[{text:'save/refresh',xtype:'button',iconCls:'refresh',
423 id:'refreshsave',
424 disabled:true,
425 handler: function(){
426     window.location.reload ();
427
428     }}}
429 ],{xtype:'tabpanel',activeTab:0,region:'center',id:'centertabpanel',
430 items:items
431
432 },{
433     xtype: 'tbtext' ,
434     //xtype:'button',
435     region: 'south' ,
436     text: '<center> © 2014 - Ε.Μ.Π Διπλωματική Εργασία - Πιστιδίκης Ιωάννης-Γεωχωρικό Σύστημα Διαχείρισης</
center>'
437
438     }}
439 });
440
441 });
442 </script>
443
444 </head>
445 <body></body>
446 </html>
447

```

```

1  <?php
2
3  require_once('../connect/connect.php');//pernei stoxeia sindesis
4  $dbconn3 = pg_connect("host=$Server port=$Port dbname=$Database user=$Username password=$Password")or die('connection failed');
5
6
7
8
9  $loginUsername = isset($_POST["User"]) ? $_POST["User"] : "";
10 $loginPassword = isset($_POST["Pass"]) ? $_POST["Pass"] : "";
11
12
13
14  if (isset($_POST['commander'])){//gia kurio diaxeiristi
15
16      $sql="SELECT \"UserName\",\"PassWord\" FROM \"public\".\"Admin\" WHERE \"UserName\"='".$loginUsername.'" AND \"PassWord\"='".$loginPassword.'" ";
17      $result=pg_query($dbconn3, $sql);
18      $i=pg_num_rows ($result);
19
20      if($i> 0){
21          if (isset($_POST['remember'])) {//gia cookies
22              setcookie('user1', $loginUsername, time()+3600);
23              setcookie('pass1', md5($loginPassword), time()+3600);
24              setcookie('mainadmin',"good",time()+3600);
25          }
26          else{// na min uparxoun cookies
27              {
28                  setcookie('user1', $loginUsername, false);
29                  setcookie('pass1', md5($loginPassword), false);
30                  setcookie('mainadmin',"good",false);
31              }
32          }
33
34          echo "{success: true,noeditable:false}";
35      }
36      else
37      {
38          echo "{success: false, errors: { reason: 'Η σύνδεση απέτυχε. Δοκίμασε ξανά.' }}";
39      }
40
41  }//telos if commander
42  else//otna den einai diaxeirisths
43  {
44      $sql="SELECT \"UserName\",\"PassWord\" FROM \"public\".\"Users\" WHERE \"UserName\"='".$loginUsername.'" AND \"PassWord\"
45      ='".$loginPassword.'" ";
46      $result=pg_query($dbconn3, $sql);
47      $i=pg_num_rows ($result);
48
49      if($i> 0){
50          if (isset($_POST['remember'])) {//gia cookies
51              setcookie('user1', $loginUsername, time()+3600);
52              setcookie('pass1', md5($loginPassword), time()+3600);
53              setcookie('mainadmin',"good",time()-3600);
54          }
55          else{// na min uparxoun cookies
56              {
57                  setcookie('user1', $loginUsername, false);
58                  setcookie('pass1', md5($loginPassword), false);
59              }
60          }
61
62          echo "{success: true,noeditable:true}";
63      }
64      else
65      {
66          echo "{success: false, errors: { reason: 'Η σύνδεση απέτυχε. Δοκίμασε ξανά.' }}";
67      }
68
69  }
70
71
72
73  ?>

```

```

1  var app;
2
3  Ext.onReady(function () {
4
5      app = new gxp.Viewer({
6          proxy : "/geoserver/rest/proxy?url=" ,
7          xtype : "panel" ,
8          portalConfig : {
9              renderTo : "tab" ,
10             layout : "border" ,
11             id : "viewer" ,
12             width : this.width,
13             height : 490,
14
15             items: [{
16                 // a TabPanel with the map and a dummy tab
17                 id : "centerpanel" ,
18                 xtype : "tabpanel" ,
19
20                 region : "center" ,
21                 activeTab : 0, // map needs to be visible on initialization
22                 border : true,
23                 items : ["mymap"]
24             }, {
25                 // container for the FeatureGrid
26                 id : "south" ,
27                 xtype : "panel" ,
28                 collapsible : true,
29                 split : false,
30                 layout : "fit",
31                 region : "south",
32                 height : 150
33             }, {
34                 region : "east" ,
35                 xtype : "panel" ,
36                 id : "east" ,
37                 collapsible : true,
38                 split : false,
39                 layout : "fit",
40                 // layoutConfig: { animate: true},
41                 //autoScroll: true,
42                 width : 200
43             }, {
44                 // container for the queryform
45                 id : "west" ,
46                 xtype : "panel" ,
47                 collapsible : true,
48                 split : false,
49                 layout : "fit",
50                 // layoutConfig: { animate: true},
51                 //autoScroll: true,
52                 region : "west",
53                 width : 200
54             }, {
55                 bbar : {id : "mybbar"}
56             }], //portalconfig
57
58             // configuration of all tool plugins for this application
59             tools : [{
60
61                 ptype : "gxp_legend" ,
62                 outputTarget : "east"
63             },
64
65             {
66                 ptype : "gxp_loadingindicator"
67             }, {
68                 ptype : "gxp_layertree" ,
69                 outputConfig : {
70                     id : "tree" ,
71                     border : true,
72                     tbar : [] // we will add buttons to "tree.bbar" later
73                 },
74                 outputTarget : "west"
75             }, {
76                 ptype : "gxp_addlayers" ,

```



```

79     actionTarget : "tree.tbar"
80 }, {
81     ptype : "gxp_removevlayer" ,
82     actionTarget : ["tree.tbar", "tree.contextMenu"]
83 }, {
84     ptype : "gxp_layerproperties" ,
85     actionTarget : ["tree.tbar", "tree.contextMenu"]
86 }, {
87     ptype : "gxp_zoomtoextent" ,
88     actionTarget : "map.tbar"
89 }, {
90     ptype : "gxp_zoom" ,
91     actionTarget : "map.tbar"
92 }, {
93     ptype : "gxp_navigationhistory" ,
94     actionTarget : "map.tbar"
95 }, { ptype: "gxp_styler",
96     actionTarget: "map.tbar",
97     width:200,
98     height:200
99
100
101
102 },
103
104 {
105     ptype : "gxp_wmsgetfeatureinfo" ,
106     outputConfig : {
107         width : 400,
108         height : 200
109     },
110     actionTarget: "map.tbar", // this is the default, could be omitted
111     toggleGroup: "layertools"
112 }, {
113     ptype : "gxp_mapproperties" ,
114     outputConfig : {
115         title : 'Map properties'
116     }
117 },
118 //////////////////////////////////////////////////
119 {
120     ptype : "gxp_measure" ,
121     actionTarget : "map.tbar"
122 },
123
124
125 {
126     // shared FeatureManager for feature editing, grid and querying
127     ptype : "gxp_featuremanager" ,
128     id : "featuremanager" ,
129     maxFeatures : 20
130 }, {
131     ptype:"gxp_zoomtolayerextent" ,
132     featureManager : "featuremanager" ,
133     appendActions : true,
134     actionTarget: ["tree.contextMenu"]
135 }, {
136     ptype : "gxp_featureeditor" ,
137     featureManager : "featuremanager" ,
138     autoLoadFeature : true, // no need to "check out" features
139     outputConfig: {panIn: false},
140     toggleGroup: "layertools"
141 }, {
142     ptype : "gxp_featuregrid" ,
143     featureManager : "featuremanager" ,
144     outputConfig : {
145         id : "featuregrid"
146     },
147     outputTarget: "south"
148 }, //////////////////////////////////
149
150 //////////////////////////////////////////////////
151 {
152     ptype : "gxp_queryform" ,
153     featureManager : "featuremanager" ,
154     outputConfig : {
155         title : "Query" ,
156         width : 320

```



```

157     },
158     actionTarget: ["featuregrid.bbar", "tree.contextMenu"],
159     appendActions: false
160 }
161
162 ],
163
164 // layer sources
165 defaultSourceType: "gxp_wmssource",
166 sources: { ol: {
167     ptype : "gxp_olsource"
168 },
169
170     "osm": {
171         ptype : "gxp_osmsource"
172     },
173     "Διαθέσιμα δεδομένα": {
174         url : "/geoserver/wms/" ,
175         version : "1.1.1"
176     },
177     "bing": {
178         ptype : "gxp_bingsource"
179     }
180 }
181
182
183 },
184 // map and layers
185 map: {
186     id: "mymap" , // id needed to reference map in portalConfig above
187     title: "ΕΠΕΞΕΡΓΑΣΙΑ ΧΑΡΤΩΝ" ,
188     projection : "EPSG:900913" ,
189     units : "m" ,
190
191     restrictedExtent : [boundary[0], boundary[1], boundary[2], boundary[3]],
192
193     center: [(boundary[0]+boundary[2])/2,(boundary[1]+boundary[3])/2],
194     zoom: 6,
195
196     controls: [
197         new OpenLayers.Control.Zoom(),
198         new OpenLayers.Control.Attribution(),
199         new OpenLayers.Control.Navigation()
200     ],
201     layers: [{
202         source : "ol" ,
203         type: "OpenLayers.Layer" ,
204         type: "OpenLayers.Layer.OSM" ,
205     args: ["OpenStreetMap"],
206         // args: ["Blank"],
207         visibility: false,
208         group: "background"
209     },
210
211     {
212         source : "osm" ,
213         name: "osmarander" ,
214         visibility : true
215     },
216
217
218 {
219     source: "bing",
220     title: "Bing Map Satelite",
221     name: "Aerial"
222 }
223 ]
224 }
225 }/map
226
227 });
228
229
230
231
232 });
233

```

```

1  //////////////////////////////////////////////////json and store of workspace store and layer////////////////////////////////////
2  //////////////////////////////////////////////////workspace.json
3  var data = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/workspaces.json");
4  var data1=Ext.decode(data);
5  var data2=data1["workspaces"]["workspace"];
6  if(data2)
7  {
8
9  }
10 else{
11     data3={"workspaces":{"workspace":[{"name":"empty","href":"empty"}]}} ;
12     var data1=Ext.decode(data3);
13     var data2=data1["workspaces"]["workspace"];
14 }
15
16 //////////////////////////////////////////////////workspace store  namespacedata2
17 var mystore = new Ext.data.JsonStore({
18     id:'mystore' ,
19     //autoload: true,
20     fields : [
21         {type: 'string', name: 'href'},
22         {type: 'string', name: 'name'}
23     ]
24 });
25
26 mystore.loadData(data2);
27
28 //////////////////////////////////////////////////namespace.json
29
30 //////////////////////////////////////////////////namespace store
31
32
33 var namespacestore = new Ext.data.JsonStore({
34     id:'namespacestore' ,
35     //autoload: true,
36     fields : [
37         {type: 'string', name: 'href'},
38         {type: 'string', name: 'name'}
39     ]
40 });
41
42
43
44
45 //////////////////////////////////////////////////layer gia namespace
46 var namespacestore1 = new Ext.data.JsonStore({
47     id:'namespacestore1' ,
48     //autoload: true,
49     fields : [
50         {type: 'string', name: 'href'},
51         {type: 'string', name: 'name'}
52     ]
53 });
54
55
56 //////////////////////////////////////////////////layer gia layer
57 var layerstore = new Ext.data.JsonStore({
58     id:'layerstore' ,
59     //autoload: true,
60     fields : [
61         {type: 'string', name: 'href'},
62         {type: 'string', name: 'name'}
63     ]
64 });
65
66 var layers = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/layers.json");
67 var layers1=Ext.decode(layers);
68 var layers2=layers1["layers"]["layer"];
69 if(layers2)
70 {
71
72 }
73 else{
74     layers3={"layers":{"layer":[{"name":"empty","href":"empty"}]}} ;
75     var layers1=Ext.decode(layers3);
76     var layers2=layers1["layers"]["layer"];
77 }
78 layerstore.loadData (layers2);

```

```

79
80 //////////////////////////////////////////////////store gia vector
81 var namespacestorevector = new Ext.data.JsonStore({
82     id:'namespacestorevector' ,
83     //autoload: true,
84     fields : [
85         {type: 'string', name: 'href'},
86         {type: 'string', name: 'name'}
87     ]
88 });
89
90
91 //////////////////////////////////////////////////createpostgisdatastore////////////////////////////////////////////////
92
93 var createpostgisdatastoreform=new Ext.FormPanel({
94     labelWidth :160,
95     url : 'lib/geoserveredit.php?postgis=1' ,
96     frame :true,
97     id: 'createpostgisdatastoreform',
98     items:[
99         {
100             xtype : 'combo' ,
101             fieldLabel : 'workspace name ' ,
102             editable :false,
103             name: 'WorkspacePostgis',
104             id:'WorkspacePostgis',
105             displayField: 'name',
106             valueField: 'name',
107             mode: 'local',
108             triggerAction: 'all',
109             listClass: 'comboalign',
110             typeAhead: true,
111             forceSelection: true,
112             selectOnFocus: true,
113             store: mystore,
114             listeners: {
115                 change : function() {
116
117
118                 }
119             }
120         },
121         {
122             xtype : 'textfield' ,
123             fieldLabel : 'Όνομα PostGIS αποθήκης ' ,
124             name : 'postgisstore' ,
125             allowBlank :false,
126             blankText: 'Το όνομα της αποθήκης είναι υποχρεωτικό',
127             autoHeight: true
128             //id:'GUser'
129         },
130         {
131
132             xtype : 'textfield' ,
133             fieldLabel : 'Όνομα Βάσης Δεδομένων ' ,
134             allowBlank :false,
135             blankText: 'Το όνομα της βάσης δεδομένων είναι υποχρεωτικό',
136             name: 'postgisname',
137             autoHeight: true
138
139         },
140     ],
141
142     {
143         xtype : 'textfield' ,
144         fieldLabel : 'Geoserver username ' ,
145         name: 'geoserveruserpostgis' ,
146         autoHeight : true
147         //id:'GUser'
148     }, {
149         xtype : 'textfield' ,
150         fieldLabel : 'Geoserver password ' ,
151         inputType : 'password' ,
152         name: 'geoserverpasspostgis'
153         // autoHeight: true
154         //id:'GPass'
155     }, {
156         xtype : 'button' ,

```



```

157 name: 'createpostgis' ,
158 text: 'Δημιουργία PostGIS' ,
159 iconCls: 'saveedit' ,
160 width: 80,
161 formBind: true,
162 handler: function() {
163     var formpostgis = Ext.getCmp('createpostgisdatastoreform').getForm();
164     if (formpostgis.isValid()) {
165         formpostgis.submit({
166             method: 'POST',
167             waitTitle: 'Connecting',
168             waitMsg: 'Sending data...',
169             success: function(form, action){
170                 obj = Ext.util.JSON.decode(action.response.responseText);
171                 Ext.Msg.alert('status', 'To PostGIS store Δημιουργήθηκε!', function(btn, text){
172                     if (btn == 'ok'){
173                         var postgisdatastoreform = Ext.getCmp('createpostgisdatastoreform').getForm();
174                         postgisdatastoreform.reset();
175                         var refreshsave = Ext.getCmp('refreshsave');
176                         refreshsave.setDisabled(false);
177                     }
178                 }); // button ok
179             }); // message
180         }); // success function
181         failure: function(form, action){
182             if (action.failureType == 'server'){
183                 obj = Ext.util.JSON.decode(action.response.responseText);
184                 Ext.Msg.alert('H σύνδεση απέτυχε!', obj.errors.reason);
185             } else {
186                 Ext.Msg.alert('Warning!', 'Authentication server is unreachable : ' + action.response.responseText);
187             }
188         }
189     }
190 }
191 //sumbit
192 //telos if isvalid
193
194 else {
195     Ext.Msg.alert("Σφάλμα!", "Τα στοιχεία δεν συμπληρώθηκαν όλα!");
196 }
197
198 }
199
200 }
201 }
202
203 ]
204
205
206 });
207 ///////////////////////////////////////////////////destoreform vector//////////////////////////////////////
208 var delstoreformvector = new Ext.FormPanel({
209     labelWidth: 160,
210     url: 'lib/geoserveredit.php?namespace=3' ,
211     frame: true,
212     id: 'delstoreformvector',
213     items: [
214         {
215             xtype: 'combo' ,
216             fieldLabel: 'workspace name' ,
217             editable: false,
218             name: 'vectorWorkspace',
219             id: 'vectorWorkspace',
220             displayField: 'name',
221             valueField: 'name',
222             mode: 'local',
223             triggerAction: 'all',
224             listClass: 'comboalign',
225             typeAhead: true,
226             forceSelection: true,
227             selectOnFocus: true,
228             store: mystore,
229             listeners: {
230                 change: function() {
231                     var vectorWorkspace = Ext.getCmp('vectorWorkspace');
232                     if (vectorWorkspace.getValue() != "")
233                     {
234                         // alert(comboworkspace.getValue());

```

```

235 var namespacedatavector = getJSONData(window.location.protocol + "/" + window.location.host + "/geoserver/rest/workspaces/" + vectorWork
space.getValue() + "/datastores.json");
236 var namespacedatavector1 = Ext.decode(namespacedatavector);
237
238 var namespacedatavector2 = namespacedatavector1['dataStores']['dataStore'];
239 //alert(Ext.encode(namespacedata2));
240 namespacestorevector.loadData(namespacedatavector2);
241 }
242
243 }
244 }
245 }, {
246 xtype : 'combo' ,
247 fieldLabel : 'namespace name' ,
248 editable : false,
249 name: 'VectorNamespaces',
250 id: 'VectorNamespaces',
251 displayField: 'name',
252 valueField: 'name',
253 mode: 'local',
254 triggerAction: 'all',
255 listClass: 'comboalign',
256 typeAhead: true,
257 forceSelection: true,
258 selectOnFocus: true,
259 store: namespacestorevector
260
261 // store: namespacestore
262 }
263 , {
264 xtype : 'textfield' ,
265 fieldLabel : 'Geoserver username' ,
266 name : 'geoserveruservector' ,
267 autoHeight : true
268 //id: 'GUser'
269 }, {
270 xtype : 'textfield' ,
271 fieldLabel : 'Geoserver password' ,
272 inputType : 'password' ,
273 name : 'geoserverpassvector'
274 // autoHeight: true
275 //id: 'GPass'
276 }, {
277 xtype : 'button' ,
278 name : 'deletebutton' ,
279 text : 'Διαγραφή' ,
280 iconCls : 'removeedit' ,
281 width : 80,
282 formBind : true,
283 handler: function() {
284     delstoreformvector.getForm().submit({
285         method: 'POST',
286         waitTitle: 'Connecting',
287         waitMsg: 'Sending data...',
288         success: function(form, action){
289             obj = Ext.util.JSON.decode(action.response.responseText);
290             Ext.Msg.alert('status', 'To store Διαγράφηκε!(message: '+obj.errors.reason+')!', function(btn, text){
291                 if (btn == 'ok'){
292                     var delstoreformvector = Ext.getCmp('delstoreform').getForm();
293                     delstoreformvector.reset();
294                     var refreshsave = Ext.getCmp('refreshsave');
295                     refreshsave.setDisabled(false);
296                 }
297             }); // button ok
298             }); // message
299             }, // success function
300             failure: function(form, action){
301
302                 if(action.failureType == 'server'){
303                     obj = Ext.util.JSON.decode(action.response.responseText);
304                     Ext.Msg.alert('Login Failed!', obj.errors.reason);
305                 } else{
306                     Ext.Msg.alert('Warning!', 'Authentication server is unreachable : ' + action.response.responseText);
307                 }
308                 var delstoreformvector = Ext.getCmp('storeform').getForm();
309                 delstoreformvector.reset();
310             }
311

```

```

312     }); //submit
313
314 }
315 }
316 ]
317 });
318
319 //////////////////////////////////////////////////delstoreform////////////////////////////////////
320
321 var delstoreform = new Ext.FormPanel({
322     labelWidth : 160,
323     url : 'lib/geoserveredit.php?namespace=2' ,
324     frame : true,
325     id : 'delstoreform',
326     items : [
327         {
328             xtype : 'combo' ,
329             fieldLabel : 'workspace name ' ,
330             editable : false,
331             name : 'RasterWorkspaces',
332             id : 'RasterWorkspaces',
333             displayField : 'name',
334             valueField : 'name',
335             mode : 'local',
336             triggerAction : 'all',
337             listClass : 'comboalign',
338             typeAhead : true,
339             forceSelection : true,
340             selectOnFocus : true,
341             store : mystore,
342             listeners : {
343                 change : function() {
344                     var RasterWorkspaces=Ext.getCmp('RasterWorkspaces');
345                     if(RasterWorkspaces.getValue()!="")
346                     {
347                         // alert(ComboBoxWorkspaces.getValue());
348                     var namespacedata = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/workspaces/" + RasterWorkspaces.getValue() + "/coveragestores.json");
349                     var namespacedata1=Ext.decode(namespacedata);
350
351                     var namespacedata2=namespacedata1['coverageStores']['coverageStore'];
352                     //alert(Ext.encode(namespacedata2));
353                     namespacestore.loadData(namespacedata2);
354                 }
355             }
356         }
357     ], {
358         }, {
359         xtype : 'combo' ,
360         fieldLabel : 'namespace name ' ,
361         editable : false,
362         name : 'RasterNamespace',
363         id : 'RasterNamespace',
364         displayField : 'name',
365         valueField : 'name',
366         mode : 'local',
367         triggerAction : 'all',
368         listClass : 'comboalign',
369         typeAhead : true,
370         forceSelection : true,
371         selectOnFocus : true,
372         store : namespacestore
373     }, {
374         // store: namespacestore
375     }, {
376         }, {
377             xtype : 'textfield' ,
378             fieldLabel : 'Geoserver username ' ,
379             name : 'geoserveruser' ,
380             autoHeight : true
381             //id : 'GUser'
382         }, {
383             xtype : 'textfield' ,
384             fieldLabel : 'Geoserver password ' ,
385             inputType : 'password' ,
386             name : 'geoserverpass'
387             // autoHeight: true
388             //id : 'GPass'

```



```

389     },{
390     xtype : 'button' ,
391     name : 'deletebutton' ,
392     text : 'Διαγραφή' ,
393     iconCls : 'removeedit' ,
394     width : 80,
395     formBind : true,
396     handler: function() {
397         delstoreform.getForm ().submit({
398             method:'POST',
399             waitTitle:'Connecting',
400             waitMsg:'Sending data...',
401             success:function(form, action){
402                 obj = Ext.util.JSON.decode(action.response.responseText);
403                 Ext.Msg.alert('status',obj.errors.reason, function(btn, text){
404                     if (btn == 'ok'){
405                         var delstoreform = Ext.getCmp('delstoreform').getForm();
406                         delstoreform.reset();
407                         var refreshsave = Ext.getCmp('refreshsave');
408                         refreshsave.setDisabled(false);
409                     }
410                 }); // button ok
411             }); //message
412             }, //success function
413             failure:function(form, action){
414                 if(action.failureType == 'server'){
415                     obj = Ext.util.JSON.decode(action.response.responseText);
416                     Ext.Msg.alert(obj.title.title, obj.errors.reason);
417                 }
418             }
419             }
420             }
421             }
422             }
423             }
424             }
425             }
426             }
427             }
428             }
429             }
430             }
431             }
432             //workspaceform
433             //workspaceform
434             var workspaceform = new Ext.FormPanel({
435                 labelWidth : 160,
436                 url : 'lib/geoserveredit.php?edit=1' ,
437                 frame : true,
438                 // bodyStyle: "background-image:url('/images/logon.jpg)",
439                 // layout: 'absolute',
440                 //height:300,
441                 id: 'workspaceform',
442                 items:[
443                     {
444                         xtype : 'textfield' ,
445                         fieldLabel : 'Workspace name ' ,
446                         name : 'GWorkspace' ,
447                         autoHeight : true
448                         //id:'GWorkspace'
449                     },
450                     {
451                         xtype : 'textfield' ,
452                         fieldLabel : 'Geoserver username ' ,
453                         name : 'geoserveruser' ,
454                         autoHeight : true
455                         //id:'GUser'
456                     },
457                     {
458                         xtype : 'textfield' ,
459                         fieldLabel : 'Geoserver password ' ,
460                         inputType : 'password' ,
461                         name : 'geoserverpass'
462                         // autoHeight: true
463                         //id:'GPass'
464                     },
465                     {
466                         xtype : 'button' ,
467                         name : 'create' ,

```

```

467 text : 'Δημιουργία' ,
468 iconCls : 'saveedit' ,
469 width : 80,
470 formBind : true,
471 handler: function() {
472     workspaceform.getForm().submit({
473         method:'POST',
474         waitTitle:'Connecting',
475         waitMsg:'Sending data...',
476         success:function(form, action){
477             Ext.Msg.alert('status', 'To workspace Δημιουργήθηκε!', function(btn, text){
478                 if (btn == 'ok'){
479                     //reset all form
480                     var delworkspaceform = Ext.getCmp('delworkspaceform').getForm();
481                     delworkspaceform.reset();
482
483                     var delstoreformvector = Ext.getCmp('delstoreformvector').getForm();
484                     delstoreformvector.reset();
485
486
487                     var delstoreform = Ext.getCmp('delstoreform').getForm();
488                     delstoreform.reset();
489
490                     var workspaceform = Ext.getCmp('workspaceform').getForm();
491                     workspaceform.reset();
492
493                     ///////////////
494                     var data = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/workspaces.json");
495                     var data1=Ext.decode(data);
496                     var data2=data1['workspaces']['workspace'];
497                     if(!data2)
498                     {
499                         data3='{ "workspaces": { "workspace": { "name": "empty", "href": "empty" } } }';
500                         var data1=Ext.decode(data3);
501                         var data2=data1['workspaces']['workspace'];
502
503                     }
504
505                     //workspacestore
506                     mystore.loadData (data2);
507                     // namespacestore.loadData(data2);
508                     // namespacestorevector.loadData(data2);
509                     var refreshsave = Ext.getCmp('refreshsave');
510                     refreshsave.setDisabled(false);
511
512             } // button ok
513             }); //message
514             }, //success function
515             failure:function(form, action){
516
517                 if(action.failureType == 'server'){
518                     obj = Ext.util.JSON.decode(action.response.responseText);
519                     Ext.Msg.alert('Login Failed!', obj.errors.reason);
520                 }else{
521                     Ext.Msg.alert('Warning!', 'Authentication server is unreachable : ' + action.response.responseText);
522                 }
523                 var workspaceform = Ext.getCmp('workspaceform').getForm();
524                 workspaceform.reset();
525             }
526
527             }); //sumbit
528
529     }
530 }
531 ]
532 });
533
534 //delworkspaceform////////////////////////////////////
535
536 //
537 var delworkspaceform = new Ext.FormPanel({
538     labelWidth : 160,
539
540     url : 'lib/geoserveredit.php?edit=2' ,
541     frame : true,
542     id: 'delworkspaceform',
543     items:[
544     {

```



```

545 xtype : 'combo' ,
546 fieldLabel : 'workspace name' ,
547 editable : false,
548 name: 'GWorkspace',
549 id: 'comboworkspace',
550 displayField: 'name',
551 valueField: 'name',
552 mode: 'local',
553 triggerAction: 'all',
554 listClass: 'comboalign',
555 typeAhead: true,
556 forceSelection: true,
557 selectOnFocus: true,
558 store: mystore
559 }, {
560     xtype : 'textfield' ,
561     fieldLabel : 'Geoserver username ' ,
562     name : 'geoserveruser' ,
563     autoHeight : true
564     //id: 'GUser'
565 }, {
566     xtype : 'textfield' ,
567     fieldLabel : 'Geoserver password ' ,
568     inputType : 'password' ,
569     name : 'geoserverpass'
570     // autoHeight: true
571     //id: 'GPass'
572 }
573 }, {
574     xtype : 'button' ,
575     name : 'create' ,
576     text : 'Διαγραφή' ,
577     iconCls : 'removeedit' ,
578     width : 80,
579     formBind : true,
580     handler: function() {
581         delworkspaceform.getForm().submit({
582             method: 'POST',
583             waitTitle: 'Connecting',
584             waitMsg: 'Sending data...',
585             success: function(form, action){
586                 //alert(data2);
587                 //message for inform
588                 obj = Ext.util.JSON.decode(action.response.responseText);
589                 //message for inform
590                 Ext.Msg.alert('status', obj.errors.reason, function(btn, text){
591                     if (btn == 'ok'){
592                         /* Ext.Msg.show({
593                             title: 'ΠΡΟΣΟΧΗ',
594                             msg: "Σας ενημερώνουμε ότι η διαγραφή του workspace θα είναι επιτυχής εφόσον είναι κενός",
595                             buttons: Ext.Msg.OK,
596                             icon: "attention"
597                         });*/
598                         //reset all form
599                         var delworkspaceform = Ext.getCmp('delworkspaceform').getForm();
600                         delworkspaceform.reset();
601
602                         var delstoreformvector = Ext.getCmp('delstoreformvector').getForm();
603                         delstoreformvector.reset();
604
605
606                         var delstoreform = Ext.getCmp('delstoreform').getForm();
607                         delstoreform.reset();
608
609                         var workspaceform = Ext.getCmp('workspaceform').getForm();
610                         workspaceform.reset();
611
612                         //message for inform
613                         var data = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/workspaces.json");
614                         var data1 = Ext.decode(data);
615                         var data2 = data1['workspaces']['workspace'];
616                         if (!data2)
617                         {
618                             data3 = '{"workspaces": {"workspace": [{"name": "empty", "href": "empty"}]}}';
619                             var data1 = Ext.decode(data3);
620                             var data2 = data1['workspaces']['workspace'];
621
622                         }

```

```

623     mystore.loadData (data2);
624     // namespacestore.loadData(data2);
625     //namespacestorevector.loadData(data2);
626     var refreshsave = Ext.getCmp('refreshsave');
627     refreshsave.setDisabled(false);
628     // var comboworkspace = Ext.getCmp('comboworkspace')
629     ///comboworkspace.bindStore(mystore);
630
631     }// button ok
632     });//message
633     };//success function
634     failure:function(form, action){
635         if(action.failureType == 'server'){
636             obj = Ext.util.JSON.decode(action.response.responseText);
637             Ext.Msg.alert(obj.title.title, obj.errors.reason);
638         }else{
639             Ext.Msg.alert("Warning!", 'Authentication server is unreachable : ' + action.response.responseText);
640         }
641         var delworkspaceform = Ext.getCmp('delworkspaceform').getForm();
642         delworkspaceform.reset();
643     }
644
645     });//submit
646
647 }
648 }
649 ]
650 });
651
652 //////////////////////////////////////////////////dellayerform////////////////////////////////////
653 var layerform = new Ext.FormPanel({
654     labelWidth : 160,
655     url : 'lib/geoserveredit.php?layer=1' ,
656     frame : true,
657     id : 'layerform',
658     items:[
659         {
660             xtype : 'combo' ,
661             fieldLabel : 'Θεματικό επίπεδο ' ,
662             editable : false,
663             name : 'glayer',
664             id : 'glayer',
665             displayField : 'name',
666             valueField : 'name',
667             mode : 'local',
668             triggerAction : 'all',
669             listClass : 'comboalign',
670             typeAhead : true,
671             forceSelection : true,
672             selectOnFocus : true,
673             store : layerstore
674
675         },
676         {
677             xtype : 'textfield' ,
678             fieldLabel : 'Geoserver username ' ,
679             name : 'geoserveruser' ,
680             autoHeight : true
681             //id : 'GUser'
682         },
683         {
684             xtype : 'textfield' ,
685             fieldLabel : 'Geoserver password ' ,
686             inputType : 'password' ,
687             name : 'geoserverpass'
688             // autoHeight : true
689             //id : 'GPass'
690         },
691         {
692             xtype : 'button' ,
693             name : 'deletebutton' ,
694             text : 'Διαγραφή' ,
695             iconCls : 'removeedit' ,
696             width : 80,
697             formBind : true,
698             handler : function() {
699                 layerform.getForm ().submit({
700                     method : 'POST',
701                     waitTitle : 'Connecting',
702                     waitMsg : 'Sending data...',

```

```

701     success:function(form, action){
702         obj = Ext.util.JSON.decode(action.response.responseText);
703         Ext.Msg.alert('status', 'To layer Διαγράφηκε!', function(btn, text){
704             if (btn == 'ok'){
705                 var layerform = Ext.getCmp('layerform').getForm();
706                 layerform.reset();
707                 var refreshsave = Ext.getCmp('refreshsave');
708                 refreshsave.setDisabled(false);
709             }
710         } // button ok
711     }); //message
712     }, //success function
713     failure:function(form, action){
714         if(action.failureType == 'server'){
715             obj = Ext.util.JSON.decode(action.response.responseText);
716             Ext.Msg.alert('Login Failed!', obj.errors.reason);
717         }else{
718             Ext.Msg.alert('Warning!', 'Authentication server is unreachable : ' + action.response.responseText);
719         }
720         //var layerform = Ext.getCmp('storeform').getForm();
721         //layerform.reset();
722     }
723 }
724
725 }); //submit
726
727 }
728 }
729 ]
730 });
731
732
733
734
735
736
737
738 ///////////////////////////////////////////////////////////////////main program/////////////////////////////////////////////////////////////////
739
740 var Geoserver=[{xtype:'tabpanel',activeTab:0,region:'center',id:'tabgeoserver',forceLayout : 'true',
741
742     items:[{
743
744         xtype : 'panel' ,
745         iconCls : 'workspace' ,
746         title : 'workspace' ,
747
748         id : 'workspace' ,
749         margins : '5 0 5 5' ,
750         split : true,
751         width: 210,
752         //layout:'accordion',
753         items:[
754             {
755                 xtype:'panel',
756                 title:'Δημιουργία workspace',
757                 iconCls: 'editworkspace',
758                 items:workspaceform
759             } //items:[formupload]
760
761             ,{
762
763                 xtype : 'panel' ,
764                 title : 'Διαγραφή workspace' ,
765                 iconCls : 'delworkspace' ,
766                 items : delworkspaceform
767
768             }
769
770         ]
771     },
772     {
773         xtype : 'panel' ,
774         iconCls : 'store' ,
775         title : 'Αποθήκη θεματικών επιπέδων' ,
776         margins : '5 0 5 5' ,
777         split : true,

```



```

779 width: 210,
780 //layout:'accordion',
781 items:[{
782     xtype:'panel' ,
783     title:'Διαγραφή Αποθήκης και θεματικού επιπέδου(για Διανυσματικά δεδομένα)' ,
784     iconCls:'delworkspace' ,
785     items:delstoreformvector
786 },
787 {
788     xtype:'panel' ,
789     title:'Διαγραφή Αποθήκης και θεματικού επιπέδου(για εικόνες)' ,
790     iconCls:'delworkspace' ,
791     items:delstoreform
792 },
793 ],
794 ],
795 },
796 ],
797 ],
798 ],
799 ],
800 ],
801 },{
802     xtype:'panel' ,
803     iconCls:'layers' ,
804     title:'Θεματικά επίπεδα' ,
805     margins:'5 0 5 5' ,
806     split:true,
807     width: 210,
808     //layout:'accordion',
809     items:[{
810         xtype:'panel',
811         title:'Διαγραφή Θεματικού επιπέδου',
812         iconCls:'delworkspace',
813         items:layerform
814     },
815     //items:storeform
816     //items:[formupload]
817 ],
818 ],
819 ],
820 ],
821 ],
822 ],
823 },
824 {
825     xtype:'container' ,
826     layout:'border' ,
827     id:'dataupload' ,
828     shallow:true,
829     force:true,
830     iconCls:'upload1',
831     title:'ΜΕΤΑΦΟΡΤΩΣΗ ΔΕΔΟΜΕΝΩΝ',
832     tbar:[],
833     items:[{
834         xtype:'gxp_layeruploadpanel' ,
835         region:'center' ,
836         proxy:"/geoserver/rest/proxy?url=" ,
837         //renderTo:"container",
838         id:'formuploadtif' ,
839         forceLayout:'true' ,
840         url:"/geoserver/rest" ,
841         width:this.width,
842         height:this.height,
843         frame:true,
844         title:"ΜΕΤΑΦΟΡΤΩΣΗ ΧΩΡΙΚΩΝ ΔΕΔΟΜΕΝΩΝ",
845         iconCls:'upload2',
846         // autoHeight: true,
847         bodyStyle:"padding: 10px 10px 0 10px;",
848         labelWidth: 65,
849         defaults: {
850             anchor:"95%",
851             allowBlank:false,

```

```

857     msgTarget: "side"
858 },
859 listeners: {
860     uploadcomplete : function(panel, detail) {
861         var refreshsave = Ext.getCmp('refreshsave');
862         refreshsave.setDisabled(false);
863         var layers = detail.layers;
864         var names = [];
865         for (var i=0, len=layers.length; i<len; ++i) {
866             names.push(layers[i].name);
867         }
868         Ext.Msg.show ({
869             title : "Success" ,
870             msg : "Added new layer" + (len !== 1 ? "s" : "") + ": " + names.join(", "),
871             minWidth: 200,
872             icon: Ext.Msg.INFO,
873             buttons: Ext.Msg.OK
874         });
875     }
876 }
877
878 }}
879
880 },{//den to exw kanei akoma
881     xtype:'panel',
882     iconCls: 'postGIS',
883     title:'Διαχείριση PostGIS',
884     margins:'5 0 5 5',
885     split:true,
886     width: 210,
887     items:createpostgisdatastoreform
888
889     })
890
891     });
892
893 // JavaScript Document

```

```

1  <?php
2
3  if (isset($_GET['edit']))
4  {
5      include "GeoserverWrapper.php";
6      if($_GET['edit']==1)//για tin proti selida
7      {
8          $workspaces=$_POST['GWorkspace'];
9          $Gusername=$_POST['geoserveruser'];
10         $Gpassword=$_POST['geoserverpass'];
11         $geoserver = new GeoserverWrapper('http://'.$_SERVER['HTTP_HOST'].'/geoserver',$Gusername, $Gpassword);
12         $checkenter=$geoserver->authGet('http://'.$_SERVER['HTTP_HOST'].'/geoserver');
13         if($checkenter!="Access denied. Check login credentials.")
14         {
15             $geoserver->createWorkspace($workspaces);
16
17             echo "{success: true}";
18         }
19
20         else
21         {
22             echo "{success: false, errors: { reason: 'Η σύνδεση απέτυχε. Δοκίμασε ξανά.' }}";
23
24         }
25
26     }//edit=1
27     if($_GET['edit']==2)//για tin proti selida
28     {
29         $workspaces=$_POST['GWorkspace'];
30         $Gusername=$_POST['geoserveruser'];
31         $Gpassword=$_POST['geoserverpass'];
32         //$workspaces="dianomes";
33         //$Gusername="admin";
34         //$Gpassword="geoserver";
35         //$datastoreName="brusoula1932_3";
36         //$layername="brusoula1932_3";
37         $geoserver = new GeoserverWrapper('http://'.$_SERVER['HTTP_HOST'].'/geoserver',$Gusername, $Gpassword);
38         $checkenter=$geoserver->authGet('http://'.$_SERVER['HTTP_HOST'].'/geoserver');
39         if($checkenter!="Access denied. Check login credentials.")
40         {
41             $message=$geoserver->deleteWorkspace($workspaces);
42             if($message=="")
43             {
44                 echo "{success: true, errors: { reason: 'To Workspace Διαγράφηκε.' }}";
45
46             }
47
48             //$geoserver->deletecoveragestores($datastoreName, $workspaces);
49
50             if($message!="")
51             {
52                 //$geoserver->deleteLayer1($layername);
53                 echo "{success: false, errors: { reason: '$message' },title:{title:'Σφάλμα!'}}";
54             }
55
56
57
58
59         }
60         else
61         {
62             echo "{success: false, errors: { reason: 'Η σύνδεση απέτυχε. Δοκίμασε ξανά.' },title:{title:'login failed!'}}";
63
64         }
65
66
67
68
69     }
70 }
71
72
73 //////////////////////////////////////////////////diagrafi namespace kai layer
74 if (isset($_GET['namespace']))
75 {
76     include "GeoserverWrapper.php";
77
78

```



```

79
80     if($_GET['namespace']==2)
81     {
82         $RasterWorkspaces=$_POST['RasterWorkspaces'];
83         $RasterNamespace=$_POST['RasterNamespace'];
84         $geoserveruser=$_POST['geoserveruser'];
85         $geoserverpass=$_POST['geoserverpass'];
86
87         /*$RasterWorkspaces='data';
88         $RasterNamespace='10_geo_peplos_2000_ana_1_2';
89         $geoserveruser='admin';
90         $geoserverpass='geoserver';
91         */
92         $geoserver1 = new GeoserverWrapper('http://'.$_SERVER['HTTP_HOST'].'/geoserver',$geoserveruser, $geoserverpass);
93         $checker=$geoserver1->authGet('http://'.$_SERVER['HTTP_HOST'].'/geoserver');
94         if($checker!="Access denied. Check login credentials.")
95         {
96
97             //$geoserver->deleteLayer1($RasterNamespace);
98             //$geoserver->deleteLayer1($RasterNamespace);
99 $message=$geoserver1->deletegeotif($RasterNamespace,$RasterWorkspaces,$RasterNamespace);
100 if($message=="")
101 {
102     #$geoserver->deletecoveragestores($RasterNamespace, $RasterWorkspaces);
103     #$geoserver->deleteLayer1($RasterNamespace);
104     echo "{success: true, errors: { reason: 'To namespace Διαγράφηκε' }}";
105 }
106
107 else
108 {
109     echo "{success: false, errors: { reason: '$message' },title:{title:'Σφάλμα'}}";
110 }
111 }
112 }
113 else
114 {
115     echo "{success: false, errors: { reason: '$checker' },title:{title:'login failed!'}}";
116 }
117 }
118 }
119 }
120
121 if($_GET['namespace']==3)
122 {
123     $vectorWorkspace=$_POST['vectorWorkspace'];
124     $VectorNamespaces=$_POST['VectorNamespaces'];
125     $geoserveruservector=$_POST['geoserveruservector'];
126     $geoserverpassvector=$_POST['geoserverpassvector'];
127
128     /*$RasterWorkspaces='data';
129     $RasterNamespace='10_geo_peplos_2000_ana_1_2';
130     $geoserveruser='admin';
131     $geoserverpass='geoserver';
132     */
133     $geoserver1 = new GeoserverWrapper('http://'.$_SERVER['HTTP_HOST'].'/geoserver',$geoserveruservector, $geoserverpassvector);
134     $checker=$geoserver1->authGet('http://'.$_SERVER['HTTP_HOST'].'/geoserver');
135     if($checker!="Access denied. Check login credentials.")
136     {
137
138         //$geoserver->deleteLayer1($RasterNamespace);
139         //$geoserver->deleteLayer1($RasterNamespace);
140 $message=$geoserver1->deleteLayerWorkspace($VectorNamespaces,$vectorWorkspace,$VectorNamespaces);
141 #$geoserver->deletecoveragestores($RasterNamespace, $RasterWorkspaces);
142 #$geoserver->deleteLayer1($RasterNamespace);
143 if($message=="")
144 {
145     #$geoserver->deletecoveragestores($RasterNamespace, $RasterWorkspaces);
146     #$geoserver->deleteLayer1($RasterNamespace);
147     echo "{success: true, errors: { reason: 'To namespace Διαγράφηκε' }}";
148 }
149
150 else
151 {
152     echo "{success: false, errors: { reason: '$message' }}";
153 }
154 }
155 }
156 else

```

```

157     {
158         echo "{success: false, errors: { reason: '$checkenter' }}";
159     }
160 }
161
162 }
163
164
165 }
166
167 if (isset($_GET['layer']))
168 {
169     include "GeoserverWrapper.php";
170
171     if (($_GET['layer']==1))
172     {
173
174         $glayer=$_POST['glayer'];
175         $geoserveruser=$_POST['geoserveruser'];
176         $geoserverpass=$_POST['geoserverpass'];
177
178         $geoserver = new GeoserverWrapper('http://'.$_SERVER['HTTP_HOST'].'/geoserver',$geoserveruser, $geoserverpass);
179         $checkenter=$geoserver->authGet('http://'.$_SERVER['HTTP_HOST'].'/geoserver');
180         if($checkenter!="Access denied. Check login credentials.")
181         {
182             $message=$geoserver->deleteLayer1($glayer);
183             echo "{success:true, errors: { reason: '$message' }}";
184         }
185         else
186         {
187             echo "{success: false, errors: { reason: '$checkenter' }}";
188         }
189     }
190 }
191
192 }
193
194
195 }
196
197 if (isset($_GET['postgis']))
198 {
199     include "../connect/connect.php";
200     include "GeoserverWrapper.php";
201
202     if (($_GET['postgis']==1))
203     {
204         /*$RasterWorkspaces='data';
205         $RasterNamespace='10_geo_peplos_2000_ana_1_2';
206         $geoserveruser='admin';
207         $geoserverpass='geoserver';
208         */
209
210         //$RasterWorkspaces=$_POST['RasterWorkspaces'];
211         $WorkspacePostgis=$_POST['WorkspacePostgis'];
212         $postgisstore=$_POST['postgisstore'];
213         $postgisname=$_POST['postgisname'];
214         $postgisusername=$Username;
215         $postgispassword=$Password;
216         $postgishost=$Server;
217         $postgisport=$Port;
218
219         $geoserveruserpostgis=$_POST['geoserveruserpostgis'];
220         $geoserverpasspostgis=$_POST['geoserverpasspostgis'];
221
222
223
224
225         $geoserver = new GeoserverWrapper('http://'.$_SERVER['HTTP_HOST'].'/geoserver',$geoserveruserpostgis, $geoserverpasspostgis);
226         $checkenter=$geoserver->authGet('http://'.$_SERVER['HTTP_HOST'].'/geoserver');
227         if($checkenter!="Access denied. Check login credentials.")
228         {
229             $message=$geoserver->createPostGISDataStore($postgisstore, $WorkspacePostgis, $postgisname, $postgisusername, $postgispassword
, $postgishost, $postgisport);
230             echo "{success:true, errors: { reason: '$message' }}";
231         }
232         else
233         {

```



```
234     echo "{success: false, errors: { reason: '$checkenter' }}";
235
236 }
237
238
239 }
240
241
242 }
243
244 ?>
```

```

1  //////////////////////////////////field //////////////////////////////////
2
3
4
5  var standart = Ext.data.Record.create([
6      {name:'id_application'
7          //type:'int'
8
9          },{
10             name: 'name' ,
11             type: 'string'
12         },
13         {
14             name: 'surname' ,
15             type: 'string'
16         },
17         },{
18             name: 'fathername' ,
19             type: 'string'
20         },
21         },
22         {
23             name: 'phone1' ,
24             type: 'string'
25         },
26         },{
27             name: 'afm' ,
28             type: 'string'
29         },{
30             name: 'identity' ,
31             type: 'string'
32         },
33         {
34             name: 'phone2' ,
35             type: 'string'
36         },
37         {
38             name:'email' ,
39             type:"string"
40         },
41         },{
42             name: 'typemap' ,
43             type: 'string'
44         },{
45             name: 'currentmap' ,
46             type: 'string'
47         },{
48             name: 'numberofland' ,
49             type: 'string'
50         },{
51             name: 'areachoice' ,
52             type: 'string'
53         },{
54             name: 'pointchoice' ,
55             type: 'string'
56         },{
57             name: 'situation' ,
58             type: 'string'
59         },{
60             name: 'user' ,
61             type: 'string'
62         }
63     ]
64
65
66
67 ]);
68
69
70
71
72 //////////////////////////////////store gia gridaction
73 var storeaction = new Ext.data.GroupingStore({
74     id:'storeaction' ,
75     reader: new Ext.data.JsonReader({fields:standart}),
76     autoLoad:true, //this w,2,3,4ill autoLoad the store on load. Or you can call
77         //store.load() later.
78     proxy: new Ext.data.HttpProxy({

```

```

79         url: 'lib/applicationtool.php?edit=1' ,
80         autoAbort :true,
81         disableCaching: true,
82         timeout: 1800000,
83         method: 'POST'
84     })
85 });
86 //////////////////////////////////////////////////store gia gridall
87
88
89
90 //////////////////////////////////////////////////grid gia aitiseis pou ekremoun////////////////////////////////
91 var gridaction = new Ext.grid.GridPanel({
92     id:'gridaction' ,
93     store : storeaction,
94     width : 600,
95     region : 'center' ,
96     margins : '0 5 5 5',
97     autoExpandColumn : 'id_application' ,
98     // plugins: [editor],
99     view : new Ext.grid.GroupingView({
100         markDirty : false
101     }),
102     columns: [
103         new Ext.grid.RowNumberer(),
104         {
105             // text:'id application',
106             width : 100,
107             header : 'id application' ,
108             id: 'id_application' ,
109             sortable : true,
110             dataIndex: 'id_application'
111         }, {
112             // xtype:"textfield",text:'id application',
113             text : 'name' ,
114             width : 100,
115             header : 'ΟΝΟΜΑ' ,
116             dataIndex : 'name' ,
117             id: 'name'
118         },
119
120
121         }, {
122             // text:'surname',
123             // xtype:"textfield",
124             width : 100,
125             header : 'Επίθετο' ,
126             dataIndex : 'surname' ,
127             id: 'surname'
128         },
129
130         }, {
131             // text:'surname',
132             // xtype:"textfield",
133             width : 100,
134             header : 'ΑΦΜ' ,
135             dataIndex : 'afm' ,
136             id: 'afm'
137         },
138
139         }, {
140             // text:'surname',
141             // xtype:"textfield",
142             width : 100,
143             header : 'Α.Δ.Τ' ,
144             dataIndex : 'identity' ,
145             id: 'identity'
146         },
147
148         }, {
149             // text:'fathername',
150             // xtype:"textfield",
151             width : 100,
152             header : 'ΟΝΟΜΑ ΠΑΤΡΟΣ' ,
153             dataIndex : 'fathername' ,
154             id: 'fathername'
155         }
156     ]

```

```

157     }, {
158         // text:'phone1',
159         // xtype:'textfield',
160         width : 100,
161         header : 'Τηλεφωνο(σταθερο)' ,
162         dataIndex : 'phone1' ,
163         id : 'phone1'
164     },
165
166     }, {
167         // text:'phone2',
168         // xtype:'textfield',
169         width : 100,
170         header : 'Τηλεφωνο(κινητο)' ,
171         dataIndex : 'phone2' ,
172         id : 'phone2'
173     },
174
175     }, {
176         // text:'email',
177         // xtype:'textfield',
178         width : 100,
179         header : 'email' ,
180         dataIndex : 'email' ,
181         id : 'email'
182     },
183
184     }, {
185         // text:'typemap',
186         // xtype:'textfield',
187         width : 100,
188         header : 'Είδος εφαρμογής' ,
189         dataIndex : 'typemap' ,
190         id : 'typemap'
191     },
192
193     }, {
194         // text:'currentmap',
195         // xtype:'textfield',
196         width : 100,
197         header : 'Τίτλος χάρτη' ,
198         dataIndex : 'currentmap' ,
199         id : 'currentmap'
200     },
201
202     }, {
203         //text:'numberofland',
204         /// xtype:'textfield',
205         width : 100,
206         header : 'Αριθμός αγροτεμαχίου' ,
207         dataIndex : 'numberofland' ,
208         id : 'numberofland'
209     },
210
211     }, {
212         // text:'areachoice',
213         // xtype:'textfield',
214         width : 100,
215         header : 'Όρια περιοχής' ,
216         dataIndex : 'areachoice' ,
217         id : 'areachoice'
218     },
219
220     }, {
221         // text:'pointchoice',
222         // xtype:'textfield',
223         width : 100,
224         header : 'Σημείο εντος αγροτεμαχίου' ,
225         dataIndex : 'pointchoice' ,
226         id : 'pointchoice'
227     },
228
229     }, {
230         // text:'situation',
231         // xtype:'textfield',
232         width : 100,
233         header : 'Κατασταση αιτησης' ,
234         dataIndex : 'situation' ,

```

```

235     id: 'situation'
236
237
238     }, {
239         // text: 'user',
240         // xtype: "textfield",
241         width : 100,
242         header : 'Χρήστης που ανέλαβε' ,
243         dataIndex : 'user' ,
244         id: 'user'
245
246
247     }
248 ],
249 tbar: [{
250     xtype: 'button' ,
251     iconCls: 'removeedit' ,
252     disabled: processcheck1, //na to kano emgfanis sto login
253     text: 'Διαγραφή αίτησης' ,
254     id: "deleteapply" ,
255     //////////////////////////////////////////////////dimiourgia function diagrafis tis aitiseis tou pelati
256     listeners: {
257         click: {
258             // scope: this,
259             fn: function () {
260                 //var grid = Ext.getCmp('grid');
261                 var selection = gridaction.getSelectionModel().getSelections();
262                 if (selection) {
263
264                     var sel = selection,
265                     model = sel[0],
266                     val = model.get('id_application');
267                     //var user=model.get("Όνομα");
268                     //////////////////////////////////
269
270                     Ext.MessageBox.show({
271                         buttons: Ext.Msg.YESNOCANCEL,
272                         title: 'Διαγραφή?' ,
273                         msg: 'Θέλετε να διαγραφή η αίτηση: '+ val,
274                         icon: Ext.Msg.QUESTION,
275                         fn: function(btn){
276                             if (btn == "no"){
277                                 storeaction.load()
278                             }
279                             if (btn == "yes"){
280                                 for(var i = 0, r; r = selection[i]; i++){
281                                     storeaction.remove (selection);
282                                 }
283                                 // store.remove(selection);
284                                 $.post ('lib/applicationtool.php?link=1',{id:val},function(data){
285
286                                     ///messagegia oloklirosei
287
288                                     Ext.Msg.alert ("ΔΙΑΓΡΑΦΗ","Η αίτηση: "+val+" "+data);
289
290                                     ///
291
292                                     }) //telos post
293
294                                     } //telos if
295                                 } //telos function
296                             });
297
298                             //////////////////////////////////
299
300                             } //telos if selection
301
302                         }
303                     }
304                 }
305             }
306
307     //////////////////////////////////
308
309     }, { xtype: 'button',
310         text: 'Ανανέωση',
311         iconCls: 'refresh',
312

```



```

313     handler: function () {
314         storeaction. load();
315         gridaction.getView().refresh();
316     }
317 },
318 {
319     xtype: 'button',
320     iconCls: 'icon-user-add',
321     text: 'Επεξεργασία',
322     id: "applicationedit",
323     listeners: {
324         click: {
325             // scope: this,
326             fn: function () {
327                 //var gridaction = Ext.getCmp('gridaction');
328                 var selection = gridaction.getSelectionModel().getSelections();
329                 if (selection) {
330
331                     var sel = selection,
332                     model = sel[0],
333                     ///epilogh opoias eggrafis apo tin epilegmeni grammi theloume
334
335                     val= model.get('id_application');
336                     var name=model.get('name');
337                     var surname=model.get('surname');
338                     var typemap=model.get('typemap');
339                     var currentmap=model.get('currentmap');
340                     var numberofland=model.get('numberofland');
341                     var areachoice=model.get('areachoice');
342                     var pointchoice=model.get('pointchoice');
343                     var situation=model.get('situation');
344                     var user=model.get('user');
345                     var afm=model.get('afm');
346                     var identity=model.get('identity');
347                     var areaprint="";
348
349                     //diaxorismos me basi to ,
350                     var coordspoint = pointchoice.split(',')
351
352                     ////////////
353                     if(situation!="f" && user!=getCookie('user1'))
354                     {
355
356
357                         Ext.Msg.show ({
358                             title: 'ΠΡΟΣΟΧΗ',
359                             msg: "Την αίτηση με αύξοντα αριθμό: "+ val+" την έχει αναλάβει ο χρήστης: "+ user,
360                             buttons: Ext.Msg.OK,
361                             icon: "attention"
362                         });
363
364
365                     }
366                     else if((situation!="t")||(situation!="f" && user==getCookie('user1')))
367                     {
368
369
370                         var vectorsarea = new OpenLayers.Layer.Vector("Περιοχη επιλογής", {isBaseLayer: false});
371                         var feature2 = new OpenLayers.Feature.Vector(
372                             OpenLayers.Geometry.fromWKT (
373                                 ""+ areachoice +""
374                             ).transform(new OpenLayers.Projection("EPSG:4121"),new OpenLayers.Projection("EPSG:900913"))
375                             );
376
377                         vectorsarea.addFeatures([feature2]);
378                         ////////////dimiourgia tou boundary
379                         var fts = vectorsarea.features;
380                         for(var i=0; i< fts.length; i++){
381                             var boundaryedit=vectorsarea.features[i].geometry.getBounds().toArray();
382                         };
383
384                         ////////////
385                         markerschoice = new OpenLayers.Layer.Markers("Σημείο Επιλογής");
386                         var size = new OpenLayers.Size(21,25);
387                         var offset = new OpenLayers.Pixel(-(size.w/2), -size.h);
388                         var icon = new OpenLayers.Icon('images/point1.gif', size, offset);
389                         var lonlat = new OpenLayers.LonLat(Number(coordspoint[0]),Number(coordspoint[1])).transform(new OpenLayers.Projection
("EPSG:4121"), new OpenLayers.Projection("EPSG:900913"));

```

```

390 markerschoice.addMarker(new OpenLayers.Marker(lonlat,icon));
391
392 //vectorsarea=vectorsarea.
393 //////////////////////////////////////////////////
394 var mapPanel, printPage;
395 //////////////////////////////////////////////////dimiourgia tou xarti me ta option
396
397
398 //////////////////////////////////////////////////dimiourgia printmodule
399
400
401 var printProvider = new GeoExt.data.PrintProvider({
402     method : "GET", // "POST" recommended for production use
403     capabilities : printCapabilities, // from the info.json script in the html
404     customParams : {
405         mapTitle : ""
406         //comment: "This is a map printed from GeoExt."
407     }
408 });
409
410 //////////////////////////////////////////////////
411 var printExtent = new GeoExt.plugins.PrintExtent({
412     printProvider : printProvider
413 });
414
415 //////////////////////////////////////////////////
416 mapPanel = new GeoExt.MapPanel({
417     title : '<center>Γεωχωρικά στοιχεία</center>' ,
418     region : 'center' ,
419
420     map : {
421
422
423     controls : [
424         new OpenLayers.Control.Zoom(),
425         new OpenLayers.Control.Attribution(),
426         new OpenLayers.Control.Navigation(),
427         new OpenLayers.Control.LayerSwitcher({ascending:false})
428     ],
429
430     displayProjection:"EPSG:900913",
431     units: 'km' ,
432     restrictedExtent: [boundaryedit[0], boundaryedit[1], boundaryedit[2], boundaryedit[3]],
433     center: [(boundaryedit[0]+boundaryedit[2])/2,(boundaryedit[1]+boundaryedit[3])/2],
434     zoom: 6,
435     numZoomLevels:20,
436     layers: [new OpenLayers.Layer.WMS( "("+typemap+") Χάρτης: "+currentmap,
437         "/geoserver/"+typemap+"/wms",
438         { layers: typemap+"."+currentmap,transparent: true,format: 'image/png'}, {
439             isBaseLayer :true,visibility: true,projection:new OpenLayers.Projection("EPSG:900913"))]]
440 },
441
442     plugins: [printExtent],
443     bbar: [{
444         text : "Εκτύπωση αποσπάσματος" ,
445         handler : function() {
446             // the PrintExtent plugin is the mapPanel's 1st plugin
447             mapPanel.plugins [0].print();
448         }
449     }
450 ]
451
452 });
453 mapPanel.map.addLayers([markerschoice]);
454 //alert(markerschoice.getDataExtent());
455 printExtent.addPage();
456 //////////////////////////////////////////////////
457
458
459 // The map we want to print
460
461 // The form with fields controlling the print output
462 var formPanel = new Ext.form.FormPanel({
463     region : "west" ,
464     split :true,
465     collapsible: true,
466     autoScroll: true,
467     width: 200,

```

```

468 bodyStyle: "padding:5px",
469 labelAlign: "top",
470 defaults: {anchor: "100%"},
471 items: [{
472     xtype: "textarea",
473     name: "comment",
474     value: "",
475     fieldLabel: "Comment",
476     plugins: new GeoExt.plugins.PrintPageField({
477         printPage: printExtent.pages [0]
478     })
479 }, {
480     xtype: "combo",
481     store: printProvider.layouts,
482     displayField: "name",
483     fieldLabel: "Layout",
484     typeAhead: true,
485     mode: "local",
486     triggerAction: "all",
487     plugins: new GeoExt.plugins.PrintProviderField({
488         printProvider: printProvider
489     })
490 }, {
491     xtype: "combo",
492     store: printProvider.outputFormats,
493     displayField: "name",
494     fieldLabel: "formats",
495     typeAhead: true,
496     mode: "local",
497     triggerAction: "all",
498     plugins: new GeoExt.plugins.PrintProviderField({
499         printProvider: printProvider
500     })
501 }, {
502     xtype: "combo",
503     store: printProvider.dpis,
504     displayField: "name",
505     fieldLabel: "Resolution",
506     tpl: '<tpl for=" "><div class="x-combo-list-item">{name} dpi</div></tpl>' ,
507     typeAhead: true,
508     mode: "local",
509     triggerAction: "all",
510     plugins: new GeoExt.plugins.PrintProviderField({
511         printProvider: printProvider
512     }),
513     // the plugin will work even if we modify a combo value
514     setValue: function(v) {
515         v = parseInt(v) + " dpi";
516         Ext.form.ComboBox.prototype.setValue.apply(this, arguments);
517     }
518 }, {
519     xtype: "combo",
520     store: printProvider.scales,
521     displayField: "name",
522     fieldLabel: "Scale",
523     typeAhead: true,
524     mode: "local",
525     triggerAction: "all",
526     plugins: new GeoExt.plugins.PrintPageField({
527         printPage: printExtent.pages [0]
528     })
529 }, {
530     xtype: "textfield",
531     name: "rotation",
532     fieldLabel: "Rotation",
533     plugins: new GeoExt.plugins.PrintPageField({
534         printPage: printExtent.pages [0]
535     })
536 }]
537 });
538 //////////////////////////////////////////////////
539
540
541 //editmap.addLayers([userlayer,vectorsarea,markerschoice,pageLayer]);
542
543
544 ////////////////////////////////////////////panel epeksergasias
545 var windowsediapply=new Ext.Window({

```



```

546 title : 'Επεξεργασία της αίτησης: ' + val,
547 width : 1000,
548 height : 600,
549 layout : 'border' ,
550 closable : false,
551 bbar: [{
552     xtype : 'button' ,
553     text : 'Ολοκλήρωση αίτησης' ,
554     handler : function(){
555
556 url = 'lib/applicationtool.php?updateedit=1' ;
557 $.post (url,{id:val},function(data){
558
559
560     Ext.Msg. alert ("ΠΡΟΣΟΧΗ!",data);
561
562     ///
563     //   profilstore.load();
564
565     })
566 }
567
568
569 }, "->" , {
570     xtype : 'button' ,
571     text : 'κλείσιμο' ,
572     iconCls : 'removeedit' ,
573     handler : function(){
574         var applicationedit = Ext.getCmp('applicationedit');
575         applicationedit.setDisabled(false);
576         windowsediapply.destroy();
577         //editmap.removeLayer(userlayer);
578         editmap.removeLayer(vectorsarea);
579         editmap.removeLayer(markerschoice);
580     }
581
582
583 }],
584 items: [
585
586     {
587         xtype : 'panel' ,
588         split : true,
589         collapsible: true,
590         autoScroll: true,
591         region: 'east',
592         id: 'infopanel',
593         frame: true,
594         title: 'Στοιχεία του ' + name + " " + surname,
595         width: 300,
596         items: [{
597             xtype : 'panel' ,
598             html : 'τα ατομικά και γεωχωρικά στοιχεία όπως φαίνονται και στον διπλανό χάρτη είναι τα εξής: <br
599 ><br><li>Α.Φ.Μ: ' + afm + '</li><li>Α.Δ.Τ: ' + identity + '</li><li>ΠΡΑΞΗ: ' + typemap + '</li><li>ΧΑΡΤΗΣ: ' + currentmap + '</li><li>ΑΓΡΟΤΕΜΑΧ
600 ΙΟ: ' + numberofland
601
602         }],
603         //layout: 'border',
604
605     }, mapPanel, formPanel
606     //, formPanel
607
608     ] //telos items
609
610
611
612 });
613
614 ///////////////////////////////////////////////////telos panels
615
616 var applicationedit = Ext.getCmp('applicationedit');
617 applicationedit.setDisabled(true);
618 windowsediapply.show();
619
620
621 } //telos else

```

```
622          ////////////
623
624          }//telos if selection
625
626          }
627      }
628  }
629
630
631
632  }
633
634
635  });
636
637
638  ////////////grid gia oles tis aitiseis
639
640
641
642
643
644
645
646
647
648
649
650
651
652
```

```

1 <?php
2 require_once ('../connect/connect.php' );//pernei stoiceia sindesis
3 $dbconn3 = pg_connect ("host=$Server port=$Port dbname=$Database user=$Username password=$Password" )or die('connection
   failed' );
4
5
6
7 if(isset ($_GET['edit'] )==1)
8 {
9
10 header ("Content-Type: application/json" );
11
12 $sql='SELECT * FROM "public"."Applications"' ;
13 $result =pg_query ($dbconn3 , $sql);
14 if (!$result ){
15     echo "An error occurred. \n" ;
16     exit ;
17
18
19 } $mydata ;
20 $i=0;
21 while ($row = pg_fetch_row ($result ))
22 {
23
24 $mydata [$i]=array ('id_application' =>$row [0], 'name' =>$row [1], 'surname' =>$row [2], 'afm' =>$row [14], 'identity' =>$row [15], 'fathername' =>
   $row [3], 'phone1' =>$row [4], 'phone2' =>$row [5], 'email' =>$row [6], 'typemap' =>$row [7], 'currentmap' =>$row [8], 'numberofland' =>$row [9], 'a
   reachoice' =>$row [10], 'pointchoic' =>$row [11], 'situation' =>$row [12], 'user' =>$row [13]);
25
26 $i++;
27 }
28 //metatropi apo array se json
29 //$out = array_values ($mydata);
30 //echo $out;
31 $out=json_encode ($mydata);
32
33 print $out;
34
35 }
36 if (isset ($_GET['link' ]))
37 {
38 if ($_GET['link' ]==1)//diagrafi apo basi
39 {
40     $id=$_POST['id'];//metafora timis id
41     $sql="DELETE FROM "public"."Applications" WHERE "id_application"=' . $id;
42     $result =pg_exec ($dbconn3 , $sql) or die (pg_errormessage ());
43     if ($result )
44     {
45         echo "διαγράφηκε επιτυχώς!" ;
46     }//if...5
47     else {
48
49         echo "δεν διαγράφηκε!" ;
50
51     }
52 }//if..link=3
53 }
54
55
56 if (isset ($_GET['updateedit' ]))
57 {
58 if ($_GET['updateedit' ]==1)//diagrafi apo basi
59 {
60     $id=$_POST['id'];//metafora timis id
61     $situation =true;
62
63     $sql="UPDATE "public"."Applications" SET ("situation","useredit") = (' . $situation . '\',' . $_COOKIE['user1' ]. '\') WHERE "id_applic
   ation"=' . $id;
64     $result =pg_exec ($dbconn3 , $sql) or die (pg_errormessage ());
65     if ($result )
66     {
67         echo "Η επεξεργασία της αίτησης ολοκληρώθηκε!" ;
68     }//if...5
69     else {
70
71         echo "Η επεξεργασία δεν ολοκληρώθηκε!" ;
72
73     }
74 }//if..link=3

```

```
75 }  
76  
77  
78  
79 ?>
```

```

1  function emailconnecting(processcheck1)
2  {
3  //dimourgoume ta paidia pou einai ta idia kai gia xristes kai gia admin
4  var emailinfo = Ext.data.Record.create([
5  {
6      name:'id_email'
7
8
9      },{
10     name: 'Host' ,
11     type: 'string'
12 },
13 {
14     name: 'Port' ,
15     type: 'string'
16
17 },{
18     name: 'SMTPSecure' ,
19     type: 'string'
20
21 },
22 {
23     name: 'SMTPAuth' ,
24     type: 'string'
25
26 },{
27     name: 'emailuser' ,
28     type: 'string'
29 },{
30     name: 'emailpassword' ,
31     type: 'string'
32 },
33 {
34     name: 'From' ,
35     type: 'string'
36 },
37 {
38     name:'FromName' ,
39     type:'string'
40
41 }
42
43 ]);
44
45
46
47 var storeemail = new Ext.data.GroupingStore({
48     id:'storeemail' ,
49     reader: new Ext.data.JsonReader({fields:emailinfo}),
50     autoLoad:true, //this w,2,3,4ill autoLoad the store on load. Or you can call
51                 //store.load() later.
52     proxy: new Ext.data.HttpProxy({
53         url: 'lib/emailinfo.php?info=1' ,
54         autoAbort :true,
55         disableCaching: true,
56         timeout: 1800000,
57         method: 'POST'
58     })
59 });
60
61
62 if(processcheck1==true)
63 {
64     var gridemailuser = new Ext.grid.GridPanel({
65         id:'gridemailuser' ,
66         store: storeemail,
67         width : 600,
68         region : 'center' ,
69         margins : '0 5 5 5',
70         autoExpandColumn : 'Host' ,
71         // plugins: [editor],
72         view: new Ext.grid.GroupingView({
73             markDirty : false
74         }),
75         columns: [
76             new Ext.grid.RowNumberer(),
77             {
78                 // text:'id application',

```

```

79 width :200,
80 header : 'id_email' ,
81 id: 'id_email' ,
82 sortable : true,
83 dataIndex: 'id_email',
84 hidden:true
85 },{
86 // text:'id application',
87 width :200,
88 header : 'Host' ,
89 id: 'Host' ,
90 sortable : true,
91 dataIndex: 'Host'
92 },{
93 // xtype:'textfield',text:'id application',
94 text : 'Port' ,
95 width : 100,
96 header : 'Port' ,
97 dataIndex : 'Port' ,
98 id: 'Port'
99
100
101
102 },{
103 // text:'surname',
104 // xtype:'textfield',
105 width : 100,
106 header : 'SMTPSecure' ,
107 dataIndex : 'SMTPSecure' ,
108 id: 'SMTPSecure'
109
110
111 },{
112 // text:'surname',
113 // xtype:'textfield',
114 width : 100,
115 header : 'SMTPAuth' ,
116 dataIndex : 'SMTPAuth' ,
117 id: 'SMTPAuth'
118
119
120 },{
121 // text:'surname',
122 // xtype:'textfield',
123 width : 200,
124 header : 'emailuser' ,
125 dataIndex : 'emailuser' ,
126 id: 'emailuser'
127
128
129 },{
130 // text:'fathername',
131 // xtype:'textfield',
132 width : 100,
133 header : 'emailpassword' ,
134 dataIndex : 'emailpassword' ,
135 id: 'emailpassword'
136
137
138 },{
139 // text:'phone1',
140 // xtype:'textfield',
141 width : 200,
142 header : 'From' ,
143 dataIndex : 'From' ,
144 id: 'From'
145
146
147 },{
148 // text:'phone2',
149 // xtype:'textfield',
150 width : 100,
151 header : 'FromName' ,
152 dataIndex : 'FromName' ,
153 id: 'FromName'
154
155
156 }

```



```

157     });
158
159
160     emailconnect=[
161     {
162         id:'emailconnect' ,
163         region : 'center' ,
164         layout : 'border' ,
165         html : 'Παραπάνω αναγράφονται τα στοιχεία του email επικοινωνίας με τους πελάτες' ,
166         width : this.width,
167         height: this.height,
168         items: [gridemailuser]
169     }
170 ];
171
172
173
174 }
175
176 else
177 {
178     var editor = new Ext.ux.grid.RowEditor({
179         saveText : 'Update' ,
180     });
181     //gia diaxeiristi
182     var gridemailadmin = new Ext.grid.GridPanel({
183         id:'gridemailadmin' ,
184         store : storeemail,
185         width : 900,
186         height : 300,
187         plugins : [editor],
188         region:'center',
189         margins: '0 5 5 5',
190         autoExpandColumn: 'Host',
191         // plugins: [editor],
192         view: new Ext.grid.GroupingView({
193             markDirty : false
194         }),
195         columns: [
196             new Ext.grid.RowNumberer(),
197             {
198                 // text:'id application',
199                 width : 200,
200                 header : 'id_email' ,
201                 id: 'id_email' ,
202                 sortable : true,
203                 dataIndex: 'id_email',
204                 hidden:true
205             },{
206                 // text:'id application',
207                 width : 200,
208                 header : 'Host' ,
209                 id: 'Host' ,
210                 sortable : true,
211                 dataIndex: 'Host',
212                 editor: {
213                     xtype : 'textfield' ,
214                     allowBlank : false,
215                     blankText: 'To Host είναι υποχρεωτικό'
216                 }
217             },{
218                 //xtype:"textfield",text:'id application',
219                 text : 'Port' ,
220                 width : 100,
221                 header : 'Port' ,
222                 dataIndex : 'Port' ,
223                 id: 'Port' ,
224                 editor : {
225                     xtype : 'textfield' ,
226                     allowBlank : false,
227                     blankText: 'To Port είναι υποχρεωτικό'
228                 }
229             }
230
231
232         ],{
233             // text:'surname',
234             // xtype:"textfield",

```

```

235     width : 100,
236     header : 'SMTPSecure' ,
237     dataIndex : 'SMTPSecure' ,
238     id : 'SMTPSecure' ,
239     editor : {
240     xtype : 'textfield' ,
241
242     allowBlank : false,
243     blankText: 'To SMTPSecure είναι υποχρεωτικό'
244 }
245
246
247 }, {
248     // text:'surname',
249     // xtype:"textfield",
250     width : 100,
251     header : 'SMTPAuth' ,
252     dataIndex : 'SMTPAuth' ,
253     id : 'SMTPAuth' ,
254     editor : {
255     xtype : 'textfield' ,
256     allowBlank : false,
257     blankText: 'To SMTPAuth είναι υποχρεωτικό'
258 }
259
260
261 }, {
262     // text:'surname',
263     // xtype:"textfield",
264     width : 100,
265     header : 'emailuser' ,
266     dataIndex : 'emailuser' ,
267     id : 'emailuser' ,
268     editor : {
269
270     xtype : 'textfield' ,
271     fieldLabel : 'Email' ,
272     allowBlank : false,
273     blankText: 'To email είναι υποχρεωτικό'
274
275
276     // anchor:'95%'
277
278
279
280 }
281
282
283 }, {
284     // text:'fathername',
285     // xtype:"textfield",
286     width : 100,
287     header : 'emailpassword' ,
288     dataIndex : 'emailpassword' ,
289     id : 'emailpassword' ,
290     editor : {
291     xtype : 'textfield' ,
292     allowBlank : false,
293     blankText: 'To password είναι υποχρεωτικό'
294 }
295
296
297 }, {
298     // text:'phone1',
299     // xtype:"textfield",
300     width : 100,
301     header : 'From' ,
302     dataIndex : 'From' ,
303     id : 'From'
304
305
306
307 }, {
308     // text:'phone2',
309     // xtype:"textfield",
310     width : 100,
311     header : 'FromName' ,
312     dataIndex : 'FromName' ,

```



```

313         id: 'FromName' ,
314         editor: {
315             xtype: 'textfield' ,
316             allowBlank : false,
317             blankText: 'To FromName είναι υποχρεωτικό'
318         }
319     },
320
321     },
322     tbar: [{ xtype:'button',
323         text: 'Ανανέωση',
324         iconCls:'refresh',
325         handler: function () {
326             storeemail.load();
327             storeemail.getView().refresh();
328         }
329     }],
330
331     text: 'Αποθήκευση αλλαγών' ,
332     ref: '../savechange' ,
333     disabled : true,
334     iconCls:'saveedit',
335     listeners: {
336         click : {
337             //scope: this,
338             fn: function () {
339
340                 //////////////////////////////////
341
342                 var gridemailadmin = Ext.getCmp('gridemailadmin');
343                 var selection = gridemailadmin.getSelectionModel().getSelections();
344                 if (selection) {
345
346                     var sel = selection,
347                     model = sel[0],
348                     val = model.get('id_email');
349                     var Host=model.get('Host');
350                     var Port=model.get('Port');
351                     var SMTPSecure=model.get('SMTPSecure');
352                     var SMTPAuth=model.get('SMTPAuth');
353                     var emailuser=model.get('emailuser');
354                     var emailpassword=model.get('emailpassword');
355                     var FromName=model.get('FromName');
356                     //////////////////////////////////
357
358                     Ext.MessageBox.show({
359                         buttons : Ext.Msg.YESNOCANCEL,
360                         title : 'Τροποποίηση?' ,
361                         msg : 'Θέλετε να τροποποιηθούν τα στοιχεία' ,
362                         icon : Ext.Msg.QUESTION,
363                         fn: function(btn){
364                             if (btn == "no"){
365
366                                 //NO
367
368                                 storeemail.load();
369                                 storeemail.getView().refresh();
370                             }
371                             if (btn == "yes"){
372
373                                 url='lib/emailinfo.php?update=1' ;
374
375                                 // store.remove(selection);
376                                 $.post(url,{id_email:val,Host:Host,Port:Port,SMTPSecure:SMTPSecure,SMTPAuth:SMTPAuth,emailuser:
emailuser,emailpassword:emailpassword,FromName:FromName},function(data){
377
378                                     ///messagegia oloklirosei
379
380                                     Ext.Msg.alert("ΤΡΟΠΟΠΟΙΗΣΗ","Τα στοιχεία του Email: "+data);
381
382                                     ////
383                                     storeemail.load();
384
385                                     }) //telos post
386
387                                     } //telos if
388                                 } //telos function
389                             });

```

```

390
391          ///////////
392
393          }//telos if selection
394
395
396          ///////////
397
398          }//function
399      }//click
400
401
402      }// listener
403
404
405
406      }}
407      });
408
409      gridemailadmin.getSelectionModel().on('selectionchange', function(sm){
410          gridemailadmin.savechange.setDisabled (sm.getCount() < 1);
411      });
412
413      emailconnect=[
414
415          {id:'emailconnect',
416            region:'center',
417            layout:'border',
418            html:'Παραπάνω αναγράφονται τα στοιχεία του email επικοινωνίας με τους πελάτες και δύναται να τροποποιηθούν από τον Διαχειριστή',
419            width:this.width,
420            height: this.height,
421            items: [gridemailadmin]
422
423          }
424      ];
425
426
427
428      }
429
430      return emailconnect;
431
432  }

```

```

1 <?php
2 require_once ('../connect/connect.php' );//pernei stoixeia sindesis
3 $dbconn3 = pg_connect ("host=$Server port=$Port dbname=$Database user=$Username password=$Password" )or die('connection failed' );
4
5 $sql='SELECT * FROM "public"."emailinfo" ' ;
6 $result =pg_query ($dbconn3 , $sql);
7
8
9 if (isset ($_GET['info' ]))
10 {
11 $i=0;
12 while ($row = pg_fetch_row ($result ))
13 {
14 //diaxisi pliroforiwn mesou json
15 $mydata ;
16 $mydata[$i]=array ('id_email' =>$row[0], 'Host' =>$row[1], 'Port' =>$row[2], 'SMTPSecure' =>$row[3], 'SMTPAuth' =>$row[4], 'emailuser' =>$row[5], 'emailpassword' =>$row[6], 'From' =>$row[5], 'FromName' =>$row[7]);
17
18 $i++;
19 }
20 $out=json_encode ($mydata);
21
22 print $out;
23
24 }
25 if (isset ($_GET['update' ]))
26 {
27
28 $id_email=$_POST['id_email' ];
29 $Host=$_POST['Host' ];
30 $Port=$_POST['Port' ];
31 $SMTPSecure=$_POST['SMTPSecure' ];
32 $SMTPAuth=$_POST['SMTPAuth' ];
33 $emailuser=$_POST['emailuser' ];
34 $emailpassword=$_POST['emailpassword' ];
35 $FromName=$_POST['FromName' ];
36
37 //metafora timis id
38 $sql='UPDATE "public"."emailinfo" SET ("Host","Port","SMTPSecure","SMTPAuth","emailuser","emailpassword","FromName") = (\'\'.$Host.\'\'.$Port.\'\'.$SMTPSecure.\'\'.$SMTPAuth.\'\'.$emailuser.\'\'.$emailpassword.\'\'.$FromName.\'\' ) WHERE "id_email"='.$id_email ;
39 $result =pg_exec ($dbconn3 , $sql) or die(pg_errormessage ());
40 if($result )
41 {
42     echo "τροποποιήθηκαν επιτυχώς!" ;
43 }//if...1
44 else{
45
46     echo "δεν Τροποποιήθηκαν!" ;
47
48 }
49
50
51 }
52 ?>

```

```

1
2
3
4
5 var user = Ext.data.Record.create([
6     {name:'id',
7       type:'string'
8     },
9     {
10        name: 'Όνομα' ,
11        type: 'string'
12    },
13    {
14        name: 'Επίθετο' ,
15        type: 'string'
16    },
17    {
18        name: 'UserName' ,
19        type: 'string'
20    },
21    {
22        name: 'PassWord' ,
23        type: 'string'
24    },
25    {
26        name: 'Position' ,
27        type: "string"
28    }
29
30
31
32
33
34
35
36
37
38
39
40
41 var profilstore = new Ext.data.GroupingStore({
42     id:'profilstore' ,
43     reader: new Ext.data.JsonReader({fields: user}
44
45     ),
46     autoLoad:true, //this w,2,3,4ill autoLoad the store on load. Or you can call
47                 //store.load() later.
48     proxy: new Ext.data.HttpProxy({
49         url: 'lib/profiltool.php?link=1' ,
50         autoAbort :true,
51         disableCaching: true,
52         timeout: 180000,
53         method: 'POST'
54     })
55 });
56
57
58 //////////////////////////////////////////////////
59
60
61
62
63
64 ///////////////
65
66 var editor = new Ext.ux.grid.RowEditor({
67     saveText : 'Update' ,
68
69 });
70
71 var profilgrid = new Ext.grid.GridPanel({
72     id:'profilgrid' ,
73     store: profilstore,
74     width : 600,
75     region : 'center' ,
76     margins : '0 5 5 5',
77     autoExpandColumn : 'Όνομα' ,
78     plugins : [editor],

```



```

79 view: new Ext.grid.GroupingView({
80     markDirty : false
81 }),
82 tbar: [{ xtype:'button',
83     text: 'Ανανέωση',
84     iconCls:'refresh',
85     handler: function () {
86         profilstore.load();
87         profilgrid.getView().refresh();
88     }
89 }, {
90
91
92     text: 'Αποθήκευση αλλαγών' ,
93     ref: '../savechange' ,
94     disabled : true,
95     iconCls:'saveedit',
96     listeners: {
97         click : {
98             //scope: this,
99             fn: function () {
100
101                 //////////////////////////////////
102
103                 var profilgrid = Ext.getCmp('profilgrid');
104                 var selection = profilgrid.getSelectionModel().getSelections();
105                 if (selection) {
106
107                     var sel = selection,
108                     model = sel[0],
109                     val = model.get('id');
110                     var name=model.get('Όνομα');
111                     var surname=model.get('Επίθετο');
112                     var username=model.get('UserName');
113                     var password=model.get('PassWord');
114                     var position=model.get('Position');
115                     //////////////////////////////////
116
117
118                     Ext.MessageBox.show({
119                         buttons : Ext.Msg.YESNOCANCEL,
120                         title : 'Τροποποίηση?' ,
121                         msg : 'Θέλετε να τροποποιηθούν τα στοιχεία του χρήστη: '+ name,
122                         icon : Ext.Msg.QUESTION,
123                         fn: function(btn){
124                             if (btn == "no"){
125
126                                 //NO
127
128                                 profilstore.load();
129                                 profilgrid.getView().refresh();
130
131                                 }
132                                 if (btn == "yes"){
133                                     if(processcheck1==false)
134                                     {
135                                         var url='lib/profiltool.php?link=2';
136                                     }
137                                     else
138                                     {
139                                         url='lib/save.php?link=4' ;
140                                     }
141                                     // store.remove(selection);
142                                     $.post (url,{id:val,name:name,surname:surname,username:username,password:password,position:positi
143                                     on},function(data){
144
145                                         //////////////////////////////////
146
147                                         ///messagegia oloklirosei
148
149                                         Ext.Msg.alert("ΤΡΟΠΟΠΟΙΗΣΗ","Ο Χρήστης: "+name+" "+data);
150
151                                         ///
152                                         profilstore.load();
153
154                                         }) //telos post
155
156                                         } //telos if
157                                     } //telos function
158                                 });
159
160                                 //////////////////////////////////
161

```

```

156         } //telos if selection
157
158
159         ////////////////
160
161         } //function
162     } //click
163
164
165     } // listener
166
167
168
169     },
170
171     columns: [
172     new Ext.grid.RowNumberer(),
173     {
174         //text : 'id',
175         width : 100,
176         header: 'ID ΧΡΗΣΤΗ',
177         //id:'name',
178         // flex: 1,
179         sortable : true,
180         dataIndex: 'id'
181     },
182     {
183     {
184         id: 'Όνομα',
185         header: 'ONOMA',
186         dataIndex: 'Όνομα',
187         width: 150,
188         sortable: true,
189         editor: {
190             xtype: 'textfield',
191             allowBlank: false
192         }
193     },
194     {
195         id: 'Επίθετο',
196         header: 'ΕΠΙΘΕΤΟ',
197         dataIndex: 'Επίθετο',
198         width: 150,
199         sortable: true,
200         editor: {
201             xtype: 'textfield',
202             allowBlank: false
203         }
204     },
205     {
206         id: 'UserName',
207         header: 'USERNAME',
208         dataIndex: 'UserName',
209         width: 150,
210         sortable: true,
211         editor: {
212             xtype: 'textfield',
213             allowBlank: false
214         }
215     },
216
217     {
218     {
219         id: 'PassWord',
220         header: 'PASSWORD',
221         dataIndex: 'PassWord',
222         width: 150,
223         sortable: true,
224         editor: {
225             xtype: 'textfield',
226             allowBlank: false
227         }
228     },
229     {
230         text: 'Position',
231         header: 'POSITION',
232         dataIndex: 'Position',
233         width: 250,

```

```
234         sortable : true
235
236
237
238
239     }
240 };
241
242 profilgrid.getSelectionModel().on('selectionchange', function(sm){
243     profilgrid.savechange.setDisabled (sm.getCount() < 1);
244 });
245
246
247 // JavaScript Document// JavaScript Document
```



```

1 <?php
2 require_once ('../../connect/connect.php' );//pernei stoixeia sindesis
3 $dbconn3 = pg_connect ("host=$Server port=$Port dbname=$Database user=$Username password=$Password" )or die ('connection
   failed' );
4 $mydata =null;
5
6
7
8
9 if($_GET['link'] ==1)
10 {
11
12 if (isset ($_COOKIE['user1' ]) && isset ($_COOKIE['pass1' ]))
13 {
14
15
16     if(isset ($_COOKIE['mainadmin' ]))
17     {
18 $sql='SELECT * FROM "public"."Admin" ' ;
19
20 $result =pg_exec ($dbconn3 , $sql ) or die (pg_errormessage ());
21
22 $i=0;
23 while ($row = pg_fetch_row ($result ))
24 {
25
26 $mydata [$i]=array ('id' =>$row [4], 'Ονομα' =>$row [0], 'Επίθετο' =>$row [1], 'UserName' =>$row [2], 'PassWord' =>$row [3], 'Position' =>'ΔΙΑΧΕΙ
   ΡΙΣΤΗΣ');
27 $i++;
28
29
30 }
31
32     }
33     else
34     {
35
36 //metafora timis id
37 $sql="SELECT * FROM \"public\".\"Users\" WHERE \"UserName\"=\"$_COOKIE['user1' ].\"\"\" ;
38
39 $result =pg_exec ($dbconn3 , $sql ) or die (pg_errormessage ());
40
41 $i=0;
42 while ($row = pg_fetch_row ($result ))
43 {
44 $mydata [$i]=array ('id' =>$row [0], 'Ονομα' =>$row [1], 'Επίθετο' =>$row [2], 'UserName' =>$row [3], 'PassWord' =>$row [4], 'Position' =>$row [5])
   ;
45 $i++;
46
47
48 }
49 //metatropi apo array se json
50 //Sout = array_values($mydata);
51 //echo $out;
52 }
53 $out=json_encode ($mydata);
54 header ("Content-Type: application/json" );
55 print $out;
56
57
58
59 }//if cookies
60
61 }//if link 1
62
63 if($_GET['link'] ==2)//diagradi apo basi
64 {
65 $id=$_POST['id'];
66 $name=$_POST['name'];
67 $surname=$_POST['surname'];
68 $username=$_POST['username'];
69 $password=$_POST['password'];
70 $position=$_POST['position'];
71
72 //metafora timis id
73 $sql="UPDATE \"public\".\"Admin\" SET (\"Name\", \"SurName\", \"UserName\", \"PassWord\") = (\"$_COOKIE['user1' ].\"$_name.\"\\\", \"$_surname.\"\\\", \"$_username.\"\\\", \"
   \"$_password.\"\\\") WHERE \"Id_Admin\"=\"$_id\" ;
74 $result =pg_exec ($dbconn3 , $sql ) or die (pg_errormessage ());

```

```
75 if($result )
76 {
77     echo "τροποποιήθηκε επιτυχώς!" ;
78     }//if...1
79 else{
80
81     echo "δεν Τροποποιήθηκε!" ;
82
83     }
84 }//if..link=4
85
86
87 ?>
88
89
90
```

```
1 // JavaScript Document
2
3 var kthmatologio=new OpenLayers.Layer.WMS(
4     "KTHMATOLOGIO" ,
5     "http://gis.ktimanet.gr/wms/wmsopen/wmsserver.aspx" ,
6     {layers: "KTBASEMAP", format: "image/png"},{isBaseLayer:true,projection:"EPSG:900913"}
7 );
8
9
10 var osm = new OpenLayers.Layer.OSM("OSM");
11
12
13
14 var options = {
15     //maxResolution: 5000,
16     displayProjection : "EPSG:900913" ,
17     units : 'km' ,
18     restrictedExtent : [boundary[0], boundary[1], boundary[2], boundary[3]],
19     center: [(boundary[0]+boundary[2])/2,(boundary[1]+boundary[3])/2],
20     zoom: 6
21 };
22
23 var maps = new OpenLayers.Map(options);
24 // map.addControl(new OpenLayers.Control.ScaleLine());
25 maps.addLayers([vectors,kthmatologio,osm]);
26
27 maps.addControl(new OpenLayers.Control.LayerSwitcher({ascending:false}));
28 ///////////////////////////////////////////////////
29 var areamap=[
30 {
31     xtype : "gx_mappanel" ,
32     //id:'maps',
33     title : "<center>ΚΑΘΟΡΙΣΜΟΣ ΠΕΡΙΟΧΗΣ ΕΥΘΥΝΗΣ</center>" ,
34     region : 'center' ,
35     map : maps,
36     tbar : [map_controls]
37
38
39
40 }
41
42 ];
43
```

```

1
2 map_controls= [];
3 map_controls=[{
4     xtype:'button',
5     iconCls:'controls_map_zoombyarea',
6     tooltip:"map Zoom By Area",
7     enableToggle: true,
8     toggleHandler:function(btn,pressed)
9     {
10         if (btn.pressed)
11         {
12
13             map_Zoom_ByArea_var =new OpenLayers.Control.ZoomBox();
14
15             maps.addControl(map_Zoom_ByArea_var);
16
17             map_Zoom_ByArea_var.activate();
18         }
19         else
20         {
21             map_Zoom_ByArea_var.deactivate ();
22         }
23     }
24 },
25 ],
26
27 { //epilogi pliroforias me basi shmeiou.
28
29     xtype:'button',
30     iconCls:'box',
31     tooltip:'Καθορισμός οριου',
32     enableToggle: true,
33     toggleHandler:function(btn,pressed)
34     {
35         if (btn.pressed)
36         {
37             boxControl = new OpenLayers.Control();
38             OpenLayers.Util.extend(boxControl, {
39                 draw: function() {
40                     this.box = new OpenLayers.Handler.RegularPolygon(boxControl,
41 {"done": this.notice}, {sides:4, irregular:true, persist:true});
42                     boxControl.box.activate();
43                 },
44
45                 notice: function(geom) {
46
47                     $.post ('lib/areamapnext.php?edit=1&geom='+geom,function(data){
48                         Ext.Msg.alert ('Καταχώρηση!', data);
49
50                         vectors.destroyFeatures();//diagrafi paliou feature gia na deixnei to kainourio
51                         var feature3 = new OpenLayers.Feature.Vector(
52                             OpenLayers.Geometry.fromWKT (
53                                 ""+ geom +""
54                             )
55                         );
56
57                         vectors.addFeatures([feature3]);
58                         var ft = vectors.features;
59                         for(var i=0; i< ft.length; i++){
60                             var boundarys=vectors.features[i].geometry.getBounds().toArray();
61                         };
62
63                         var extent = new OpenLayers.Bounds(boundarys[0], boundarys[1], boundarys[2], boundarys[3]);
64                         maps.setOptions({restrictedExtent: extent});
65                         //var mymaps = Ext.getCmp('mymap');
66
67                         Ext.getCmp('mymap').map.setOptions({restrictedExtent: extent}); //kahtorizw tin idiotita kai stous allous xartes
68
69                         maps.zoomToMaxExtent();
70                         Ext.getCmp('mymap').map.zoomToMaxExtent();
71                     });
72                 }
73             },{
74                 });
75             maps.addControl( boxControl);

```

```

79
80
81     }
82     else
83     {
84
85         boxControl.box.deactivate ();
86
87     }
88 }
89
90     },{
91     xtype:'button' ,
92
93     iconCls:'controls_map_zoomtomapextent' ,
94     tooltip:"ZoomMaxExtent" ,
95     handler:function(){
96
97         maps.zoomToMaxExtent ();
98
99     }
100 },{
101     xtype:'button' ,
102     //text:'reset',
103     iconCls:'reset' ,
104     tooltip:'RESET YOUR AREA MAP' ,
105     handler:function()
106     {
107 var geoms="POLYGON((2090925.1210202998 4130266.3624164,2090925.1210202998 5310172.7469602,3244859.2338484 5310172.7469
602,3244859.2338484004 4130266.3624164,2090925.1210202998 4130266.3624164))";
108     $.post('lib/areamapnext.php?edit=2',function(data){
109         Ext.Msg.alert('Καταχώρηση!', data);
110         vectors.destroyFeatures();//diagrafi paliou feature gia na deixnei to kainourio
111         var feature3 = new OpenLayers.Feature.Vector(
112         OpenLayers.Geometry.fromWKT (
113             ""+ geoms +""
114         )
115         );
116
117         vectors.addFeatures([feature3]);
118         var ft = vectors.features;
119         for(var i=0; i< ft.length; i++){
120             var boundarys=vectors.features[i].geometry.getBounds().toArray();
121         };
122
123         var extent = new OpenLayers.Bounds(boundarys[0], boundarys[1], boundarys[2], boundarys[3]);
124         maps.setOptions({restrictedExtent: extent});
125         Ext.getCmp('mymap').map.setOptions({restrictedExtent: extent}); //kahtorizw tin idiotita kai stous allous xartes
126         maps.zoomToMaxExtent();
127         Ext.getCmp('mymap').map.zoomToMaxExtent();
128     });//telod function data
129
130
131     }
132 }
133 ];
134

```



```
1 <?php
2
3 if (isset($_GET['edit']))
4 {
5     if($_GET['edit']==1)//gia tin proti selida
6     {
7
8     $geom=$_GET['geom'];
9     //anoigma arxeiou
10    $fp=fopen('../../areamap/area.php','w');
11    //bazoume to polygono epilosis
12    $data="<?php \n " ;
13    $data=$data."$". "area=" . $geom . ";\n ?>" ;
14
15    fwrite ($fp, $data);
16    fclose ($fp);
17
18    echo "Επιτυχής καταχώρηση!" ;
19    }
20
21    if($_GET['edit']==2)//gia tin proti selida
22    {
23    $fp=fopen('../../areamap/area.php','w');
24    //bazoume to polygono epilosis
25    $data="<?php \n " ;
26    $data=$data."$". "area='POLYGON((2090925.1210202998 4130266.3624164,2090925.1210202998 5310172.7469602,3244859.2338484 531
0172.7469602,3244859.2338484004 4130266.3624164,2090925.1210202998 4130266.3624164));\n ?>" ;
27
28    fwrite ($fp, $data);
29    fclose ($fp);
30
31    echo "Επαναφορά ορίου περιοχής!" ;
32    }
33 }
34 ?>
```

```

1
2
3
4
5 var user = Ext.data.Record.create([
6     {name:'id',
7       type:'string'
8
9       },{
10        name: 'Όνομα' ,
11        type: 'string'
12    },
13    {
14        name: 'Επίθετο' ,
15        type: 'string'
16    },
17    },
18    {
19        name: 'UserName' ,
20        type: 'string'
21    },
22    },
23    {
24        name: 'PassWord' ,
25        type: 'string'
26    },
27    {
28        name:'Position' ,
29        type:"int"
30    }
31
32    ]
33
34
35
36 });
37
38
39
40 // hideous function to generate employee data
41 var storecombo=new Ext.data.GroupingStore({
42
43     reader: new Ext.data.JsonReader({fields: [{name:'Id_Posistion'}, {name:'Position'}]}
44
45     ),
46     // autoLoad:true,
47     proxy: new Ext.data.HttpProxy({
48         url: 'lib/save.php?link=5' ,
49         autoAbort :true,
50         disableCaching: true,
51         timeout: 180000,
52         method: 'POST'
53     })
54
55
56 });
57
58
59 var store = new Ext.data.GroupingStore({
60     id:'store' ,
61     reader: new Ext.data.JsonReader({fields: user}
62
63     ),
64     autoLoad:true, //this w,2,3,4ill autoLoad the store on load. Or you can call
65         //store.load() later.
66     proxy: new Ext.data.HttpProxy({
67         url: 'lib/save.php?link=1' ,
68         autoAbort :true,
69         disableCaching: true,
70         timeout: 180000,
71         method: 'POST'
72     })
73
74 });
75
76
77 var editor = new Ext.ux.grid.RowEditor({
78     saveText : 'Update' ,

```



```

79
80 });
81
82 var grid = new Ext.grid.GridPanel({
83     id: 'grid' ,
84     store : store,
85     width : 600,
86     region : 'center' ,
87     margins : '0 5 5 5' ,
88     autoExpandColumn : 'Όνομα' ,
89     plugins : [editor],
90     view: new Ext.grid.GroupingView({
91         markDirty : false
92     }),
93     tbar: [{
94         iconCls : 'icon-user-add' ,
95         text : 'Προσθήκη χρήστη' ,
96         handler : function(){
97
98             app=new Ext.Window({
99                 title : "Δημιουργία νέου χρήστη" ,
100                 iconCls : 'newuser' ,
101                 plain : true,
102                 closable: false,
103                 layout: 'border',
104                 height: 250, width: 360,
105                 items: [
106
107                     {
108                         xtype : 'ttext' ,
109                         region : 'south' ,
110                         text : ' © 2014'
111
112                     },
113
114                     {
115                         xtype : 'form' ,
116                         region : 'center' ,
117                         frame : true,
118                         id: 'form',
119                         items: [
120
121                             {
122
123                                 xtype : 'textfield' ,
124                                 fieldLabel : 'Όνομα' ,
125                                 allowBlank : false,
126                                 blankText: 'Το όνομα είναι υποχρεωτικό',
127                                 name: 'name',
128                                 id: 'name'
129                             },
130                             {
131                                 xtype : 'textfield' ,
132                                 fieldLabel : 'Επίθετο' ,
133                                 allowBlank : false,
134                                 blankText: 'Το επίθετο είναι υποχρεωτικό',
135                                 name: 'surname',
136                                 id: 'surname'
137                             },
138                             {
139                                 xtype : 'textfield' ,
140                                 fieldLabel : 'Username' ,
141                                 allowBlank : false,
142                                 blankText: 'Το username είναι υποχρεωτικό',
143                                 name: 'username',
144                                 id: 'username'
145                             },
146                             {
147                                 xtype : 'textfield' ,
148                                 fieldLabel : 'Password' ,
149                                 inputType : 'password' ,
150                                 allowBlank : false,
151                                 blankText: 'Το password είναι υποχρεωτικό',
152                                 name: 'password',
153                                 id: 'password'
154                             },
155                             {
156                                 xtype : 'textfield' ,

```

```

157
158     fieldLabel : 'Θέση' ,
159     name : 'position' ,
160
161
162     value : 1
163 }
164
165 ],//telos item tou form
166 buttons: [
167 {
168     text : 'Δημιουργία' ,
169     iconCls : 'saveedit' ,
170     handler : function() {
171         var form = Ext.getCmp("form").getForm();
172         // var form = this.up('form').getForm();
173         if (form.isValid()) {
174             form.submit({
175                 ///////////////
176                 method : 'POST' ,
177                 url : 'lib/save.php?link=2' ,
178                 reset : false,
179                 waitTitle : "Please Wait",
180                 waitMsg : 'Saving ...',
181
182                 ///////
183                 success: function(form, action) {
184                     Ext.Msg.alert('Προσθήκη', action.result ? action.result.message : 'Η προσθήκη του χρήστη ολοκληρώθηκε');
185                     store.load();
186                     grid.getView().refresh();
187                     grid.getView().refresh();
188                     form.reset();
189                 },
190                 failure: function(form, action) {
191                     Ext.Msg.alert('Failed', action.result ? action.result.message : 'No response');
192                 }
193             });
194         } else {
195             Ext.Msg.alert("Error!", "Η αίτηση δεν είναι σωστή!");
196         }
197     }
198 },//telos tou prosthiki
199 {
200     text : 'Κλείσιμο' ,
201     iconCls : 'removeedit' ,
202     handler : function() {
203         app.close ();
204     }
205 },//telos tou close
206 ],//telos tou button
207 }
208 ],//telo item tou windows
209 });
210 app.show();
211
212
213 }
214 },{
215     ref: './removeBtn' ,
216     iconCls : 'icon-user-delete' ,
217     text : 'Αφαίρεση χρήστη' ,
218     disabled : true,
219     handler: function(){
220         editor.stopEditing ();
221
222
223     },
224     listeners: {
225         click : {
226             // scope: this,
227             fn: function () {
228                 //var grid = Ext.getCmp('grid');
229                 var selection = grid.getSelectionModel().getSelections();
230                 if (selection) {
231
232                     var sel = selection,
233                     model = sel[0],
234                     val = model.get('id');

```

```

235 var user=model.get("Όνομα");
236 //////////////////////////////////////////////////
237
238
239 Ext.MessageBox.show({
240     buttons : Ext.Msg.YESNOCANCEL,
241     title : 'Διαγραφή?' ,
242     msg : 'Θέλετε να διαγραφεί ο χρήστης: '+ user,
243     icon : Ext.Msg.QUESTION,
244     fn: function(btn){
245         if (btn == "no"){
246             store.load()
247         }
248         if (btn == "yes"){
249             for(var i = 0, r; r = selection[i]; i++){
250                 store.remove (selection);
251             }
252             // store.remove(selection);
253             $.post ('lib/save.php?link=3',{id:val},function(data){
254
255                 ///messagegia oloklirosei
256
257                 Ext.Msg.alert("ΔΙΑΓΡΑΦΗ","Ο Χρήστης: "+user+" "+data);
258
259                 ///
260
261             }) //telos post
262
263             } //telos if
264         } //telos function
265     });
266
267     //////////////////////////////////
268
269     } //telos if selection
270
271     }
272 }
273 }
274 }, { xtype:'button',
275     text: 'Ανανέωση',
276     iconCls:'refresh',
277     handler: function () {
278         store.load();
279         grid.getView().refresh();
280     }
281 }, "->",{
282
283
284     text: 'Αποθήκευση αλλαγών' ,
285     ref: '../savechange' ,
286     disabled : true,
287     iconCls:'saveedit',
288     listeners: {
289         click : {
290             //scope: this,
291             fn: function () {
292
293                 //////////////////////////////////
294
295                 var grid = Ext.getCmp('grid');
296                 var selection = grid.getSelectionModel().getSelections();
297                 if (selection) {
298
299                     var sel = selection,
300                     model = sel[0],
301                     val = model.get('id');
302                     var name=model.get("Όνομα");
303                     var surname=model.get("Επίθετο");
304                     var username=model.get("UserName");
305                     var password=model.get("PassWord");
306                     var position=model.get("Position");
307                     //////////////////////////////////
308
309
310                     Ext.MessageBox.show({
311                         buttons : Ext.Msg.YESNOCANCEL,
312                         title : 'Τροποποίηση?' ,

```

```

313         msg: 'Θέλετε να τροποποιηθούν τα στοιχεία του χρήστη: '+ name,
314         icon: Ext.Msg.QUESTION,
315         fn: function(btn){
316             if (btn == "no"){
317                 //NO
318                 store.load();
319                 grid.getView().refresh();
320             }
321             if (btn == "yes"){
322                 // store.remove(selection);
323                 $.post('lib/save.php?link=4',{id:val,name:name,surname:surname,username:username,password:passw
324 ord,position:position},function(data){
325
326                 ///messagegia oloklirosei
327
328                 Ext.Msg.alert("ΤΡΟΠΟΠΟΙΗΣΗ","Ο Χρήστης: "+name+" "+data);
329
330                 ////
331                 store.load();
332
333                 }) //telos post
334
335                 } //telos if
336             } //telos function
337         });
338
339         ////////////
340
341         } //telos if selection
342
343         //////////////////////////////////////
344
345         } //function
346     } //click
347
348     } // listener
349
350
351
352
353
354     },
355
356     columns: [
357         new Ext.grid.RowNumberer(),
358         {
359             //text : 'id',
360             width : 100,
361             header: 'ID ΧΡΗΣΤΗ',
362             //id:'name',
363             // flex: 1,
364             sortable : true,
365             dataIndex: 'id'
366         },
367     ],
368     {
369         id: 'Όνομα',
370         header: 'ΟΝΟΜΑ',
371         dataIndex: 'Όνομα',
372         width: 150,
373         sortable : true,
374         editor: {
375             xtype: 'textfield' ,
376             allowBlank : false
377         }
378     },
379     {
380         id: 'Επίθετο',
381         header: 'ΕΠΙΘΕΤΟ',
382         dataIndex: 'Επίθετο',
383         width: 150,
384         sortable : true,
385         editor: {
386             xtype: 'textfield' ,
387             allowBlank : false
388         }
389     },

```

```

390 {
391     id: 'UserName',
392     header: 'USERNAME',
393     dataIndex: 'UserName',
394     width: 150,
395     sortable: true,
396     editor: {
397         xtype: 'textfield',
398         allowBlank: false
399     }
400 },
401
402 {
403     id: 'PassWord',
404     header: 'PASSWORD',
405     dataIndex: 'PassWord',
406     width: 150,
407     sortable: true,
408     editor: {
409         xtype: 'textfield',
410         allowBlank: false
411     }
412 }
413 }, {
414     header: 'POSITION',
415     dataIndex: 'Position',
416     width: 250,
417     sortable: true,
418     editor: {
419         xtype: 'textfield',
420         allowBlank: false,
421         valueField: 1,
422         editable: false
423     }
424 }
425 }
426
427
428
429 ]]
430 });
431
432 grid.getSelectionModel().on('selectionchange', function(sm){
433     grid.removeBtn.setDisabled (sm.getCount() < 1);
434     grid.savechange.setDisabled(sm.getCount() < 1);
435 });
436
437
438 // JavaScript Document

```



```

1  <?php
2  if (isset($_GET['link']))
3  {
4      require_once ('../connect/connect.php' );//pernei stoixeia sindesis
5      $dbconn3 = pg_connect ("host=$Server port=$Port dbname=$Database user=$Username password=$Password" )or die('connectio
n failed' );
6
7      if($_GET['link'] ==1)//gia tin proti selida
8      {
9
10
11      header("Content-Type: application/json" );
12
13
14
15
16      $sql='SELECT * FROM "public"."Users"' ;
17      $result=pg_query ($dbconn3 , $sql);
18      if (!$result ) {
19          echo "An error occured. \n" ;
20          exit ;
21
22
23      } $mydata ;
24      $i=0;
25      while ($row = pg_fetch_row ($result ))
26      {
27
28      $mydata [$i]=array ('id'=>$row [0], 'Όνομα' =>$row [1], 'Επίθετο' =>$row [2], 'UserName' =>$row [3], 'PassWord' =>$row [4], 'Position' =>$row [5]
);
29      $i++;
30
31
32      }
33      //metatropi apo array se json
34      //$out = array_values($mydata);
35      //echo $out;
36      $out=json_encode ($mydata );
37
38      print $out ;
39
40      } //if...link=1
41      if($_GET['link'] ==2)//gia tin proti selida
42      {
43          // post metabliton apo parathiro eisagwgeis neou user
44          $Name=$_POST['name' ];
45          $Surname=$_POST['surname' ];
46          $Username2=$_POST['username' ];
47          $Password2=$_POST['password' ];
48          $position=$_POST['position' ];
49
50      $sql='INSERT INTO "public"."Users"("Name","SurName","UserName","PassWord","Id_Position") VALUES ("' . $Name . '\',' . $Surname .
'\',' . $Username2 . '\',' . $Password2 . '\',' . $position . '\')';
51      $result=pg_exec ($dbconn3 , $sql) or die(pg_errormessage ());
52
53
54
55      } //if...link=2
56
57      if($_GET['link'] ==3)//diagrafi apo basi
58      {
59          $id=$_POST['id' ]; //metafora timis id
60          $sql='DELETE FROM "public"."Users" WHERE "Id_Users"=' . $id ;
61          $result=pg_exec ($dbconn3 , $sql) or die(pg_errormessage ());
62          if($result )
63          {
64              echo "διαγράφηκε επιτυχώς!" ;
65              } //if...5
66          else {
67
68              echo "δεν διαγράφηκε!" ;
69
70          }
71      } //if...link=3
72
73      if($_GET['link'] ==4)//diagrafi apo basi
74      {
75          $id=$_POST['id' ];

```

```

76 $name=$_POST['name'];
77 $surname=$_POST['surname'];
78 $username=$_POST['username'];
79 $password=$_POST['password'];
80 $position=$_POST['position'];
81
82 //metafora timis id
83 $sql="UPDATE "public"."Users" SET ("Name","SurName","UserName","PassWord","Id_Position") = ('
      . $name . '\,\' . $surname . '\,\' .
      $username . '\,\' . $password . '\,\' . $position . '\') WHERE "Id_Users"=' . $id;
84 $result=pg_exec($dbconn3, $sql) or die(pg_errormessage ());
85 if($result)
86 {
87     echo "τροποποιήθηκε επιτυχώς!" ;
88 } //if...1
89 else{
90
91     echo "δεν Τροποποιήθηκε!" ;
92
93 }
94 } //if..link=4
95
96 if($_GET['link']==5) //json gia position
97 {
98     header("Content-Type: application/json" );
99     //metafora timis id
100 $sql='SELECT * FROM "public"."Position" ;
101 $result=pg_exec($dbconn3, $sql) or die(pg_errormessage ());
102 $mydata;
103 $i=0;
104 while ($row = pg_fetch_row ($result))
105 {
106
107 $mydata[$i]=array('Id_Posistion' =>$row[0], 'Position' =>$row[1]);
108 $i++;
109
110 }
111 }
112 //metatropi apo array se json
113 //$out = array_values($mydata);
114 //echo $out;
115 $out=json_encode ($mydata);
116
117 print $out;
118
119 } //if..link=5
120
121
122 } //if....1
123 ?>

```


ΕΦΑΡΜΟΓΗ ΑΙΤΗΣΗ ΧΑΡΤΩΝ

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.d
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>Αίτηση Χαρτών</title>
6 <link rel="icon" href="images/custom-reports.png" type="image/x-icon">
7 <link rel="stylesheet" type="text/css" href="css/main.css"/>
8 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/geoext/resources/css/popup.css">
9 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/resources/css/ext-all.css" />
10 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/grid/grid-examples.css" />
11 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/shared/examples.css" />
12 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/openlayers/theme/default/style.css" />
13 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/openlayers/theme/default/google.css" />
14 <link rel="stylesheet" type="text/css" href="../external/gxp-master/externals/geoext/resources/css/geoext-all.css" />
15 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/resources/css/xtheme-access2.css" />
16 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/ux/css/RowEditor.css" />
17 <link rel="stylesheet" type="text/css" href="../external/gxp-master/src/theme/all.css"/>
18 <link rel="stylesheet" type="text/css" href="../external/ext-3.4.0/examples/ux/statusbar/css/statusbar.css" />
19 <script src="../external/gxp-master/externals/openlayers/lib/OpenLayers.js"></script>
20 <script type="text/javascript" src="../external/proj4js/proj4js-compressed.js"></script>
21 <?php include "../areamap/area.php"; ?>
22 <script type="text/javascript">
23 Proj4js.defs ["EPSG:4121"] = "+proj=tmerc +lat_0=0 +lon_0=24 +k=0.9996 +x_0=500000 +y_0=0 +ellps=GRS80 +towgs84=-199.87,74.79,24
6.62,0,0,0,0 +units=m +no_defs";
24
25 function load()
26 {
27 Ext.onReady(function () {
28 Ext.Msg.show({
29 title : 'ΠΡΟΣΟΧΗ',
30 msg : "Σας ενημερώνουμε ότι η γεωαναφορά των χαρτών έγινε κατά προσέγγιση για την ηλεκτρονική εξυπηρ
31 έτησή σας, ενώ τα πραγματικά δεδομένα μπορείτε να τα παραλάβετε κατόπιν αίτησης!",
32 buttons : Ext.Msg.OK,
33 icon : "attention"
34 });
35 });
36 }
37
38
39
40 //kathorismo tou oriou perioxhs
41 var boundarymap="<?php echo $area; ?>";
42
43 var vectors = new OpenLayers.Layer.Vector("vector", {isBaseLayer: false});
44 var feature2 = new OpenLayers.Feature.Vector(
45 OpenLayers.Geometry.fromWKT (
46 ""+ boundarymap +""
47 )
48 );
49
50 vectors.addFeatures([feature2]);
51 var ft = vectors.features;
52 for(var i=0; i< ft.length; i++){
53 var boundary=vectors.features[i].geometry.getBounds().toArray();
54 };
55
56 </script>
57
58
59
60
61 <script src="../external/ext-3.4.0/adaptor/ext/ext-base.js"></script>
62 <script src="../external/ext-3.4.0/ext-all.js"></script>
63
64
65
66 <script src="../external/gxp-master/externals/geoext/lib/GeoExt.js"></script>
67
68 <script src="../external/gxp-master/externals/geoext/lib/overrides/override-ext-ajax.js"></script>
69 <script src="../external/gxp-master/src/script/loader.js"></script>
70
71 <script type="text/javascript" src="../external/gxp-master/examples/viewer/ux/RowExpander.js"></script>
72 <script type="text/javascript" src="viewer.js"></script>
73
74 <link rel="stylesheet" type="text/css" href="../external/gxp-master/src/theme/all.css"/>
75 <link rel="stylesheet" type="text/css" href="../external/gxp-master/examples/viewer.css"/>
76 <script type="text/javascript" src="../external/jquery-1.11.0.min.js"></script>

```

```

77
78
79 <link href="css/style_login.css" rel="stylesheet" type="text/css" />
80 <script src="../external/gxp-master/src/script/locale/el.js"></script>
81 <script type="text/javascript" src="../external/ext-3.4.0/examples/ux/RowEditor.js"></script>
82 <script type="text/javascript" src="../external/ext-3.4.0/examples/ux/statusbar/StatusBar.js"> </script>
83 <script type="text/javascript" src="../external/ext-3.4.0/examples/ux/statusbar/ValidationStatus.js"> </script>
84 <script type="text/javascript" src="../external/function.js"></script>
85
86 <script type="text/javascript">
87
88 Ext.onReady(function () {
89
90     load();
91     ///////////////////////////////////workspaces////////////////////////////////////
92     var data = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/workspaces.json");
93     var data1=Ext.decode(data);
94     var data2=data1['workspaces']['workspace'];
95     if(data2)
96     {
97
98     }
99     else{
100         data3={"workspaces":{"workspace":[{"name":"empty","href":"empty"}]}} ;
101         var data1=Ext.decode(data3);
102         var data2=data1['workspaces']['workspace'];
103     }
104     //////////////////////////////////workspace store namespacesdata2
105     var mystore = new Ext.data.JsonStore({
106         id:'mystore' ,
107         //autoload: true,
108         fields : [
109             {type: 'string', name: 'href'},
110             {type: 'string', name: 'name'}
111         ]
112     });
113
114     mystore.loadData(data2);
115     //////////////////////////////////
116     var namespacestore = new Ext.data.JsonStore({
117         id:'namespacestore' ,
118         //autoload: true,
119         fields : [
120             {type: 'string', name: 'href'},
121             {type: 'string', name: 'name'}
122         ]
123     });
124
125 });
126
127
128
129 //////////////////////////////////dimiourgia formas aitishs
130
131 var questionform=new Ext.FormPanel({
132     id:'questionform' ,
133     url:'lib/editrequest.php?email=1' ,
134     labelAlign : 'top' ,
135     frame :true,
136     title: 'Παρακαλώ επικοινωνήστε για οποιοδήποτε πρόβλημα',
137     bodyStyle:'padding:5px 5px 0',
138     width: 600,
139     height:400,
140     items: [{
141         layout : 'column' ,
142         items : [{
143             columnWidth : .5,
144             layout : 'form' ,
145             items : [{
146                 xtype : 'textfield' ,
147                 fieldLabel : 'Όνομα' ,
148                 allowBlank : false,
149                 blankText: 'Το όνομα είναι υποχρεωτικό',
150                 name: 'firstname',
151                 anchor:'95%'
152             },{
153                 xtype : 'textfield' ,
154                 fieldLabel : 'Θεμα',

```



```

155         allowBlank :false,
156         blankText: 'Το Θέμα είναι υποχρεωτικό',
157         name: 'subject',
158         id: 'subject',
159         anchor:'95%'
160     }
161 },{
162     columnWidth :.5,
163     layout : 'form' ,
164     items : [{
165         xtype : 'textfield' ,
166         fieldLabel : 'Επίθετο' ,
167         allowBlank :false,
168         blankText: 'Το Επίθετο είναι υποχρεωτικό',
169         name: 'lastname',
170         id: 'lastname',
171         anchor:'95%'
172     },{
173         xtype : 'textfield' ,
174         fieldLabel : 'Email' ,
175         allowBlank :false,
176         blankText: 'Το email είναι υποχρεωτικό',
177         name: 'emailaddress',
178         id: 'emailaddress',
179         vtype:'email',
180         anchor:'95%'
181     }
182     ]
183 },{
184     xtype : 'htmleditor' ,
185     id : 'question' ,
186     allowBlank :false,
187     name: 'questiondata',
188     id: 'questiondata',
189     fieldLabel: "",
190     height: 200,
191     anchor: '98%'
192 },
193
194 buttons: [{
195     text : 'Αποστολή' ,
196     iconCls : 'saveedit' ,
197     handler : function(){
198         var questionform = Ext.getCmp('questionform');
199         if(questionform.getForm().isValid()){
200             var sb = Ext.getCmp('form-statusbar1');
201             sb.showBusy('Παρακαλώ περιμένετε...');
202             questionform.getEl().mask();
203             questionform.getForm().submit({
204                 url : 'lib/editrequest.php?email=1' ,
205                 success : function(form,action){
206                     obj = Ext.util.JSON.decode (action.response.responseText);
207                     questionform.getForm().reset();
208                     sb.setStatus({
209                         text : obj.message,
210                         iconCls : "",
211                         clear : true
212                     });
213                     questionform.getEl().unmask();
214                 }
215             });
216         }
217     }
218 }
219
220 },{
221     text : 'Κλείσιμο' ,
222     iconCls : 'removeedit' ,
223     handler : function(){
224         questionwindow.hide ();
225         questionform.getForm().reset();
226     }
227 }
228 });
229
230 //////////////////////////////////////////////////
231
232

```

```

233 var formpanel = new Ext.FormPanel({
234 url : 'lib/editrequest.php?edit=1' ,
235 labelWidth : 150,
236 frame : true,
237 id : 'formpanel',
238 defaultType : 'textfield',
239 items : [ {
240     xtype : 'panel' ,
241     title : '<center>ΑΙΤΗΣΗ</center>' ,
242
243     items : [{
244         title : 'ΠΡΟΣΩΠΙΚΑ ΣΤΟΙΧΕΙΑ' ,
245         layout : 'form' ,
246         defaults : {width: 230},
247         defaultType : 'textfield',
248         items : [{
249             fieldLabel : 'Όνομα' ,
250             name : 'first' ,
251             allowBlank : false,
252             selectOnFocus : true,
253             //msgTarget: 'side',
254             blankText : 'Το όνομα είναι υποχρεωτικό'
255             // value: 'Jack'
256         }, {
257             fieldLabel : 'Επίθετο' ,
258             name : 'last' ,
259             allowBlank : false,
260             selectOnFocus : true,
261             //msgTarget: 'side',
262             blankText : 'Το επίθετο είναι υποχρεωτικό'
263         }, {
264             fieldLabel : 'ΑΦΜ' ,
265             name : 'afm' ,
266             allowBlank : false,
267             selectOnFocus : true,
268             //msgTarget: 'side',
269             blankText : 'Το ΑΦΜ είναι υποχρεωτικό'
270
271
272
273         }, {
274             fieldLabel : 'Α.Δ.Τ' ,
275             name : 'identity' ,
276             allowBlank : false,
277             selectOnFocus : true,
278             //msgTarget: 'side',
279             blankText : 'Η ταυτότητα είναι υποχρεωτική'
280
281
282
283         }, {
284             fieldLabel : 'Όνομα Πατρός' ,
285             name : 'fatherfirst'
286
287         } ] //
288     },
289     //////////////////////////////////////
290
291     {
292         title : 'ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ' ,
293         layout : 'form' ,
294         defaults : {width: 230},
295         defaultType : 'textfield',
296
297         items : [{
298             xtype : "numberfield" ,
299             fieldLabel : 'Σταθερο τηλέφωνο' ,
300             name : 'homephone' ,
301             id : 'homephone'
302
303
304         }, {
305             xtype : "numberfield" ,
306             fieldLabel : 'Κινητό τηλέφωνο' ,
307             name : 'mobilephone' ,
308             id : 'mobilephone'
309
310         }, {

```

```

311         fieldLabel : 'Email' ,
312         name : 'email' ,
313         vtype : 'email' ,
314         id : 'email'
315     } //
316 },
317 //////////////////////////////////////////////////
318
319 {
320
321     title : 'ΓΕΩΧΩΡΙΚΑ ΣΤΟΙΧΕΙΑ ' ,
322     //collapsible: true,
323     layout : 'form' ,
324     id : 'geo' ,
325     defaults : {width: 230},
326     defaultType : 'textfield',
327     items:[{
328         xtype : 'combo' ,
329         store : mystore,
330         displayField : 'name' ,
331         valueField : 'name' ,
332         mode : 'local' ,
333         triggerAction : 'all' ,
334         listClass : 'comboalign' ,
335         typeAhead : true,
336         forceSelection: true,
337         selectOnFocus: true,
338         editable:false,
339         fieldLabel: 'Περιοχή',
340         id: 'area',
341         allowBlank: false,
342         blankText: 'Απαιτείται επιλογή περιοχής',
343         name: 'area',
344         listeners: {
345             change : function() {
346                 var area=Ext.getCmp('area');
347                 if(area.getValue()!="")
348             {
349                 Ext.getCmp ('RasterNamespace').reset();
350                 // alert(comboworkspacess.getValue());
351                 var namespacedata = getJSONData(window.location.protocol + "://" + window.location.host + "/geoserver/rest/workspaces/" + area.getValue() +
352                 "/coveragestores.json");
353                 var namespacedata1=Ext.decode(namespacedata);
354                 var namespacedata2=namespacedata1["coverageStores"]["coverageStore"];
355                 //alert(Ext.encode(namespacedata2));
356                 namespacestore.loadData(namespacedata2);
357             }
358         }
359     }
360 }
361
362 },{
363     xtype : 'combo' ,
364     fieldLabel : 'Συγκεκριμένος Χάρτης' ,
365     editable : false,
366     name: 'RasterNamespace',
367     id: 'RasterNamespace',
368     displayField: 'name',
369     valueField: 'name',
370     mode: 'local',
371     triggerAction: 'all',
372     listClass: 'comboalign',
373     typeAhead: true,
374     forceSelection: true,
375     selectOnFocus: true,
376     allowBlank: false,
377     blankText: 'Απαιτείται το όνομα του φύλλου χάρτη',
378     store: namespacestore
379
380     // store: namespacestore
381 },{
382     xtype : "numberfield" ,
383     fieldLabel : 'Αριθμός αγροτεμαχίου' ,
384     name : 'numberarea' ,
385     allowBlank : false,
386     blankText: 'Απαιτείται αριθμός αγροτεμαχίου',
387     id: 'numberarea'

```

```

388
389 },{
390     fieldLabel : 'Επιλογή περιοχής' ,
391     allowBlank : false,
392     blankText: 'Απαιτείται περιοχή επιλογής',
393     name: 'areachoice',
394     id:'areachoice',
395     editable:false,
396     // allowBlank:false,
397     selectOnFocus: true,
398     listeners: {
399         'render': function() {
400             var areachoice=Ext.getCmp('areachoice');
401             areachoice.getEl().on('click', function(){
402                 Ext.Msg. alert("", "Επέλεξε περιοχή ενδιαφέροντος από το χάρτη<br>με χρήση του εργαλείου: <img src='images/box.png'>");
403                 // var homenumber=Ext.getCmp('home');
404                 var boxarea=Ext.getCmp('boxarea');
405                 boxarea.setDisabled(false);
406                 //alert("hello");
407                 // homenumber.setValue("playe");
408             }
409         )
410     }
411 }
412
413 },{
414     fieldLabel : 'Επιλογή σημείου εντός αγροτεμαχίου' ,
415     allowBlank : false,
416     blankText: 'Απαιτείται σημείο εντός αγροτεμαχίου',
417     name: 'pointchoice',
418     id:'pointchoice',
419     editable:false,
420     // allowBlank: false,
421     selectOnFocus: true,
422     listeners: {
423         'render': function() {
424             var pointchoice=Ext.getCmp('pointchoice');
425             pointchoice.getEl().on('click', function(){
426                 Ext.Msg. alert("", "Επέλεξε σημείο εντός του αγροτεμαχίου ενδιαφέροντος από το χάρτη<br>με χρήση του εργαλείου: <img src='images/point.png'>");
427                 // var homenumber=Ext.getCmp('home');
428                 var point=Ext.getCmp('point');
429                 point.setDisabled(false);
430                 //alert("hello");
431                 // homenumber.setValue("playe");
432             }
433         )
434     }
435 }
436 //msgTarget: 'side',
437 //blankText: 'Name is required'
438 }
439
440 ]
441
442
443 }
444
445 ]//
446
447 },
448 {
449     xtype : 'compositefield' ,
450     items : [
451         { xtype : 'checkbox',
452           id:'term',
453           name:'term',
454
455           checked:false,
456           // selectOnFocus: true,
457           listeners : {
458               check : {
459                   fn : function(checkbox, checked){
460                       //your logic
461                       if(checkbox.checked==true)
462                           {
463

```



```

464     var sendrequest=Ext.getCmp('sendreq');
465
466     Ext.Msg.show({
467         title : 'ΠΡΟΣΟΧΗ',
468         msg : "Με τη συγκεκριμένη επιλογή δηλώνεται υπεύθυνα η εγκυρότητα των στοιχείων που θα υποβληθούν!!",
469         buttons : Ext.Msg.OK,
470         icon : "attention"
471     });
472     sendrequest.setDisabled(false);
473
474 }
475 if(checkbox.checked==false)
476 {
477     Ext.getCmp ('sendreq').setDisabled(true);
478 }
479
480
481
482
483 }
484 // scope : your-scope //---> Optional
485 }
486 }
487
488 },
489 { xtype : 'label' , text : 'Αποδοχή όρων υποβολής αίτησης' }
490 ]
491 }
492
493 ],
494
495
496 buttons: [{
497     text : 'Αποστολή αίτησης' ,
498     iconCls : 'saveedit' ,
499     id:'sendreq' ,
500     disabled :true,
501     handler: function(){
502         //if(Ext.getCmp("term").getValue()==false)
503         //{
504             //alert("hello");}
505
506         if(formpanel.getForm().isValid()){
507             var sb = Ext.getCmp("form-statusbar");
508             sb.showBusy('Αποστολή αίτησεις..');
509             formpanel.getEl().mask();
510             formpanel.getForm().submit({
511                 url : 'lib/editrequest.php?edit=1' ,
512                 //////////////////////////////////////////////////
513                 success : function(form,action){
514                     obj = Ext.util.JSON.decode (action.response.responseText);
515                     formpanel.getForm().reset();
516                     sb.setStatus({
517                         text : obj.message,
518                         iconCls : "",
519                         clear : true
520                     });
521                     formpanel.getEl().unmask();
522                 },
523
524                 //////////////////////////////////////////////////
525                 failure:function(form, action){
526                     obj = Ext.util.JSON.decode(action.response.responseText);
527                     formpanel.getForm().reset();
528                     sb.setStatus({
529                         text : obj.message,
530                         iconCls : "",
531                         clear : true
532                     });
533                     formpanel.getEl().unmask();
534
535                 }
536                 //////////////////////////////////////////////////
537             });
538             //} //telos if(checkbox)
539             //else
540             //{
541             /* Ext.Msg.show({

```

```

542         title:'ΠΡΟΣΟΧΗ',
543         msg: "Δεν έχετε επιβεβαιώσει τους όρους χρήσης!",
544         buttons:Ext.Msg.OK,
545         icon: "attention"
546     });
547     */
548     //}
549 }
550 }
551 },{
552     text : 'Ακυρο' ,
553     iconCls : 'removeedit' ,
554     handler : function(){
555         var formpanel=Ext.getCmp('formpanel');
556         formpanel.getForm().reset();
557     }
558 }]
559
560 });
561
562
563 //////////////////////////////////////
564
565 //////////////////////////////////////windows////////////////////////////////////
566
567
568 var questionwindow=new Ext.Window({
569     id:'questionwindow' ,
570     title : 'ΠΑΡΑΘΥΡΟ ΕΠΙΚΟΙΝΩΝΙΑΣ' ,
571     layout : 'fit' ,
572     closable : false,
573     width: 600,
574     //height:400,
575     height:this.height,
576     bbar: new Ext.ux.StatusBar({
577         id : 'form-statusbar1' ,
578         defaultText : 'Συμπλήρωση στοιχείων...' ,
579         plugins : new Ext.ux.ValidationStatus({form:'questionform'})
580     }),
581     items:questionform
582
583 });
584
585
586 //////////////////////////////////////
587
588 itemsrequest=[
589 {xtype:'panel',title:'ΧΑΡΤΗΣ ΕΥΡΕΣΗΣ ΓΕΩΧΩΡΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ',region:'center',
590     html:'<div id="tab"></div>',
591     id:'mappanels',
592     iconCls: 'editlayer',
593
594     items: []
595
596     },
597 {
598     title:'ΣΤΟΙΧΕΙΑ ΑΙΤΗΣΗΣ',
599
600
601     margins: '5 5 0 0',
602     xtype:'panel',
603     region:'north',
604     //layout:'fit',
605     iconCls: 'titleearth',
606     tbar:['->',
607     {
608         xtype : 'button' ,
609         iconCls : 'help' ,
610         text : 'Βοήθεια' ,
611         handler :function(){
612
613             window .open ('helpclient.pdf', 'newwindow', 'width=700, height=600');
614         }
615
616
617     },{
618
619

```

```

620     xtype : 'button' ,
621     iconCls : 'email' ,
622     text : 'Επικοινωνία' ,
623     handler : function(){
624
625         questionwindow.show ();
626
627     }
628
629 },
630
631 },{
632     xtype : 'ttext' ,
633     region : 'south' ,
634     text : '<center> © 2014 - Ε.Μ.Π Διπλωματική Εργασία - Πισπιδίκη Ιωάννης-Γεωχωρικός Χάρτης Αιτήσεων</cen
ter>'
635
636     },{
637         xtype : 'panel' ,
638
639         region : 'east' ,
640         frame : true,
641         autoScroll : true,
642         bbar : new Ext.ux.StatusBar({
643             id : 'form-statusbar' ,
644             defaultText : 'Συμπλήρωση αίτησης..' ,
645             plugins : new Ext.ux.ValidationStatus({form:'formpanel'})
646         }),
647         items : formpanel,
648         width : 430,
649         collapsible : true,
650         autoScroll : true,
651         collapsed : true
652
653
654
655     }
656
657
658 ];
659
660
661 var viewport=new Ext.Viewport({layout:"border",id:'viewport',items:itemsrequest
662
663
664 });
665
666 // Ext.getCmp('home').on('click', function()
667
668 });
669
670
671
672 </script>
673 </head>
674
675 <body>
676
677 </body>
678 </html>

```

```

1
2
3 <?php
4
5
6
7     if (isset($_GET['email']))
8     {
9
10
11     require_once('../connect/connect.php');//pernei stoixeia sindesis
12     $dbconn3 = pg_connect("host=$Server port=$Port dbname=$Database user=$Username password=$Password")or die('connection failed');
13
14     $sql='SELECT * FROM "public"."emailinfo"';
15     $result=pg_query($dbconn3, $sql);
16
17
18
19
20     $i=0;
21     while ($row = pg_fetch_row($result))
22     {
23
24
25         $host= $row[1];
26         $port= $row[2];
27         $SMTPSecure = $row[3];
28         $SMTPAuth = $row[4];
29         $emailuser = $row[5];
30         $emailpassword = $row[6];
31         $From = $emailuser;
32         $FromName = $row[7];
33
34
35         $i++;
36     }
37     $firstname=$_POST['firstname'];
38     $lastname=$_POST['lastname'];
39     $emailaddress=$_POST['emailaddress'];
40     $subject=$_POST['subject'];
41     $questiondata=$_POST['questiondata'];
42     $questiondata=wordwrap($questiondata, 70);
43
44
45     require ("../external/PHPMailer-master/class.phpmailer.php");//einai gia na leitourgei to phpmail-master
46
47     $results_messages = array();
48
49     $mail = new PHPMailer(true);
50     $mail->CharSet = 'utf-8';
51     ///////////////
52     $mail1 = new PHPMailer(true);
53     $mail1->CharSet = 'utf-8';
54
55     class phpmailerAppException extends phpmailerException {}
56
57     try {
58         $to = $emailaddress;
59         if(!PHPMailer::validateAddress($to)) {
60             // throw new phpmailerAppException("Email address " . $to . " is invalid -- aborting!");
61             echo "{success: true,message:'Email address " . $to . " is invalid -- aborting!'}";
62         }
63     }
64     $mail->isSMTP();
65     $mail1->isSMTP();
66     //$mail->SMTPDebug = 2;
67     $mail->Host = $host;
68     $mail->Port = $port;
69     $mail->SMTPSecure = $SMTPSecure;
70     $mail->SMTPAuth = $SMTPAuth;
71     $mail->Username = $emailuser;
72     $mail->Password = $emailpassword;
73     //$mail->addReplyTo("pispidikisj@yahoo.gr", "john");
74     $mail->From = $emailuser;
75     $mail->FromName = $FromName ;
76     $mail->addAddress($emailuser, $FromName );
77     $mail->Subject = $subject;
78     //$mail->Subject =

```



```

79 $body = "<font size='4'><b>ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ</b></font><br>ΟΝΟΜΑ: ".$firstname."<br>".ΕΠΙΘΕΤΟ: ".$lastname."<br>".EMAIL ADDRESS: ".$emailaddress."</table><br><br><font size='4'><b>ΠΕΡΙΕΧΟΜΕΝΟ ΘΕΜΑΤΟΣ</b></font><br>".$questiondata;
80 ///////////////////////////////////////////////////
81 $mail1->Host = $host;
82 $mail1->Port = $port;
83 $mail1->SMTPSecure = $SMTPSecure;
84 $mail1->SMTPAuth = $SMTPAuth;
85 $mail1->Username = $emailuser;
86 $mail1->Password = $emailpassword;
87 // $mail->addReplyTo("pispidikisj@yahoo.gr", "john");
88 $mail1->From = $emailuser;
89 $mail1->FromName = "Διεύθυνση";
90
91 $mail1->addAddress($emailaddress, "Διεύθυνση");
92 $mail1->Subject = "Απάντηση σε ".$subject;
93 // $mail->Subject =
94 $body1 = "Ευχαριστούμε που επικοινωνήσατε!!! Θα επικοινωνήσουμε σύντομα μαζί σας!<br><br>Με εκτίμηση,<br>Η Διεύθυνση";
95
96
97
98 //good
99 //EOT;
100 $mail->WordWrap = 80;
101 $mail->msgHTML($body, dirname(__FILE__), true); //Create message bodies and embed images
102
103
104 try {
105     $mail->send();
106     // $results_messages[] = "Message has been sent using SMTP";
107     echo "{success: true,message:'Το μήνυμα στάλθηκε...'}";
108 }
109 catch (phpmailerException $e) {
110     throw new phpmailerAppException("Unable to send to: ' . $to. ': ' . $e->getMessage());
111 }
112
113 ///////////////////////////////////////////////////
114
115 $mail1->WordWrap = 80;
116 $mail1->msgHTML($body1, dirname(__FILE__), true); //Create message bodies and embed images
117
118
119 try {
120     $mail1->send();
121 }
122 }
123 catch (phpmailerException $e) {
124     throw new phpmailerAppException("Unable to send to: ' . $to. ': ' . $e->getMessage());
125 }
126 }
127 catch (phpmailerAppException $e) {
128     $results_messages[] = $e->errorMessage();
129 }
130 } //telos email
131
132 if (isset($_GET['edit']))
133 {
134     require_once("../connect/connect.php");
135     $dbconn3 = pg_connect("host=$Server port=$Port dbname=$Database user=$Username password=$Password") or die('connection failed');
136
137     ///////////////////////////////////////////////////metafora metabliton
138
139     $nameclient= isset($_POST['first']) ? $_POST['first']: "null";
140     $surname=isset($_POST['last']) ? $_POST['last']: "null";
141
142     $afm= isset($_POST['afm']) ? $_POST['afm']: "null";
143     $identity=isset($_POST['identity']) ? $_POST['identity']: "null";
144
145     $fathername=isset($_POST['fatherfirst']) ? $_POST['fatherfirst']: "";
146     $homenumber=isset($_POST['homephone']) ? $_POST['homephone']: 0;
147     if(!$homenumber)
148     {
149         $homenumber=0;
150     }
151     $mobilenumber=isset($_POST['mobilephone']) ? $_POST['mobilephone']: 0;
152     if(!$mobilenumber)
153     {
154         $mobilenumber=0;
155     }

```

```

156     $email=isset($_POST['email']) ? $_POST['email']: "";
157     $typemap=isset($_POST['area']) ? $_POST['area']: "null";
158     $currentmap=isset($_POST['RasterNamespace']) ? $_POST['RasterNamespace']: "null";
159
160     $numberofland=isset($_POST['numberarea']) ? $_POST['numberarea']: 0;
161     $areachoice=isset($_POST['areachoice']) ? $_POST['areachoice']: "";
162     $pointchoice=isset($_POST['pointchoice']) ? $_POST['pointchoice']: "";
163     $situation='false';
164     $useredit="";
165
166
167     $sql="INSERT INTO \"public\".\"Applications\"(\"nameclient\",\"surname\",\"fathername\",\"homenumber\",\"mobilenumber\",\"email\",\"typemap\",\"current
map\",\"numberofland\",\"areachoice\",\"pointchoice\",\"situation\",\"useredit\",\"afm\",\"adt\") VALUES (\". $nameclient.\",\". $surname.\",\". $fathername.\",\".
\". $homenumber.\",\". $mobilenumber.\",\". $email.\",\". $typemap.\",\". $currentmap.\",\". $numberofland.\",\". $areachoice.\",\". $pointchoice.\",\". $
situation.\",\". $useredit.\",\". $afm.\",\". $identity.\");
168     $result=pg_exec($dbconn3, $sql)or die(pg_errormessage());
169
170
171
172     echo "{success: true,message:'Οι αιτησεις σας στάλθηκε με επιτυχία...'}";
173
174
175
176
177
178 }
179
180
181
182 ?>

```

```

1  var app;
2  var markers=null;
3
4  Ext.onReady(function () {
5
6
7
8  var osm = new OpenLayers.Layer.OSM("OSM");
9
10
11
12  app = new gxp.Viewer({
13    //proxy: "/geoserver/rest/proxy?url=",
14    xtype: "panel" ,
15    portalConfig : {
16      renderTo : "tab" ,
17      layout : "border" ,
18      id:'viewer' ,
19      // width: this.width,
20      height :570,
21
22      //hidden:true,
23      //disabled:true,
24
25      // by configuring items here, we don't need to configure portalItems
26      // and save a wrapping container
27      items : [{
28        // a TabPanel with the map and a dummy tab
29        id: "centerpanel" ,
30        xtype: "tabpanel" ,
31        margins : '5 5 0 0',
32        region: "center" ,
33        activeTab : 0, // map needs to be visible on initialization
34        border : true,
35        items: ["mymap"]
36        //tbar:[]
37      }, {
38        region : 'east' ,
39        xtype : 'panel' ,
40        id : 'east' ,
41        collapsible : true,
42        autoScroll: true,
43        collapsed: true,
44        margins: '5 5 0 0',
45        //split: true,
46        //expanded:true,
47        layout: 'fit',
48
49        width: 200
50
51      }, {
52        // container for the queryform
53        id: "west" ,
54        xtype: "panel" ,
55        margins : '5 5 0 0',
56        collapsible : true,
57        autoScroll: true,
58        collapsed: true,
59        //split: true,
60        //expanded:false,
61
62
63        layout: 'fit',
64        // layoutConfig: { animate: true},
65        //autoScroll: true,
66        region: "west",
67        width: 200
68      },
69    ],
70    bbar: {id: "mybbar"}
71  },//portalconfig
72
73  // configuration of all tool plugins for this application
74  tools: [{
75
76    ptype:'gxp_legend' ,
77    outputTarget : "east"
78  },

```



```

79
80 {
81   ptype : "gxp_loadingindicator"
82 }, {
83   ptype : "gxp_layertree" ,
84   outputConfig : {
85     id : "tree" ,
86     border : true,
87     tbar : [] // we will add buttons to "tree.bbar" later
88   },
89   outputTarget: "west"
90 }, {
91   ptype : "gxp_addlayers" ,
92   actionTarget : "tree.tbar"
93 }, {
94   ptype : "gxp_removelayer" ,
95   actionTarget : ["tree.tbar", "tree.contextMenu"]
96 }, {
97   ptype : "gxp_layerproperties" ,
98   actionTarget : [ "tree.contextMenu"]
99 },
100
101
102 {
103   ptype : "gxp_zoomtoextent" ,
104   actionTarget : "map.tbar"
105 },
106 {
107   ptype : "gxp_wmsgetfeatureinfo" ,
108   infoFormat : "application/vnd.ogc.gml" ,
109   outputConfig : {
110
111     width : 400,
112     height : 200
113   },
114   actionTarget: "map.tbar", // this is the default, could be omitted
115   toggleGroup: "layertools"
116 },
117 //////////////////////////////////////////////////
118 {
119   ptype : "gxp_measure" ,
120   actionTarget : "map.tbar"
121 },
122 {
123   // shared FeatureManager for feature editing, grid and querying
124   ptype : "gxp_featuremanager" ,
125   id : "featuremanager" ,
126   maxFeatures : 20
127 }, {
128   ptype : "gxp_wmsrasterstylesdialog" ,
129   featureManager : "featuremanager" ,
130   actionTarget : ["map.tbar"]
131 },
132 {
133   ptype : "gxp_zoomtolayerextent" ,
134   featureManager : "featuremanager" ,
135   appendActions : true,
136   actionTarget : ["tree.contextMenu"]
137 },
138 //////////////////////////////////////////////////
139 {
140   ptype : "gxp_queryform" ,
141   featureManager : "featuremanager" ,
142   outputConfig : {
143     title : "Query" ,
144     width : 320
145   },
146   actionTarget : [ "tree.contextMenu"],
147   appendActions: false
148 }
149 //,
150 //{
151   // not a useful tool - just a demo for additional items
152   // actionTarget: "mybbar", // ".bbar" would also work
153   // actions: [{text: "Click me - I'm a tool on the portal's bbar"}]
154 //}
155 ],
156

```

```

157 // layer sources
158 defaultSourceType: "gxp_wmssource",
159 sources: { ol: {
160     ptype: "gxp_olsource"
161 },
162
163     "osm": {
164 ptype: "gxp_osmsource"
165 },
166     local: {
167         url: "/geoserver/wms/" ,
168         version: "1.1.1"
169     },
170     "bing": {
171 ptype: "gxp_bingsource"
172 }
173
174
175 },
176
177 // map and layers
178 map: {
179     id: "mymap" , // id needed to reference map in portalConfig above
180     title: "Βοηθητικός Χάρτης" ,
181     projection: "EPSG:900913" ,
182     units: "m" ,
183     tbar:[
184     {
185 xtype: 'button' ,
186     iconCls: 'box' ,
187     tooltip: "Επιλογή περιοχής ενδιαφέροντος" ,
188     enableToggle: true,
189     id: 'boxarea',
190     disabled: true,
191     toggleHandler: function(btn,pressed)
192     {
193
194 if (btn.pressed)
195     {
196         //var homenumber=Ext.getCmp('home');
197         //homenumber.setValue("playeeeeee");
198
199         boxControl = new OpenLayers.Control();
200 OpenLayers.Util.extend(boxControl, {
201 draw: function() {
202     this.box = new OpenLayers.Handler.RegularPolygon(boxControl,
203 {"done": this.notice}, {sides:4, irregular:true, persist:true});
204 boxControl.box.activate();
205 },
206 notice: function(geom) {
207 alert(geom);
208 var geoms=geom.transform(new OpenLayers.Projection("EPSG:900913"),new OpenLayers.Projection("EPSG:4121"));
209 alert(geoms);
210 var areachoice=Ext.getCmp('areachoice');
211 areachoice.setValue(geom);
212
213
214 }
215 });
216 var maps=Ext.getCmp('mymap');
217 maps.map.addControl( boxControl);
218
219
220
221
222     }
223     else
224     {
225         boxControl.box.deactivate ();
226         var boxarea=Ext.getCmp('boxarea');
227 boxarea.setDisabled(true);
228     }
229 }
230 },{
231
232     xtype: 'button' ,
233     iconCls: 'point' ,
234     tooltip: "Επιλογή σημείου εντος αγροτεμαχίου" ,

```

```

235     enableToggle : true,
236     id:'point',
237     disabled:true,
238     toggleHandler:function(btn,pressed)
239     { var maps=Ext.getCmp('mymap');
240
241
242     if (btn.pressed)
243     {
244
245
246         maps.map.events.register ("click",maps.map,map_GetCoord);
247         var pointchoice=Ext.getCmp('pointchoice');
248
249
250     }
251     else
252     {
253         maps.map.events.unregister ("click",maps.map,map_GetCoord);
254         markers.destroy();
255
256         var point=Ext.getCmp('point');
257         point.setDisabled(true);
258
259
260     }
261
262 }
263
264
265 },->",{
266 xtype:'button',
267 iconCls:'controls_map_zoombyarea',
268 tooltip:"map Zoom By Area",
269 enableToggle: true,
270 toggleHandler:function(btn,pressed)
271 {
272     if (btn.pressed)
273     {
274
275         map_Zoom_ByArea_var =new OpenLayers.Control.ZoomBox();
276
277         Ext.getCmp('mymap').map.addControl(map_Zoom_ByArea_var);
278
279         map_Zoom_ByArea_var.activate();
280
281     }
282     else
283     {
284         map_Zoom_ByArea_var.deactivate ();
285     }
286 }
287 }
288
289 }],
290
291
292
293 // maxExtent: [boundary[0], boundary[1], boundary[2], boundary[3]],
294 // maxExtent: Ext.getCmp('mymap').map.zoomToMaxExtent(),
295 restrictedExtent: [boundary[0], boundary[1], boundary[2], boundary[3]],
296 // maxResolution: 2.5,
297 //numZoomLevels: 20,
298 center: [(boundary[0]+boundary[2])/2,(boundary[1]+boundary[3])/2],
299 zoom: 6,
300
301 controls: [
302     new OpenLayers.Control.Zoom(),
303     new OpenLayers.Control.Attribution(),
304     new OpenLayers.Control.Navigation()
305 ],
306 layers: [{
307     source : "ol" ,
308     type : "OpenLayers.Layer" ,
309     type : "OpenLayers.Layer.OSM" ,
310 args: ["OpenStreetMap"],
311 // args: ["Blank"],
312 visibility: false,

```

```

313         group: "background"
314     },
315     {
316         source: "osm" ,
317         name: "osmarander" ,
318         visibility : true
319     },
320
321
322     {
323         source: "bing",
324         title: "Bing Map Satellite",
325         name: "Aerial"
326     }
327 }
328     ]
329 } //map
330
331 });
332
333
334
335 //Ext.getCmp('mymap').map.addLayers([kthmatologio]);
336
337
338 });
339
340 //////////////////////////////////////////////////function
341 function map_GetCoord(evt)
342 {
343     var maps=Ext.getCmp('mymap');
344     var coord=maps.map.getLonLatFromViewPortPx(evt.xy).transform(new OpenLayers.Projection("EPSG:900913"),new OpenLayers.Projection("EPSG:4121"));
345     var x=coord.lon.toFixed(2);
346     var y=coord.lat.toFixed(2);
347
348     var size = new OpenLayers.Size(21,25);
349
350
351     markers = new OpenLayers.Layer.Markers("Σημείο Επιλογής");
352     maps.map.addLayer(markers);
353
354     maps.map.addLayer(markers);
355     var offset = new OpenLayers.Pixel(-(size.w/2), -size.h);
356     var icon = new OpenLayers.Icon('images/point1.gif', size, offset);
357
358     var lonlat = new OpenLayers.LonLat(x,y).transform(new OpenLayers.Projection("EPSG:4121"), new OpenLayers.Projection("EPSG:900913"));
359
360     markers.addMarker(new OpenLayers.Marker(lonlat,icon));
361
362     // alert(x+","+y);
363     var pointchoice=Ext.getCmp('pointchoice');
364     pointchoice.setValue(x+","+y);
365
366     var bounds = markers.getDataExtent();
367     maps.map.zoomToExtent(bounds);
368 }
369
370

```

ΕΦΑΡΜΟΓΗ ΓΙΑ ΕΓΚΑΤΑΣΤΑΣΗ ΣΥΣΤΗΜΑΤΟΣ

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>installation</title>
6 </head>
7 <?php
8
9 header('Location: pages/page1.php');
10
11 ?>
12 <body>
13 </body>
14 </html>
```



```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Διαδικασία εγκατάστασης</title>
5   <?php
6
7     $geo_url="";
8   ?>
9     <meta charset="utf-8">
10    <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">
11    <link rel="icon" href="../images/page.png" type="image/x-icon">
12    <link rel="stylesheet" href="../lib/css/demo.css">
13    <link rel="stylesheet" href="../lib/css/sky-forms.css">
14    <link rel="stylesheet" href="../lib/css/sky-forms-black.css">
15
16  </head>
17  <body class="bg-black">
18    <div class="body">
19
20      <!-- Red color scheme -->
21      <form class="sky-form">
22        <header> Στοιχεία Βάσης Δεδομένων PostGIS</header>
23
24        <fieldset>
25        <section>
26          <p>
27            <label class="label">SERVER (Για τοπικό server: localhost)</label>
28            <label class="input">
29              <input type="text" id="server">
30            </label>
31          </p>
32          <p>&nbsp;</p>
33        </section>
34
35
36
37        <section></section>
38      </fieldset>
39
40      <fieldset>
41        <p>
42          <section></section>
43
44          <section></section>
45          <label class="label">PORT (π.χ: 5432) </label>
46          <label class="input">
47            <input type="text" id="port">
48
49          </label>
50        </p>
51        <p>&nbsp;</p>
52      </fieldset>
53
54      <fieldset>
55        <p>
56          <section></section>
57
58          <section></section>
59
60          <section></section>
61          <label class="label">DATABASE (Όνομα Βάσης Δεδομένων)</label>
62          <label class="input">
63            <input type="text" id="database">
64          </label>
65        </p>
66        <p>&nbsp;</p>
67      </fieldset>
68
69      <fieldset>
70        <label class="label">USERNAME(π.χ postgres)</label>
71        <label class="input">
72          <input type="text" id="username">
73        </label>
74      </p>
75      <p>&nbsp;</p>
76    </fieldset>
77
78  </div>

```



```
79
80 <label class="label">password</label>
81 <label class="input">
82     <input type="password" id="password">
83 </label>
84 </p>
85     <p>&nbsp;</p>
86
87
88 </fieldset>
89 <footer><div id="result"></div>
90 <button type="button" class="button" style="visibility:hidden" id="next">Επομενο</button>
91 <button type="button" class="button button-secondary" id="checkDB" ;>Ελεγχος</button>
92 </footer>
93 </form>
94 <!--/ Red color scheme -->
95 </div>
96 <script type="text/javascript" src="../../lib/jquery-1.11.0.min.js"></script>
97 <script type="text/javascript" src="../../lib/check.js"></script>
98 </body>
99 </html>
```

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Διαδικασία εγκατάστασης</title>
5
6     <meta charset="utf-8">
7     <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">
8     <link rel="icon" href="../images/page.png" type="image/x-icon">
9     <link rel="stylesheet" href="../lib/css/demo.css">
10    <link rel="stylesheet" href="../lib/css/sky-forms.css">
11    <link rel="stylesheet" href="../lib/css/sky-forms-black.css">
12
13  </head>
14  <body class="bg-black">
15    <div class="body">
16
17      <!-- Red color scheme -->
18      <form class="sky-form">
19        <header> Στοιχεία Κύριου Διαχειριστή</header>
20
21        <fieldset>
22          <section>
23            <label class="label">Όνομα Διαχειριστή</label>
24            <input class="input">
25              <input type="text" id="Name">
26            </label>
27          </section>
28
29          <section>
30            <label for="file" class="input input-file">
31              </label>
32            </section>
33
34            <section></section>
35          </fieldset>
36
37          <fieldset>
38            <p>
39              <section></section>
40
41              <section></section>
42              <label class="label">Επώνυμο Διαχειριστή</label>
43              <label class="input">
44                <input type="text" id="Surname">
45              </label>
46            </p>
47            <p>&nbsp;</p>
48          </fieldset>
49
50          <fieldset>
51            <p>
52              <section></section>
53
54              <section></section>
55
56              <section></section>
57              <label class="label">UserName</label>
58              <label class="input">
59                <input type="text" id="Username2">
60              </label>
61            </p>
62            <p>&nbsp;</p>
63          </fieldset>
64
65          <fieldset>
66            <label class="label">Password</label>
67            <label class="input">
68              <input type="password" id="Password2">
69            </label>
70          </p>
71          <p>&nbsp;</p>
72        </fieldset>
73
74        <fieldset>
75
76          <label class="label">Retype Password </label>
77          <label class="input">
78            <input type="password" id="RPassword2">

```

```
79         </label>
80     </p>
81     <p>&nbsp;</p>
82
83
84     </fieldset>
85     <footer><div id="result2"></div>
86
87     <button type="button" class="button" id="Next1" style="visibility:visible">Next</button>
88     <button type="button" class="button button-secondary" onclick="window.history.back();" id="back">Πίσω</button>
89     </footer>
90 </form>
91 <!--/ Red color scheme -->
92 </div>
93 <script type="text/javascript" src="../../lib/jquery-1.11.0.min.js"></script>
94 <script type="text/javascript" src="../../lib/check.js"></script>
95 </body>
96 </html>
```

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Διαδηκασία εγκατασταση</title>
5
6     <meta charset="utf-8">
7     <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">
8     <link rel="icon" href=" ../images/page.png" type="image/x-icon">
9     <link rel="stylesheet" href=" ../lib/css/demo.css">
10    <link rel="stylesheet" href=" ../lib/css/sky-forms.css">
11    <link rel="stylesheet" href=" ../lib/css/sky-forms-black.css">
12
13  </head>
14  <body class="bg-black">
15    <div class="body">
16
17      <!-- Red color scheme -->
18      <form class="sky-form">
19        <header> Στοιχεία Εξυπηρετητή Χαρτών GeoServer</header>
20
21        <fieldset>
22          <section>
23            <label class="label"><strong>Προκειμένου να είναι προσβάσιμος ο Geoserver χρειάζονται τα παρακάτω βήματα:</strong><br>
24            <br>
25            1) Άνοιγμα του httpd.conf του εξυπηρετητή διαδικτύου apache          <br>
26            2) Αντιγραφή στο τέλος του αρχείου:</label>
27          </section>
28
29          <section></section>
30
31          <section></section>
32        </fieldset>
33        <fieldset>
34          <p>
35            <section></section>
36
37            <section></section>
38
39            <section></section>
40            <span class="label">&lt;VirtualHost *:80&gt;<br>
41 ProxyRequests Off<br>
42 ProxyPreserveHost On<br>
43 &lt;Proxy *&gt;<br>
44 Order deny,allow<br>
45 Allow from all<br>
46 &lt;/Proxy&gt;<br>
47 ProxyPass /geoserver http://localhost:8085/geoserver<br>
48 ProxyPassReverse /geoserver http://localhost:8085/geoserver<br>
49 &lt;/VirtualHost&gt; </span></p>
50 </fieldset>
51
52        <fieldset>
53          <label class="label"><strong>ΕΠΕΞΗΓΗΣΗ</strong><br>
54            1) http://localhost:8085/geoserver : url εκτέλεσης του GeoServer<br>
55            2)&lt;VirtualHost *:80&gt;: (80) η θύρα εκτέλεσης του δικού μας server<br>
56          </label>
57        </fieldset>
58        <footer><div id="result2"></div>
59
60        <button type="button" class="button" id="return3" style="visibility:visible" >Επιστροφή Στη Διαχείριση</button>
61      </footer>
62    </form>
63    <!--/ Red color scheme -->
64  </div>
65  <script type="text/javascript" src=" ../lib/jquery-1.11.0.min.js"></script>
66  <script type="text/javascript" src=" ../lib/check.js"></script>
67 </body>
68 </html>

```



```

1  ///sinartisi gia elexo dedomenon gia thn page1
2
3  $('#checkDB').click(function (){
4
5      var server=$('#server').val();
6      var port=$('#port').val();
7      var database=$('#database').val();
8      var username=$('#username').val();
9      var password=$('#password').val();
10
11     if(server=="||port=="||database=="||username=="||password=="")
12     {
13
14         document.getElementById ('result').innerHTML="<font color=#FF9900>ΠΡΟΣΟΧΗ!!! Δεν συμπληρώθηκαν όλα τα στοιχεία</font>";
15
16     }
17     else
18     {
19         $.post ('../db/dbcheck.php?link=1',{server:server,port:port,database:database,username:username,password:password},function(data){
20             if(data!="connection failed")
21             {
22                 if(data!="")
23                 {
24                     document.getElementById ('next').style.visibility="visible";
25                 }
26             }
27
28             document.getElementById ('result').innerHTML=data;
29
30         }
31
32     }
33
34 })
35
36
37 }
38
39 });
40
41
42 ///sinartisi gia elexo dedomenon gia thn page2
43
44 $('#next').click(function (){
45
46     location.href = "page2.php" ;
47 });
48
49
50 ///page2
51 //1)check gia kena pedia
52 $('#Next1').click(function (){
53
54     var Name=$('#Name').val();
55     var Surname=$('#Surname').val();
56     var Username2=$('#Username2').val();
57     var Password2=$('#Password2').val();
58     var RPassword2=$('#RPassword2').val();
59
60     if(Name=="||Surname=="||Username2=="||Password2=="||RPassword2=="")
61     {
62
63         document.getElementById ('result2').innerHTML="<font color=#FF9900>ΠΡΟΣΟΧΗ!!! Δεν συμπληρώθηκαν όλα τα στοιχεία</font>";
64
65     }
66     else
67     {
68         if(Password2!=RPassword2)
69         {
70             document.getElementById ('result2').innerHTML="<font color='red'>Password mismatch</font>";
71
72         }
73         else
74         {
75             $.post ('../db/dbcheck.php?link=2',{Name:Name,Surname:Surname,Username2:Username2>Password2:Password2},function(data)
76             {
77                 //document.getElementById('result2').innerHTML=data;
78                 if(data!="Επιτυχής καταχώρηση")

```

```

78     {
79         document.getElementById ('result2').innerHTML=data;
80
81     }
82     else
83     {
84         alert (data);
85         location.href = "page3.php";
86     }
87 } //telos post
88
89 } //telos else
90 }
91
92 // location.href = "page2.php";
93 });
94
95
96
97 $('#RPassword2').on('keyup',function(){
98
99
100
101 var Password2=$('#Password2').val();
102 var RPassword2=$('#RPassword2').val();
103 if(Password2!=RPassword2)
104 {
105     document.getElementById ('result2').innerHTML="<font color='#FF9900'>Password mismatch</font>";
106
107 }
108 if(Password2==RPassword2)
109 {
110     document.getElementById ('result2').innerHTML="<font color='green'>Password OK</font>";
111
112 }
113
114 });
115
116
117 ///page 3
118 $('#return3').click(function (){
119     alert (" Η εγκατάσταση ολοκληρώθηκε!\nΠροσοχή! Σβήστε το φάκελο install");
120
121     location.href = "../../admin/index.php";
122 });

```

```

1  <?php
2
3  if (isset($_GET['link']))
4  {
5      if($_GET['link']==1)//gia tin proti selida
6      {
7
8          $server=$_POST['server'];
9          $port=$_POST['port'];
10         $database=$_POST['database'];
11         $username=$_POST['username'];
12         $password=$_POST['password'];
13         $dbconn3 = pg_connect ("host=$server port=$port dbname=$database user=$username password=$password" )or die('connection fail
ed');
14         if($dbconn3 )
15         {
16             $string = "<font color='green'>Επιτυχής σύνδεση</font>" ;
17             //echo $string;
18
19             $fp=fopen ('../../connect/connect.php' , 'w');
20             $data="<?php \n " ;
21             $data=$data."$ " .Server=" " . $server . " , \n " . "$ " .Port=" " . $port . " , \n " . "$ " .Database=" " . $database . " , \n " . "$ " .Username=" " . $username . " , \
n " . "$ " .Password=" " . $password . " ; \n ?> " ;
22
23             fwrite ($fp, $data);
24             fclose ($fp);
25
26             ///////////////////////////////////////////////////
27             //dimiourgia pinakvn apo arxeio////
28             //1)anoigma arxeiou
29             $myFile = "sql.txt" ;
30             $fh = fopen ($myFile , 'r');
31             $theData = fread ($fh, filesize ($myFile ));
32             fclose ($fh);
33             //dimiourgia sql
34             $sql=$theData;
35             //echo $theData;
36             $result=pg_exec ($dbconn3 , $sql) or die(pg_errormessage ());
37             if($result )
38             {
39                 $string=$string ." <font color='green'>και καταχώρηση πινάκων</font>" ;
40
41             }
42         }
43         echo $string ;
44     }
45
46 } //2 if
47 if($_GET['link']==2)//gia tin deuteri selida
48 {
49     ///////////////
50     $Name=$_POST['Name'];
51     $Surname=$_POST['Surname'];
52     $Username2=$_POST['Username2'];//ta bazw 2 gia na min mplekei me connect
53     $Password2=$_POST['Password2'];
54
55
56
57     ///////////////
58
59
60     require_once ('../../connect/connect.php' );//pernei stoixeia sindesis
61     $dbconn3 = pg_connect ("host=$Server port=$Port dbname=$Database user=$Username password=$Password" )or die('connecti
on failed' );
62
63     $sql='INSERT INTO "public"."Admin"("Name","SurName","UserName","PassWord") VALUES (\'' . $Name . '\',\' ' . $Surname . '\',\' ' . $Usern
ame2 . '\',\' ' . $Password2 . '\')';
64
65     $result=pg_exec ($dbconn3 , $sql) or die(pg_errormessage ());
66
67     if($result )
68     {
69         $string="Επιτυχής καταχώρηση" ;
70         echo $string ;
71     }
72 }
73
74 } //3 if

```



```
75  
76 }//proti if  
77 ?>
```