

**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ  
ΠΟΛΥΤΕΧΝΕΙΟ**



**NATIONAL TECHNICAL  
UNIVERSITY OF ATHENS**

Σχολή Αγρονόμων & Τοπογράφων Μηχανικών  
**ΓΕΩΠΛΗΡΟΦΟΡΙΚΗ**  
ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ

School of Rural & Surveying Engineering  
**GEOINFORMATICS**  
POST-GRADUATE PROGRAMME

## **Αξιοποίηση Γεωγραφικών Μεταδεδομένων στη Διαχείριση της Αγοράς Ακινήτων (Real Estate)**

**ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

του

**ΠΑΙΓΝΙΓΙΑΝΝΗ ΜΙΧΑΛΗ**

**Επιβλέπων :** Τιμολέων Σελλής  
Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2008



## **Αξιοποίηση Γεωγραφικών Μεταδεδομένων στη Διαχείριση της Αγοράς Ακινήτων (Real Estate)**

### **ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

του

**ΠΑΙΓΝΙΓΙΑΝΝΗ ΜΙΧΑΛΗ**

**Επιβλέπων :** Τιμολέων Σελλής  
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 3η Νοέμβριου 2008.

.....  
Τιμολέων Σελλής  
Καθηγητής Ε.Μ.Π.

.....  
Μαρίνος Κάβουρας  
Καθηγητής Ε.Μ.Π.

.....  
Λύσανδρος Τσούλος  
Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2008

.....  
**ΜΙΧΑΛΗΣ ΠΑΙΓΝΙΓΙΑΝΝΗΣ**

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π, 2001  
MRes Mobile & Satellite Communications, University of Surrey, 2002,

Copyright © Μιχάλης Παϊνιγιάννης, 2008

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

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

## Πρόλογος

Μετά από δύο σχεδόν χρόνια κατόρθωσα να ολοκληρώσω με τη συγκεκριμένη μεταπτυχιακή εργασία το διατμηματικό μεταπτυχιακό πρόγραμμα του ΕΜΠ της Γεωπληροφορικής. Το κίνητρό μου ήταν το βαθύτερο ενδιαφέρον, για τη γεωγραφικά κατανομημένη πληροφορία, που ανέπτυξα κατά τις προπτυχιακές μου σπουδές, αλλά και κατά την εργασιακή μου απασχόληση με δεδομένα, που είχαν χωρική υπόσταση.

Το βασικό ενδιαφέρον μου για τις Τηλεπικοινωνίες δεν μου είχε καλύψει στο βαθμό που επιθυμούσα την ανάγκη για βαθύτερη ενασχόληση με τα GIS. Τα GIS έχουν αποτελέσει ένα σχεδόν αχώριστο προσωπικό συνεργάτη σε κάθε στροφή της καριέρας μου. Ξεκινώντας από την LDK ECO εν έτει 2003 με το αυτοσχέδιο λογισμικό δρομολόγησης απορριμματοφόρων για τον Καπποδιστριακό Δήμο Κερκυραίων, περνώντας στο κομμάτι της ραδιοκάλυψης δικτύου GSM, LMDS και στις μελέτες / μετρήσεις ιδιοπαρεμβολών και διαχείρισης φάσματος στην QTELECOM (2004-2006), και εν συνεχεία στη δυναμική απεικόνιση σε μιμικό διάγραμμα των επιπέδων συμφόρησης όλων των κόμβων της QTELECOM (κυκλωμάτων κίνησης E1s όσο και κυκλωμάτων σηματοδοσίας SS7) για την «οπτική» διευκόλυνση της διαδικασίας μετάβασης της εθνικής περιαγωγής από το δίκτυο της Vodafone σε αυτό της WIND (2007) και σήμερα πλέον στην υποστήριξη των υπολογιστικών αναγκών - λειτουργιών του Οργανισμού Αστικών Συγκοινωνιών Αττικής (2008), το ενδιαφέρον για τα GIS είναι συνεχές και αποτέλεσε την αιτία για την οποία μπήκα στη διαδικασία να αφιερώσω όλον τον ελεύθερό μου και τις κανονικές μου άδειες των 2 τελευταίων ετών για να παρακολουθήσω το εν λόγω μεταπτυχιακό πρόγραμμα.

Θα ήθελα να ευχαριστήσω τους δικούς μου (Νίκο και Ιωάννα) γιατί με στήριξαν όταν εξαντλημένος από την προσπάθεια ήμουν έτοιμος να τα παρατήσω. Να ευχαριστήσω όλους τους καθηγητές αλλά και τη γραμματέα κα Παλιάτσου για το μοναδικό χαμόγελο και ευγένειά της. Να ευχαριστήσω τον κο Σελλή για την τιμή, που μου έκανε να εκπονήσω ένα θέμα μεταπτυχιακή εργασίας, που ονειρευόμουν εδώ και χρόνια. Οι όποιες ατέλειες στην εργασία μου βαρύνουν αποκλειστικά εμένα. Ευχαριστώ για τη βοήθειά του, τον παλιό συμφοιτητή Διδ. Σπ. Αθανασίου που με καθοδήγησε στον «θαυμαστό» κόσμο των web2.0 εφαρμογών. Τέλος να εκφράσω ένα ευχαριστώ στην Δ. Θ. Καμύρη για το De Profundis του O.Wilde.



## Περίληψη

Η παρούσα μεταπτυχιακή εργασία έχει ως στόχο τη διερεύνηση της ενσωμάτωσης γεωγραφικών μεταδεδομένων σε διαδικασίες ανεύρεσης ακινήτων προς πώληση σε συστήματα Κτηματομεσιτικών υπηρεσιών (on-line Real Estate) στο Διαδίκτυο. Μέρος της εργασίας αποτέλεσε η υλοποίηση μιας πρότυπης διαδικτυακής εφαρμογής για την απόδειξη της εφικτότητας και της ωφέλειας μιας τέτοιας προσέγγισης.

Βασικό κίνητρο για τη συγκεκριμένη ιδέα-εφαρμογή αποτέλεσε η προσωπική ταλαιπωρία κατά τη φάση αναζήτησης ακινήτου προς ενοικίαση / πώληση. Δεδομένου ότι παραπλήσιες λύσεις έχουν αναπτυχθεί παγκοσμίως, η ιδέα δεν διαγκωνίζεται τόσο για την πρωτοτυπία της όσο για την ευαισθητοποίηση της ελληνικής αγοράς. Η ιδέα της δωρεάν ή με πολύ μικρό κόστος παροχής πληροφόρησης για τα υποψήφια προϊόντα, χωρίς την υποχρέωση για καταβολή υπέρογκων ποσοστών επί του τιμήματος είναι πράξη άκρως «δημοκρατική» στη λογική παροχής ευρυζωνικών υπηρεσιών σε ολόκληρο τον πληθυσμό. Δεδομένου ότι η οικοδομή αποτελεί τον βασικό πυλώνα ανάπτυξης της εθνικής μας οικονομίας (ευτυχώς ή δυστυχώς), η διάχυση της πληροφορίας θα επιταχύνει την ανάπτυξη μειώνοντας το κόστος αγοράς κατοικίας (απαλείφεται η ανάγκη των μεσαζόντων – μεσιτών). Τουλάχιστον για τα μικρότερα ακίνητα οι πολίτες θα επιλέγουν το ακίνητο που πραγματικά ικανοποιεί τις ανάγκες τους στην καλύτερη δυνατή τιμή. Επιπλέον, ο ίδιος ο κλάδος ακινήτων εξελίσσεται επιτρέποντας την εξειδίκευση των μεσιτών και την ενασχόληση μόνο όσων έχουν την απαιτούμενη επαγγελματική κατάρτιση.

Αρχικά γίνεται σύντομη αναφορά σε θέματα της αγοράς ακινήτων, που τεκμηριώνουν με βάση την κλασική οικονομική θεωρία την ανάγκη για ενημέρωση των καταναλωτών και τον προσδιορισμό της αγοραίας αξίας. Θέματα σχετικά με GIS αναπτύσσονται πάντα έχοντας κατά νου την αγορά ακινήτων.

Στη συνέχεια περιγράφονται οι ιδέες, που αναπτύσσονται από την παγκόσμια κοινότητα λογισμικού σχετικά με το WEB2.0 και την ανάπτυξη εφαρμογών προσβάσιμων από οπουδήποτε χωρίς να υπάρχουν “SILO” και proprietary βάσεις ή διεπαφές (SOAP, AJAX κλπ). Στο ίδιο μήκος κύματος κινείται και η Ευρωπαϊκή κοινότητα του GIS με το πρόγραμμα INSPIRE. Η περιγραφή των βασικότερων εργαλείων, που χρησιμοποιήθηκαν για την ανάπτυξη του λογισμικού θα αναφερθούν εδώ. Γενικότερη αναφορά θα γίνει στη μεγάλη διάδοση των Google Maps API και σε παρεμφερή APIs απεικόνισης γεωγραφικών πληροφοριών σε χάρτες.

Στην Τρίτη ενότητα θα γίνει μια ανασκόπηση των υπάρχουσών λύσεων, που υπάρχουν μέχρι και σήμερα στον παγκόσμιο ιστό και αφορούν τη συγκεκριμένη ιδέα. Η ιδέα είναι σχετικά ώριμη σε αγορές της Αμερικής ή και της Αγγλίας. Χαρακτηριστικά παραδείγματα θα αναφερθούν.

Στην τελευταία ενότητα θα περιγραφεί η δομή της πλατφόρμας της λύσης, που ανέπτυξα στα πλαίσια της μεταπτυχιακής εργασίας. Να τονίσω ότι σε καμία περίπτωση η πλατφόρμα δεν επιδιώκει τα σκήπτρα της καλύτερης εφαρμογής online-real estate. Αποτελεί απλά ένα testbed ή αλλιώς η απόδειξη (proof-of-concept) ότι η ανάπτυξη ενός τέτοιου εργαλείου δεν απαιτεί υπέρογκη προσπάθεια. Να σημειώσω σε αυτό το σημείο ότι η συγκεκριμένη ιδέα είχε προταθεί στον Πανελλήνιο Διαγωνισμό Innovation 2006 της OTENET.

**Λέξεις Κλειδιά:** γεωγραφικά μεταδεδομένα, γεωκωδικοποίηση, γεωεπεξεργασία, γεωσυσχέτιση, Google Maps, AJAX, Real Estate, GIS.



## Summary

The present thesis makes an effort to embed geographic metadata in operational functions of online Real Estate GIS applications. Part of the current work was the development of a functional application proving the viability of such an approach. The whole effort was more or less as a proof-of-concept approach rather than a full commercial effort.

Basic motive for the particular idea-application was the personal discomfort during the phase of real estate search for renting/sale. Due to the fact that, similar solutions have been developed worldwide, the idea does not compete so much for her originality as for promoting best practises in the Greek market. The idea of using such an online service for free or for a very small fee without extra charges based on the price of the real estate, is an action extremely "democratic" because the information becomes available to the masses. Since Dwelling Construction constitutes the basic pylon of growth for the greek economy, the availability and diffusion of information will accelerate the growth, decreasing the purchase cost of residence (no need for RE agents) at least for the smaller estates. Individuals will select the estate that will really satisfy their needs in the best possible price. Moreover, the sector of RE will mature through the specialisation of brokers / RE agents, in areas like RE leasing and Investment assessment.

To start with, a short report on issues regarding the market of real estate explains that the need for consumers' ability to easily reach unbiased Real Estate (RE) information stems from the classic economic theory. GIS needs are shortly analysed here.

In the second part of the thesis, we explain in detail the world's-software-community WEB 2.0 ideas and the online web applications that are accessible from anywhere. The development based on vertical and proprietary databases will be opposed to the open (SOAP, AJAX etc) interconnected way of developing web applications. The European community drives the specification processes towards the same direction. These efforts are grouped under the INSPIRE project. The basic tools and architectures that were used during the development of the software testbed will be explained in detail here. This description will enlighten widely used APIs, like Google Maps and similar GIS technologies.

In the Third part, we examine existing solutions in the world web relevant to the RE industry. The idea is relatively mature in markets such as USA or UK. Characteristic examples will be given.

In the last part, we describe the technicalities of the developed testbed in depth. In no case does the application developed seek to be recognised as the best (in terms of functionality and user friendliness or completeness) online application in RE. It simply constitutes a testbed and a proof (proof - of - concept) that developing online applications does not pose to big difficulties.

**Words Keys:** geographic metadata, geocoding, geoprocessing, geo-correlation, Google Maps, AJAX, Real Estate, GIS

## Πίνακας περιεχομένων

<b>1</b>	<b>Εισαγωγή.....</b>	<b>11</b>
1.1	Αντικείμενο της μεταπτυχιακής εργασίας .....	11
1.2	Οργάνωση του τόμου.....	11
<b>2</b>	<b>Τα οικονομικά του Real Estate – Ανάγκη για GIS.....</b>	<b>15</b>
2.1	Ιδιοκτησία στο πέρασμα των αιώνων .....	15
2.1.1.1	Γενικά.....	15
2.1.1.2	Ιδιοκτησία και Κοινωνικά Συστήματα .....	16
2.1.1.3	Frédéric Bastiat - property is value .....	17
2.1.1.4	Ιδιοκτησία στο σήμερα – ανάγκη για Πληροφόρηση .....	18
2.2	Ηλεκτρονικές Δημοπρασίες .....	20
2.2.1.1	Εισαγωγή.....	20
2.2.1.2	Ηλεκτρονικές Δημοπρασίες Στα Ακίνητα .....	21
<b>3</b>	<b>Τεχνολογίες WEB2.0 .....</b>	<b>23</b>
3.1	Εισαγωγή σε Τεχνολογίες WEB -- HTTP and HTML .....	23
3.2	PHP και λοιπές Server-Side Τεχνολογίες .....	25
3.2.1.1	Java Framework .....	25
3.2.1.2	.NET Framework.....	26
3.3	JavaScript και άλλες Client-Side Technologies.....	26
3.4	Φιλικότητα προς το χρήστη .....	27
3.5	Κατανοώντας την AJAX.....	28
3.5.1.1	Τεχνικές λεπτομέρειες της AJAX .....	29
3.5.1.2	Μειονεκτήματα της AJAX.....	31
3.5.1.3	Παράδειγμα AJAX από τον κώδικα του EMESITI.....	31
3.6	XML-SOAP .....	32
3.6.1.1	Πλεονεκτήματα του SOAP .....	33
3.6.1.2	Μειονεκτήματα του SOAP.....	33
3.7	OGC GIS Web Services – Συνδυασμός AJAX και SOAP .....	35
3.7.1.1	Παράδειγμα AJAX μαζί με SOAP GIS Web υπηρεσία (Παν. Indiana) .....	37
3.7.1.2	Parsing XML δεδομένων.....	38
3.7.1.3	Συμπεράσματα από τη χρήση AJAX μαζί με SOAP GIS Web υπηρεσία από Πανεπιστήμιο Indiana και επιπλέον υλοποιήσεις WEB GIS υπηρεσιών .....	40
3.7.1.4	Συμπεράσματα από τη χρήση AJAX μαζί με SOAP GIS Web υπηρεσία από Πανεπιστήμιο Indiana και επιπλέον υλοποιήσεις WEB GIS υπηρεσιών .....	40

<b>4</b>	<b>Υφιστάμενες Υπηρεσίες και Εφαρμογές.....</b>	<b>43</b>
4.1	Google Maps API .....	43
4.2	Απλό παράδειγμα σελίδας με χρήση Google Maps API.....	46
4.3	Εναλλακτικές λύσεις με SOAP - Microsoft Mappoint .....	47
4.3.1.1	Λειτουργικότητα .....	47
4.4	WEB GIS εφαρμογές στο χώρο του Real Estate .....	48
4.4.1.1	HousingMaps.com .....	48
4.4.1.2	<a href="http://www.nestoria.co.uk">http://www.nestoria.co.uk</a> .....	49
4.4.1.3	<a href="http://www.zillow.com">http://www.zillow.com</a> .....	50
4.4.1.4	<a href="http://www.rento.gr">www.rento.gr</a> .....	51
4.4.1.5	<a href="http://www.xe.gr">www.xe.gr</a> -Εφημερίδα «ΧΡΥΣΗ ΕΥΚΑΙΡΙΑ» .....	52
4.4.1.6	<a href="http://www.realosophy.com">www.realosophy.com</a> .....	53
<b>5</b>	<b>Πρότυπη Εφαρμογή: E-MESITIS.....</b>	<b>55</b>
5.1	Εισαγωγή .....	55
5.2	E-MESITIS Βασική εφαρμογή .....	55
5.3	Λογική Αρχιτεκτονική & Εργαλεία Ανάπτυξης .....	56
5.3.1.1	Λογική Αρχιτεκτονική .....	56
5.3.1.2	Εργαλεία Ανάπτυξης.....	58
5.4	Δομή Βάσης Δεδομένων .....	63
5.4.1.1	Πίνακες της Βάσης.....	63
5.4.1.2	Επεξηγήσεις Σχήματος Βάσης Δεδομένων .....	64
5.4.1.3	Χωρικοί Τελεστές Αναζήτησης.....	67
5.5	Οδηγός Χρήσης.....	68
5.5.1.1	Μη εγγεγραμμένος χρήστης – επισκέπτης .....	69
5.5.1.2	Είσοδος εγγεγραμμένου επισκέπτη .....	70
5.5.1.3	Εισαγωγή νέου ακινήτου προς πώληση εγγεγραμμένου χρήστη.....	70
5.5.1.4	Εισαγωγή σημείου ενδιαφέροντος εγγεγραμμένου χρήστη.....	72
5.5.1.5	Αναζήτηση ακινήτου με βάση κριτήρια .....	72
<b>6</b>	<b>Επίλογος.....</b>	<b>75</b>
6.1	Σύνοψη και συμπεράσματα.....	75
6.2	Μελλοντικές επεκτάσεις .....	76
<b>7</b>	<b>Παραρτήματα.....</b>	<b>77</b>
7.1	PHP Κώδικας αναζήτησης.....	77
7.2	Κώδικας HTML – αρχείο frommarkup2.html .....	85
	<b>Βιβλιογραφία.....</b>	<b>117</b>

# **1** *Εισαγωγή*

## **1.1 Αντικείμενο της μεταπτυχιακής εργασίας**

Η σχετικά πρόσφατη ανάπτυξη διαδικτυακών τόπων που προσφέρουν εύκολη πρόσβαση σε γεωγραφική πληροφορία και διαδικτυακές υπηρεσίες με σκοπό τη διάχυση γεωγραφικών δεδομένων και πληροφοριών (Google Maps, Yahoo Maps), έχει οδηγήσει στην άνθιση εφαρμογών που συνθέτουν περιεχόμενο και υπηρεσίες από διάφορες πηγές & διαδικτυακές υπηρεσίες προκειμένου να προσφέρουν θεματικό περιεχόμενο με γεωγραφική υπόσταση.

Στόχος της μεταπτυχιακής εργασίας είναι η συγκέντρωση όλων των εργαλείων και τεχνικών απαραίτητων για την ανάπτυξη Δικτυακών GIS Εφαρμογών. Η ανάπτυξη εφαρμογής σχετικής με το χώρο του Real Estate (R.E. - Κτηματομεσιτικά) αποτελεί απλά ένα πεδίο εφαρμογής με ιδιαίτερα έντονη ζήτηση, που έρχεται να καλύψει ένα μεγάλο κενό στην ελληνική κτηματαγορά. Δεδομένης της απουσίας κεντρικά συγκεντρωμένης πληροφορίας για το σύνολο των συναλλαγών ακινήτων / γεωτεμαχίων η μόνη ίσως βιώσιμη και μικρού κόστους λύση είναι η ad-hoc συγκέντρωση των ενδιαφερόμενων σε ένα δικτυακό Portal R.E. . Το μόνο μειονέκτημα αναφορικά με την εμπορική επέκταση του συγκεκριμένου μοντέλου είναι τα έξοδα διαφήμισης, που απαιτούνται προκειμένου να γίνει γνωστό στο ευρύ καταναλωτικό κοινό μια τέτοια προσπάθεια.

## **1.2 Οργάνωση του τόμου**

Ο τόμος της μεταπτυχιακής εργασίας αποτελείται από έξι κεφάλαια, συν το παράρτημα. Σε αυτά γίνεται ανάλυση των υφιστάμενων προτύπων για την ανάπτυξη εφαρμογών, που κάνουν χρήση μεταδεδομένων γεωγραφικού χαρακτήρα στο Διαδίκτυο αλλά και των ήδη ανεπτυγμένων λύσεων

που δεν περιορίζονται μόνο στο χώρο της Κτηματαγοράς (R.E). Παρουσιάζεται τέλος η εφαρμογή που αναπτύξαμε αναδεικνύοντας καινοτόμους τρόπους χρήσης των τεχνολογιών (AJAX) προκειμένου να υπάρχει πραγματική φιλικότητα προς το χρήστη και άμεση συσχέτιση μεταξύ της περιήγησης του χρήστη και της εμφανιζόμενης πληροφορίας στο χάρτη.

Το 1<sup>ο</sup> κεφάλαιο είναι εισαγωγικό και περιγράφει σε γενικές γραμμές το αντικείμενο της εργασίας.

Στο 2<sup>ο</sup> κεφάλαιο τεκμηριώνεται με βάση την κλασική οικονομική θεωρία γιατί μια εφαρμογή ενημέρωσης του καταναλωτικού κοινού σχετικά με τα προς πώληση ακίνητα είναι απαραίτητη για τη σωστή λειτουργία της αγοράς. Σύντομα αναφέρονται τα προβλήματα και οι στρεβλώσεις της Ελληνικής κτηματαγοράς. Η ανάπτυξη της τελευταίας έχει στηριχτεί στην ελεγχόμενη παροχή πληροφόρησης από μεσίτες, που συχνά πυκνά δεν διαθέτουν καν τα απαραίτητα πιστοποιητικά-εγγύα ή επαγγελματικούς τίτλους. Το θεσμικό κενό διευκολύνει καταστάσεις του τύπου «ότι δηλώσεις είσαι», που δυστυχώς είναι γνωστές στην Ελλάδα και οδηγεί στον εύκολο πλουτισμό και στην παραοικονομία.

Στο 3<sup>ο</sup> κεφάλαιο αναπτύσσονται τα εργαλεία και οι αρχιτεκτονικές της παγκόσμιας κοινότητας αναφορικά με την ανάπτυξη δικτυακών υπηρεσιών. Η κατάργηση των κατακόρυφων και ιδιόκτητων (proprietary) silo πληροφορίας είναι ίσως μονόδρομος για την ανάπτυξη αλληλοσυνδεδεμένων εφαρμογών με αυξημένη ευχρηστία και παροχή καινοτόμων υπηρεσιών. Τέτοιες τεχνολογίες όπως AJAX, SOAP, XML κλπ αναπτύσσονται σε βαθμό, που ο ενδιαφερόμενος αναγνώστης να ανατρέξει στις πάμπολες πηγές, που υπάρχουν πλέον διαθέσιμες για αυτόν τον λόγο.

Στο 4<sup>ο</sup> κεφάλαιο παρουσιάζονται μερικά εργαλεία σχετικά με την ενσωμάτωση γεωγραφικών μεταδεδομένων στις δικτυακές εφαρμογές του τρίτου κεφαλαίου. Δεν γίνεται εξαντλητική διερεύνηση αλλά μελετώνται τα πλέον αντιπροσωπευτικά, όπως Google Maps, Google Earth και Microsoft Mappoint. Τέλος παρουσιάζονται παραδείγματα από τον παγκόσμιο ιστό, διάφορων δικτυακών εφαρμογών σχετικών με την περιοχή του R.E , στα οποία γίνεται χρήση γεωγραφικών πληροφοριών. Χαρακτηριστικό κοινό μειονέκτημά τους είναι η μη άμεση ενημέρωση των απεικονιζόμενων χαρτών μετά από μετακίνηση του χρήστη. Οι υλοποιήσεις δεν παρέχουν πλήρη GIS αίσθηση και φιλικότητα στο χρήστη. Αυτό το πρόβλημα, που αποτελεί και την καινοτομία της δικής μας υλοποίησης, αντιμετωπίζεται στον E-MESITI.

Στο 5<sup>ο</sup> κεφάλαιο περιγράφονται οι τεχνικές λεπτομέρειες της υλοποίησής μας και τα εργαλεία και η υπολογιστική υποδομή που χρησιμοποιήσα. Εικόνες (screenshots) παρατίθενται μαζί με οδηγίες χρήσης. Αναφέρονται διεξοδικά και τα σημεία δυσκολίας ή όσες περιοχές χρήζουν περαιτέρω ανάπτυξης.

Το 6<sup>ο</sup> κεφάλαιο συνοψίζει τα συμπεράσματα της παρούσας μεταπτυχιακής εργασίας και αναλύει πιθανές μελλοντικές επεκτάσεις, της εφαρμογής του E-MESITI.





# 2

## *Τα οικονομικά του Real Estate – Ανάγκη για GIS*

### *2.1 Ιδιοκτησία στο πέρασμα των αιώνων*

#### *2.1.1.1 Γενικά*

Η αξία της γης [WIKI08] και οι τρόποι εκτίμησής της είναι θέμα που απασχόλησε τον άνθρωπο από την αρχαιότητα μέχρι σήμερα με πλήθος φιλοσοφικών ρευμάτων (Αριστοτέλης, Ρωμαίοι, Εκκλησιαστικοί συγγραφείς, Κλασσικοί Μεταρρυθμιστές, Μαρξιστές κ.ο.κ.). Ο Αριστοτέλης στα «Πολιτικά» διαχώρισε την αξία της γης σε αξία χρήσης και ανταλλακτική αξία.

Ο Adam Smith (1723-1790) ιδρυτής της σύγχρονης οικονομικής επιστήμης διαχώρισε ακόμα περισσότερο την έννοια της αξίας ενώ ο D.Ricardo (1772-1823) εισήγαγε την έννοια της γαιοπροσόδου. Ο Karl Marx (1818-1883) ανέπτυξε τη θεωρία της υπεραξίας αλλά και την ιδέα ότι αφού το έδαφος δεν αποτελεί προϊόν είναι άνευ αξίας, ενώ η εργασία επί του εδάφους είναι αυτή που παρέχει το δικαίωμα στην ιδιοκτησία. Ανεξαρτήτως της εκάστοτε επικρατούσας θεωρίας, η οικονομία χαρακτηρίζεται από δυναμικά και όχι στατικά φαινόμενα συνεπώς κάθε θεωρία δεν αποτελεί ακριβή απεικόνιση της αλήθειας αλλά λειτουργεί απλουστευτικά. Συνεπώς η ισχύς της εκάστοτε θεωρίας είναι πεπερασμένη και όχι διαχρονική.

Ο άνθρωπος ζεί και διαμορφώνει κάθε φορά τις συνθήκες που εξυπηρετούν τη διαβίωσή του. Οι ανάγκες του είναι απεριόριστες, πολλαπλασιάζονται και μετασχηματίζονται δια μέσου του χρόνου λαμβάνοντας υπόψη και την αύξηση του παγκόσμιου πληθυσμού. Η λειτουργία της οικονομίας με όλες τις μορφές – εκφάνσεις της είναι ένα μέσο εξασφάλισης των περιορισμένων

διαθέσιμων πόρων για την ικανοποίηση όσο περισσότερων αναγκών. Στο Ρωμαϊκό Δίκαιο η ιδιοκτησία αποτελούσε βασικό στοιχείο της «ελευθερίας του συμβάλλεσθαι» σχετικά με τα ανταλλάξιμα αγαθά. Οι διάφοροι τύποι ιδιοκτησίας περιλαμβάνουν την πραγματική ιδιοκτησία (έδαφος), την προσωπική ιδιοκτησία (λοιπά αγαθά) και στις μέρες μας και την πνευματική ιδιοκτησία (δικαιώματα επί καλλιτεχνικών δημιουργιών, εφευρέσεων, κ.λπ.). Το δικαίωμα της κατοχής είναι στενά συνυφασμένο με την έννοια της ιδιοκτησίας και δίνει στον ιδιοκτήτη το δικαίωμα να αξιοποιήσει, διανείμει, μεταφέρει την ιδιοκτησία με όποιον τρόπο αυτός κρίνει.

Η δημόσια ιδιοκτησία είναι οποιαδήποτε ιδιοκτησία ελεγχόμενη από το κράτος ή από μια κοινότητα. Η ιδιωτική ιδιοκτησία είναι εκείνη η ιδιοκτησία που δεν είναι δημόσια και είναι υπό έλεγχο ενός μεμονωμένου ή από μια ομάδα ατόμων. Τα σύγχρονα συστήματα ιδιοκτησιακού δικαίου δίνουν δικαιώματα ιδιοκτησίας σε νομικά πρόσωπα, το οποίο είναι ευρύτερο από τα φυσικά πρόσωπα και περιλαμβάνει εταιρείες. Η ιδιοκτησία προστατεύεται τόσο από τη Διεθνή συνθήκη των Ανθρωπίνων Δικαιωμάτων των Ηνωμένων Εθνών Άρθρο 17, αλλά και από την Ευρωπαϊκή συνθήκη ανθρωπίνων Δικαιωμάτων Πρωτόκολλο 1 (European Convention on Human Rights - ECHR).

### ***2.1.1.2 Ιδιοκτησία και Κοινωνικά Συστήματα***

Σύμφωνα με τον Adam Smith, η κεντρική ιδέα του καπιταλισμού είναι η προσδοκία του κέρδους. "Η βελτίωση του αποθέματος κεφαλαίου" στηρίζεται στα ιδιοκτησιακά δικαιώματα, και την πεποίθηση ότι τα δικαιώματα ιδιοκτησιών ενθαρρύνουν τους κατόχους ιδιοκτησίας στο να αναπτύξουν την ιδιοκτησία, να παράγουν πλούτο και να διαθέσουν αποτελεσματικά τους πόρους. Συνεπώς η λειτουργία της αγοράς στηρίζεται στην κεντρική ιδέα της κεφαλαιοκρατίας, που είναι η ιδιοκτησία. Σε αυτή τη λογική, ο νόμος προστατεύει την ιδιοκτησία με την προσδοκία της παραγωγής περισσότερου πλούτου και άρα και καλύτερων επιπέδων διαβίωσης για τον πληθυσμό. Η βάση του καπιταλισμού είναι η θεωρία της εργασίας σύμφωνα με την οποία «σου ανήκει η ζωή σου, άρα και ότι παράγεις εν ζωή και όλα τα προϊόντα είναι ανταλλάξιμα στην ελεύθερη αγορά».

Αντίθετα ο Σοσιαλισμός στηρίζει τη φιλοσοφία του σε μια κριτική της έννοιας της ιδιοκτησίας, δηλώνοντας, μεταξύ άλλων, ότι το κόστος προάσπισης της ιδιοκτησίας είναι υψηλότερο από τα οφέλη από την ιδιωτική ιδιοκτησία, και ότι η ενθάρρυνση του ιδιοκτήτη-κατόχου να αναπτύξει την ιδιοκτησία του, παράγοντας πλούτο, δεν συνεπάγεται την αύξηση του συλλογικού επιπέδου διαβίωσης παρά μόνο του προσωπικού. Στα πλαίσια αυτά, αναπτύχθηκε και η ιδέα της

προσωρινής ιδιοκτησίας με τη μορφή της χρησικτησίας. Για παράδειγμα οι εργαλειομηχανές ανήκουν στους εργάτες, που τις χρησιμοποιούν ενώ δεν νοείται ιδιοκτήτης εξ αποστάσεως.

Από την άλλη πλευρά, ο κομμουνισμός υποστηρίζει ότι μόνο η συλλογική ιδιοκτησία των μέσων της παραγωγής μέσω πολιτικών ενεργειών (όχι απαραίτητως ενός κράτους) θα εξασφαλίσει την ελαχιστοποίηση των άνισων ή άδικων διαχωρισμών και τη μεγιστοποίηση των οφελών, και ότι η ιδιωτική ιδιοκτησία (που στην κομμουνιστική θεωρία περιορίζεται στο κεφάλαιο) πρέπει να καταργηθεί. Διαφοροποίηση γίνεται μόνο για την προσωπική ιδιοκτησία του προλεταριάτου, που χαρακτηρίζεται ως «σκληρά - κερδισμένη, με ίδια εργασία και ίδια μέσα» (κομμουνιστικό Μανιφέστο).

Ο Charles Comte, στο «Traité de la propriété» (1834), δικαιολόγησε τη νομιμότητα της ιδιοκτησίας μόνο στην περίπτωση που δεν προκαλεί δεινά σε κανένα. Υποστήριξε ότι ιστορικά η ιδιοκτησία, εν μέρει αλλά όχι ολόκληρη, είναι νόμιμη. Ο Comte όπως και ο Proudhon αργότερα απέρριψαν το Ρωμαϊκό δίκαιο και την έννοια της δουλείας. Στην πραγματεία του το 1849 «What is Property?», ο Pierre Proudhon διαχώρισε την ιδιοκτησία σε de jure ιδιοκτησία (legal title) και de facto ιδιοκτησία (φυσική κατοχή) και θεώρησε την πρώτη ως παράνομη ακόμα και την υπεκμίσθωσή της. Οι θεωρίες του Proudhon επηρέασαν τους εκκολαπτόμενους τότε αναρχικούς όπως τον Mikhail Bakunin.

### **2.1.1.3 Frédéric Bastiat - property is value**

Ο Frédéric Bastiat στο βιβλίο του «Economic Harmonies» (1850) εισήγαγε την καινοτόμο αντίληψη ότι η ιδιοκτησία δεν αποτελεί σχέση μεταξύ ατόμων και αντικειμένων αλλά μεταξύ διαφορετικών ατόμων όσον αφορά στα αντικείμενα. Στην ουσία, αυτό που κατέχει ο ιδιοκτήτης δεν είναι το ίδιο το αντικείμενο αλλά η αξία του αντικειμένου (αγοραία αξία). Ο Bastiat σε αντίθεση με τον Proudhon υποστήριξε ότι εξαιτίας της τεχνολογικής εξέλιξης οι ώρες εργασίας, που απαιτούνται για το ίδιο έργο, μειώνονται, άρα η ιδιοκτησία «αυτοκαταστρέφεται» μετατρέπόμενη σε συλλογική πρόοδο. Η ιδιοκτησία βέβαια δεν θα καταργηθεί ποτέ δεδομένης της συνεχούς δημιουργίας νέων αναγκών (βλ. σύγχρονη εποχή απόκτηση και Α' κατοικίας και Β' κατοικίας και αυτοκινήτου κ.λ.π.).

#### **2.1.1.4 Ιδιοκτησία στο σήμερα – ανάγκη για Πληροφόρηση**

Αν και σήμερα εξακολουθεί αρκετά έντονη η συζήτηση κυρίως με αφορμή την ανάπτυξη του Τρίτου Κόσμου, μια από τις πλέον πολυσηζητημένες θεωρίες είναι αυτή του σύγχρονου Περουβιανού οικονομολόγου Hernando de Soto (αδερφού του Alvaro). Αν και οι πολέμοί του, τον κατηγορούν για νεο-φιλελεύθερες ιδέες, η βασική του αρχή για καταπολέμηση της φτώχειας συνίσταται στην επίσημη αναγνώριση τίτλων ιδιοκτησίας στους φτωχούς των χωρών του Αναπτυσσόμενου Κόσμου. Η δυνατότητα ανταλλαγής / υποθήκευσης αυτών των τίτλων και ο δανεισμός σύμφωνα με τον De Soto θα οδηγήσει στην ανάπτυξη των χωρών αυτών με ίδια μέσα και στην έξοδο των παρειών των οικονομιών αυτών από την παραοικονομία.

Σημαντικό στοιχείο στη ρητορική του De Soto είναι η κρατική προστασία της περιουσίας και η τυποποίηση των διαδικασιών μεταβιβάσεων και οργανωμένου αρχείου, ώστε να αναπτυχθεί η οικονομία επιτρέποντας το δανεισμό και προσφέροντας μέτρο αξιολόγησης της φερεγγυότητας και του επενδυτικού ρίσκου των δανείων.

Η ανάγκη για πληροφόρηση δεν περιορίζεται μόνο στον αναπτυσσόμενο κόσμο αλλά αποτελεί και βασικό παράγοντα καθορισμού της εκάστοτε οικονομικής δραστηριότητας και στις ανεπτυγμένες οικονομίες, πολύ περισσότερο στην εγχώρια ελληνική αγορά. Η έλλειψη πληροφόρησης, οι συχνές αλλαγές στο νομοθετικό και φορολογικό πλαίσιο, η απουσία χωροταξικού σχεδιασμού, η σε πολλές περιπτώσεις άναρχη και αυθαίρετη επέκταση των πόλεων συνεπεία της αστικοποίησης, της εντονότατης οικιστικής πίεσης μετά την Μικρασιατική Καταστροφή και της κατάρρευσης των τοπικών γεωργικών οικονομιών (κρίση σταφίδας με ταυτόχρονη κατάρρευση εξαγωγικών λιμένων Πελοποννήσου – Μονεμβασιάς, Κυπαρισσίας – αποσάρθρωση τοπικών οικονομιών της Ηπείρου και Ζαγοροχωρίων) αλλά και αργότερα της συνολικής μεταστροφής της Ελληνικής οικονομίας σε μικρής κλίμακας βιοτεχνική παραγωγή, και η απουσία κρατικού σχεδιασμού δημιούργησε μια οικονομία με έντονες κερδοσκοπικές τάσεις [ΣΙΟΛ07]. Η εμπορία της γής αναπτύχθηκε σε μια ασταθή και επιρρεπή στην κερδοσκοπία ελληνική οικονομία και οδήγησε στον πολυτεμαχισμό της γής, άναρχη ανάπτυξη πόλεων με παντελή απουσία υποδομών δυσχεραίνοντας την ανάπτυξη οργανωμένων αστικών οικισμών με δημόσιες υποδομές εκτινάσσοντας το ύψος των επενδύσεων που απαιτούνταν για απαλλοτριώσεις. [ΖΑΜΠ08]

Το άρθρο 948 του Αστικού Κώδικα (Α.Κ) ορίζει ότι Ακίνητα είναι το έδαφος και τα συστατικά μέρη αυτού, όπως ορίζονται στο άρθρο 953 Α.Κ ή τα παραρτήματά του, όπως ορίζονται στο άρθρο 956 Α.Κ. Συνεπώς στα Ακίνητα συμπεριλαμβάνεται η γή και οι όποιες βελτιώσεις επ’

αυτής. Ομοίως το γεωτεμάχιο (ή οικόπεδο στον αστικό χώρο) ορίζεται το συνεχόμενο τμήμα της επιφανείας του εδάφους με τα συστατικά του μέρη και παραρτήματα με ενιαία λειτουργικότητα ή χρήση, που περιγράφεται αυτοτελώς σε συμβόλαιο [ZAMΠ08].

Η έννοια της Αξίας Αγοράς Ακινήτων έχει κατά καιρούς διαφορετικό περιεχόμενο. Ο πιο αντιπροσωπευτικός ορισμός, αναφορικά με την μέτρησή της, είναι ο ορισμός της Αξίας Ανοικτής Αγοράς (Open Market Value), σύμφωνα με την οποία έχουμε τέλεια ανταγωνιστική αγορά χωρίς υποκειμενικές εκτιμήσεις, αποκλείοντας τις περιπτώσεις αγοροπωλησιών, όπου υπήρξε εξαναγκασμός, χρονικός περιορισμός ή περιορισμένος αριθμός αγοραστών. Βέβαια στην πράξη σπάνια έχουμε τέλεια ανταγωνιστική αγορά και συνήθως έχουμε αγορά υπό περιορισμούς. Σε αυτή την περίπτωση η πλέον αντιπροσωπευτική τιμή ενός Ακινήτου είναι η Αγοραία Αξία (Market Value), η οποία είναι η υψηλότερη πιθανή τιμή σε ρευστό ή ισοδύναμο ρευστού, που προσφέρεται σε ένα ακίνητο με τις ισχύουσες συνθήκες ανταγωνισμού σε συγκεκριμένη χρονική στιγμή μέσα σε εύλογο χρονικό διάστημα από τον υποψήφιο αγοραστή υπό την προϋπόθεση ότι οι συμβαλλόμενοι είναι **καλά πληροφορημένοι** δρουν ορθολογικά με σύνεση προς το συμφέρον τους χωρίς περιορισμούς. [ZAMΠ08].

Σε αυτό το σημείο αξίζει να σημειώσουμε ότι το θεσμικό πλαίσιο, που διέπει το επάγγελμα των μεσιτών είναι το λιγότερο «χαλαρό» [ΣΟΕ07]. Η έλλειψη νομοθετικού πλαισίου που να καθορίζει σαφή επαγγελματικά δικαιώματα και ελάχιστα τυπικά προσόντα για το επάγγελμα του μεσίτη έχει οδηγήσει σε καταστάσεις που χαρακτηρίζονται από την κατάτμηση της αγοράς σε μικρομεσίτες, με αφανή έσοδα (το κόστος της μεσιτείας φτάνει το ποσοστό του 4% επί του συνολικού τιμήματος χωρίς την νόμιμη καταβολή φόρων «κοινώς μαύρα»). Το επιπλέον αυτό κόστος επιβαρύνει την οικοδομή και συνολικά την ανάπτυξη. Η δημιουργία ενός συστήματος άμεσης επαφής αγοραστών και πωλητών χωρίς την ύπαρξη αμφιβόλου κύρους μεσαζόντων, που χειραγωγούν την αγορά κατά το δικό τους συμφέρον διαχειριζόμενοι την πληροφόρηση θα αποβεί θετική για την εθνική οικονομία.

Στην κατεύθυνση της πληροφόρησης βασικό ρόλο μπορούν να διαδραματίσουν οι δικτυακές (online) εφαρμογές Γεωπληροφοριακών Συστημάτων (GIS). Δεδομένων των συνεχώς αυξανόμενων αναγκών για αστικό χώρο (μέχρι το 2030 4,9 δισεκατομμύρια άνθρωποι θα κατοικούν σε πόλεις δηλαδή το 60% του παγκόσμιου πληθυσμού [IEEESP707], η χρήση των GIS στο χώρο των κτηματομεσιτικών θα βοηθήσει σε

- ✓ Αύξηση παραγωγικότητας και αξιοπιστίας
- ✓ Μικρότερο κόστος πληροφόρησης

- ✓ Βελτίωση ανταγωνισμού

Οι δυνατότητες των GIS έχουν ως ακολούθως

- ✓ Χαρτογραφική απεικόνιση
  - Δημιουργία Θεματικών χαρτών
  - Geocoding – Γεωκωδικοποίηση: διαδικασία ανάθεσης γεωγραφικών αναγνωριστικών σε γνωρίσματα ενός χάρτη, όπως για παράδειγμα διευθύνσεις. Αναφέρεται και ως Γεωσήμανση (geotagging) από web2.0 εφαρμογές.
- ✓ Οργάνωση Βάσεων Δεδομένων
- ✓ Χωρική Ανάλυση πέραν των κλασσικών μεθόδων στατιστικής ανάλυσης
- ✓ Βελτίωση ανταγωνισμού

## **2.2 Ηλεκτρονικές Δημοπρασίες**

### **2.2.1.1 Εισαγωγή**

Μια μορφή άμεσης πληροφόρησης αποτελούν και οι Ηλεκτρονικές Δημοπρασίες [WIKI08] [COSMOONE]. Δεν σχετίζονται επί της παρούσης άμεσα με GIS αλλά θα μπορούσαν να συνδυαστούν στο μέλλον σε ένα ολοκληρωμένο Portal Ακινήτων. Αποτελούν επέκταση των κλασσικών δημοπρασιών, οι οποίες έχουν αποδειχτεί ως ένα δοκιμασμένο μέσο διευκόλυνσης αγορών ή πωλήσεων. Οι δημοπρασίες επιτρέπουν σε πολύ σύντομο χρόνο και με απλές διαδικασίες την διεξαγωγή συναλλαγών. Οι διάφορες δημοπρασίες, που έχουν αναπτυχθεί κατά τους αιώνες εξυπηρετούν τις ιδιαιτερότητες συγκεκριμένων αγορών προϊόντων π.χ. Ολλανδική δημοπρασία (Αγορά τουλίπας), Αγγλική Δημοπρασία (ξεχωριστή τεχνική και οικονομική προσφορά), Δημοπρασία Vichery κλπ.

Αξίζει να επισημανθεί πως οι Ηλεκτρονικές Δημοπρασίες μεταξύ επιχειρήσεων (B2B) διαφέρουν σημαντικά τόσο στο σχεδιασμό, όσο και στη φιλοσοφία, από τις δημοπρασίες οι οποίες αφορούν διάφορα καταναλωτικά αγαθά (B2C). Αρκετές εταιρείες δραστηριοποιούνται και στον ελληνικό κλάδο. Για παράδειγμα η cosmoONE (θυγατρική του ΟΤΕ) παρέχει υπηρεσίες Ηλεκτρονικών Δημοπρασιών αποκλειστικά για συναλλαγές μεταξύ επιχειρήσεων (B2B).

Οι B2B δημοπρασίες μπορούν να παραμετροποιηθούν σύμφωνα με τις εκάστοτε ανάγκες αγοράς ή πώλησης και μπορεί να είναι πλειοδοτικές ή μειοδοτικές (forward ή reverse). Τα κριτήρια των

δημοπρασιών μπορούν να διαφέρουν ανάλογα με τις εξειδικευμένες ανάγκες του κάθε διαγωνισμού. Έτσι μία δημοπρασία ενδέχεται να έχει ως κριτήριο επιλογής μόνο την τιμή (English Auction) ή το συνδυασμό τιμής και προσφερόμενης ποσότητας (Yankee ή Dutch Auction) ή το συνδυασμό τιμής και τεχνικών ή οικονομικών χαρακτηριστικών (Multi Variable Bids). Από την άλλη πλευρά οι δημοπρασίες B2C είναι αποκλειστικά πλειοδοτικές και επιδιώκεται η μεγαλύτερη δυνατή συμμετοχή.

### **2.2.1.2 Ηλεκτρονικές Δημοπρασίες Στα Ακίνητα**

Η αξιοποίηση της Ακίνητης Περιουσίας είναι από τα θέματα που θα απασχολήσουν έντονα την Αγορά στο άμεσο μέλλον είτε με τη μορφή Εταιρειών Επενδύσεων σε Ακίνητη Περιουσία, είτε ως "Leasing Ακίνητης Περιουσίας", είτε με τον όρο "Sale and Leaseback Ακινήτων" εξαιτίας της έλλειψης εναλλακτικών επενδύσεων, του προβλήματος ρευστότητας και των φορολαπαλαγών κ.α. Η διαπραγμάτευση των ακινήτων, αλλά και η επιλογή της κατάλληλης επένδυσης μπορούν να υλοποιηθούν μέσα από τις Ηλεκτρονικές Δημοπρασίες [COSMOONE], είτε πρόκειται για πώληση ακινήτου είτε πρόκειται για αγορά ή εκμίσθωση.

**Πώληση ακινήτου:** Στη διαδικασία πώλησης ενός ακινήτου περιλαμβάνονται οι εξής φάσεις:

- 1. Ενημέρωση του επενδυτικού κοινού.**
- 2. Παροχή πληροφοριών.** – Έλεγχος νομικών εγγράφων
- 3. Υποβολή ενδιαφέροντος.**
- 4. Διεξαγωγή της Δημοπρασίας**

Η υπηρεσία των Ηλεκτρονικών Δημοπρασιών μπορεί να δράσει καταλυτικά στην τέταρτη φάση, ενώ μπορεί να λειτουργήσει επικουρικά και στις τρεις πρώτες. Τα κύρια πλεονεκτήματα που διαπιστώνονται στην εφαρμογή της υπηρεσίας είναι:

- Επίτευξη της βέλτιστης τιμής.**
- Απόλυτη διαφάνεια.**
- Οικονομία χρόνου.**

Στην παρούσα εργασία δεν ασχολήθηκα με την υλοποίηση συστήματος δημοπρασίας. Τα παραπάνω αναφέρθηκαν ως ενδεχόμενες μελλοντικά πεδία διερεύνησης.





# 3

## *Τεχνολογίες WEB2.0*

### *3.1 Εισαγωγή σε Τεχνολογίες WEB -- HTTP and HTML*

Προτού μιλήσουμε για τον ηλεκτρονικό μεσίτη ακινήτων θα αναφέρουμε τεχνολογίες στις οποίες στηρίχτηκε. Ο παγκόσμιος ιστός όπως τον γνωρίζουμε έχει στηριχτεί σε δύο βασικά δομικά στοιχεία το πρωτόκολλο HTTP και τους κανόνες μορφοποίησης / παρουσίασης του περιεχομένου HTML [AJAXGIS WEB SERVICES] [WK HTML08]. Το Hypertext Transfer Protocol (HTTP) [WK HTTP08] είναι ένα πρωτόκολλο επικοινωνιών για τη μεταφορά πληροφορίας στο World Wide Web. Η ανάπτυξή του συντονίστηκε από το World Wide Web Consortium (W3C) και το Internet Engineering Task Force (IETF), προσπάθειες που κορυφώθηκαν στην έκδοση σειράς Request for Comments (RFCs), όπως τα RFC 2616 (Ιούνιος 1999) - HTTP/1.1, που χρησιμοποιείται ευρέως. Το HTTP είναι ένα πρότυπο επικοινωνίας τύπου request/response μεταξύ client και server στο οποίο στηρίζονται όλες οι ιστοσελίδες <http://www.mozilla.org>. Αποτελεί κομμάτι του Application Layer (Layer 4 στο TCP/IP stack), και μπορεί να χρησιμοποιηθεί με οποιοδήποτε Transmission layer πρωτόκολλο και όχι αποκλειστικά και μόνο με το TCP (default πόρτα 80). Διαθέτει 8 μεθόδους / αιτήσεις εκ των οποίων οι πιο σημαντικές είναι οι

GET

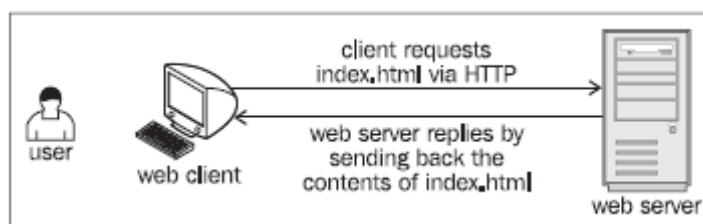
Με τη μέθοδο αυτή ο client ζητάει την αποστολή κάποιας απάντησης. Κανονικά η εντολή ανήκει στην ομάδα εντολών που χαρακτηρίζονται ως ασφαλείς (safe) και δεν πρέπει να προκαλεί server side αλλαγές, ώστε διαδοχικές κλήσεις της να επιστρέφουν το ίδιο πάντα αποτέλεσμα. Αυτό πολλές φορές δεν τηρείται από τους προγραμματιστές που με αυτόν τον τρόπο υλοποιούν online υπηρεσίες –εφαρμογές. Για παράδειγμα σε εντολές get αποστέλλονται και παράμετροι για τη σύνταξη SQL commands που υλοποιούν διαγραφές από μια βάση δεδομένων με ακίνητα. Δεν υπάρχει αυστηρή τήρηση του συγκεκριμένου κανόνα

## POST

Αποτελεί εξ ορισμού μη ασφαλή διαδικασία. Σε αντίθεση με την GET είναι ο πλέον «ασφαλής» τρόπος υποβολής αιτημάτων και υλοποίησης online εφαρμογών αφού δεν αποστέλλει τα δεδομένα inline στο αίτημα στο πεδίο του HTTP που περιλαμβάνει το URL αλλά μπορεί να εφαρμόζει κρυπτογράφηση γιατί τα δεδομένα περιέχονται στο σώμα του μηνύματος. [WK HTML08]

Ο πλέον γνωστός τύπος εγγράφου στον Παγκόσμιο Ιστό (Internet) είναι το HyperText Markup Language (HTML), και στηρίζεται σε μια μορφή markup κατανοητή στους φυλλομετρητές (web browsers) και περιγράφει τη μορφοποίηση και το περιεχόμενο. Το HTML δεν σχεδιάστηκε για την ανάπτυξη online εφαρμογών με δυναμικό περιεχόμενο γιατί κάθε μετάβαση σε νέα ιστοσελίδα HTML μέσω HTTP, απαιτεί πλήρη ανανέωση της σελίδας (page reload). Συνεπώς, απαιτείται η νέα σελίδα όπως και η διεύθυνσή της να προϋπάρχει ως στατικό αντικείμενο πριν από το αίτημα. [AJAX&PHP]

Παρόλα αυτά, τα δύο μαζί HTTP and HTML αποτελούν πολύ επιτυχημένο συνδυασμό. Η απλή αυτή συνεργασία φαίνεται στο επόμενο Εικόνα 1 Ένα απλό HTTP Request [AJAX&PHP]:



**Εικόνα 1 Ένα απλό HTTP Request [AJAX&PHP]**

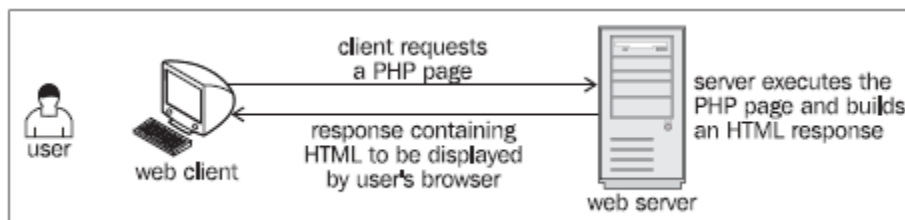
Προκειμένου να αντιμετωπιστούν τα μειονεκτήματα με το στατικό περιεχόμενο κλπ αναπτύχθηκαν διάφορες μέθοδοι. Όλες οι μέθοδοι χρησιμοποιούν το HTTP για την μεταφορά των δεδομένων. Η διαφοροποίηση έγκειται είτε στην πλευρά του server (Server side τεχνολογίες) με δυνατότητες δυναμικής παραγωγής σελίδων π.χ. με βάση τα περιεχόμενα κάποιας βάσης δεδομένων είτε με Client-side τεχνολογίες, που με διάφορες επεκτάσεις στο HTML επιχειρείται μια διευρυμένη εμπειρία σε σχέση με την απλή απεικόνιση ιστοσελίδων. Ανεξαρτήτως των εσωτερικών διεργασιών που λαμβάνουν χώρα στο server ο χρήστης θα πρέπει να πάρει ως απάντηση ένα HTML αντικείμενο, που θα μπορέσει να το δει στο φυλλομετρητή του. Server-side web technologies επιτρέπουν σε ένα web server να εκτελέσει πολύπλοκους υπολογισμούς να εκτελέσει αντικειμενοστραφή προγράμματα, να συνεργαστεί με βάσεις δεδομένων πέρα από το να εμφανίζει στατικές ιστοσελίδες. Κάποιες από τις server side λύσεις είναι οι PHP, το ASP.NET (Active Server Pages, τεχνολογία που υποστηρίζεται από τη Microsoft), η Java Server Pages (JSP), η Perl, το ColdFusion, το Ruby on Rails, και άλλα.

## 3.2 PHP και λοιπές Server-Side Τεχνολογίες

Η PHP (αναγραμματισμός του Hypertext Preprocessor) είναι μια scripting γλώσσα, που σχεδιάστηκε εξ αρχής για την παραγωγή δυναμικών ιστοσελίδων. Αρχικός δημιουργός της το 1995 ήταν ο Rasmus Lerdorf. Πλέον αναπτύσσεται από το PHP Group ως ελεύθερο λογισμικό υπό την σχετική αδειοδότηση από την Free Software Foundation. Η PHP είναι ευρέως διαδεδομένη και μπορεί να ενσωματωθεί σε HTML κώδικα. Εκτελείται στο server λαμβάνοντας ως είσοδο τον PHP κώδικα και παράγοντας ως αποτέλεσμα ιστοσελίδες. Είναι συμβατή με κάθε λειτουργικό και web server. Χρησιμοποιείται σε άνω των 20 εκατομμυρίων ιστοσελίδες παγκοσμίως και 1 εκατομμύριο web servers Ιδιαίτερα δημοφιλής στον Apache web server. Η τελευταία έκδοση της PHP είναι η 5.2.6 εκδοθείσα την 1η Μαΐου, 2008

Στην

Εικόνα 2: Client Requests a PHP Page [AJAX&PHP] απεικονίζεται η λειτουργία ενός αιτήματος για μια PHP σελίδα με όνομα index.php. Σε αντίθεση με την εικόνα 1, εδώ δεν επιστρέφονται τα περιεχόμενα της σελίδας index.php, αλλά ο server εκτελεί το index.php και επιστρέφει τα αποτελέσματα. Τα αποτελέσματα είναι σε μορφή HTML ή άλλη (πχ XML), που μπορεί να καταλάβει ο web client.



Εικόνα 2: Client Requests a PHP Page [AJAX&PHP]

Στην πλευρά του server μπορεί να υπάρχει κάποια βάση δεδομένων (Oracle, MySQL ή άλλη) για δυναμικό περιεχόμενο των σελίδων. Ακόμα και με την PHP η σελίδα είναι στατική και χωρίς δυνατότητες απεικόνισης πολυμέσων (video, music streaming κλπ). Η ανάγκη αυτή καλύπτεται όχι με την PHP αλλά με τεχνολογίες που έχουν αναπτυχθεί στην πλευρά του πελάτη (client-side technologies).

### 3.2.1.1 Java Framework

Αξίζει να σημειωθεί ότι υπάρχουν διαθέσιμα frameworks μετατροπής από Java σε Javascript.

Τέτοιο είναι και το γνωστό Google Web Toolkit<sup>1</sup>, που περιλαμβάνει μια μεγάλη βιβλιοθήκη εργαλείων - widgets και τον αντίστοιχο Java σε JavaScript compiler. Αποτελεί το βασικότερο ανταγωνιστή της PHP.

### **3.2.1.2 .NET Framework**

Το αντίπαλο δέος του κόσμου του ελεύθερου λογισμικού είναι η γνωστή Microsoft. Η δική της προσέγγιση στις web εφαρμογές στηρίζεται στο framework .asp με τις ομώνυμες ιστοσελίδες. Αποτελεί μέρος της λύσης της Microsoft με τον κωδικό .NET. Παρόλο που έχουν και αυτές ακολουθήσει τη λογική της AJAX που θα μελετήσουμε παρακάτω η φιλοσοφία τους είναι αρκετά διαφορετική.

## **3.3 JavaScript και άλλες Client-Side Technologies**

Οι διάφορες Client-side τεχνολογίες διαφέρουν από πολλές απόψεις, ξεκινώντας από τον τρόπο που φορτώνονται και εκτελούνται από τον web client. Το JavaScript<sup>2</sup> είναι μια scripting γλώσσα, της οποίας ο κώδικας γράφεται ως απλό κείμενο και μπορεί να ενσωματωθεί στις σελίδες HTML για να τις εμπλουτίσει. Όταν ένας χρήστης ζητά μια σελίδα HTML, εκείνη η σελίδα HTML μπορεί να περιέχει JavaScript. Το JavaScript υποστηρίζεται από όλους τους σύγχρονους φυλλομετρητές Ιστού χωρίς απαίτηση για επιπλέον εγκατάσταση.

Η JavaScript<sup>3</sup> δεν είναι μια μεταγλωττισμένη γλώσσα με συνέπεια να μην είναι κατάλληλη για λόγους ταχύτητας για εκτέλεση εντατικών υπολογισμών ή για οδηγούς συσκευών και η φάση της ερμηνείας (interpretation) εκτελείται στον φυλλομετρητή με αποτέλεσμα να εγείρονται και κάποια θέματα ασφαλείας (ακεραιότητα κώδικα κλπ). Απλά παραδείγματα Javascript είναι η

---

<sup>1</sup> [http://en.wikipedia.org/wiki/Google\\_Web\\_Toolkit](http://en.wikipedia.org/wiki/Google_Web_Toolkit)

<sup>2</sup> [JAVABIBLE], [JAVAIIDIOT], [JAVADUM], [AJAX&PHP], [WKJAVA]

<sup>3</sup> Παρά το όνομά της, η JavaScript σχετίζεται ελάχιστα με την Java. Ξεκίνησε από την Netscape Communications Corporation το 1995 στην προσπάθεια του Brendan Eich να απλουστεύσει την διεπαφή της Java του Navigator σε άπειρους χρήστες. Μετά την αρχική επιτυχία η Microsoft δημιούργησε τη δική της εκδοχή της JavaScript, που αποκάλεσε Jscript και η οποία είχε ασυμβατότητες με την έκδοση της Netscape.. Μετά την προσφυγή της Netscape και της Sun στο European Computer Manufacturers Association (ECMA) για προτυποποίηση της JavaScript, η standardized εκδοχή ονομάζεται ECMAScript. Οι περισσότερες εκδοχές της JavaScript συγκλίνουν προς την ECMAScript 3.0.

επικύρωση δεδομένων που εισάγει ένας χρήστης σε μια ιστοσελίδα εισαγωγής προσωπικών στοιχείων στην φάση της πληκτρολόγησης αποφεύγοντας την αποστολή της μισοσυμπληρωμένης αίτησης στο server για επικύρωση από το php script πριν την εισαγωγή στη DB. Η επαναποστολή της σελίδας στο χρήστη με το σχετικό μήνυμα λάθους θα οδηγούσε σε χειροκίνητη συμπλήρωση από τον χρήστη εκ νέου όλων των πεδίων. Με αυτόν τον τρόπο η web εφαρμογή γίνεται λίγο πιο φιλική προς τον χρήστη.

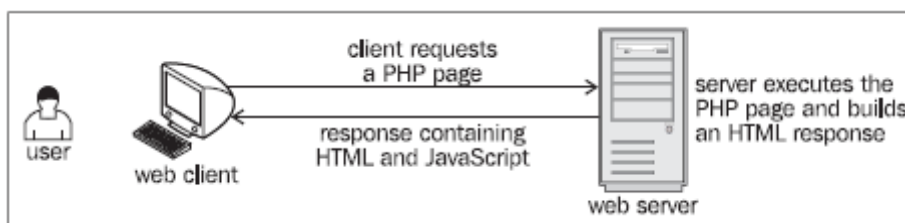
Άλλες ανταγωνιστικές τεχνολογίες στην πλευρά του χρήστη είναι τα Java applets και το Macromedia Flash. Τα Java applets είναι γραμμένα σε Java, και εκτελούνται μέσω μιας Java Virtual Machine, η οποία όμως πρέπει να εγκατασταθεί ξεχωριστά σε κάθε υπολογιστή πελάτη. Τα Java applets είναι μονόδρομος για πιο πολύπλοκα projects, αλλά έχουν χάσει την αρχική δημοτικότητά τους στο χώρο των web applications λόγω του ότι καταναλώνουν πολλούς πόρους ή απαιτούν μεγάλους χρόνους αρχικής εκκίνησης και υπερκαλύπτουν τις ανάγκες μικρών και απλών web εφαρμογών.

Το Macromedia Flash διαθέτει πολύ ισχυρά εργαλεία για δημιουργία animations και γραφικών εφέ (graphical effects), και αποτελεί de-facto standard για τέτοιου είδους προγράμματα. Το Flash όπως και η JVM απαιτεί από κάθε client να εγκαταστήσει κάποιο browser plug-in.

Συνδυάζοντας HTML με κάποια server-side τεχνολογία αλλά και κάποια client-side τεχνολογία , δημιουργεί τη βάση για την ανάπτυξη πολύ ισχυρών web εφαρμογών.

### 3.4 Φιλικότητα προς το χρήστη

Για την περίπτωση των απλών εφαρμογών, όπου δεν γίνεται χρήση των τεχνολογιών που θεωρούνται πιο «βαριές» όπως JAVA, ή FLASH, ο μηχανισμός λειτουργίας φαίνεται στην Εικόνα 3 Η απάντηση είναι μια σελίδα HTML με JavaScript και έχει κατασκευαστεί δυναμικά από την PHP.



Εικόνα 3: HTTP, HTML, PHP, και JavaScript σε λειτουργία [AJAX&PHP]

Το πρόβλημα με αυτή τη λύση είναι ότι κάθε φορά που ο client θέλει νέα δεδομένα από το server, ένα νέο HTTP request αίτημα πρέπει να αποσταλεί και να ξαναφορτώσει το σύνολο της σελίδας παγώνοντας κάθε άλλη τρέχουσα εργασία του χρήστη. Το ονομαζόμενο **page reload** αποτελεί το πλέον διαδεδομένο σύγχρονο πρόβλημα και η λύση είναι η τεχνολογία AJAX.

### 3.5 Κατανοώντας την AJAX

Το AJAX είναι ένα ακρωνύμιο για το **Asynchronous JavaScript and XML**. Δίνει απλά τη δυνατότητα στην client-side JavaScript να αποστέλλει αιτήματα στο server στο παρασκήνιο (background server calls) καθώς και να παραλάβει τα νέα δεδομένα ενημερώνοντας τα αντίστοιχα τμήματα μιας ιστοσελίδας χωρίς να προκαλεί full page reloads. Μια τυπική απεικόνιση της λειτουργίας αυτής φαίνεται στην Εικόνα 5, ενώ γνωστά παραδείγματα είναι τα Google Suggest<sup>4</sup>, το Yahoo! Instant Search, το GMail αποτέλεσε την πρώτη web email AJAX εφαρμογή με τις Yahoo! Mail και Hotmail να ακολουθούν, και επιπλέον τα Google Maps και Yahoo Maps.

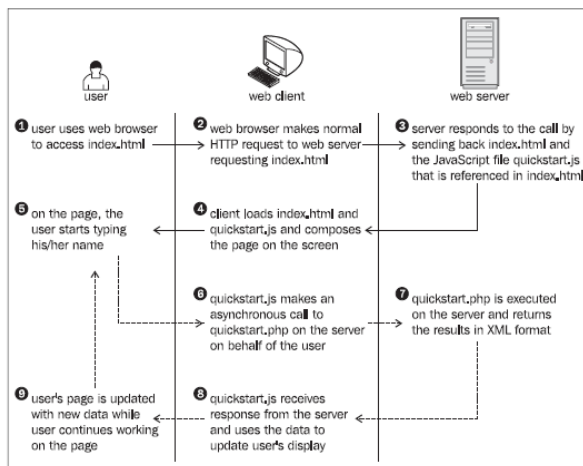


Εικόνα 4 Άρθρο Garrett

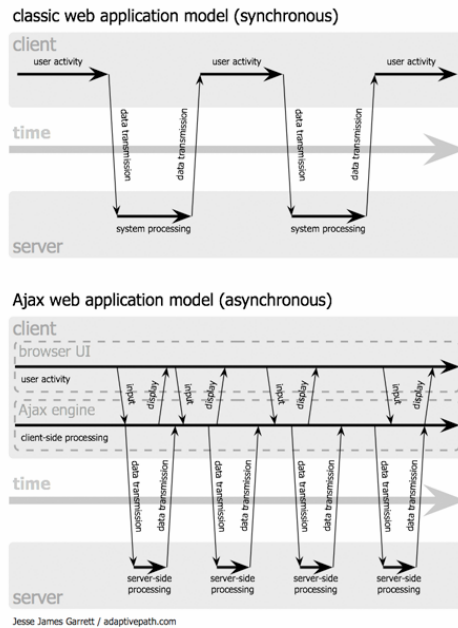
Κανένα συστατικό της AJAX δεν είναι καινούριο, αλλά όλα προϋπήρχαν ήδη από το 1998. Η AJAX έγινε γνωστή μετά το εξαιρετικό άρθρο του Jesse James Garret's το 2005 στο [AJAXGAR1], και διαδόθηκε αφότου η Google ξεκίνησε να το χρησιμοποιεί ευρέως. Αξίζει να σημειώσουμε ότι και η ίδια η Microsoft αναπτύσσει AJAX εργαλεία μέσα από το Project ATLAS (πλέον .NET AJAX).

Η AJAX κατορθώνει να εξισορροπήσει τη λειτουργικότητα που προσφέρουν οι client-side τεχνολογίες με τις server-side τεχνολογίες. Χωρίς τη χρήση της, οι δύο θεωρούνταν ανεξάρτητες και εκτελούνταν διαδοχικά αλλά με κοινό σκοπό την εξυπηρέτηση των αιτημάτων του χρήστη. Με την AJAX εξισορροπείται ο φόρτος εργασίας μεταξύ client και server side κώδικα επιτρέποντας την μεταξύ τους επικοινωνία, ενόσω ο χρήστης συνεχίζει να εργάζεται απρόσκοπτα στην ιστοσελίδα. Στη συνέχεια και για να γίνει πιο κατανοητή η χρήση αναφέρουμε παράδειγμα με data validation:

<sup>4</sup> <http://www.google.com/webhp?complete=1>, <http://instant.search.yahoo.com>, <http://www.gmail.com>, <http://maps.google.com>, <http://maps.yahoo.com>.



Εικόνα 5 Τυπική κλήση AJAX [AJAX&PHP]



Εικόνα 6 HTTPRequest in AJAX [AJAXGAR2]

Στα πλαίσια του παραδείγματος του data validation (Εικόνα 5 Τυπική κλήση AJAX [AJAX&PHP]) των προσωπικών δεδομένων τα πλεονεκτήματα της χρήσης του AJAX είναι ότι η επικύρωση των πεδίων μπορεί να γίνει όπως και με την απλή Javascript στην πλευρά του χρήστη αλλά χωρίς την υπερβολική καθυστέρηση της αρχικής έναρξης της σελίδας (π.χ. εξαιτίας της ενημέρωσης με όλες τις δυνατές πόλεις, χώρες, νομούς κλπ) για τα πεδία των διευθύνσεων). Με την AJAX, η web εφαρμογή μπορεί να επικυρώσει τα δεδομένα που έχει εισάγει ο χρήστης στο υπόβαθρο ενώ ο χρήστης συνεχίζει να πληκτρολογεί. Από τα προαναφερθέντα γίνεται σαφές ότι το μοντέλο AJAX ταιριάζει κατά κύριο λόγο σε δικτυακές εφαρμογές με μεγάλο όγκο πληροφορίας σε κάθε σελίδα, όπου η παραμικρή τροποποίηση θα επέφερε τρομερό όγκο δεδομένων. Τέτοιες εφαρμογές είναι και οι εφαρμογές γεωγραφικών δεδομένων μιας και οι χάρτες και γενικά οι εικόνες υποβάθρου αλλά και τα σημεία απεικόνισης απαιτούν μεγάλο όγκο πληροφορίας.

### 3.5.1.1 Τεχνικές λεπτομέρειες της AJAX

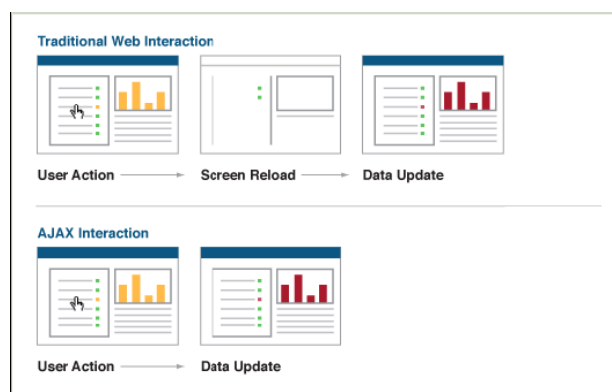
Οι τεχνικές που χρησιμοποιεί η AJAX είναι ήδη υλοποιημένες σε όλους τους σύγχρονους web browsers συμπεριλαμβανομένων των Mozilla Firefox, Internet Explorer και Opera. Δεν υπάρχει ανάγκη για εγκατάσταση επιπλέον λογισμικού στους client. Πιο αναλυτικά χρησιμοποιείται:

- JavaScript είναι το βασικό συστατικό της AJAX, επιτρέποντας την ανάπτυξη της client-side λειτουργικότητας. Στις JavaScript functions γίνεται ιδιαίτερη χρήση του Document Object Model (DOM) προκειμένου να ελεγχθούν προγραμματιστικά τα διάφορα στοιχεία των HTML σελίδων.

- Το XMLHttpRequest αντικείμενο επιτρέπει στην JavaScript να καλέσει τον server ασύγχρονα (asynchronously), στο υπόβαθρο χωρίς να διακόπτει τον χρήστη από την τρέχουσα εργασία του. Το αίτημα εξακολουθεί να είναι ένα απλά HTTP αίτημα χωρίς να απαιτεί ειδικές ρυθμίσεις στο τοπικό firewall του χρήστη.
- Κάποια server-side τεχνολογία απαιτείται να εξυπηρετήσει τα αιτήματα που έχουν αποσταλεί μέσω της Javascript. Η πιο διαδεδομένη λύση είναι η PHP.

Να σημειώσουμε ότι η απάντηση είναι σε μορφή (format), που μπορεί να κατανοήσει η JavaScript στον client. Το προτεινόμενο format και αυτό που χρησιμοποιήθηκε και στην μεταπτυχιακή εργασία είναι το XML (Extensible Markup Language), και αναλύεται διεξοδικότερα στην παράγραφο §3.6 παρακάτω XML-SOAP. Εδώ απλώς να αναφέρουμε ότι η XML έχει το πλεονέκτημα ότι υποστηρίζεται ευρέως και υπάρχουν αρκετές βιβλιοθήκες, που κάνουν εύκολο το parsing των απαντήσεων. Εναλλακτικά μπορούν να χρησιμοποιηθούν και άλλα format ακόμα και απλό κείμενο. Μια σχετικά δημοφιλής εναλλακτική στην XML είναι το JavaScript Object Notation (JSON).

**XMLHttpRequest (XHR)** [WK XMLHTTP08] είναι ένα API (Application Programming Interface) που μπορεί να χρησιμοποιηθεί από την JavaScript και άλλες web browser γλώσσες για να μεταφέρει XML και άλλα δεδομένα κειμένου ή ακόμα HTML και JSON μεταξύ web server και φυλλομετρητή. Αναπτύχθηκε αρχικά από την Microsoft ως μέρος του Outlook Web Access 2000, σαν server side API κλήση. Πλέον το υποστηρίζουν όλοι οι web browsers. Η υλοποίηση της Microsoft ονομάζεται XMLHttpRequest και υποστηρίζεται από τον Internet Explorer 5.0 και μετά. Είναι προσβάσιμη μέσω JScript, VBScript και άλλες scripting γλώσσες του IE. Το Mozilla project ενσωμάτωσε την πρώτη υλοποίηση του XMLHttpRequest στον Mozilla 1.0 το 2002.



**Εικόνα 7 Βασικό Πλεονέκτημα AJAX**



Σε αυτό το σημείο αξίζει να αναφέρουμε ότι η προσπάθεια της Microsoft να ενσωματώσει την AJAX στα προϊόντα-frameworks της Ajax.NET Professional και το ASP.NET AJAX<sup>5</sup> (μέχρι πρόσφατα η κωδική ονομασία ήταν *Microsoft Atlas*) στην .NET πλατφόρμα (.NET framework 3.5 και Visual Studio 2008) όπως το UpdatePanel διαφέρει ως προς το ότι στην ουσία καλύπτει την πολλαπλή αποστολή XMLHttpRequest και απλά «κοροϊδεύει» τον browser να μην κάνει reload ολόκληρη τη σελίδα. Η φιλοσοφία της Microsoft έγκειται στο ότι οι προγραμματιστές πρέπει να ρίχνουν το βάρος στην ανάπτυξη του server ή αλλιώς business side logic και όχι στην ανάπτυξη του client side, αδιαφορώντας για τα υπόλοιπα πλεονεκτήματα της AJAX (μικρότερο bandwidth, λιγότερα αιτήματα στο server για css ή άλλα συχνά επαναλαμβανόμενα components των ομοειδών σελίδων).

### **3.5.1.2 Μειονεκτήματα της AJAX**

Επειδή τίποτα δεν είναι τέλει σε αυτόν τον δύσκολο κόσμο, αναφέρω ενδεικτικά μερικά από τα προβλήματα της AJAX:

- ✓ Χάνεται η έννοια του bookmark αφού δεν αλλάζει ο τίτλος (url) της ιστοσελίδας. Όπως και στις εφαρμογές γραφείου δεν υπάρχει η παλιά έννοια του bookmark.
- ✓ Μηχανές αναζήτησης δεν μπορούν να καταλογογραφήσουν όλες τις AJAX σελίδες ενός site (χρήση bots και web-crawlers κλπ) γιατί δεν εκτελούν scripts σε javascript.
- ✓ Το κουμπί οπισθοχώρησης (Back button) και το ιστορικό στους browsers, απαιτεί ειδική τροποποίηση [AJAXDUM] για να έχει τα αναμενόμενα αποτελέσματα αφού η σελίδα παραμένει η ίδια.
- ✓ Ίσως οι browsers σε κάποια PDAs ή κινητά να μην είναι συμβατοί με το XMLHttpRequest αντικείμενο.

### **3.5.1.3 Παράδειγμα AJAX από τον κώδικα του EMESITI**

Η ακόλουθη συνάρτηση ορίζει ένα περιτύλιγμα που επιτρέπει την ασύγχρονη κλήση οποιουδήποτε php ή άλλου script :

---

<sup>5</sup> <http://en.wikipedia.org/wiki/Ajax.NET> και [http://en.wikipedia.org/wiki/ASP.NET\\_AJAX](http://en.wikipedia.org/wiki/ASP.NET_AJAX)

```

<script language = "javascript" type="text/javascript" >
    function getDataPOST(mydataSource, mydata)
    {
        var XMLHttpRequestObject = false;
        var mytarget = 0;
        if (window.XMLHttpRequest) {
            XMLHttpRequestObject = new XMLHttpRequest();
        } else if (window.ActiveXObject) {
            XMLHttpRequestObject = new ActiveXObject("Microsoft.XMLHttp");
        }
        if(XMLHttpRequestObject) {
            XMLHttpRequestObject.open("POST", mydataSource, true);
            XMLHttpRequestObject.setRequestHeader('Content-type',
            XMLHttpRequestObject.onreadystatechange = function()
            {
                if (XMLHttpRequestObject.readyState == 4 &&
                    XMLHttpRequestObject.status == 200) {
                    mytarget = XMLHttpRequestObject.responseText;
                    delete XMLHttpRequestObject;
                    XMLHttpRequestObject = null;
                }
            }
            XMLHttpRequestObject.send(mydata);
        }
        return mytarget;
    }
</script>

```

Και αντίστοιχα η ασύγχρονη κλήση π.χ. για να γίνει η επικύρωση του username και του password – το αποτέλεσμα απλά περιλαμβάνει απλό κείμενο – yes ή no ανάλογα με το αν είναι σωστά τα στοιχεία:

```

var urlGET = "loginGET.php?username=" + username + "&password=" +password ;
GETRequest(urlGET,password);

```

### 3.6 XML-SOAP

Η αιτία της καθυστερημένης ανάπτυξης των web εφαρμογών είναι η ύπαρξη των ιδιωτικών silo πληροφορίας κοινώς οι proprietary βάσεις δεδομένων των επιχειρήσεων, που κατά κανόνα είναι απομονωμένες και χωρίς διεπαφές με τον έξω κόσμο. Χαρακτηριστικά αναφέρεται το παράδειγμα των ασφαλιστικών εταιρειών. Αν τα συστήματά τους ήταν ανοικτής αρχιτεκτονικής τότε θα μπορούσε να αναπτυχθεί μια εφαρμογή από κάποιον τρίτο στη μορφή παρατηρητηρίου τιμών που θα καθιστούσε εφικτή την επιλογή του βέλτιστου ασφαλιστικού προγράμματος με βάση τις απαιτήσεις του εκάστοτε χρήστη.

Το επονομαζόμενο δεύτερης γενεάς ηλεκτρονικό εμπόριο (WEB2.0) είναι στην ουσία το άνοιγμα των απομονωμένων βάσεων δεδομένων σε εφαρμογές τρίτων. Οι τεχνολογίες που υποστηρίζουν μια τέτοια μετάβαση είναι οι XML και SOAP [WIKI08], [IEEESP204]. Η XML (Extensible Markup Language) αποτελεί εξέλιξη της HTML (Hypertext Markup Language) και δίνει νέες δυνατότητες διευκολύνοντας την αναζήτηση και τη μεταφορά δεδομένων μεταξύ μηχανών. Εντολές μορφοποίησης (formatting tags) δεν εμπλέκονται όπως στην HTML με το περιεχόμενο. Η διαφορά οφείλεται στο ότι η HTML ήταν κυρίως γλώσσα περιγραφής

του τρόπου εμφάνισης ιστοσελίδων και δεν έλυσε το πρόβλημα της διαλειτουργικότητας μεταξύ βάσεων δεδομένων. Σε αυτό το σημείο πρέπει να σημειωθεί ότι ούτε και η XML λύνει το πρόβλημα διαφορετικών schema διαφορετικών βάσεων, αλλά τυποποιεί τη δομή των αποτελεσμάτων.

Η πρώτη εκδοχή του SOAP ήταν το πολύ απλό XML-RPC.


Υπάρχουν αρκετοί τύποι μηνυμάτων στο SOAP, αλλά ο πιο διαδεδομένος είναι το Remote Procedure Call (RPC), στην οποία ο υπολογιστής –πελάτης (the *client*) αποστέλλει το αίτημα στον εξυπηρετητή (*server*) και ο τελευταίος επιστρέφει την απάντηση. SOAP είναι η συνέχεια του XML-RPC.

<b>3.6.1.1 Πλεονεκτήματα του SOAP</b>	
✓	Η χρήση του SOAP πάνω από το <b>HTTP</b> επιτρέπει ευκολότερη επικοινωνία μέσω proxies και firewalls από άλλες τεχνολογίες απομακρυσμένης εκτέλεσης ρουτινών.
✓	Το SOAP είναι αρκετά ευέλικτο ώστε να επιτρέπει τη χρήση διαφορετικών Layer4–(ορολογία OSI - transport) πρωτοκόλλων. Οι πιο συνηθισμένες στοίβες πρωτοκόλλων χρησιμοποιούν το HTTP ως transport protocol, χωρίς να αποκλείεται και η χρήση των (e.g. SMTP, RSS).
✓	Το SOAP είναι ανεξάρτητο πλατφόρμας ανάπτυξης.
✓	Το SOAP είναι ανεξάρτητο γλώσσας προγραμματισμού, ενώ είναι
✓	Απλό και επεκτάσιμο.

<b>3.6.1.2 Μειονεκτήματα του SOAP</b>	
✓	Εξαιτίας της ευγλωττίας της XML, το SOAP είναι αρκετά πιο αργό στην εκτέλεση σε σχέση με άλλες middleware τεχνολογίες όπως είναι η CORBA, το οποίο δεν αποτελεί πρόβλημα όταν αποστέλλονται μηνύματα μικρά σε μέγεθος. Για τη βελτίωση της

	ταχύτητας της XML στην περίπτωση embedded binary objects έχει αναπτυχθεί το Message Transmission Optimization Mechanism.
✓	Τα μηνύματα του SOAP χρησιμοποιούν το TCP/IP προκειμένου να παραδώσουν το μήνυμά τους στο σωστό σημείο, με συνέπεια οι ρόλοι των εμπλεκόμενων πλευρών να είναι καθορισμένοι (client server) - ερωτών και αποκρινόμενος. Η λειτουργικότητα που απαιτείται για κάθε πακέτο ορίζεται με out-of-message πληροφορία στις SoapAction HTTP επικεφαλίδες. Ανταγωνιστική τεχνολογία όπως η WS-Addressing (Web Services) ενσωματώνει όλους τους HTTP out-of-message information, (i.e., SoapAction HTTP επικεφαλίδες στο XML πακέτο).
✓	Η χρήση του HTTP οδηγεί σε υποβέλτιστες λύσεις, αφού στο επίπεδο εφαρμογής απαιτείται η γνώση για το αν θα χρησιμοποιηθεί POST-based binding ή η μέθοδος GET).

Μια πρόσθετη τεχνολογία, που επέτρεψε την αναζήτηση ακόμα και μεταξύ διαφορετικών εταιρειών (extranets) των παρεχόμενων Web Services είναι η UDDI (Universal Description Discovery and Integration). Η UDDI πρόκειται για μια XML-based κατάλογο-χρυσό οδηγό για εταιρείες παγκοσμίως, που ενδιαφέρονται να παρουσιάσουν τις υπηρεσίες τους στον παγκόσμιο ιστό. Πρόκειται για μια ανοικτή προσπάθεια, υποστηριζόμενη από την OASIS, που βοηθά τις εταιρείες να δημοσιοποιήσουν σε καταλόγους τις προσφερόμενες υπηρεσίες τους αλλά και να αναζητήσουν δικτυακές υπηρεσίες που προσφέρουν άλλες συνεργαζόμενες εταιρείες. Αποτελείται από τις White Pages — διευθύνσεις και στοιχεία επικοινωνίας, τις Yellow Pages — κατηγοριοποιήσεις με βάση συγκεκριμένες ταξινομίες και τις Green Pages — τεχνικές λεπτομέρειες αναφορικά με τις διαφημιζόμενες υπηρεσίες. Η UDDI αρχικά προτάθηκε ως βασικό στανταρ για τις Web service. Στηρίζεται σε SOAP μηνύματα και την γλώσσα WSDL - Web Services Description Language. Μέχρι και τον Ιανουάριο του 2006 υπήρχε ένα δημόσια προσβάσιμος κατάλογος με την ονομασία Universal Business Registry. Από το 2/2006 και μετά οι βασικοί υποστηρικτές του, οι IBM, Microsoft και SAP ανακοίνωσαν ότι έκλειναν τους δικούς τους UDDI κόμβους παρόλο που η πενταετής προσπάθεια στέφθηκε με απόλυτη επιτυχία με την επίσημη δικαιολογία ότι οι εμπορικές συμφωνίες μεταξύ εταιρειών αναφορικά με την χρήση Web Services απαιτούσε ακόμα τον ανθρώπινο παράγοντα.



FUNCTION	TRADITIONAL WEB ACTIVITIES	WEB SERVICES
Find a Web site	Search engine	UDDI
Site description	Search engine site description	WSDL
Transport protocol	HTTP	SOAP
Data format	XML	

Εικόνα 8 Ακρονύμια στο Web2.0 [IEEE SP204]

Η εφαρμογή που αναπτύχθηκε στα πλαίσια της παρούσας μεταπτυχιακής εργασίας σε καμία περίπτωση δεν προσπάθησε να δημιουργήσει Web Service. Όπως θα δούμε και σε επόμενο κεφάλαιο η δυσκολία στην Ελλάδα έγκειται στην έλλειψη συγκεντρωμένης πληροφορίας, είτε αυτή αφορά αγοραπωλησίες είτε στατιστικά στοιχεία ακινήτων σε κάποιο κεντρικό φορέα ή Επαγγελματικό Σύλλογο μεσιτών. Λύσεις με την παραπάνω λογική των Web Services σίγουρα θα έδιναν διαφορετική προοπτική στην εθνική οικονομία. [AJAXGIS WEB SERVICES]

### 3.7 OGC GIS Web Services – Συνδυασμός AJAX και SOAP

Δεδομένου ότι στην εν λόγω μεταπτυχιακή εργασία μελετάμε την ανάπτυξη web εφαρμογών με γεωγραφικά δεδομένα και απεικόνιση σε χάρτη είμαστε πλέον σε θέση να αναπτύξουμε τις προαναφερθείσες τεχνολογίες υπό το πρίσμα των γεωγραφικών δεδομένων και εφαρμογών.

Το Open Geospatial Consortium (OGC) ως διεθνής μη κερδοσκοπικός οργανισμός διεθνούς προτυποποίησης γεωγραφικών δεδομένων ορίζει έναν αριθμό προτύπων standards, τόσο για δομές δεδομένων (data models) όσο και για δικτυακές υπηρεσίες (online services).

Τα GIS εισάγουν μεθόδους οπτικοποίησης, διαχείρισης και ανάλυσης γεωχωρικών δεδομένων. Η φύση των γεωγραφικών δεδομένων απαιτεί τη συνεργασία και διαλειτουργικότητα πληθώρας παρόχων δεδομένων. Προκειμένου να εξασφαλιστεί η διαλειτουργικότητα, η OGC έχει εισάγει πρότυπα δημοσιοποιώντας specifications για GIS υπηρεσίες. Η έλευση των Web Service βοηθά

στο να ξεπεραστούν οι δυσκολίες – μειονεκτήματα της παραδοσιακής τεχνικής Distributed Object παρέχοντας διαλειτουργικότητα μεταξύ διαφορετικών πλατφορμών και γλωσσών σε καταναμημένα περιβάλλοντα.

Με τα GIS services μέσω των Web Service approach πολλοί GIS vendors μοιράζονται data stores και εφαρμογές σε καταναμημένα περιβάλλοντα. Η ανάπτυξη των OGC services με βάση το μοντέλο των Web Services προσφέρει τα ακόλουθα πλεονεκτήματα [AJAXGIS WEB SERVICES]:

**Distribution:** Ευκολότερο το μοίρασμα χωρικών δεδομένων και εφαρμογών (platform & language neutral).

**Integration:** Ευκολότερη ενσωμάτωση σε εφαρμογές τρίτων χωρικών δεδομένων και λειτουργιών με εύκολη κλήση online υπηρεσιών.

**Infrastructure:** Εκμετάλλευση της τεράστιας επένδυσης σε εξοπλισμό.

Οι πλέον γνωστές OGC GIS services είναι οι Υπηρεσίες GIS Χαρτογράφησης και Υπηρεσίες GIS data. Η ορολογία OGC για τις πρώτες είναι Web Map Services (WMS) και για τις δεύτερες Web Feature Services (WFS), Web Coverage Services (WCS). Η WFS παρέχει feature data σε vector format κωδικοποιημένο σε Geographic Markup Language (GML) ενώ η WCS παρέχει coverage data σε raster format.

Οι Cubewerx, Demis και Intergraph είναι εμπορικές εταιρείες GIS που ασχολούνται με τεχνολογίες Web Services. Η Google Maps και η Ka-Map έχουν ενσωματώσει το AJAX model στα συστήματα παρουσίασης χαρτών τους (GIS visualization systems). Η Ka-Map χρησιμοποιεί τον εξυπηρετητή MapServer<sup>6</sup> με παράλληλη χρήση AJAX και PHP. Σε όλες τις προαναφερθείσες προσπάθειες οι Web Services και η τεχνική AJAX χρησιμοποιούνται ξέχωριστά. Το ECMAScript (αναφέρθηκε στην υποσημείωση 3στη σελίδα 26 ) με το XML E4X είναι η μόνη λύση που ενοποιεί και τα δύο. Το E4X είναι μια επέκταση της JavaScript, που διευκολύνει το XML scripting. Με το E4X δεν χρειάζονται ειδικά XML APIs όπως το DOM ενώ τα XML έγγραφα αποτελούν πλέον εγγενή τύπο της ίδιας της JavaScript. Οι ιδιότητες του E4X επιτρέπουν την απευθείας κλήση Web Services μέσα από τον browser, αν και ο μόνος browser που υποστηρίζει E4X μέχρι στιγμής είναι ο **Mozilla 1.8**. [AJAXGIS WEB SERVICES –

---

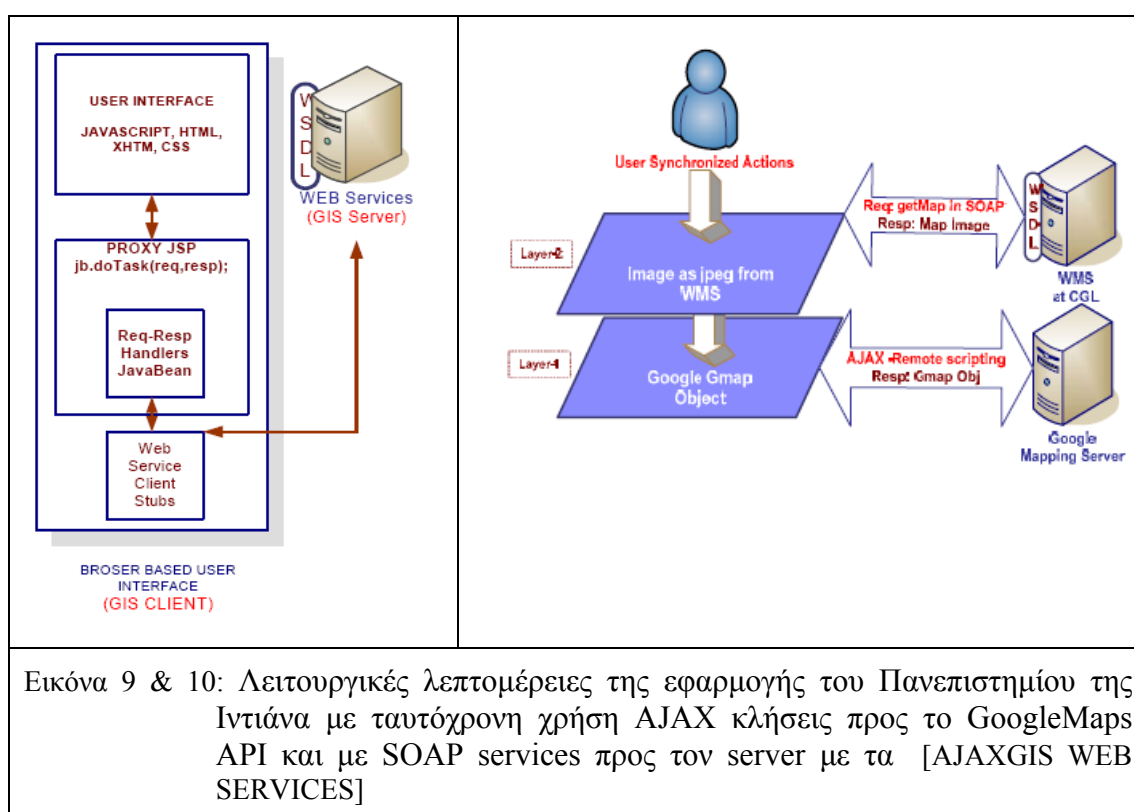
<sup>6</sup> MapServer official web site <http://ms.gis.umn.edu/>.

μεταγενέστερο του 2005 – Στις 6/2008 η τρέχουσα έκδοση του Mozilla Firefox 2.0.0.15 χρησιμοποιούσε το Mozilla rv:1.8.1.15, άρα οι παρατηρήσεις είναι επίκαιρες].

### 3.7.1.1 Παράδειγμα AJAX μαζί με SOAP GIS Web υπηρεσία (Παν. Indiana)

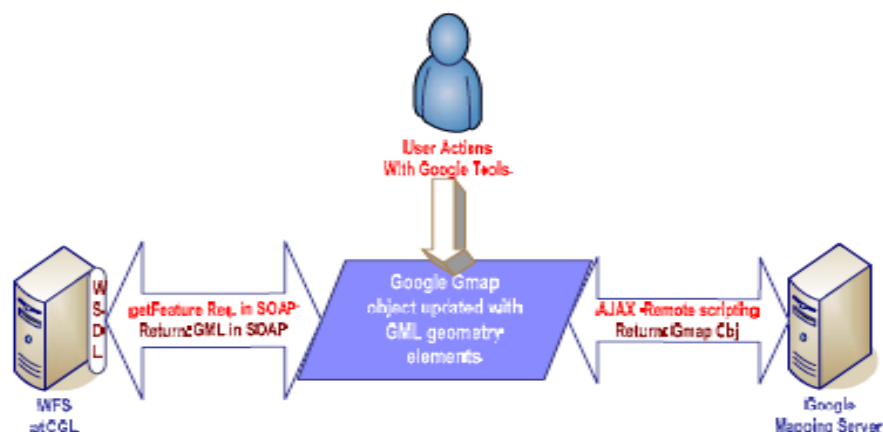
Στο άρθρο “Integrating AJAX Approach into GIS Visualization Web Services” [AJAXGIS WEB SERVICES] η ομάδα από το πανεπιστήμιο της Indiana περιγράφει την ανάπτυξη εφαρμογής που αξιοποιεί τόσο τη δικτυακή υπηρεσία Google Maps μέσω AJAX όσο και WFS μέσω SOAP. Η αρχιτεκτονική λύση φαίνεται στο ακόλουθο σχήμα. Πιο αναλυτικά η εφαρμογή του πελάτη αναπτύχθηκε με JSP. Η JSP διαθέτει κατάλληλο Interface για να λαμβάνει τα XMLHttpRequests από τη Javascript και να επικοινωνεί με τον WFS server.

Συνεπώς στη λύση τους έχουν συμπεριλάβει τις βασικές λειτουργίες GIS (μετακίνησης, zoom, υπόβαθρα κλπ) μέσω GoogleMaps και από την WEB SERVICES GIS εφαρμογή απλά καλούν επιπλέον layers.



Η επικάλυψη των αποτελεσμάτων του google με αυτά του WDS server γίνεται με χρήση τεχνικών JavaScript. Ο client δεν χρειάζεται να κάνει εργασίες rendering ή άλλες χαρτογραφικές εργασίες για την παραγωγή της εικόνας του χάρτη. Ο χάρτης επιστρέφεται από το WMS έτοιμος

για χρήση σε κάποιο format όπως JPEG ή PNG ή TIFF. These images in different formats are converted to a JavaScript object before overlaying.



**Εικόνα 11 Ενσωμάτωση GML αποτελεσμάτων της SOAP υπηρεσίας του WFS με το Gmap object του GoogleMaps API [AJAXGIS WEB SERVICES]**

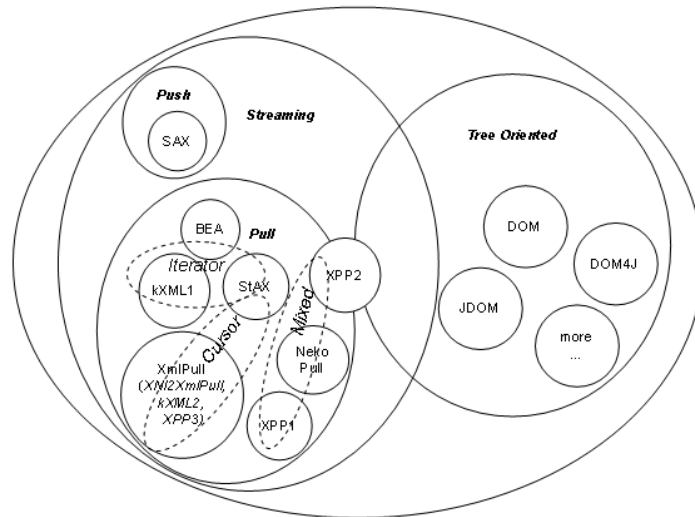
Σε αυτό το σημείο πρέπει να τονίσουμε ότι στο άρθρο εσφαλμένα αναφέρονται οι πρωτότυπες συναρτήσεις του GoogleMaps API με τα ονόματα Gmap, GPoint, GMarker. Το σωστό όπως αναφέρεται και στην [GOOGLEMAPS] είναι το `google.maps.Map2` αντί για `GMap2`.

Τα αποτελέσματα από την ασύγχρονη κλήση του XMLHttpRequest API επιστρέφουν σε XML το οποίο απαιτεί γνώση του DOM για το parsing. Υπάρχει περίπτωση τα αποτελέσματα να είναι πολύ ογκώδη και να οδηγήσουν σε μήνυμα λάθους "Out of Memory" το οποίο και επέλυσαν με Pull Parsing. Χωρίς να εισέλθω σε λεπτομέρειες η μορφή αυτή του parsing αντιμετωπίζει τις αρκετά μνημοβόρες δομές, που απαιτούνται κατά τη φάση της μετατροπής του σειριακής ακολουθίας XML tags σε δομή δέντρου. Το pull parsing αξιοποιεί τη λογική του streaming δεδομένων προς τον client και λαμβάνει μέρη για συσκευές μικρών δυνατοτήτων (π.χ. κινητά). Στην παρακάτω Εικόνα 12 Μέθοδοι για XML parsing - <http://www.xmlpull.org/history/diagram.gif> απεικονίζονται όλες οι μέθοδοι για parsing των XML απαντήσεων, που επιστρέφει ένας server, προκειμένου να ξαναδημιουργηθούν οι δομές που απαιτεί το business logic του προγράμματος από το stream των δεδομένων, που ελήφθησαν.

### 3.7.1.2 Parsing XML δεδομένων

Μετά το parsing της GML απάντησης από τον WFS μέσω της Java, το αποτέλεσμα προωθείται στον browser μέσω της διεπαφής Javascript-Java. Άρα η κλήση της XMLHttpRequest στη JavaScript, επιστρέφει τα κομμάτια της σελίδας που πρέπει να αλλάξουν.





Εικόνα 12 Μέθοδοι για XML parsing - <http://www.xmlpull.org/history/diagram.gif>

Ο αντίστοιχος κώδικας, που χρησιμοποίησα στον κώδικά μου για την επεξεργασία των XML δεδομένων (στοιχεία ακινήτων) φαίνεται ακολούθως:

```

if (navigator.appName == "Microsoft Internet Explorer") {
    xmlDoc=new ActiveXObject("Microsoft.XMLDOM");
    xmlDoc.async="false";
    xmlDoc.loadXML(localurl); }
else {
    if (navigator.appName == "Netscape") {
        arser=new DOMParser();
        xmlDoc=parser.parseFromString(localurl,"text/xml");
        // epestrepse to dom object apo tin xml apantisi
    }
}
//telos tou parsing tou xml

```

και για να αποθηκεύσω το array με τα σπίτια που επεστράφησαν:

```
AvailableSpitiemarkers = xmlDoc.documentElement.getElementsByTagName("marker");
```

Το διάβασμα των στοιχείων του κάθε σπιτιού-marker έγινε με το ακόλουθο (πχ για τον όροφο)

```
var orofos = parseFloat(AvailableSpitiemarkers[i].getAttribute("orofos"));
```

### **3.7.1.3 Συμπεράσματα από τη χρήση AJAX μαζί με SOAP GIS Web υπηρεσία από**

#### ***Πανεπιστήμιο Indiana και επιπλέον υλοποιήσεις WEB GIS υπηρεσιών***

Σχετικά με τα συμπεράσματα από την μελέτη του Παν Indiana, το Google Map παρέχει το χάρτη με γρήγορη απόκριση αλλά τα προβλήματα καθυστερήσεων υπάρχουν στις WMS και στις WFS Web Services , εξαιτίας των χαρακτηριστικών και του όγκου των γεωγραφικών δεδομένων αλλά και χρονοβόρων και επίπονων αλγορίθμων για το rendering algorithms των τελικών χαρτών. Να σημειωθεί ότι δεν χρησιμοποιήθηκαν high performance servers και ιδιωτικά δίκτυα. Για βελτίωση των χρόνων προτείνουν streaming version των WMS και WFS και αποστολή των γεωγραφικών δεδομένων σε μορφή εικόνας ή GML αντίστοιχα με χρήση message middleware systems όπως το NaradaBrokering, που έχει αναπτυχθεί υπό το CGL (Community Grids Lab.) στο ίδιο πανεπιστήμιο. Το NaradaBrokering<sup>7</sup> αποτελεί ένα σύστημα που χρησιμοποιεί κατακευματισμένους κόμβους στο διαδίκτυο (large-scale distributed streaming systems ) για την ανάπτυξη δικτύων διανομής περιεχομένου με έμφαση στην ασφάλεια και την ανθεκτικότητα σε βλάβες μεμονωμένων κόμβων. Ειδικά στην περιοχή του GIS προσφέρει Quality of Service (QoS) και ασφάλεια στα ανταλλασσόμενα μηνύματα ανεξαρτήτως του τρόπου μεταφοράς της πληροφορίας (TCP, UDP, Multicast, SSL, RTP, HTTP). Μερικά αρκετά εντυπωσιακά παραδείγματα που έχουν ήδη υλοποιηθεί είναι τα:

### **3.7.1.4 Συμπεράσματα από τη χρήση AJAX μαζί με SOAP GIS Web υπηρεσία από**

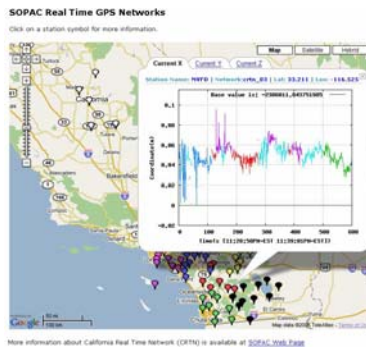
#### ***Πανεπιστήμιο Indiana και επιπλέον υλοποιήσεις WEB GIS υπηρεσιών***

Σχετικά με τα συμπεράσματα από την μελέτη του Παν Indiana, το Google Map παρέχει το χάρτη με γρήγορη απόκριση αλλά τα προβλήματα καθυστερήσεων υπάρχουν στις WMS και στις WFS Web Services , εξαιτίας των χαρακτηριστικών και του όγκου των γεωγραφικών δεδομένων αλλά και χρονοβόρων και επίπονων αλγορίθμων για το rendering algorithms των τελικών χαρτών. Να σημειωθεί ότι δεν χρησιμοποιήθηκαν high performance servers και ιδιωτικά δίκτυα. Για βελτίωση των χρόνων προτείνουν streaming version των WMS και WFS και αποστολή των γεωγραφικών δεδομένων σε μορφή εικόνας ή GML αντίστοιχα με χρήση message middleware systems όπως το NaradaBrokering, που έχει αναπτυχθεί υπό το CGL (Community Grids Lab.)

---

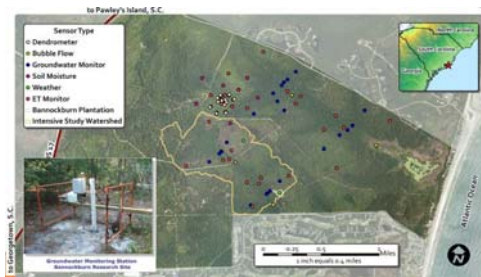
<sup>7</sup> <http://www.naradabrokering.org/>

στο ίδιο πανεπιστήμιο. Το NaradaBrokering<sup>8</sup> αποτελεί ένα σύστημα που χρησιμοποιεί καταναμημένους κόμβους στο διαδίκτυο (large-scale distributed streaming systems ) για την ανάπτυξη δικτύων διανομής περιεχομένου με έμφαση στην ασφάλεια και την ανθεκτικότητα σε βλάβες μεμονωμένων κόμβων. Ειδικά στην περιοχή του GIS προσφέρει Quality of Service (QoS) και ασφάλεια στα ανταλλασσόμενα μηνύματα ανεξαρτήτως του τρόπου μεταφοράς της πληροφορίας (TCP, UDP, Multicast, SSL, RTP, HTTP). Μερικά αρκετά εντυπωσιακά παραδείγματα που έχουν ήδη υλοποιηθεί είναι τα:



### Earthquake Science<sup>9</sup>

Το υπό της NASA χρηματοδοτούμενο QuakeSim project χρησιμοποιεί το NaradaBrokering για να συνδυάσει ροές πληροφορίας που διασυνδέουν καταναμημένα συστήματα και GPS συσκευές που δίνουν δεδομένα πραγματικού χρόνου.



### Environmental Monitoring

Το Πανεπιστήμιο του Clemson ερευνά θέματα περιβαλλοντικής βιωσιμότητας sustainability αναφορικά με την παρόχθια ανάπτυξη (παρακολούθηση υπόγειων υδάτων, ποιότητα επιφανειακών υδάτων, καιρικά φαινόμενα.

<sup>8</sup> <http://www.naradabrokering.org/>

<sup>9</sup> <http://quakesim.jpl.nasa.gov/>

## Audio-Video conferencing

Το GlobalMMCS10 project αποτελεί ένα A/V σύστημα τηλεδιασκέψεων που ενσωματώνει πληθώρα multimedia codecs και υποστηρίζει τη μείξη πολλαπλών video streams σε ένα μεμονωμένο single mixed stream. Το GlobalMMCS στηρίζεται και αυτό στο σύστημα του NaradaBrokering.



Το θετικό σημείο στο συνδυασμό AJAX και Web Services είναι ότι και τα δύο χρησιμοποιούν XML για την αίτηση και αποστολή των δεδομένων επιτρέποντας στους προγραμματιστές την εύκολη διασύνδεση των δύο. Η αποκλειστική χρήση Google Maps έχει δυσκολίες στην εξαγωγή και απεικόνιση πληροφορίας (feature), που σχετίζεται με κάποιο σημείο του χάρτη. Χρησιμοποιώντας GIS Web Services στην ίδια εφαρμογή και αναθέτοντας το κομμάτι αυτό στο WMS, το τελικό αποτέλεσμα είναι πολύ ταχύτερο μιας και εξαλείφεται το μειονέκτημα του Google Maps. faster.

---

<sup>10</sup> <http://www.globalmmcs.org>

# 4

## *Υφιστάμενες Υπηρεσίες και Εφαρμογές*

Στην παρούσα ενότητα παρουσιάζονται υφιστάμενες εμπορικές υπηρεσίες και εφαρμογές που χρησιμοποιούνται για τη διάχυση γεωγραφικής πληροφορίας με επιπλέον έμφαση σε υπηρεσίες σχετικές με την αγορά ακινήτων. Θα γίνει περιληπτική ανασκόπηση κυρίως του Google Maps API που χρησιμοποιήσα και στην υλοποίηση. Αναλυτικότερα, παρουσιάζονται υφιστάμενες δικτυακές υπηρεσίες με τις αντίστοιχες προγραμματιστικές διεπαφές (APIs) για την απεικόνιση πληροφοριών σε χάρτες και την ενσωμάτωση χαρτών σε ιστοτόπους (mash-ups).

### *4.1 Google Maps API*

Το Google Maps API [GGLMAP], [RAMGOOGL2] είναι μια δωρεάν και ελεύθερη δικτυακή υπηρεσία – Application Programming Interface που εστιάζει σε θέματα χαρτογράφησης (web-mapping) και GIS εφαρμογών. Μαζί με το Google Earth και Google Maps προσφέρεται από την Google δωρεάν.



Το Google Maps API αποτελεί τον πυρήνα – προγραμματιστικό εργαλείο με το οποίο αναπτύχθηκε και το Google Maps. Αποτελεί στην πράξη εργαλείο για την ανάπτυξη άλλων web σελίδων και εφαρμογών (άσχετων ως προς την google). Τα υπόβαθρα, που προσφέρει, ποικίλουν από οδικούς χάρτες σε δορυφορικές εικόνες και υβριδικούς. Οι δορυφορικές εικόνες είναι στην πλειοψηφία τους τουλάχιστον ενός έτους, ενώ μερικές από αυτές έχουν υποστεί λογοκρισία από τις εθνικές κυβερνήσεις κάποιων κρατών.

Τα βασικά εργαλεία που παρέχονται είναι για μετατόπιση (pan), εστίαση (zoom), εύρεση διεύθυνσης / τοπωνυμίου, προσθήκης νέου Layer, απεικόνιση σημείων ενδιαφέροντος με markers, δρομολόγηση με βάση το οδικό δίκτυο, οδηγίες οδήγησης από το αρχικό σημείο μέχρι το τελικό σημείο προορισμό κλπ. Οι πηγές των δεδομένων της Google Maps & Google Maps API είναι η Tele Atlas, η NAVTEQ, κυβερνητικές πηγές ενώ οι δορυφορικές εικόνες υψηλής ανάλυσης παρέχονται από την DigitalGlobe και τον δορυφόρο QuickBird.

Όλες οι εφαρμογές της Google έχουν βασιστεί κατά μεγάλο ποσοστό σε γλώσσα JavaScript. Αυτό έχει σαν αποτέλεσμα τη δυνατότητα αλληλεπίδρασης με το χρήστη, κάνοντας χρήση ασύγχρονων δικτυακών αιτήσεων στον εξυπηρετητή - Ajax κλήσεις.. Συγκεκριμένα, το Google Maps δημιουργήθηκε με χρήση του AjaxSLT framework. Να σημειώσουμε ότι το Google Earth στην ουσία δεν επιτρέπει την κατασκευή εφαρμογών αλλά απαιτεί το δικό του client για να λειτουργήσει.

Το Google Maps API προσφέρει μια ίδια μέθοδο για τη δημιουργία XmlHttpRequest αντικειμένων τα οποία συμβατή με τις εκδόσεις των Internet Explorer, Firefox, και Safari. Παρέχει επίσης τη δυνατότητα για parsing σε XML έγγραφα με χρήση της GXml.parse. Το Google Maps API δεν απαιτεί τη χρήση AJAX κλήσεων (XmlHttpRequest) για να λειτουργήσει και βασίζεται σε JavaScript/DHTML API, αλλά οι νέες εκδόσεις έχουν αντικαταστήσει τις παλιές κλάσεις με αντικείμενα που υποστηρίζουν πλέον AJAX.

Προς το παρόν οι υπηρεσίες, που παρέχονται από την Google είναι διαθέσιμες μόνο για κατασκευή ιστοσελίδων και δεν μπορούν να χρησιμοποιηθούν από άλλη εφαρμογή, αφού δεν βασίζονται σε κάποιο ανοιχτό πρότυπο, όπως το SOAP/XML.

Οι προαναφερθείσες υπηρεσίες είναι δωρεάν (για web sites με μέγιστο αριθμό επισκέψεων 50000/ημέρα) και το μόνο που χρειάζεται , είναι η καταχώρηση του domain name στο σύστημα της Google προκειμένου να του παραχωρηθεί κωδικός πρόσβασης (API key). Μοναδική υποχρέωση οι προσφερόμενες υπηρεσίες να μην διατίθενται επί πληρωμή αλλά να είναι δωρεάν προς τους επισκέπτες του ιστοχώρου του.

Δεν θα αναλύσουμε λεπτομερώς τη δομή των αντικειμένων του Google Maps αφού υπάρχουν κατατοπιστικότερα βιβλία και ιστότοποι. Απλά θα αναφέρουμε τις σημαντικότερες δομές δεδομένων που παρέχει σε ένα προγραμματιστή:

➤ **The GMap class:** Πρόκειται για την βασική κλάση που υλοποιεί ένα την οντότητα ενός

χάρτη. Κάθε στιγμιότυπο της κλάσης αντιστοιχεί σε ένα χάρτη στη σελίδα. Το μέγεθος του χάρτη καθορίζεται από το μέγεθος τους αντικειμένου που το περιβάλλει.

- **Info Window:** Κάθε χάρτης έχει ένα μόνο «παράθυρο πληροφοριών», το οποίο εμφανίζει HTML περιεχόμενο σε ένα παράθυρο πάνω από το χάρτη.
- **Overlays:** Τα overlays είναι αντικείμενα πάνω στο χάρτη τα οποία είναι «συνδεδεμένα» σε γεωγραφικές συντεταγμένες. Το Google Maps API περιλαμβάνει δύο τύπους overlays, τα markers και τα polylines.
- **Controls:** Το Maps API έχει ενσωματωμένους τα εξής χειριστήρια GLargeMapControl, GSmallMapControl, GSmallZoomControl, GmapTypeControl.

Τα ονόματα των τοποθεσιών και οι λεπτομέρειες των δρόμων ποικίλουν αρκετά από μέρος σε μέρος. Είναι οι ακριβέστεροι στις ΗΠΑ και την Ευρώπη, αλλά οι τακτικές αναπροσαρμογές χαρτογράφησης βελτιώνουν την κάλυψη και στα υπόλοιπα μέρη.

Η Google έχει επιλύσει πολλές ανακρίβειες στη διανυσματική χαρτογράφηση από την πρώτη απελευθέρωση του λογισμικού, χωρίς την απαίτηση ενημέρωσης στο ίδιο το πρόγραμμα. Ένα τέτοιο παράδειγμα ήταν η απουσία της περιοχής Nunavut στον Καναδά. Πρόσφατες ενημερώσεις συνετέλεσαν, επίσης, στην αύξηση της κάλυψης λεπτομερούς αεροφωτογραφίας ιδιαίτερα στην δυτική Ευρώπη. Σε ορισμένες περιοχές της ξηράς τα σύννεφα και οι σκιές (συμπεριλαμβανομένου και των σκιών από τα βουνά) καθιστούν δύσκολη ή και αδύνατη την παρατήρηση των λεπτομερειών.

Γενικά τα γεωγραφικά χαρακτηριστικά του Google Maps είναι:

- Coordinate System and Projection
  - Το εσωτερικό σύστημα αναφοράς συντεταγμένων της εφαρμογής Google Earth είναι οι γεωγραφικές συντεταγμένες (latitude/longitude) στο World Geodetic System του 1984 (WGS84) datum
- Βασική ανάλυση:
  - ΗΠΑ: 15μ.
  - Γερμανία: 1μ.
  - Παγκόσμια: εν γένει 15μ. (μερικές περιοχές όπως οι χώρες της Νότιας Αμερικής έχουν πολύ χαμηλή ανάλυση).
- Υψηλή ανάλυση:

- ΗΠΑ.: 1μ., 0.6μ, 0.3μ, 0.15μ (πολύ σπάνια, για παράδειγμα στο Cambridge, στο Google Campus, ή στο Glendale, CA.)
- Ευρώπη: 0.3μ, 0.15μ (για παράδειγμα στο Βερολίνο, στη Ζυρίχη και στο Αμβούργο)

## 4.2 Απλό παράδειγμα σελίδας με χρήση Google Maps API

Η Google διαθέτει αρκετά AJAX APIs που μπορούν να χρησιμοποιηθούν σε μία ιστοσελίδα χωρίς χρήση server side code. Τέτοια είναι τα Maps API, the AJAX Search API, και το AJAX Feed API. Για να χρησιμοποιηθούν απαιτείται απλά ένα κλειδί (registration στο site της Google για συγκεκριμένο domain):

```
<script type="text/javascript"
src="http://www.google.com/jsapi?key=ABCDEFGG"></script>
```

Η συνάρτηση google.load φορτώνει τα στοιχεία που θέλουμε:

```
<script type="text/javascript">
  google.load("maps", "2");
  google.load("search", "1");
</script>
```

άρα μια σχετικά απλή ιστοσελίδα είναι η ακόλουθη:

```
<html>
  <head>
    <script type="text/javascript"
src="http://www.google.com/jsapi?key=ABCDEFGG"></script>
    <script type="text/javascript">
      google.load("maps", "2");
      google.load("search", "1");

      // Call this function when the page has been loaded
      function initialize() {
        var map = new google.maps.Map2(document.getElementById("map"));
        map.setCenter(new google.maps.LatLng(37.4419, -122.1419), 13);

        var searchControl = new google.search.SearchControl();
        searchControl.addSearcher(new google.search.WebSearch());
```



```

        searchControl.addSearcher(new google.search.NewsSearch());
        searchControl.draw(document.getElementById("searchcontrol"));
    }
    google.setOnLoadCallback(initialize);
</script>

</head>
<body>
    <div id="map" style="width: 200px; height: 200px"></div>
    <div id="searchcontrol"></div>
</body>

</html>

```

Για την πληρότητα της συζήτησης αξίζει να αναφέρουμε και το Google Earth, που αποτελεί τρισδιάστατη απεικόνιση της γήινης επιφάνειας [RAMGOOGL1] και απαιτεί την εγκατάσταση ειδικού client στον υπολογιστή του τελικού χρήστη.

### ***4.3 Εναλλακτικές λύσεις με SOAP - Microsoft Mappoint***

Η διαδικτυακή υπηρεσία MapPoint είναι μια XML-based διαδικτυακή υπηρεσία με ευελιξία όσον αφορά τον προγραμματισμό της και παρέχεται από τη Microsoft. Για την εφαρμογή αυτή έχει χρησιμοποιηθεί η **Visual Studio .NET**. Χρησιμοποιείται ευρέως από εταιρίες και ανεξάρτητους προγραμματιστές σε εφαρμογές λογισμικού με στόχο την ολοκλήρωση (integration) σε location-based υπηρεσίες όπως χάρτες, οδηγίες προς οδηγούς και αναζήτηση της πιο κοντινής διαδρομής σε εφαρμογές λογισμικού και διαδικτυακές πύλες (portals).

Επιπροσθέτως, η διαδικτυακή υπηρεσία MapPoint μπορεί να προσφέρει ένα εκτεταμένο σύνολο περιεχομένων που σχετίζονται με την χαρτογράφηση (όπως κατατάξεις επιχειρήσεων καθώς και σημείων ενδιαφέροντος (Points Of Interest-POIs) που είναι πιθανό να διατεθούν ως συστατικό εφαρμογών. Αυτή τη στιγμή υποστηρίζονται από τη δικτυακή υπηρεσία MapPoint περισσότερες από 15.000.000 δοσοληψίες σε καθημερινή βάση.

#### ***4.3.1.1 Λειτουργικότητα***

Η διαδικτυακή υπηρεσία MapPoint προσφέρεται μέσω ενός **SOAP (Simple Object Access Protocol) API**. Κάθε μια από τις προσφερόμενες συναρτήσεις του MapPoint API στέλνει και

λαμβάνει ως ορίσματα XML δεδομένα. Η υποστήριξη του SOAP και άλλων ανοιχτών πρότυπων δίνει στη διαδικτυακή υπηρεσία MapPoint το **χαρακτηριστικό και προνόμιο της διαλειτουργικότητας (interoperability), με αποτέλεσμα να μπορεί εύκολα να συνδυαστεί και να ολοκληρωθεί με άλλες εφαρμογές.**

Έτσι παραθέτει σε μικρό χρονικό διάστημα μια λίστα σημείων ενδιαφέροντος με κριτήριο την εγγύτητά τους σε μια επιλεγμένη τοποθεσία. Οι αναζητήσεις αυτές μπορούν να εκτελούνται για σημεία ενδιαφέροντος κατά μήκος μιας συγκεκριμένης δοσμένης διαδρομής. Η δικτυακή υπηρεσία MapPoint εμπεριέχει μια βάση δεδομένων με περισσότερα από 15.000.000 σημεία ενδιαφέροντος (POI), ενώ και οι πελάτες μπορούν να συνεισφέρουν εισάγοντας στη βάση(στον Microsoft server,όπως προαναφέρθηκε) άλλα συνήθη σημεία ενδιαφέροντος.

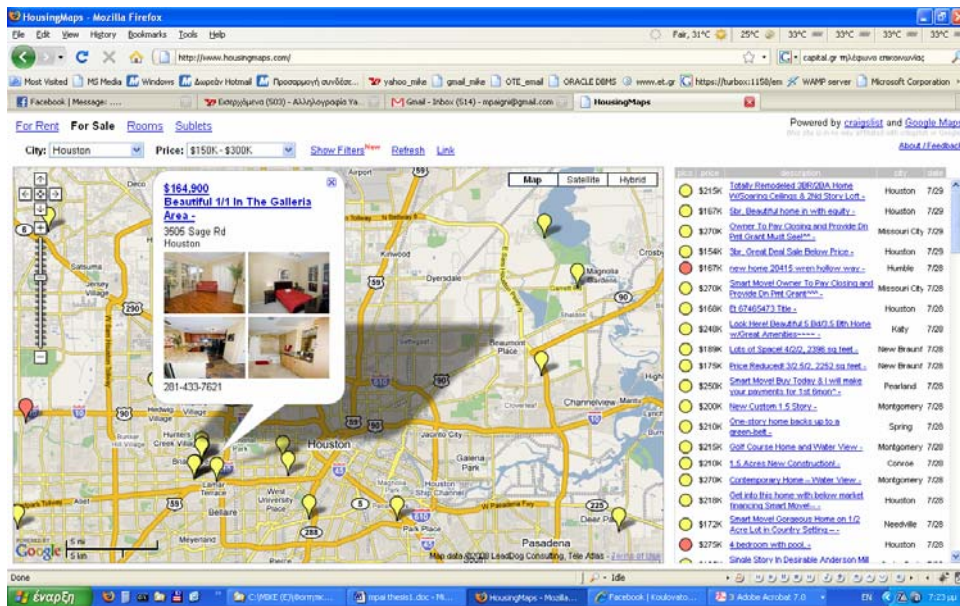
## ***4.4 WEB GIS εφαρμογές στο χώρο του Real Estate***

### ***4.4.1.1 HousingMaps.com***

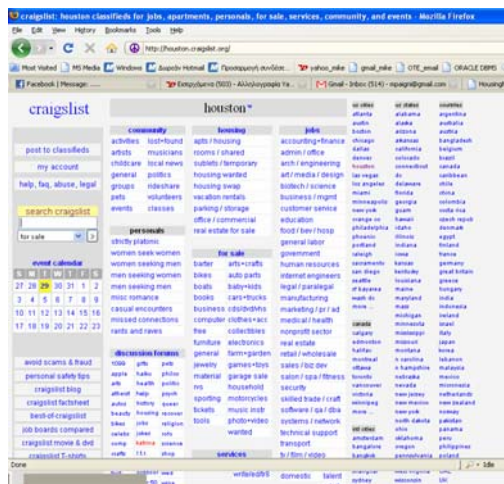
Αποτελεί στην ουσία ένα από τα πρώτα Mashups που κατασκευάστηκαν με χρήση του Google Maps API. Αξιοποίησε την διάσημη λίστα ακινήτων προς ενοικίαση / αγορά που υπάρχει στην Αμερική craigslist ([www.Craigslist.org](http://www.Craigslist.org)) απεικονίζοντας τα ακίνητα σε χάρτη [RAMGOOGLE]<sup>11</sup>. Δίνεται η δυνατότητα για φιλτράρισμα των αποτελεσμάτων ανάλογα με την τιμή, τη γεωγραφική θέση ή και άλλα κριτήρια. Διαθέτει και λίγα ακίνητα στην ευρύτερη περιοχή του Λονδίνου. Δυστυχώς δεν αξιοποιεί πλήρως τις δυνατότητες του AJAX. Για παράδειγμα δεν ενημερώνεται η λίστα ακινήτων κάθε φορά που μετακινούμαστε στο χάρτη. Η επιλογή refresh σχετίζεται μόνο με τις λεκτικές επιλογές (η γεωγραφική περιοχή επιλέγεται με βάση την πολιτεία μόνο και σε σχετικά μικρό zoom).

---

<sup>11</sup> <http://housingmaps.com>,



Εικόνα 13 HousingMaps.com χωρική αναζήτηση και infowindow με φωτογραφίες



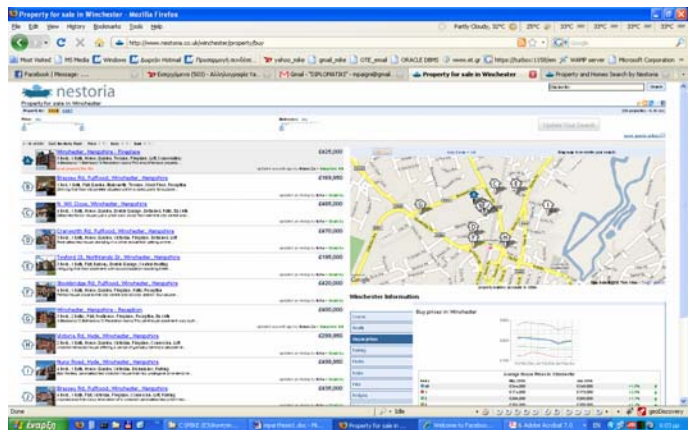
Εικόνα 14 Craigslist για την περιοχή του Houston, USA

#### 4.4.1.2 <http://www.nestoria.co.uk>

Βασισμένο σε [yui.yahooapis.com](http://yui.yahooapis.com) πρόκειται για πολύ αισθητικά όμορφο site με AJAX δυνατότητες. Στηρίζεται στις λίστες του <http://www.homes24.co.uk> και για mapping tools χρησιμοποιεί το <http://www.mapstraction.com/>. Απευθύνεται στην Αμερικάνικη αγορά ακινήτων. Το τελευταίο είναι μια προσπάθεια άμβλυνσης των διαφορών μεταξύ online mapping APIs από τους κύριους αντιπάλους στον συγκεκριμένο τομέα (π.χ. Google, Yahoo!, Microsoft, Multimap και Map24), προκειμένου να εξασφαλιστεί η εικόλια αλλαγής provider ανά πάσα στιγμή.

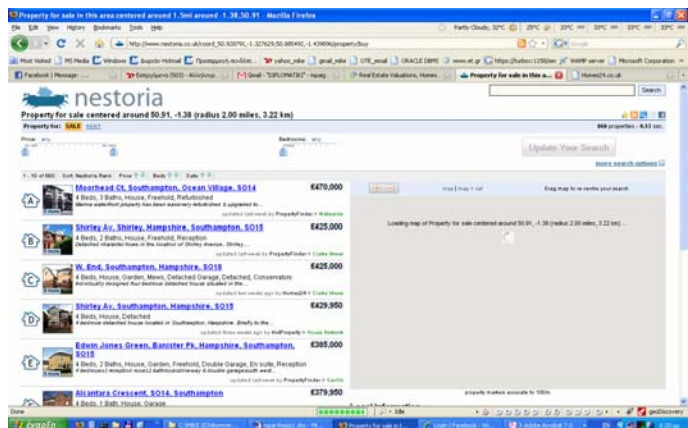


Εικόνα 15 Αρχική σελίδα nestoria



Εικόνα 16 Σελίδα αναζήτησης Nestoria

Αξίζει να σημειώσουμε ότι όπως φαίνεται και στη Εικόνα 17 Σελίδα αναζήτησης μετά από το μετακίνηση στο χάρτη και πάτημα του πλήκτρου update το interface δεν αξιοποιεί πλήρως τις δυνατότητες του AJAX. Γεωγραφική μετακίνηση στο χάρτη δεν ενημερώνει αυτόματα τα αποτελέσματα αναζήτησης. Απλά ενεργοποιείται κουμπί με τίτλο UPDATE στην κορυφή του χάρτη, το οποίο και ενημερώνει χάρτη και αποτελέσματα όπως φαίνεται παραπλεύρως:

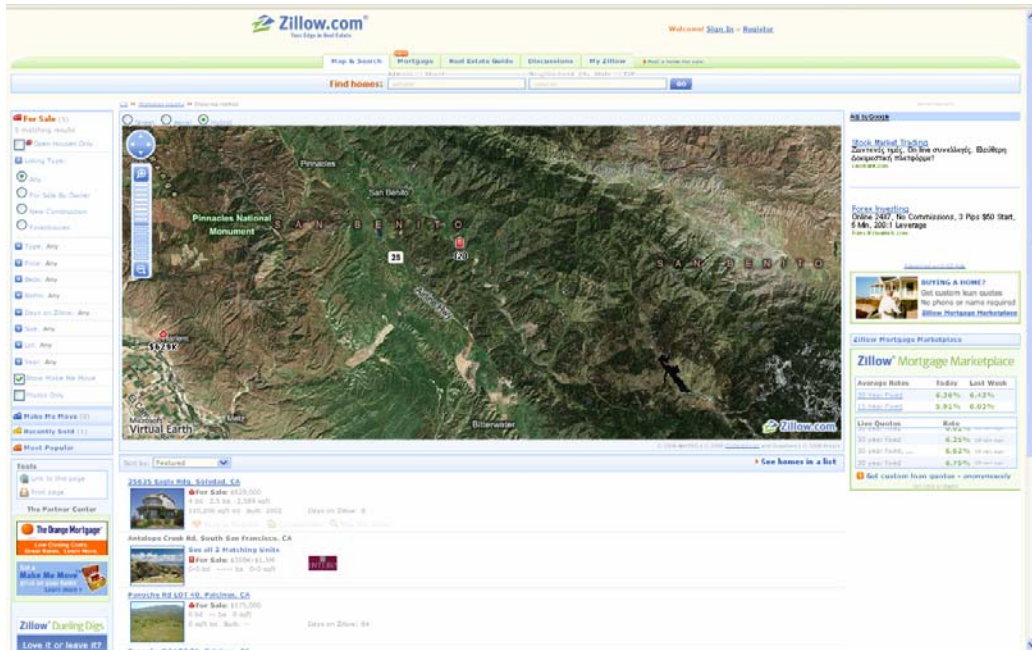


Εικόνα 17 Σελίδα αναζήτησης μετά από το μετακίνηση στο χάρτη και πάτημα του πλήκτρου update

Πολύ αξιοσημείωτο ο πίνακας με τα στατιστικά και τις τιμές ακινήτων κάτω από το χάρτη.

#### 4.4.1.3 <http://www.zillow.com>

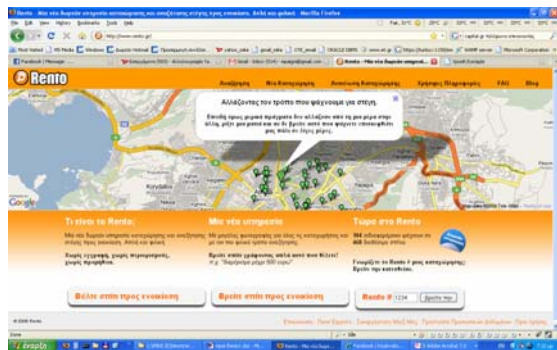
Άλλο ένα site που απευθύνεται στην Αμερικάνικη αγορά ακινήτων, με διάφορα εργαλεία και πληροφορίες για δάνεια, ασφάλεια κατοικίας, και άλλες συμβουλές σχετικές με την αγορά ακινήτων. Έχει στηριχτεί στο Microsoft Virtual Earth. Το zillow διαθέτει και APIs για στατιστικά για τη χρήση σε άλλες ιστοσελίδες - δυστυχώς τα στατιστικά είναι για την Αμερικάνικη αγορά



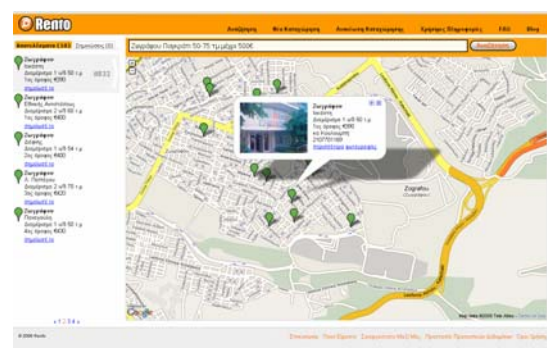
Εικόνα 18 Zillow.com

#### 4.4.1.4 www.rento.gr

Πολύ αξιόλογη ελληνική προσπάθεια. Περιορίζεται στην αγορά ενοικιάσεων και όχι σε αγοραπωλησίες (προφανώς για να αποφύγει την διένεξη με τους μεσίτες, οι οποίοι αξιοποιούν άλλα μέσα όπως θα αναφερθεί και στην παράγραφο §4.4.1.5 στη σελίδα 52).



Εικόνα 19 Rento.gr



Εικόνα 20 Rento.gr σελίδα αποτελεσμάτων

Ομοίως και εδώ υπάρχει το ίδιο πρόβλημα με την μετακίνηση επί του χάρτη – δεν ενημερώνονται τα αποτελέσματα. Προβληματάκια με τη χρήση του back button και αποθήκευσης url. Πολύ χρήσιμη η δυνατότητα σημειωματαρίου. Η περιήγηση στο site είναι δωρεάν. Χρεώνεται μόνο η φωτογράφιση σε περίπτωση που κάποιος δεν αναρτήσει τις φωτογραφίες του μόνος του. Η εταιρεία που το ανέπτυξε είναι η Sense & Funtasia και στηρίχτηκε σε τεχνολογία :

- \* Ruby on Rails με επεκτάσεις σε C++ για τη server side λογική
- \* Το frontend σε javascript
- \* Map rendering μέσω του Google Maps
- \* Χρήση του .NET Compact Framework και για βάση δεδομένων την PostgreSQL
- \* Debian Linux servers και OpenBSD firewalls



Εικόνα 21 Δωρεάν νέες καταχωρήσεις

#### 4.4.1.5 [www.xe.gr](http://www.xe.gr) -Εφημερίδα «ΧΡΥΣΗ ΕΥΚΑΙΡΙΑ»

Πρόκειται για αναβαθμισμένη έκδοση του site της εφημερίδας μικρών αγγελιών. Αξίζει να τονιστεί ότι στις εφημερίδες μικρών αγγελιών το μεγαλύτερο ποσοστό προέρχεται από μεσίτες που δεν εμφανίζουν ούτε φωτογραφίες ούτε ακριβή διεύθυνση προκειμένου να αποφύγουν την απευθείας επικοινωνία πωλητή και ενδιαφερόμενου. Εξάλλου η πληρωμή των μεσιτών ανέρχεται περίπου στο 4% της αγοραίας αξίας και στην πλειοψηφία των περιπτώσεων δεν εμφανίζεται με νόμιμα έγγραφα (φοροδιαφυγή). Συνεπώς οι εφημερίδες δεν έχουν άμεσο όφελος από την απεικόνιση της ακριβούς θέσης και των ακριβών στοιχείων ενός ακινήτου. Βέβαια πρέπει να σημειωθεί ότι υπάρχει προσπάθεια χρήσης γεωγραφικής απεικόνισης με το Google maps κάτι που μέχρι πριν μερικούς μήνες ήταν ανύπαρκτο.



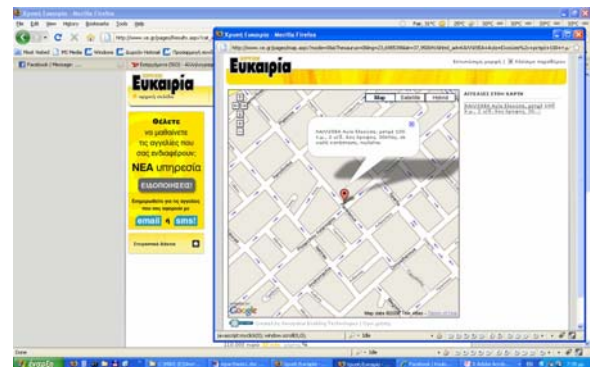
Εικόνα 22 Επιλογή περιφέρειας Ελλάδος



Εικόνα 23 Επιλογή λοιπών χαρακτηριστικών κριτηρίων αναζήτησης ακινήτου



Εικόνα 24 Αποτελέσματα αναζήτησης Ελλάδος



Εικόνα 25 Απεικόνιση ενός αποτελέσματος στο χάρτη της Αθήνας

#### 4.4.1.6 *www.realsophy.com*

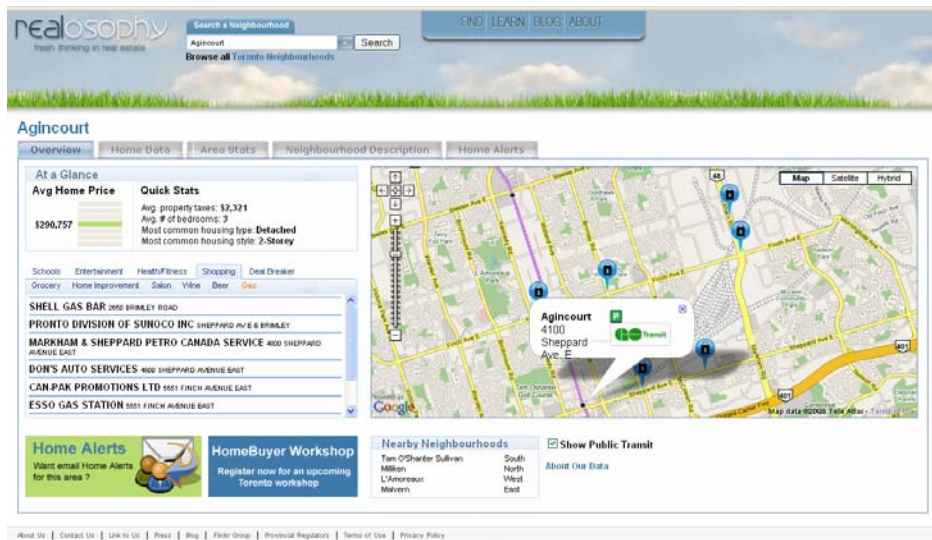
Το [www.realsophy.com](http://www.realsophy.com) στοχεύει περισσότερο στην ανεύρεση γειτονιάς και όχι τόσο στην επιλογή σπιτιού. Διαθέτει πλήθος στατιστικών στοιχείων όπως κατάταξη σχολείων, στατιστικά στοιχεία μαθητών, τοπική εγκληματικότητα, και πλήθος γεωγραφικών στοιχείων όπως σχολεία, βενζινάδικα, διασκέδαση, starbucks κλπ. Και αυτό χρησιμοποιεί Google Maps API.



Εικόνα 26 realosophy.com Επιλογή γειτονιάς



Εικόνα 27 Στατιστικά στοιχεία γειτονιάς



Εικόνα 28 Χάρτης Realosophy με σημεία ενδιαφέροντος



# 5

## *Πρότυπη Εφαρμογή: E-MESITIS*

### *5.1 Εισαγωγή*

Στο παρόν κεφάλαιο εστιάζουμε στα τεχνικά θέματα – επιλογές του σχεδιασμού και της υλοποίησης της πρότυπης εφαρμογής με τίτλο *E-MESITIS*. Στόχος της υλοποίησης αυτής ήταν η αξιολόγηση της δυσκολίας αξιοποίησης γεωγραφικών μεταδεδομένων και αναζήτησης ακινήτων ή αγροτεμαχίων προς πώληση με απευθείας αναζήτηση σε χάρτη χωρίς τη μεσολάβηση μεσάζοντα (μεσίτη) με διαδραστικό τρόπο.

### *5.2 E-MESITIS Βασική εφαρμογή*

Η παρούσα διαδικτυακή εφαρμογή υλοποιήθηκε με σκοπό την απόδειξη (α) της εφικτότητας και (β) της ωφέλειας από την αναζήτηση γεωσυσχετισμένων ακινήτων / αγροτεμαχίων. Παρόλο που παγκοσμίως υπάρχει πληθώρα σχετικών εφαρμογών, στην Ελλάδα υπάρχει αποσπασματική κάλυψη είτε της αγοράς ενοικιάσεων είτε της αγοράς των μικρών μαγαζιών. Δυστυχώς δεν υπάρχει κατάλογος με τα υπονήφια προς πώληση /ενοικίαση ακίνητα όπως είναι η craigslist στην Αμερική, ώστε να διευκολύνει το μαζικό data entry και να δημιουργήσει την κρίσιμη μάζα ακινήτων, που θα προσελκύσουν το κοινό και θα αξιοποιήσουν μεθόδους διάδοσης στόμα-με-στόμα προκειμένου να αποφευχθεί η μεγάλη διαφημιστική δαπάνη. Τεχνολογικά αποτελεί μια υβριδική διαδικτυακή εφαρμογή, δηλαδή ένα mashup<sup>12</sup>

---

<sup>12</sup> Ο όρος *Mashup* αναφέρεται σε μια διαδικτυακή εφαρμογή που συνδυάζει στοιχεία από περισσότερες από μια πηγές για την δημιουργία ενός νέου ενιαίου εργαλείου. Ο όρος *Mashup*, πρωτοχρησιμοποιήθηκε για να χαρακτηρίσει παραλλαγές μουσικών κομματιών που συνθέτονταν από περισσότερα του ενός τραγούδια.

Ως mashup αναφέρεται στη συγχώνευση περιεχομένου από διαφορετικές πηγές, χρησιμοποιώντας το Διαδίκτυο. Πιο συγκεκριμένα, η εφαρμογή έχει σχεδιαστεί να αντλεί χάρτες από την Google Maps, ακίνητα με όλα τα στοιχεία τους από τη βάση δεδομένων που αναπτύξαμε και στοιχεία αναζήτησης από την Google. Στην πράξη, οι χάρτες από την Google Maps συγχωνεύονται με τα στοιχεία από τις προαναφερθείσες πηγές σε έναν ενιαίο χάρτη.

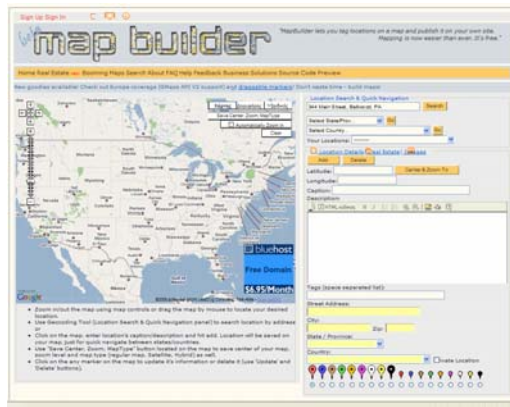
## **5.3 Λογική Αρχιτεκτονική & Εργαλεία Ανάπτυξης**

### **5.3.1.1 Λογική Αρχιτεκτονική**

Η ανάπτυξη της εφαρμογής στηρίχθηκε στα ευρέως αποδεκτά παραδείγματα καλής πρακτικής για την ανάπτυξη mashups και ειδικότερα, σε τεχνικές προγραμματισμού AJAX (Asynchronous JavaScript and XML) οι οποίες παρουσιάζονται στη συνέχεια της ενότητας. Σύμφωνα με τα παραπάνω, η επιχειρηματική ευφυΐα της εφαρμογής υλοποιείται τόσο στον εξυπηρετητή (στην περίπτωση μας PHP) όσο και στον browser (JavaScript), ενώ οι πηγές δεδομένων είναι ο κεντρικός εξυπηρετητής (ORACLE Spatial 11g), όσο και το διαδίκτυο (Google Maps). Η server-side πλευρά υλοποιήθηκε με χρήση της PHP και ORACLE γιατί:

- Η PHP "συνεργάζεται" άριστα με βάσεις δεδομένων που υποστηρίζουν ερωτήματα SQL, όπως είναι οι MySQL, Microsoft SQL server, PostgreSQL, Oracle κ.ά. Αυτό επιτρέπει τη δημιουργία αξιόλογων εφαρμογών που επιτρέπουν την αποθήκευση δεδομένων και ανταλλαγή τους μεταξύ online χρηστών.
- Η Oracle αποτελεί μια επαγγελματική λύση με τεράστιες δυνατότητες σε χωρικούς τελεστές, τις οποίες δεν διαθέτουν Open source Λύσεις. Ειδικότερα, η ανταγωνιστική MySQL η πλέον διαδεδομένη δικτυακή ΒΔ, παρόλο που υποστηρίζει τύπο δεδομένων geometry (POINT, LINESTRING, POLYGON), αλλά και μερικές χωρικές συναρτήσεις όπως StartPoint(ls), Intersection(g1,g2), Difference(g1,g2) κ.ά.. δεν διαθέτει ακόμα συναρτήσεις για τον έλεγχο χωρικών σχέσεων μεταξύ των γεωμετριών Distance() και Related(). Επειδή η απόσταση ήταν καθοριστικής σημασίας ερώτημα για την εύρεση ακινήτου (απόσταση από άλλο σημείο ενδιαφέροντος που ορίζει ο χρήστης) επελέγη η Oracle 11g. Η διανομή είναι δωρεάν εφόσον χρησιμοποιείται για ερευνητικούς λόγους.
- Για το rendering των χαρτών αλλά και τα υπόβαθρα χρησιμοποιήθηκε το GOOGLE MAPS API. Να σημειώσουμε ότι διερευνήσα τη χρήση wrapper APIs για το rendering

των χαρτών. Τέτοια είναι τα Openlayers και MapBuilder<sup>13</sup>. Την εποχή που αποφάσιζα τα εργαλεία ανάπτυξης το Mapbuilder ανακοίνωσε νέα έκδοση στις 20/12/2007 και έκτοτε κάνει rendering με το Openlayers. Τα Openlayers δυστυχώς δεν είχαν το ίδιο όμορφο αισθητικά αποτέλεσμα. Αξίζει να αναφέρουμε ότι το MapBuilder διαθέτει με πληρωμή API εξειδικευμένο στην αγορά του Real Estate MapBuilder™STANDARD συνδρομή κοστίζει \$19.95/month και περιορίζεται σε 500 ενεργές τοποθεσίες. Επέλεξα τελικά το ίδιο το Google Maps API εξαιτίας της πολύ μεγάλης βάσης χρηστών που διαθέτει μιας και το Openlayers ήταν ακόμα σε στάδιο ανάπτυξης.



Εικόνα 29 Map Builder

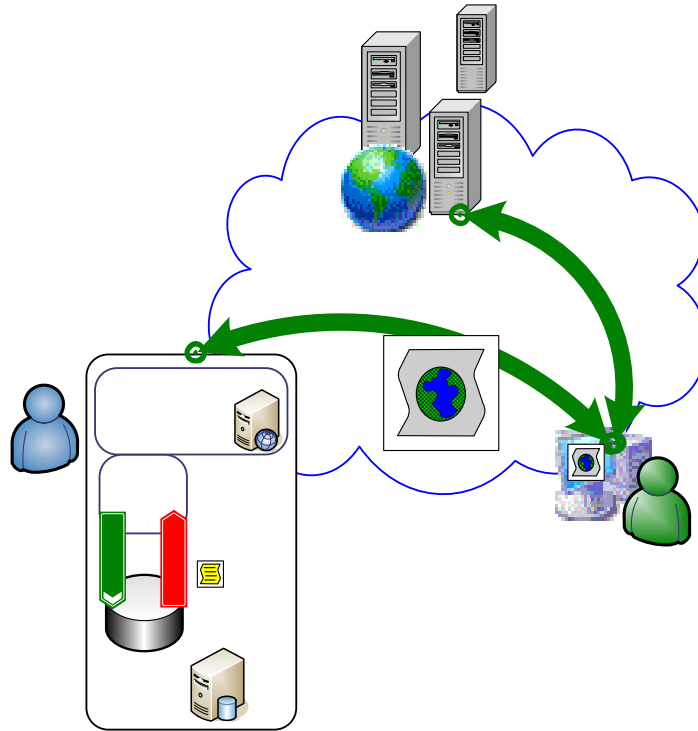


Εικόνα 30 Openlayers

Το Openlayers είναι μια JavaScript βιβλιοθήκη για την παρουσίαση χαρτών χωρίς την εξαρτήσεως τύπου server-side. OpenLayers υλοποιεί ένα JavaScript API για την κατασκευή πλούσιων web-based geographic applications, παρόμοιων με τα Google Maps και MSN Virtual Earth APIs, με την ουσιαστική διαφορά ότι είναι δωρεάν και αναπτύσσεται από την Open Source κοινότητα λογισμικού. Τα OpenLayers υλοποιούν καθιερωμένες μεθόδους πρόσβασης σε χωρικά δεδομένα, όπως τα OpenGIS Consortium's Web Mapping Service (WMS) και Web Feature Service (WFS) πρωτόκολλα. Χρησιμοποιεί τμήματα από Prototype.js και την Rico βιβλιοθήκη. Τα OpenLayers διαχωρίζει τα χαρτογραφικά εργαλεία από τα χωρικά δεδομένα επιτρέποντας στα εργαλεία να λειτουργούν ανεξαρτήτως πηγής προέλευσης των δεδομένων και καταργώντας τα «proprietary silos».

Στη συνέχεια παρουσιάζεται η λογική αρχιτεκτονική της εφαρμογής που έχει αναπτυχθεί, στην οποία αποτυπώνονται αφενός οι τεχνολογίες που έχουν αναπτυχθεί για την ανάπτυξη του συστήματος και αφετέρου η πολυπλοκότητα της υλοποίησης.

<sup>13</sup> <http://www.openlayers.org/> και <http://www.mapbuilder.net/index.php>



**Σχήμα 1: Λογική αρχιτεκτονική**  
(εγγεγραμμένος χρήστης)

### 5.3.1.2 Εργαλεία Ανάπτυξης

Αναλυτικότερα, για την ανάπτυξη της εφαρμογής χρησιμοποίησα τα ακόλουθα:

- Βάση δεδομένων ORACLE Spatial 11g, στην οποία αποθηκεύονται τόσο σχεσιακές όσο και χωρικές πληροφορίες των ακινήτων προς πώληση / ενοικίαση. Έχουμε αξιοποιήσει χωρικούς τελεστές που προσφέρει η Oracle.
  - `SDO_GEOM.SDO_DISTANCE (point1.shape, pointB.shape, 100) <$.BUFFER_METERS )` αλλά και
  - `SDO_RELATE(pointB.shape, area1.shape, 'MASK=INSIDE+COVEREDBY')='TRUE'`

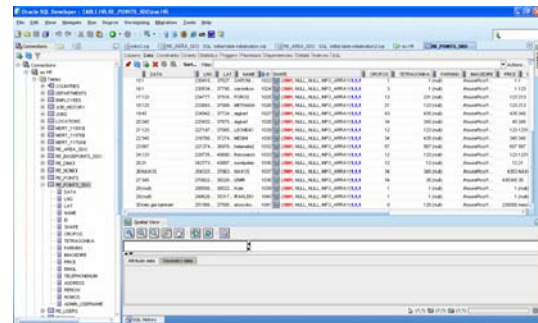
Για το χειρισμό της βάσης – δημιουργία tables, έλεγχος ορθότητας queries, επαλήθευση αποτελεσμάτων της PHP, εισαγωγή dummy στοιχείων κλπ χρησιμοποίησα το Oracle SQL Developer 1.5.0.53 μια Java based εφαρμογή για τη διαχείριση της Oracle. Για το καθημερινή συντήρηση της βάσης χρησιμοποίησα το Enterprise Manager 11g (μέσα από το web interface - <https://turbox:1158/em> όπου turbox =localhost και 1158 η TCP πόρτα

**Admin**

που ακούει ο listener της βάσης του EMESITI). Η έκδοση της Java(TM) Platform που χρησιμοποιήσα ήταν η JDK 1.5.0\_06.

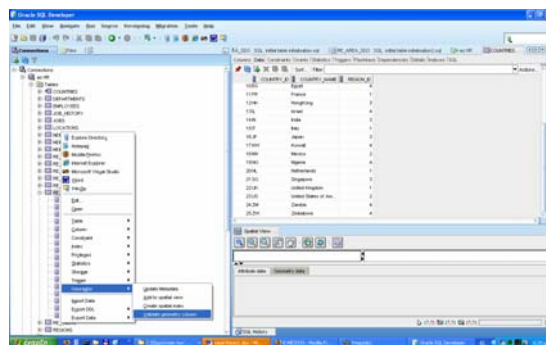


Εικόνα 31 Enterprise Manager



Εικόνα 32 SQL Developer

Σε αυτό το σημείο να αναφέρουμε το πολύ καλό add-on του SQL Developer με όνομα Georaptor (<http://sourceforge.net/projects/georaptor/>) το οποίο δίνει κάποιες GIS δυνατότητες οπτικοποίησης και γεωμετρικών ελέγχων μέσα στο ίδιο το εργαλείο ανάπτυξης SQL. Ίσως το πιο σημαντικό είναι γραφικό “Validation” των στηλών με τα SDO objects (σημεία, γραμμές, πολύγωνα). Δυστυχώς όπως φαίνεται και παρακάτω στην Εικόνα 33 στην νέα έκδοση του SQL Developer δεν λειτουργεί ο χάρτης και δυστυχώς έχει σταματήσει η ενημέρωση του georaptor από τις αρχές του 2007.



Εικόνα 33 Georaptor add-in στο SQL Developer Oracle.

- PHP Version 5.2.5, για τη διασύνδεση με τη βάση δεδομένων, την εκτέλεση επερωτήσεων και την παραγωγή αποτελεσμάτων XML για ενσωμάτωση σε μέρη της σελίδας που χρειάζονται τροποποίηση. Να τονίσω ότι δεν ανέπτυξα PHP σελίδες αλλά όλος ο κώδικας υπάρχει ως HTML αφού τα θέματα ασφαλείας μιας πειραματικής εφαρμογής στην παρούσα φάση δεν ήταν τόσο απειλητικά.

PHP Version 5.2.5	
System	Windows NT TURBO5.1 build 2600
Build Date	Dec 11 2007 17:00:32
Configure Command	escrpt /nologo configure --enable-openssl --enable-pdftoc C:\cygwin\bin\build core-php-nts-5.2.5.2916 /core-php-nts-5.2.5.2916 --with-openssl=C:\cygwin\bin\build core-php-nts-5.2.5.2916 /core-php-nts-5.2.5.2916 --with-extra-include=C:\cygwin\bin\build core-php-nts-5.2.5.2916 /core-php-nts-5.2.5.2916 --with-extra-lib=C:\cygwin\bin\build core-php-nts-5.2.5.2916 /core-php-nts-5.2.5.2916 --disable-ipv6 --disable-bcmath --disable-calendar --disable-ftp --without-gd --disable-odbc --without-sqlite --disable-tk --with-iconv --with-krnl --disable-force-cgi-redirect --enable-fastcgi --enable-ipv6 --enable-pdftoc --disable-xml --disable-xmlwriter
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configure Alias (php.ini Path)	C:\WINDOWS
Loaded Configure File	C:\Program Files\ZendCore For Oracle\etc\php.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	disabled
Registered PHP Streams	ftp, ftps, ftps, ftp, data, http, ftp, compress.zlib
Registered Streams Sockets	tcp, udp, ssl, ssh2, ssh2, fs
Transports	

Directive	Local Value	Master Value
mysql.default_host	no value	no value
mysql.default_port	3306	3306
mysql.default_pw	no value	no value
mysql.default_socket	no value	no value
mysql.default_user	no value	no value
mysql.max_links	Unlimited	Unlimited
mysql.reconnect	Off	Off

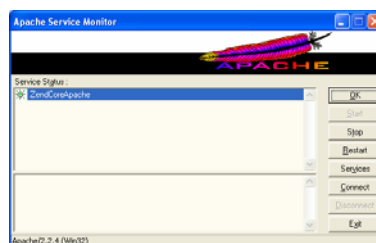
oci8		
OCI8 Support	enabled	
Version	1.2.3	
Revision	\$Revision: 1.260.2.16.2.30 \$	
Active Persistent Connections	0	
Active Connections	0	
Temporary Lob support	enabled	
Collections support	enabled	

Directive	Local Value	Master Value
oci8.default_prefetch	10	10
oci8.max_persistent	-1	-1
oci8.old_oci_close_semantics	0	0
oci8.persistent_timeout	-1	-1
oci8.ping_interval	60	60
oci8.privileged_connect	Off	Off
oci8.statement_cache_size	20	20

Εικόνα 34 PHP configuration με την αντίστοιχη βιβλιοθήκη oci8 για κλήσεις Oracle

- Apache Web Server, ο οποίος αναλαμβάνει την εκτέλεση των PHP σελίδων και τη φιλοξενία του διαδικτυακού τόπου. Ελλείπει δεύτερου υπολογιστή και για να μην προβώ σε εγκατάσταση εικονικού μηχανήματος με virtualization (Virtual PC της Microsoft ή Virtual Box της SUN ή λύση της VMWARE) χρησιμοποίησα τη δυνατότητα localhost με χρήση loopback network adapter. Με αυτόν τον τρόπο δημιουργήθηκε μια δικτυακή υπηρεσία, όπου server και client συγκατοικούν στο ίδιο μηχανήμα. Ως εξυπηρετητής χρησιμοποιήθηκε ο Apache HTTP Server, στην έκδοση 2.2.3. Ο Apache είναι ένας cross-platform open source HTTP server, ο οποίος, μεταξύ άλλων, είναι διαθέσιμος για Windows, UNIX και MacOS X. Αναπτύχθηκε από την Apache Software Foundation και διατίθεται δωρεάν στο διαδίκτυο στη σελίδα <http://httpd.apache.org/download.cgi>. Χρησιμοποίησα την υλοποίηση της ZEND εξαιτίας της άμογης συνεργασίας της με την ORACLE. Wamp server δεν συνεργαζόταν τόσο άμεσα. Για τον έλεγχο λειτουργίας του ήταν πολύ χρήσιμο το Apache Monitor.



Εικόνα 35 Apache monitor

- EMESITIS web site, που περιλαμβάνει τη διεπαφή που προσφέρεται προς τους χρήστες. Η εμφάνιση των ιστοσελίδων ελέγχεται μέσω κατάλληλου CSS που έχει αναπτυχθεί για το σκοπό αυτό και ενσωματώνεται στην αρχή του κυρίως html αρχείου. Για την

ανάπτυξή του δεν χρησιμοποιήθηκε κάποιο έτοιμο CMS (Content Management System τύπου Joomla, Drupal) ή άλλου λόγω της επιπλέον πολυπλοκότητας και του περιορισμένου χρόνου. Για την ενσωμάτωση έτοιμων λειτουργικών στοιχείων (components) χρησιμοποίησα τη βιβλιοθήκη με Javascript εργαλεία, που παρέχει η Yahoo με το όνομα Yahoo User Interface (YUI)<sup>14</sup>. Από τη βιβλιοθήκη έγινε χρήση τόσο του εργαλείου Multitab όσο και του εργαλείου autocomplete. Αντίστοιχη βιβλιοθήκη με εργαλεία διαθέτει και η google αλλά το αισθητικό αποτέλεσμα της Yahoo ήταν καλύτερο και η χρήση του δικού της multitab πιο εύκολη. Για να επιδείξω τη δυνατότητα χρήσης και ενσωμάτωσης εργαλείων από διαφορετικούς προμηθευτές χρησιμοποίησα το εργαλείο της Google<sup>15</sup> slideshow το οποίο χρησιμοποιεί AJAX calls της βιβλιοθήκης της Google για να φορτώσει νέες φωτογραφίες στο slideshow χωρίς να γίνει refresh της σελίδας.

```
<!-- Dependencies -->
<!-- Sam Skin CSS for TabView -->
<link rel="stylesheet" type="text/css" href="http://yui.yahooapis.com/2.5.2/build/tabview/assets/skins/sam/tabview.css">

<!-- JavaScript Dependencies for Tabview: -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.5.2/build/yahoo-dom-event/yahoo-dom-event.js"></script>
<script type="text/javascript" src="http://yui.yahooapis.com/2.5.2/build/element/element-beta-min.js"></script>

<!-- OPTIONAL: Connection (required for dynamic loading of data) -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.5.2/build/connection/connection-min.js"></script>

<!-- Source file for TabView -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.5.2/build/tabview/tabview-min.js"></script>

<script type="text/javascript">
var myTabs = new YAHOO.widget.TabView("demo");
</script>

<div id="demo" class="yui-navset">
  <ul class="yui-nav">
    <li class="selected"><a href="#tab1"><em>Tab One Label</em></a></li>
    <li><a href="#tab2"><em>Tab Two Label</em></a></li>
    <li><a href="#tab3"><em>Tab Three Label</em></a></li>
  </ul>
  <div class="yui-content">
    <div><p>Tab One Content</p></div>
    <div><p>Tab Two Content</p></div>
    <div><p>Tab Three Content</p></div>
  </div>
</div>
```

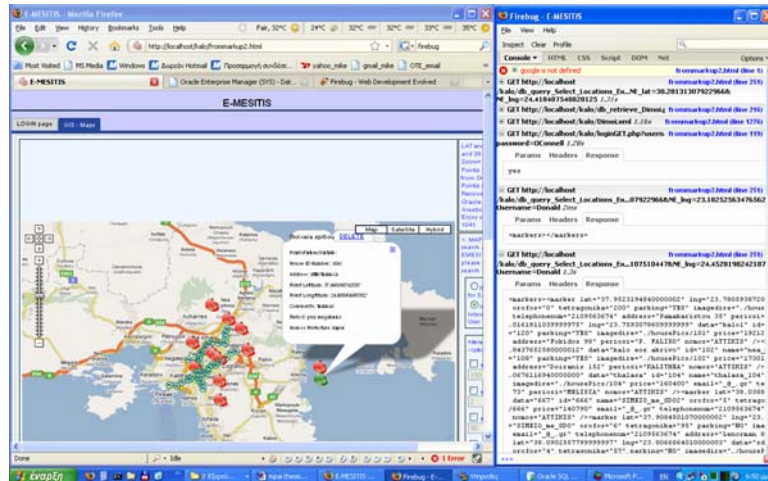
### Πίνακας 1 Javascript κώδικας για ενσωμάτωση του εργαλείου Yahoo Multitab σε ιστοσελίδα

- Web browser, στον οποίο εκτελείται ο JavaScript κώδικας των σελίδων που αναλαμβάνουν τη σύνδεση με το Google Maps και με άλλες πηγές δεδομένων, καθώς και γενικότερα με θέματα επεξεργασίας και εμφάνισης πληροφορίας (π.χ. XML parsing), έχει ελεγχθεί λειτουργικότητα τόσο με Mozilla 2, Mozilla 3 όσο και με Internet Explorer 6 και 7. Οι δυνατότητες εκτέλεσης Javascript ήταν ενεργοποιημένες. Για debugging για την Javascript αλλά και για απευθείας έλεγχο των αποτελεσμάτων της PHP

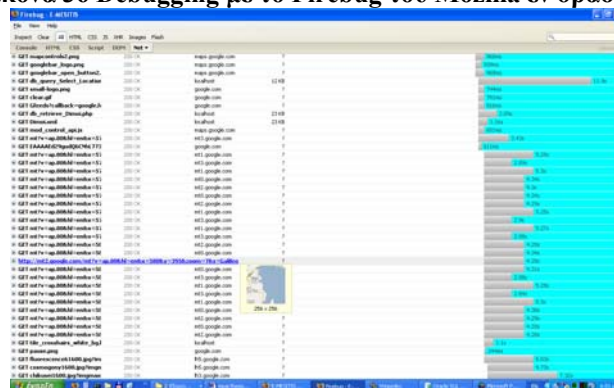
<sup>14</sup> <http://developer.yahoo.com/yui/>

<sup>15</sup> <http://www.google.com/uds/solutions/slideshow/sample.rss>

χρησιμοποίησα το FIREBUG add-on του Mozilla (<http://getfirebug.com/>) αλλά και το built-in εργαλείο του Mozilla error console. Ο συνδυασμός των δύο ήταν πολύ πιο εύχρηστος και διεισδυτικός σε σχέση με το εργαλείο, που διαθέτει ο Internet Explorer. Ενδεικτικά παραθέτω μερικές εικόνες και με χρόνους απόκρισης ανά component της ιστοσελίδας μας.



**Εικόνα 36 Debugging με το Firebug του Mozilla εν δράσει...**



**Εικόνα 37 Χρόνοι απόκρισης με το Firebug του Mozilla εν δράσει...  
χμμμ έχουμε καθυστερήσεις**

- Πηγές δεδομένων, όπως το Google Maps, οι οποίες ανάλογα με το API που προσφέρουν, ενσωματώνονται είτε στον client, είτε στον server. Τα υπόβαθρα με τους χάρτες και τα layers, που επιλέγει ο χρήστης παρέχονται απευθείας από την Google χωρίς παρέμβαση του server μας. Η ενσωμάτωσή τους γίνεται στον client (client side ενσωμάτωση). Αντίθετα τα δεδομένα με τα ακίνητα παρέχονται από την Oracle (server side ενσωμάτωση)



## 5.4 Δομή Βάσης Δεδομένων

### 5.4.1.1 Πίνακες της Βάσης



Όλοι οι πίνακες και ορισμοί συσχετίσεων υπάρχουν στο συνοδευτικό μαγνητικό μέσο (δισκέτα για περιορισμό των υπό εκτύπωση σελίδων). Εδώ παραθέτουμε τα σημαντικότερα πεδία. Μετά την παράθεση των πεδίων των πινάκων ακολουθεί σύντομη επεξήγηση.

**Πίνακας RE\_AREA\_SDO** με πεδία DATA (VARCHAR 1024Bytes NOT NULL), NAME (VARCHAR VARCHAR 1024Bytes NOT NULL), ID (NUMBER ακρίβειας 32 NOT NULL) και πεδίο SHAPE (SDO\_GEOMETRY από το schema MDSYS). Έχω ορίσει ως index το πεδίο SHAPE RE\_AREA\_SDO\_GI. Πρωτεύον κλειδί το id.

**Πίνακας RE\_BASEPOINTS\_SDO** με πεδία: DATA {VARCHAR2(1024 CHAR)}, LNG{NUMBER(38,0)}, LAT{NUMBER(38,0)}, NAME{VARCHAR2(1024 CHAR)}, ID{NUMBER(32,0) NOT NULL}, SHAPE{SDO\_GEOMETRY}, ADDRESS {VARCHAR2(1024 CHAR)}, PERIOXI {VARCHAR2(255 CHAR)}, NOMOS {VARCHAR2(255 CHAR)}, ADMIN\_USERNAME{VARCHAR2(20 BYTE)}. Το πεδίο SHAPE έχει οριστεί ως SPATIAL INDEX. Πρωτεύον κλειδί το id.

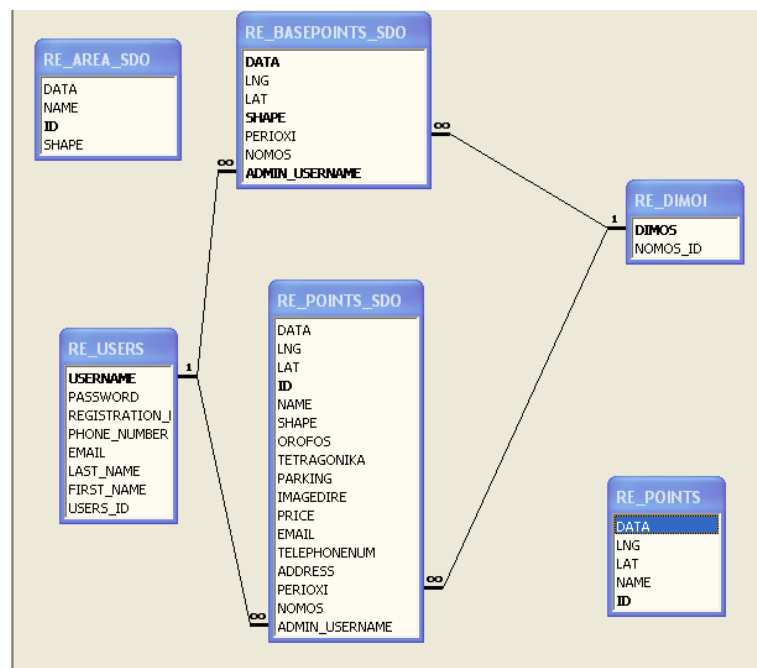
**Πίνακας RE\_DIMOI** με πεδία: ID{NUMBER(38,0)}, DIMOS{VARCHAR2(40 BYTE)}, NOMOS { VARCHAR2(40 BYTE)}

**Πίνακας RE\_POINTS** με πεδία: DATA{VARCHAR2(255 BYTE)}, LNG{NUMBER(32,0)}, LAT{NUMBER(32,0)}, NAME{VARCHAR2(255 BYTE)}, ID{NUMBER(32,0)}. Όλα είναι χαρακτηρισμένα ως NOT NULL.

**Πίνακας RE\_POINTS\_SDO** με πεδία: DATA{VARCHAR2(1024 CHAR)}, LNG{NUMBER(38,0)}, LAT{NUMBER(38,0)}, NAME{VARCHAR2(1024 CHAR)}, ID{NUMBER(32,0), NOT NULL}, SHAPE{SDO\_GEOMETRY}, OROFOS{NUMBER(38,0)}, TETRAGONIKA {NUMBER(32,0)}, PARKING{VARCHAR2(20 BYTE)}, IMAGEDIRE{VARCHAR2(1024 CHAR)}, PRICE{NUMBER(32,0)}, EMAIL{VARCHAR2(100 BYTE)}, TELEPHONENUM{NUMBER(15,0)}, ADDRESS{VARCHAR2(1024 CHAR)},

PERIOXI{VARCHAR2(255 CHAR)}, NOMOS{VARCHAR2(255 CHAR) },  
 ADMIN\_USERNAME{VARCHAR2(20 BYTE)},

**Πίνακας RE\_USERS** με πεδία: USERNAME{VARCHAR2(20 BYTE)},  
 PASSWORD{VARCHAR2(20 BYTE)}, REGISTRATION\_DATE{VARCHAR2(20 BYTE)},  
 PHONE\_NUMBER{VARCHAR2(20 BYTE)}, EMAIL{VARCHAR2(20 BYTE)},  
 LAST\_NAME{VARCHAR2(20 BYTE)}, FIRST\_NAME{VARCHAR2(20 BYTE)},  
 USERS\_ID{NUMBER(20,0)}. Όλα τα πεδία είναι NOT NULL. Ο πίνακας γέμισε με τα  
 παραδείγματα ονομάτων που διέθετε η ORACLE στο Tablespace Users HR:EMPLOYEES.



Εικόνα 38 Λιάγραμμα Συσχετίσεων Βάσης Δεδομένων

#### 5.4.1.2 Επεξηγήσεις Σχήματος Βάσης Δεδομένων

**Πίνακας (σχέση) RE\_POINTS\_SDO** αποθηκεύει τα ακίνητα προς πώληση μαζί με όλα τα στοιχεία τους (όροφο, τετραγωνικά, parking, ιδιοκτήτη, στοιχεία επικοινωνίας για το εν λόγω ακίνητο και γεωγραφικές συντεταγμένες τόσο σε αντικείμενο της ORACLE όσο και σε δύο στοιχεία τύπου Long για ευκολότερη διαχείρισή τους – σε κάθε περίπτωση οι χωρικές συναρτήσεις υλοποιούνται με τους χωρικούς τελεστές της ORACLE αλλά για όλες τις άλλες κλήσεις όπως παρουσίαση αποτελεσμάτων στην HTML σελίδα, χρησιμοποιούνται τα αριθμητικά πεδία, τόσο για λόγους ταχύτητας αλλά και επειδή η HTML δεν διαθέτει σύνθετο πεδίο τύπου σημείου. Τα στοιχεία επικοινωνίας του ιδιοκτήτη αναφέρονται εδώ γιατί ο ιδιοκτήτης μπορεί να

έχει δώσει συγκεκριμένο τηλ επικοινωνίας για τους ενδιαφερόμενους ενός ακινήτου διαφορετικού από το προσωπικό του τηλέφωνο ή το προσωπικό email, τα οποία και αποθηκεύονται στον πίνακα χρηστών.

Η εισαγωγή των στοιχείων γίνεται μέσω του db\_insert\_marker2.php script. Η διαγραφή αντίστοιχα (μόνο από τον ιδιοκτήτη που το εισήγαγε) γίνεται από το db\_delete\_location2.php. Το πεδίο data διατηρεί μια σύντομη περιγραφή με την οποία διαφημίζει το τρέχον ακίνητο ο ιδιοκτήτης του. Το name αποτελεί ένα όνομα που αποδίδει ο ιδιοκτήτης πχ «οροφδιαμέρισμα 3<sup>ο</sup> ορόφου Σπάρτης 12» και αποτελεί μέρος του υποψήφιου κλειδιού γιατί στο ίδιο γεωγραφικό σημείο ο ίδιος ιδιοκτήτης μπορεί να διαθέτει και έτερο ακίνητο π.χ. ο κατασκευαστής μιας πολυκατοικίας. Το υποψήφιο κλειδί αποτελείται από τα πεδία admin\_username, shape, name. Προκειμένου να απλοποιήσουμε τα πράγματα ορίσαμε αυτόματο μετρητή id τον οποίο και βάλαμε ως id (τύπος ακέραιος). Η ρουτίνα εισαγωγής φροντίζει να εντοπίσει την πλειάδα με το μεγαλύτερο ακέραιο ως id και να θέσει το id του νέου σημείου max\_id+1. Ο δεύτερος λόγος για τον οποίο οδηγήθηκα στη χρήση του id παρόλο που στο μοντέλο ΟΣ (Entity Relationships model) δεν είχε καθοριστεί τέτοια ανάγκη, ήταν το ότι για την αποθήκευση των φωτογραφιών έπρεπε να οριστεί ένα directory. Να σημειώσουμε εδώ ότι μετά από συζήτηση καταλήξαμε να μην υλοποιήσουμε τις φωτογραφίες ως blobs (δηλαδή αντικείμενα αποθηκευμένα εντός της ORACLE), αλλά η βάση να αποθηκεύει ένα Link (directory) της μορφής «./housePics/ID». Αυτό το string το αποθηκεύουμε στο πεδίο IMAGEDIRE.

Σχετικά με το πεδίο PERIOXI, υπήρχε η σκέψη για σύνδεση με τον πίνακα των Δήμων και αυτόματη εισαγωγή με χρήση του autocomplete javascript αντικειμένου της google. Όπως εξηγούμε και παρακάτω αυτό δεν επετεύχθη με αποτέλεσμα να μην υλοποιήσουμε τον περιορισμό αναφοράς (ξένο κλειδί) με το πρωτεύον κλειδί του πίνακα Δήμων. Η απλούστερη εκδοχή με όνομα RE\_POINTS προέκυψε από τη φάση ανάπτυξης του front end της εφαρμογής και προτού γίνει χρήση των χωρικών τελεστών και δυνατοτήτων της ORACLE SPATIAL. Η αντίστοιχη εισαγωγή γινόταν με το db\_insert\_marker.php script. Συνεπώς δεν θα μας απασχολήσει περαιτέρω.

**Πίνακας RE\_BASEPOINTS\_SDO** αποθηκεύει τα σημεία ενδιαφέροντος των χρηστών με βάση τα οποία θα εκτελεστούν τα όποια τυχόν ερώτημα εγγύτητας (π.χ. τόπος εργασίας, τόπος κατοικίας γονέων, σχολείο παιδιών, εξοχικό κλπ). Σε αυτόν τον πίνακα έχουν αποθηκευτεί και οι σταθμοί του METPO. Οι τελευταίοι έχουν σημαδευτεί με τέτοιο τρόπο ώστε να μην μπορούν να διαγραφούν από κανένα χρήστη. Η εισαγωγή των σημείων ενδιαφέροντος πραγματοποιείται από το db\_insert\_QueryPoints.php ενώ η διαγραφή από το db\_delete\_Baselocation2.php . Και εδώ

έχουμε την ίδια λογική που είχαμε και στον πίνακα RE\_POINTS\_SDO αναφορικά με την επιλογή των πεδίων και το πρωτεύον κλειδί και τα ξένα κλειδιά και δεν θα τα επαναλάβουμε.

**Πίνακας RE\_AREA\_SDO** αποθηκεύει τις γεωγραφικές διαστάσεις του ορθογώνιας γήινης επιφάνειας , που είναι ορατό στο παράθυρο του φυλλομετρητή του χρήστη. Το πεδίο data αποθηκεύει την default τιμή area\_extent όπως και το name το mixalis\_search και δεσμεύτηκαν για μελλοντική χρήση (cookie και όνομα χρήστη βλ.επε παρακάτω), ενώ το shape αποθηκεύει την οντότητα του ορατού παραθύρου. Προφανώς υποψήφιο κλειδί είναι ο συνδυασμός και των τριών. Για να απλουστεύσουμε το πρωτεύον κλειδί δημιουργήσαμε ένα πεδίο id (τύπου ακεραίου) το οποίο φροντίζουμε να το αυξάνουμε αυτόματα. Οι διαστάσεις αυτές χρησιμοποιούνται από το script db\_query\_Select\_Locations\_Extent.php κατά τη διαδικασία ανανέωσης της σελίδας μετά από μετατόπιση/μετακίνηση του χρήστη ώστε να εντοπιστούν μόνο τα ακίνητα εκείνα από τον πίνακα RE\_POINTS\_SDO που εμπίπτουν στο εν λόγω παράθυρο. Δυστυχώς επερώτηση με ad-hoc κατασκευή του πλαισίου εντός του επερωτήματος της db\_query\_Select\_Locations\_Extent.php χωρίς την ανάγκη για αποθήκευση στον πίνακα RE\_AREA\_SDO δεν λειτουργεί γιατί ο ορισμός του χωρικού τελεστή SDO\_RELATE απαιτεί το ένα όρισμα να είναι οπωσδήποτε γεωμετρικό στοιχείο πίνακα και το άλλο πίνακας ή προσωρινή οντότητα (transient instance) – σε διαφορετική περίπτωση εμφανιζόταν σφάλμα ODBC στον SQL\_DEVELOPER. Η κατασκευή του RE\_AREA\_SDO και αποθήκευση πρώτα της οντότητας extent και μετά εκτέλεση της επερώτησης ήταν μονόδρομος.

Προσοχή αξίζει να τονιστεί στο σημείο αυτό ότι στην παρούσα φάση δεν έχει υλοποιηθεί το db\_query\_Select\_Locations\_Extent.php ως ασφαλές transaction (εξαιτίας ακριβώς της παραπάνω αποθήκευσης του extent στο RE\_AREA\_SDO). Δεδομένου ότι το php script φροντίζει να σβήνει όλες τις προηγούμενες εγγραφές πριν αποθηκεύσει το νέο «ορατό» ορθογώνιο πλαίσιο του τρέχοντος επερωτήματος δεν καθόρισα κλειδί. Με αυτόν τον τρόπο εξηγείται γιατί υποψήφιο κλειδί είναι και τα πεδία cookie και username. Χωρίς την χρήση cookies προκειμένου να ταυτοποιηθεί το session του χρήστη (δεν αρκεί απλά το username για να ταυτοποιήσει το παράθυρο που εκκινεί το rendering του χάρτη), η εφαρμογή μας θα εμφανίσει πρόβλημα σε πολύ βαριά χρήση από πολλούς ταυτόχρονους clients. Όσα πειράματα έγιναν με παράλληλη χρήση / πρόσβαση χρηστών και από το ίδιο LAN δυστυχώς δεν ανέδειξαν κάποιο πρόβλημα. Σε κάθε περίπτωση πιθανή εμφάνιση λανθασμένου αποτελέσματος δεν οδηγεί σε κατάρρευση απλά απαιτεί μια νέα μικρομετακίνηση του χρήστη στο χάρτη ώστε να επανεκτελεστεί το επερώτημα και να ενημερωθούν σωστά τα εμφανιζόμενα αποτελέσματα. Συνεπώς δυσανεξία των χρηστών ίσως να εμφανίζεται σε βεβαρημένους servers. Παρόλ' αυτά μελλοντική εμπορική ανάπτυξη θα

πρέπει να λάβει υπόψη διεξοδικά stress tests με εργαλεία όπως το QA Wizard Pro. Λύση μπορεί να δοθεί με τροποποίηση του πίνακα RE\_AREA\_SDO και αποθήκευση random αλφαριθμητικού (cookie), που θα χαρακτηρίζει μονοσήμαντα το session (θα συνδέεται με cookie στον browser του χρήστη). Δεδομένου ότι κάτι τέτοιο ξεφεύγει από την παρούσα μεταπτυχιακή εργασία δεν διερευνήθηκε περαιτέρω. Για την επεξήγηση του γιατί υλοποίησα το επερώτημα με χρήση της SDO\_GEOM.SDO\_DISTANCE και όχι του SDO\_WITHIN\_DISTANCE operator δες την επόμενη παράγραφο §5.4.1.3.

Οι Πίνακες **DIMOI** και **REGIONS** έχουν αναπτυχθεί προκειμένου να λειτουργήσουν τα πεδία εισαγωγής (autocomplete), ώστε να περιορίσουμε την πιθανότητα λάθους κατά την εισαγωγή στοιχείων (πχ Δήμος Νέας Σμύρνης, και όχι Δημ. Ν.Σμύρνης ή Νέα Σμύρνη κλπ). Η λειτουργικότητα είναι έτοιμη απλά δεν προχωρήσαμε στην ενσωμάτωση στο user interface. Το javascript αντικείμενο *autocomplete* της Yahoo δεν κατόρθωσα να το κάνω να συνεργαστεί πολύ καλά με AJAX κλήσεις. Χρειαζόταν η παράθεση των πεδίων Inline στον κώδικα της HTML διογκώνοντας το κυρίως αρχείο, χωρίς να παρέχει πραγματική δυναμικότητα με Ajax κλήση (αντίστοιχη δυσκολία είχα και με το autocomplete της Google). Περαιτέρω διερεύνηση απαιτείται μελλοντικά..

#### 5.4.1.3 Χωρικοί Τελεστές Αναζήτησης

Το βασικό σημείο που διαφοροποιεί την εφαρμογή μας από τον «ανταγωνισμό» είναι όπως είπαμε η αυτόματη ενημέρωση των αποτελεσμάτων κάθε φορά, που ο χρήστης μετακινείται επί του χάρτη. Αυτό επιτυγχάνεται με χρήση κατάλληλης συνάρτησης σε Javascript που πυροδοτεί αντίστοιχο PHP script το οποίο εφαρμόζει το ορατό παράθυρο ως μάσκα φιλτραρίσματος των αποτελεσμάτων με χρήση του ακόλουθου: SDO\_RELATE(point1.shape, area1.shape, 'MASK=INSIDE+COVEREDBY')='TRUE', όπου area1 ο πίνακας RE\_AREA\_SDO και point1 ο πίνακας RE\_POINTS\_SDO. Αντίστοιχα για τα σημεία ενδιαφέροντος SDO\_RELATE(pointB.shape, area1.shape, 'MASK=INSIDE+COVEREDBY')='TRUE'.

Εκτός από τα διάφορα φιλτραρίσματα με βάση τιμή, ύψος ορόφου, parking κλπ έχουμε εφαρμόσει κριτήριο εγγύτητας με βάση συγκεκριμένη ζώνη επιρροής (buffer zone) από τα σημεία που μας δίνει ο χρήστης. Αξίζει να σημειώσουμε ότι δεν χρησιμοποίησα το SDO\_WITHIN\_DISTANCE γιατί όπως αναφέρεται στο “ORACLE Spatial User Guide and Reference 9.2 March 2002”:

```
SDO_WITHIN_DISTANCE(geometry1
MDSYS.SDO_GEOMETRY, aGeom MDSYS.SDO_GEOMETRY, params
VARCHAR2) ;
```

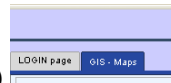
The **SDO\_WITHIN\_DISTANCE** operator is not suitable for performing spatial joins. That is, a query such as *Find all parks that are within 10 distance units from coastlines* will not be processed as an index-based spatial join of the COASTLINES and PARKS tables. Instead, it will be processed as a nested loop query in which each COASTLINES instance is in turn a reference object that is buffered, indexed, and evaluated against the PARKS table. Thus, the **SDO\_WITHIN\_DISTANCE** operation is performed *n* times if there are *n* rows in the COASTLINES table.

Αντ' αυτού οι Χωρικοί τελεστές που χρησιμοποιήσαμε είναι οι: **SDO\_GEOM.SDO\_DISTANCE** (point1.shape, pointB.shape, 100) <.\$BUFFER\_ METERS )

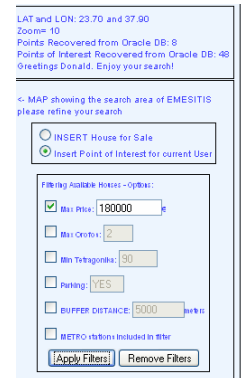
## 5.5 Οδηγός Χρήσης

Τα βασικά δομικά στοιχεία της εφαρμογής είναι τα ακόλουθα:

- Το multitab μενού (κατασκευασμένο με το YUI).



Εικόνα 39 Κυρίως χάρτης



Εικόνα 40 Επιλογές χρήστη

id	Name	Address	Orfos	m <sup>2</sup>	Parking	Price	Email	Telephone Number	County	Prefecture
102	nea_penteli_mazoneta	Dosiatis 152	2	108	YES	173010	@_gr	2109593674	KALITHEA	ATTIKIS
666	SIMEIO_ma_5202	Siggriva 140	5	85	NO	140790	@_gr	2109593674	KALITHEA	ATTIKIS
666	SIMEIO_ma_5203	Kilissas 85	4	57	NO	123653	@_gr	2109593674	MAKRISSI	ATTIKIS
672	SIMEIO_ma_5207	Adamas 2	0	75	NO	162075	@_gr	2109593674	NISSIA	ATTIKIS
666	32432	Pasaidonia 12	6	105	YES	157500	@_gr	2109593674	MENIDI	ATTIKIS
1012	ID_megallitris_aps1000	Pasaidonia 12	6	105	YES	157500	@_gr	2109593674	MENIDI	ATTIKIS
1027	alpinat	435	43	435		345	345	345	345	345
1042	karites	subdast	1	800		12000	pann@bar.gr	231312313	karites	Evia

Εικόνα 41 Πίνακας αποτελεσμάτων

Τα τρία αυτά στοιχεία διασυνδέονται δυναμικά μέσω κλήσεων AJAX (Javascript).

Η ιστοσελίδα της εφαρμογής απευθύνεται σε εγγεγραμμένους χρήστες οι οποίοι έχουν δικαίωμα αποθήκευσης /διαγραφής σημείων ενδιαφέροντος και εκτέλεσης αναζητήσεων διαθέσιμων ακινήτων στο χάρτη. Οι μη εγγεγραμμένοι χρήστες εμποδίζονται σε πλήρη με την υπέρθεση

σχετικού αδιαφανούς μηνύματος επί του χάρτη και την απενεργοποίηση όλων των επιλογών πλήκτρων. Η εγγραφή αυτομάτως απενεργοποιεί τα ανωτέρω «εμπόδια». Για την είσοδο-authentication δεν χρησιμοποίησα τις πιο ασφαλείς PUT μεθόδους αποστολής του password (SSL κλπ) δεδομένου ότι η εφαρμογή αποτελεί ένα απλό proof of concept. Η τροποποίηση θα ήταν πολύ μικρή.

Ο μη εγγεγραμμένος χρήστης μπορεί να ενημερωθεί σχετικά με την εφαρμογή, αλλά δεν λήφθηκε μέριμνα για την εγγραφή του, αφού κάτι τέτοιο ήταν εκτός πλαισίου της μεταπτυχιακής εργασίας στον τομέα της Γεωπληροφορικής. Δεδομένου ότι υπάρχουν έτοιμα Content Management Systems (CMS) όπως το πολύ γνωστό Joomla, Drupal κλπ δεν κρίθηκε σκόπιμο να αφιερωθεί χρόνος στο εικαστικό κομμάτι. Το κύριο μέλημα ήταν η δημιουργία της βασικής εφαρμογής. Η εικαστική παρέμβαση μπορεί να γίνει ανά πάσα στιγμή με ενσωμάτωση του παραχθέντος κώδικα σε ένα έτοιμο CMS σκελετό.



### 5.5.1.1 Μη εγγεγραμμένος χρήστης – επισκέπτης

Ο εγγεγραμμένος και μη εγγεγραμμένος χρήστης που επιλέγει να επισκεφτεί την ιστοσελίδα της εφαρμογής, αντικρίζει την ακόλουθη εικόνα:

The screenshot shows the E-MESITIS website interface. At the top, there is a navigation bar with 'LOGIN page' and '015 - Main'. The main content area features a map with several red markers and the text 'ΠΑΡΑΚΑΛΟΥΜΕ ΕΓΓΡΑΦΕΙΤΕ' (PLEASE REGISTER) repeated multiple times. To the right of the map, there is a search filter panel with options for 'PROPERTY NUMBER - TYPE', 'City Name', 'No. of Units', 'No. of Bedrooms', 'Price', 'Number of Floors', and 'BETWEEN: Minimum and Maximum'. Below the map and search panel, there is a table with the following columns: 'sq. ID', 'Name', 'Address', 'Ordnos', 'm<sup>2</sup>', 'Parking', 'Price', 'Email', 'Telephone Number', 'County', and 'Prefecture'. The table contains 10 rows of data.

sq. ID	Name	Address	Ordnos	m <sup>2</sup>	Parking	Price	Email	Telephone Number	County	Prefecture
0	επιμ_ε_αλλεσι	Falakas 35	0	200	YES	320000	_@_#	2109903074	ΑΝΑΤΟΛΙΑ	ΑΤΤΙΚΗΣ
1	επιμ	Falakas 68	3	120	YES	192120	_@_#	2109903074	Π. ΦΑΛΓΟ	ΑΤΤΙΚΗΣ
2	επιμ_επιμ_επιμ_επιμ_επιμ	Επίτιμοι 152	2	108	YES	173018	_@_#	2109903074	ΧΑΛΚΙΔΙΚΗ	ΑΤΤΙΚΗΣ
3	επιμ_επιμ_104	Αλεξάνδρ 75	1	100	YES	160400	_@_#	2109903074	ΜΕΛΙΣΣΑ	ΑΤΤΙΚΗΣ
4	επιμ_επιμ_επιμ_επιμ_επιμ	Σίγγοι 140	5	95	NO	140790	_@_#	2109903074	ΧΑΛΚΙΔΙΚΗ	ΑΤΤΙΚΗΣ
5	επιμ_επιμ_επιμ_επιμ_επιμ	Λαοκωπει 68	9	64	NO	209990	_@_#	2109903074	ΑΤΗΝΕΣ	ΑΤΤΙΚΗΣ
6	επιμ_επιμ_επιμ_επιμ_επιμ	Επίτιμοι 95	4	87	NO	123820	_@_#	2109903074	ΑΝΑΤΟΛΙΑ	ΑΤΤΙΚΗΣ
7	επιμ_επιμ_επιμ_επιμ_επιμ	Παλατινή 3	1	68	YES	208410	_@_#	2109903074	ΠΕΡΙΣΤΕΡΙΑ	ΑΤΤΙΚΗΣ
8	επιμ_επιμ_επιμ_επιμ_επιμ	Συριζοί & Ριζοί 1	90	YES	199480	_@_#	2109903074	ΡΗΓΑΣ	ΑΤΤΙΚΗΣ	
9	επιμ_επιμ_επιμ_επιμ_επιμ	Αδωνειδ 2	0	75	NO	162970	_@_#	2109903074	ΗΡΕΙΑ	ΑΤΤΙΚΗΣ

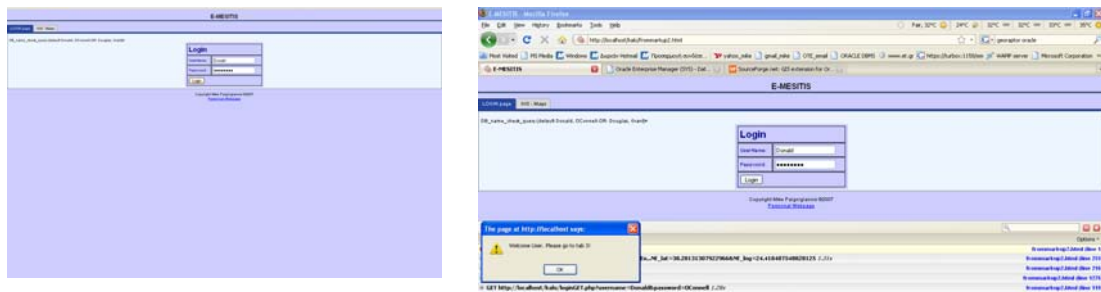
Σχήμα 2: Αρχική Σελίδα με αδιαφανή εικόνα

Έχουν απεικονιστεί με  τα σπίτια προς πώληση / ενοικίαση ενώ με  οι σταθμοί του METRO. Όλες οι εισαγωγές των σταθμών του METRO έγιναν χειροκίνητα με την λειτουργία εισαγωγής σημείων ενδιαφέροντος – θα αναφερθούμε στην συνέχεια. Απλά για τους σταθμούς έγινε χειροκίνητη παρέμβαση στη βάση ώστε να μην μπορεί να τους σβήσει κανένας χρήστης. Στο

κάτω μέρος της σελίδας κάτω από το χάρτη παρουσιάζεται με AJAX σε ένα div HTML στοιχεία δυναμικά περιεχόμενα του χάρτη. Μετακίνηση του χάρτη πυροδοτεί αυτόματα καινούρια αναζήτηση στο τρέχον ορατό πλαίσιο και ενημέρωση και στον πίνακα. Με αυτόν τον τρόπο επιλύσαμε το πρόβλημα δυσλειτουργίας που αντιμετωπίζαν όλα τα site, που είδαμε στα υπάρχοντα παραδείγματα.

### 5.5.1.2 Είσοδος εγγεγραμμένου επισκέπτη

Σε αυτό το σημείο να αναφέρουμε ότι η εγγραφή δεν συνεπάγεται την αποστολή κάποιου cookie με αποτέλεσμα ο browser να μην μπορεί να διατηρήσει τη σύνδεση σε περίπτωση εκτέλεσης F5 (refreshing ή κίνησης back-forward). Αυτό θα μελετηθεί σε κάποιο επόμενο στάδιο εμπορικής ανάπτυξης.



Σχήμα 3: Σελίδες εισόδου εγγεγραμμένου χρήστη με Firebug εν δράσει.


### 5.5.1.3 Εισαγωγή νέου ακινήτου προς πώληση εγγεγραμμένου χρήστη

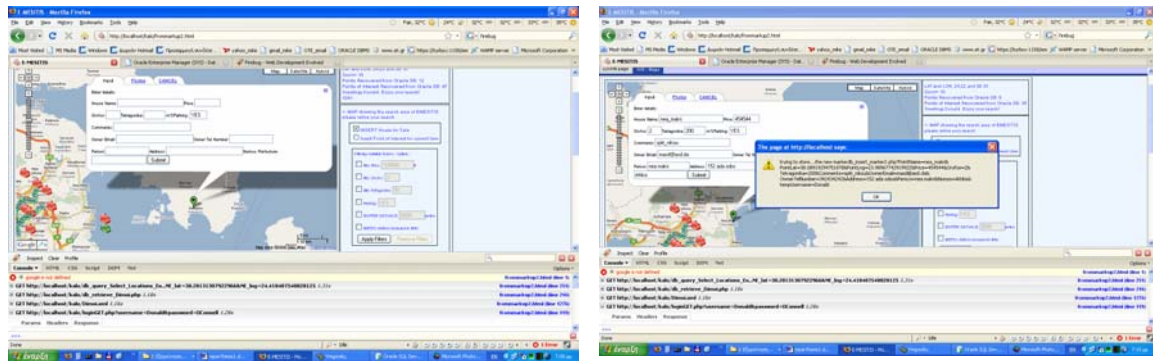
Επιλέγοντας το πρώτο radio button «Insert House for Sale» και κλικ πάνω στο χάρτη είναι δυνατή η αποθήκευση νέων ακινήτων προς πώληση. Υποχρεωτικά πεδία προς συμπλήρωση με διάφορα λεκτικά εμφανίζονται εντός του InfoWindow.



Εικόνα 42 Εικόνες ακινήτων ! Φωτοαγγελιές με ένα κλικ!

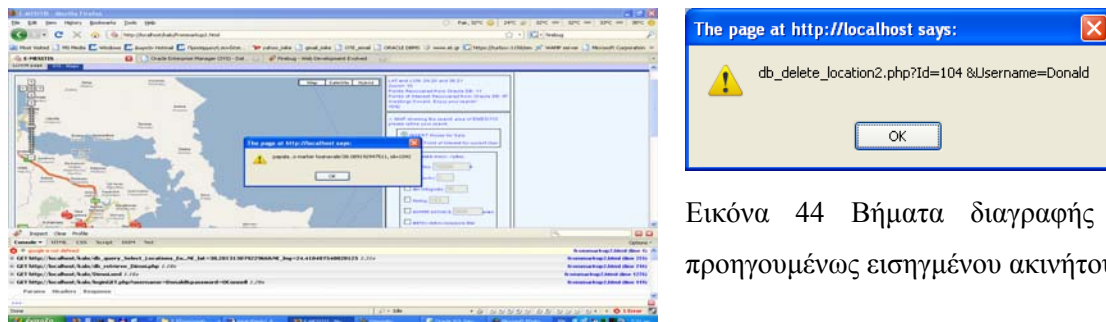


Έχει προβλεφθεί και δυνατότητα uploading φωτογραφιών. Επειδή δεν υπάρχει AJAX τρόπος για να εκτελεστεί το upload και για να αποφύγω το pop-up window αλλά απενεργοποίησα τη δυνατότητα, ώστε να προσαυτευτεί ο ιδιοκτήτης του χώρου από πιθανώς άσεμνο ή παραπλανητικό υλικό που θα φορτωθεί στη σελίδα.. Φωτογραφίες υπαρχόντων ακινήτων φαίνονται στο σχήμα Με το πράσινο δάκρυ απεικονίζεται αρχικά το υπό εισαγωγή νέο ακίνητο. Μετά την επιτυχή εισαγωγή των στοιχείων και την αποθήκευση το νέο ακίνητο απεικονίζεται και αυτό με .




Εικόνα 43 Βήματα καταχώρησης νέου ακινήτου προς πώληση στη Νέα Μάκρη

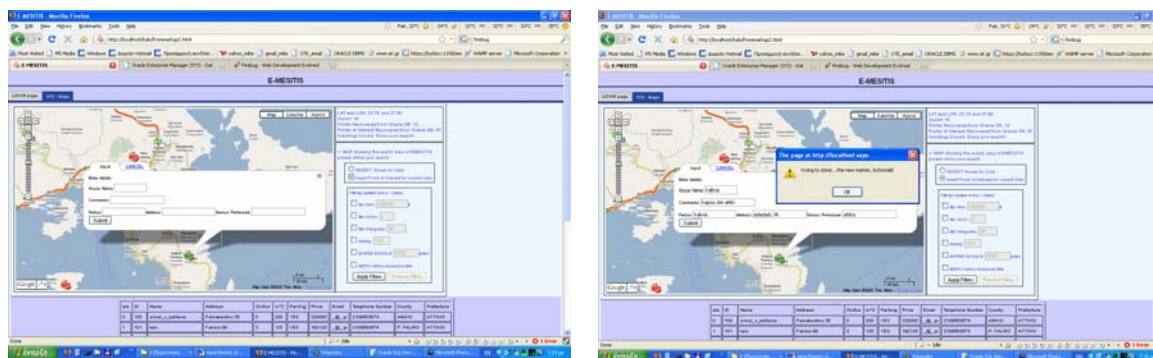
Να σημειώσουμε ότι ακίνητα προς πώληση που έχει εισάγει ο χρήστης μπορούν και να διαγραφούν από το αντίστοιχο μενού. Κάθε χρήστης έχει τα δικά του ακίνητα προς πώληση, που παρόλο που βλέπουν όλοι μόνο ο ίδιος/α μπορεί να διαγράψει.




Εικόνα 44 Βήματα διαγραφής του προηγουμένως εισηγμένου ακινήτου.

#### 5.5.1.4 Εισαγωγή σημείου ενδιαφέροντος γεωγραφμένου χρήστη

Με το πράσινο  απεικονίζονται τα σημεία ενδιαφέροντος, δηλαδή τα σημεία εκείνα με βάση τα οποία θα πραγματοποιηθούν οι χωρικές αναζητήσεις με βάση την απόσταση. Κάθε χρήστης έχει τα δικά του σημεία ενδιαφέροντος που μόνο αυτός βλέπει ή μπορεί να διαγράψει. Για την εισαγωγή ενός σημείου επιλέγεται το δεύτερο radio button “Insert Point of Interest for current User” και γίνεται κλικ στο κατάλληλο σημείο στο χάρτη. Σημείο ενδιαφέροντος μπορεί να είναι για παράδειγμα η διεύθυνση εργασίας (θέλω να βρω σπίτι που να απέχει x απόσταση από τη δουλειά μου), διεύθυνση συγγενών (π.χ. σπίτι που να απέχει 5km ώστε να είναι κοντά οι παπούδες και γιαγιάδες για τη φύλαξη των παιδιών), απόσταση από το εξοχικό ή από το σπίτι του αδερφού κλπ.



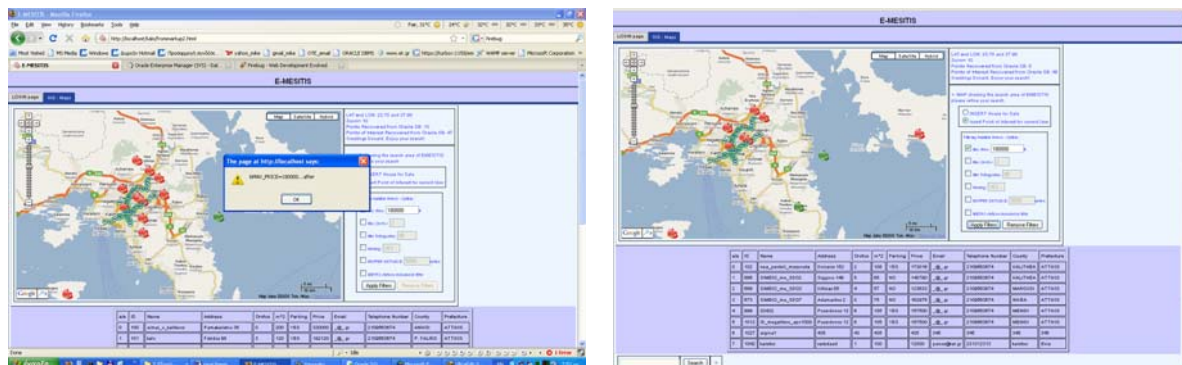
Εικόνα 45 Βήματα εισαγωγής σημείου ενδιαφέροντος

Μετά την επιτυχή εισαγωγή των στοιχείων και την αποθήκευση το σημείο ενδιαφέροντος απεικονίζεται μόνιμα με το: . Να σημειώσουμε ότι σημεία ενδιαφέροντος είναι και οι στάσεις του METRO οι οποίες όμως δεν μπορούν να διαγραφούν. Απλά υπάρχει η επιλογή όπως θα δούμε αργότερα να μην συμπεριληφθούν στην αναζήτηση με βάση την απόσταση.

#### 5.5.1.5 Αναζήτηση ακινήτου με βάση κριτήρια

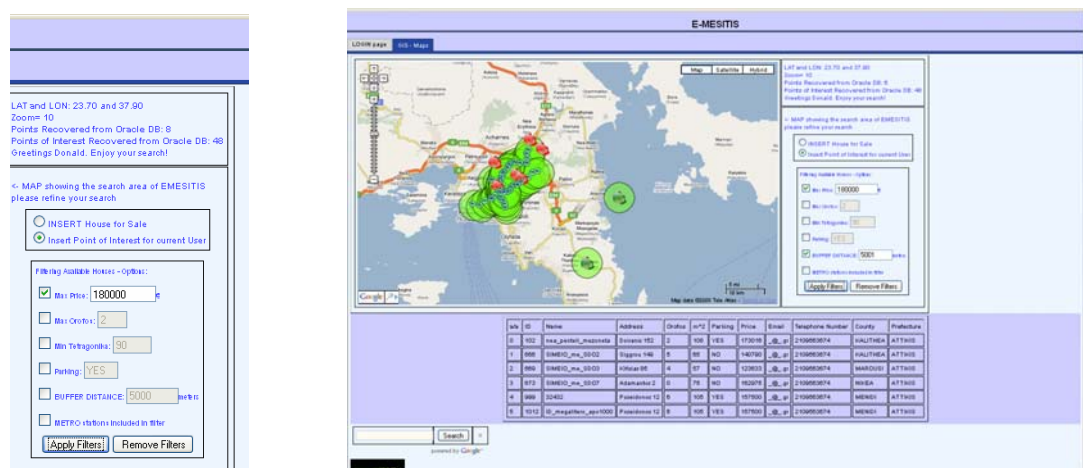
Η διαδικασία αναζήτησης μπορεί να γίνει με πληθώρα κριτηρίων. Αξίζει να τονίσουμε ότι για λόγους ασφαλείας και αποφυγής των πολύ απειλητικών και προσφάτως αναπτυχθέντων επιθέσεων “SQL code injection” σε site, κρίθηκε απαραίτητη η μη κατασκευή εντός της HTML σελίδας του query ώστε αφενός να είναι αφανή τα ονόματα των tables και του σχήματος της βάσης και αφετέρου να έχουμε τη δυνατότητα ελέγχου των εισηγμένων τιμών στην PHP. Το query χιτίζεται δυναμικά εντός της αντίστοιχης PHP ρουτίνας, όπως φαίνεται 7.1 παρακάτω στην ενότητα §7.1 «PHP Κώδικας αναζήτησης».

Πχ περιορισμός των αποτελεσμάτων από τα 13 που υπάρχουν στο ορατό πλαίσιο σε όσα δεν υπερβαίνουν την τιμή των 180000€ φαίνεται ακολούθως



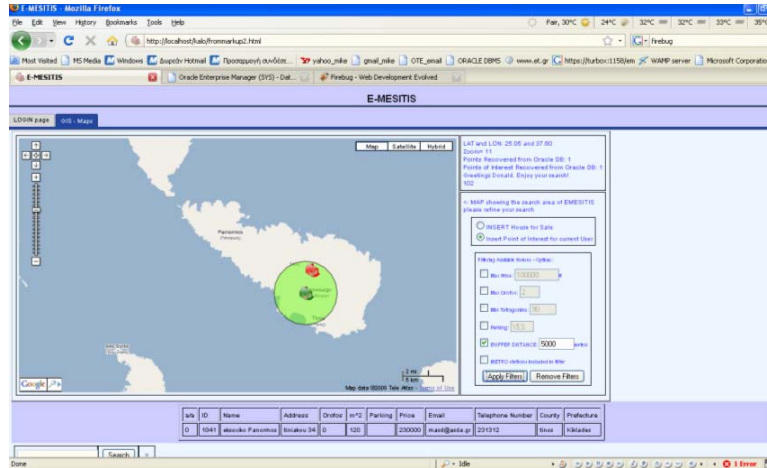
Εικόνα 46 Διαδικασία φιλτραρίσματος γεωγραφικών αναζήτησης με βάση τιμή πώλησης

Τα φίλτρα διαθέτουν προεπιλεγμένες τιμές στις οποίες επανέρχονται μετά από επιλογή της κατάργησης όλων των φίλτρων.



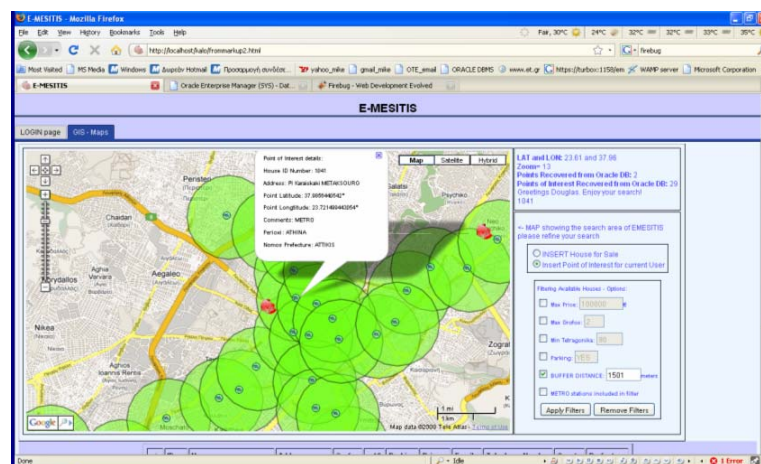
Εικόνα 47 Κριτήρια επιλογής με βάση την τιμή πώλησης εν δράσει...

Αντίστοιχα η αναζήτηση μπορεί να γίνει με φιλτράρισμα χωρικό με βάση την επιθυμητή απόσταση από τα σημεία ενδιαφέροντος (λογική δημιουργίας buffers με την ίδια ακτίνα). Στην επόμενη εικόνα φαίνεται η επιλογή ακτίνας 5km γύρω από όλα τα σημεία ενδιαφέροντος του εν λόγω χρήστη συμπεριλαμβανομένων και των σταθμών του METRO.



**Εικόνα 48** Εύρεση ακινήτου στην Πάνορμο

Αντίστοιχα η αναζήτηση μπορεί να γίνει με φιλτράρισμα χωρικό με βάση την επιθυμητή απόσταση από τα σημεία ενδιαφέροντος (λογική δημιουργίας buffers με την ίδια ακτίνα). Στην επόμενη εικόνα φαίνεται η επιλογή ακτίνας 5km γύρω από όλα τα σημεία ενδιαφέροντος του εν λόγω χρήστη συμπεριλαμβανομένων και των σταθμών του ΜΕΤΡΟ. Αντίστοιχα φαίνεται και η χωρική αναζήτηση χωρίς τους σταθμούς του ΜΕΤΡΟ.



**Εικόνα 49** Εύρεση ακινήτου εντός ακτίνας 1500m από το σταθμό Μεταξουργείου

# 6 *Επίλογος*

## **6.1 Σύνοψη και συμπεράσματα**

Ανακεφαλαιώνοντας, ο αρχικός στόχος επιτεύχθηκε. Το proof-of-concept απέδειξε ότι εντός έξι μηνών για ένα έμπειρο χρήστη<sup>16</sup> είναι δυνατή η διερεύνηση - ανάπτυξη μιας πρωτότυπης εφαρμογής με γεωγραφικά δεδομένα χωρίς κανένα κόστος εφόσον δεν πρόκειται για επιχειρησιακό προϊόν (άδεια Oracle, server αδιάλειπτης λειτουργίας, και τηλεπικοινωνιακά κόστη και πιθανά τέλη για χρήση των υπηρεσιών Google Maps σε περίπτωση που ξεπεραστεί το ημερήσιο όριο χρήσης).

Από τεχνικής απόψεως, η τεχνολογία AJAX, αποδείχθηκε αρκετά εύκολη και με θεαματικά αποτελέσματα. Ίσως ανασταλτικός παράγοντας είναι το πλήθος των διαφορετικών τεχνολογιών που χρησιμοποιήσαμε (μεταξύ άλλων: Oracle SQL Spatial, PHP, JavaScript, HTML, XML, Google Maps API, YUI κλπ), και η πληθώρα εργαλείων ειδικά αν σκεφτούμε τα πιο εξειδικευμένες περιβάλλοντα όπως Eclipse, Rico κλπ). Το ευκολότερο βήμα σε όλα αυτά ίσως ήταν το API των Google Maps.

Ευελπιστώ ότι με η παρούσα εφαρμογή θα αποτελέσει έναυσμα για συνεργασία μεταξύ των μεσιτών, του Σώματος Ορκωτών Εκτιμητών, των εφημερίδων «Μικρών Αγγελιών», αλλά και της συγκροτημένης πολιτείας στα πλαίσια της Στατιστικής υπηρεσίας ή της Εφορίας αλλά και των ενώσεων καταναλωτών για τη δημιουργία μιας βάσης δεδομένων με όλες τις αγοραπωλησίες και

---

<sup>16</sup> Επειδή απαιτήθηκε εκμάθηση τόσο Javascript, PHP, Oracle & Apache Admin, ο υποφαινόμενος χρειάστηκε σχεδόν ένα ημερολογιακό έτος part time εργασίας για την ολοκλήρωση (6 μήνες διερεύνηση-επιλογή εργαλείων και 6 μήνες ανάπτυξη).

την ολοκληρωμένη αντιμετώπιση της αγοράς ακινήτων μακριά από μικροσυμφέροντα και silo πληροφόρησης. Η μείωση του κόστους απόκτησης του πιο βασικού αγαθού (στέγη) είναι ουσιώδους σημασίας για την εθνική οικονομία προκειμένου να βρεθούν πόροι για πιο παραγωγικές διαδικασίες.

## **6.2 Μελλοντικές επεκτάσεις**

Προκειμένου να αναπτυχθεί μια πραγματικά Open-source λύση χωρίς κανένα άλλο κόστος αδειοδότησης και να αντιμετωπιστεί ο περιορισμός που θέτει το Google Maps ως προς το μέγιστο αριθμό ημερήσιων επισκέψεων κρίνεται σκόπιμη η ανεύρεση εναλλακτικών λύσεων είτε μέσω Openlayers είτε τρίτης open-source λύσης. Δυστυχώς η ESRI με το IMS δεν είναι ούτε αυτή δωρεάν. Αναφορικά με τη χρήση της βάσης και δεδομένων των ελλείψεων της MySQL σε χωρικούς τελεστές θα πρέπει να διερευνηθεί η PostgreSQL που είναι opensource Λογισμικό. Ενδεχόμενη εναλλακτική αρχιτεκτονική προσέγγιση με .NET ενόψει της νέας έκδοσης του MS SQL server που θα έχει χωρικούς τελεστές και δυνατότητες.

Χρήσιμα εργαλεία που θα έκαναν ευκολότερη την ανάπτυξη κώδικα JavaScript σύμφωνα με την τεχνολογία AJAX και θα έδιναν περισσότερες δυνατότητες για ένα πιο εύχρηστη και περίτεχνη διεπαφή χρήστη είναι το Google Web Toolkit (το οποίο παράγει αυτομάτως κώδικα Javascript από Java) ή άλλα Content Management Systems (CMS – Joomla, Drupal) ώστε να ενοποιηθεί και να απλουστευτεί η ενημέρωση του site, η προσθήκη νέων components, η διαχείριση χρηστών, τα forums, ρουτίνες από το Mapbuilder για στατιστικά του Real Estate κλπ. Πέρα από τον τομέα των τεχνολογικών επεκτάσεων και αλλαγών, πρέπει να επισημάνουμε επεκτάσεις που αφορούν στα παρακάτω:

- Ολοκλήρωση της εφαρμογής που υλοποιήσαμε με άλλες διαδικτυακές υπηρεσίες και ανάπτυξη της σαν widget που θα έβρισκε χρήση για παράδειγμα, στη διαδικτυακή υπηρεσία facebook. Δυνατότητες forum ώστε οι αγοραστές να σχολιάζουν τα προς πώληση / ενοικίαση ακίνητα.
- Τροποποίηση της εφαρμογής, ώστε να μπορεί να χρησιμοποιείται από χρήστες κινητών τηλεφώνων, τα οποία είναι σε θέση να γνωρίζουν τη θέση τους.
- Επέκταση της εφαρμογής, ώστε να ενσωματωθεί δυνατότητα αποστολής email (skype-voip κλήσης στον ιδιοκτήτη για κλείσιμο συμφωνίας).

Σίγουρα, οι ιδέες που μπορούν να παραχθούν είναι ποικίλες προσφέροντας κατ' αυτόν τον τρόπο πλούσιες εφαρμογές και υπηρεσίες στο χώρο του συνεχώς αναπτυσσόμενου Web 2.0.

# 7

## Παραρτήματα

### 7.1 PHP Κώδικας αναζήτησης



Η βασική σελίδα με τον κώδικα HTML και JAVASCRIPT. Όλος ο κώδικας τόσο σε PHP όσο και τα tables της βάσης παρατίθενται στο μαγνητικό μέσο (δισκέτα για περιορισμό των υπό εκτύπωση σελίδων).

```
<?php // File:
$debugtext = '<debugginInfo>';
if ($conn = oci_connect("HR", "a", "//localhost/orcl1")) {
    $debugtext .= 'Successfully connected to Oracle Database !';
}
else {
    $errmsg = oci_error();
    $debugtext .= 'Oracle connect error: ' . $errmsg['message'];
}
```

```

// ===== EXOUME TO NEO EXTENT POU THA EISAGOUME STON PINAKA RE_AREA_SDO =====
$SW_lat = number_format($_GET['SW_lat'],16)*10000000000000000;
$SW_lng = number_format($_GET['SW_lng'],16)*10000000000000000;
$NE_lat = number_format($_GET['NE_lat'],16)*10000000000000000;
$NE_lng = number_format($_GET['NE_lng'],16)*10000000000000000;

$debugtext .= "ta inputs sto php einai ".$SW_lat." ".$SW_lng." ".$NE_lat." ".$NE_lng." ";

// ===== EISAGOGE NEOU EXTENT STON PINAKA RE_AREA_SDO afou diagrapso to palio oste na min auksanetai aperiorista to id=====
// ara to neo pou eisago tha exei id panta 1 PROSOXI!!!
$delete_oldest_Extent_as_searchquery = " DELETE FROM RE_AREA_SDO WHERE ID=(SELECT MIN(ID) FROM RE_AREA_SDO) ";
$stmtid_del = oci_parse($conn, $delete_oldest_Extent_as_searchquery);
$r = oci_execute($stmtid_del, OCI_DEFAULT);
// echo "to query".$insert_Extent_as_searchquery."etrekse OK.".$r." to dipla";

$insert_Extent_as_searchquery = " INSERT INTO RE_AREA_SDO(data, name, id, shape) VALUES('extent_anazitisis', 'mixali_search', 1, MDSYS.SDO_GEOMETRY(2003, NULL, NULL, MDSYS.SDO_ELEM_INFO_ARRAY(1,1003,3), MDSYS.SDO_ORDINATE_ARRAY('".$SW_lat."', ".$SW_lng."', ".$NE_lat."', ".$NE_lng.")))";
$stmtid = oci_parse($conn, $insert_Extent_as_searchquery);
$r = oci_execute($stmtid, OCI_DEFAULT);
$debugtext .= "to query".$insert_Extent_as_searchquery."etrekse OK.".$r." to dipla";
oci_commit($conn);
if ($r)
{
    $debugtext .= " Location successfully inserted!";
    oci_commit($conn);
    oci_close($conn);
}

```



```

else
{
    $debugtext .= "<html> <head> </head> <body> <font color=#FF0080> There was a problem inserting the location!</font> </body></html>";
    var_dump(oci_error($stid));
}

// PROSOXI META TO INSERT KAI TO COMMIT TO TABLE PERIEXEI NEO MAX(ID) exei enimerothei automata!!! PROSEXE STO EPOMENO QUERY

$whereclause = "";
$wherePlithosApaitiseon = 0;
$query2 = "SELECT point1.ID, point1.NAME, point1.LAT, point1.LNG, point1.DATA, point1.OROFOS, point1.TETRAGONIKA, point1.PARKING, point1.IMAGEDIRE, point1.PRICE, point1.EMAIL, point1.TELEPHONENUM,
point1.ADDRESS, point1.PERIOXI, point1.NOMOS, (SELECT MAX(ID) FROM RE_POINTS_SDO) AS MAXID FROM RE_AREA_SDO area1, RE_POINTS_SDO point1 ";

//filtrara kai tous dio pinakes toso ta spitia oso kai ta simeia endiaferontos me to orato parathiro gia na min pairno perierga apotelesmata

if (isset($_GET['Username'])) { $Username = $_GET['Username']; $debugtext .= " $Username=".$Username; }
if (isset($_GET['MAX_PRICE'])) { $MaxPrice = $_GET['MAX_PRICE']; $debugtext .= $MaxPrice; }
if (isset($_GET['MAX_OROFOS'])) { $MaxOrofos = $_GET['MAX_OROFOS']; $debugtext .= $MaxOrofos; }
if (isset($_GET['MIN_TETRAGONIKA'])) { $MinTetragonika = $_GET['MIN_TETRAGONIKA']; $debugtext .= $MinTetragonika; }
if (isset($_GET['FILTER_PARKING'])) { $FilterParking = $_GET['FILTER_PARKING']; $debugtext .= $FilterParking; }
if (isset($_GET['BUFFER_IN_METERS'])) { $BUFFER_IN_METERS = $_GET['BUFFER_IN_METERS']; $debugtext .= $BUFFER_IN_METERS; }
if (isset($_GET['METRO_BUFFER'])) { $METRO_BUFFER = $_GET['METRO_BUFFER']; $debugtext .= $METRO_BUFFER; }

// ===== GEOGRAFIKO QUERY ME VASI TO NEO EXTENT STON PINAKA RE_POINTS_SDO =====
$conn2 = oci_connect("HR", "a", "localhost/orcl1");// OXI peristent one (afou exo pthreads) - DISTIXOS tin persistent DEN mporeis na tin kliseis me oci_close sto telos

// palio kai aplo $query2 = "SELECT point1.ID, point1.NAME, point1.LAT, point1.LNG, point1.DATA, point1.OROFOS, point1.TETRAGONIKA, point1.PARKING, point1.IMAGEDIRE, point1.PRICE, point1.EMAIL, point1.TELEPHONENUM,
point1.ADDRESS, point1.PERIOXI, point1.NOMOS, (SELECT MAX(ID) FROM RE_POINTS_SDO) AS MAXID FROM RE_AREA_SDO area1, RE_POINTS_SDO point1 WHERE area1.id=(SELECT MAX(ID) FROM RE_AREA_SDO) AND
SDO_RELATE(point1.shape, area1.shape, 'MASK=INSIDE+COVEREDBY')='TRUE' ORDER BY point1.id";

$query2 = "SELECT DISTINCT point1.ID, point1.NAME, point1.LAT, point1.LNG, point1.DATA, point1.OROFOS, point1.TETRAGONIKA, point1.PARKING, point1.IMAGEDIRE, point1.PRICE, point1.EMAIL, point1.TELEPHONENUM,
point1.ADDRESS, point1.PERIOXI, point1.NOMOS, (SELECT MAX(ID) FROM RE_POINTS_SDO) AS MAXID FROM RE_AREA_SDO area1, RE_POINTS_SDO point1 ";

// Oposdipote to DISTINCT oste na min epistrefo duplicates apo osa ikanopoioioun tin apostasi apo parapano spitia.

```





```

$text .= '<marker lat="'.number_format(($row2['LAT']/1000000000000000),16,$locale['decimal_point'],$locale['thousands_sep'])."' .
    'lng="'.number_format(($row2['LNG']/1000000000000000),16,$locale['decimal_point'],$locale['thousands_sep'])."' .
    'data="'.htmlentities($row2['DATA'])."' .
    'id="'.htmlentities($row2['ID'])."' .
    'name="'.htmlentities($row2['NAME'])."' .
    'orofos="'.htmlentities($row2['OROFOS'])."' .
    'tetragonika="'.htmlentities($row2['TETRAGONIKA'])."' .
    'parking="'.htmlentities($row2['PARKING'])."' .
    'imagedire="'.htmlentities($row2['IMAGEDIRE'])."' .
    'price="'.htmlentities($row2['PRICE'])."' .
    'email="'.htmlentities($row2['EMAIL'])."' .
    'telephonenumber="'.htmlentities($row2['TELEPHONENUM'])."' .
    'address="'.htmlentities($row2['ADDRESS'])."' .
    'perioxi="'.htmlentities($row2['PERIOXI'])."' .
    'nomos="'.htmlentities($row2['NOMOS'])."' .
    />';

    $debugtext .= " NEO RECORD ".$i;
    $i = $i + 1;
    $MAXID= htmlentities($row2['MAXID']);//ok se kathe grammi tha to ksanatheto alla mikro to kako...(oles oi grammes exoun to idio)
} //to while
} //to if(r2)
//$text .= '<MAXID value="'. $MAXID.'" />'; den to vazou den einai toso simantiko telika--afou tha ginetai prota elegxos ton fotos apo anthropo

// ===== GEOGRAFIKO QUERY ME VASI TO NEO EXTENT STON PINAKA RE_BASEPOINTS_SDO =====

```



```

'lng="'.number_format($row3['LNG']/1000000000000000,16,$locale['decimal_point'],$locale['thousands_sep'])."' .
'data="'.htmlentities($row3['DATA'])."' .
'id="'.htmlentities($row3['ID'])."' .
'name="'.htmlentities($row3['NAME'])."' .
'address="'.htmlentities($row3['ADDRESS'])."' .
'perioxi="'.htmlentities($row3['PERIOXI'])."' .
'nomos="'.htmlentities($row3['NOMOS'])."' .
'/>';

$debugtext .= " NEO BASE RECORD ". $i;
$i = $i + 1;

$MAXBASEID= htmlentities($row3['MAXID']);//ok se kathe grammi tha to ksanatheto alla mikro to kako...(oles oi grammes exoun to idio)
} //tou while
} //tou if(r3)
// -----

$text .= "</markers>";
echo $text; //SOS AUTO EDO EINAI OLA TA LEFTA EDO IPARXOUN OLA TA APOTELESMATA

$debugtext = '</debuginfo>';
$text = $debugtext;

if (fwrite($handle, $text) == FALSE) {
    $debugtext = "Could not create file.txt.";
}
else {
    $debugtext = "Created file.txt OK for debugging reasons just to have the last operation in a file should a bad guy hits.";
    // thimistou oti an doulepseis me auto to arxeio xaneis ti dinatotita tautoxrnis xrisis      tou site apo pollous users

```

```

    }
fclose($handle); //PROSOXI
oci_close($conn2);
?>

```

## 7.2 Κώδικας HTML – αρχείο *frommarkup2.html*



Η βασική σελίδα με τον κώδικα HTML και JAVASCRIPT. Όλος ο κώδικας τόσο σε PHP όσο και τα tables της βάσης παρατίθενται στο μαγνητικό μέσο (δισκέτα για περιορισμό των υπό εκτύπωση σελίδων).

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml">

<head>
<link rel="shortcut icon" href="favicon.ico" > <!-- // gia to eikonidio pou fainetai sto tab tou mozilla & sto parathiro -->
<meta name="robots" content="index, follow" > <!-- // gia na einai prosvasimo apo psaxtiria -->

<meta name="" http-equiv="content-type" content="text/html; charset=utf-8"><title> E-MESITIS</title>

<!-- //////////////////////////////////////////////////////////////////// -->
<!-- To gss einai gia to box ton fotografion apo to slideshow toy google -->
<link rel="stylesheet" type="text/css" href="style.css">
<style type="text/css">
/*margin and padding on body element
can introduce errors in determining
element position and are not recommended;
we turn them off as a foundation for YUI
CSS treatments. */
body {
margin:0;
padding:0;
}

.gss a img {border : none;}

```

```

.gss {
  width: 100px;
  height: 100px;
  color: #ddddd;
  background-color: #000000;
  padding: 8px;
}

#map {font-size:8pt;}

</style>

<link rel="stylesheet" type="text/css" href="frommarkup_clean_files/fonts-min.css">
<link rel="stylesheet" type="text/css" href="frommarkup_clean_files/tabview.css">
<script type="text/javascript" src="frommarkup_clean_files/yahoo-dom-event.js"></script>
<script type="text/javascript" src="frommarkup_clean_files/element-beta.js"></script>
<script type="text/javascript" src="frommarkup_clean_files/tabview.js"></script><!--there is no custom header content for this example-->

<script charset="utf-8" id="injection_graph_func" src="frommarkup_clean_files/injection_graph_func.js"></script>
<link href="frommarkup_clean_files/highlighter.css" type="text/css" rel="stylesheet">
<link href="frommarkup_clean_files/highlighter_002.css" type="text/css" rel="stylesheet">

<!-- ////////////////////////////////////////////////// -->
<!-- GOOGLE MAPS Localhost key -->
<script type="text/javascript" src="http://www.google.com/jsapi?key=ABQIAAAATyqlpfuy2d29ZX-cXsFdvht2yXp_ZAY8_ufC3CFXhHIE1NvwkxSybDTw517GCVHI0V-YwHd_DqCCog&hl=gr"></script>

<script src="http://www.google.com/uds/solutions/slideshow/gfslideshow.js"
  type="text/javascript">
  //fortono kai to slideshow gia na deixno fotografies apo rss/atom feed tis perioxis endiaferontos apo flickr
</script>
<!-- GOOGLE MAPS Localhost key END -->
<!-- ////////////////////////////////////////////////// -->

<!-- ////////////////////////////////////////////////// -->
<!-- GENIKA GIA AJAX multithreading -->

<script language = "javascript" type="text/javascript" >
  function getDataPOST(mydataSource, mydata)
  {
    var XMLHttpRequestObject = false;
    var mytarget = 0;
    if (window.XMLHttpRequest) {
      XMLHttpRequestObject = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
      XMLHttpRequestObject = new ActiveXObject("Microsoft.XMLHttp");
    }
    if(XMLHttpRequestObject) {
      XMLHttpRequestObject.open("POST", mydataSource, true);
      XMLHttpRequestObject.setRequestHeader('Content-type','application/x-www-form-urlencoded');
    }
  }
}

```



```

XMLHttpRequestObject.onreadystatechange = function()
{
    if (XMLHttpRequestObject.readyState == 4 && XMLHttpRequestObject.status == 200) {
        mytarget = XMLHttpRequestObject.responseText;
        delete XMLHttpRequestObject;
        XMLHttpRequestObject = null;
    }
}
XMLHttpRequestObject.send(mydata);
}
return mytarget;
}

```

```
<!-- script GENIKA GIA AJAX multithreading END -->
```

```
<!-- ////////////////////////////////////// -->
```

```
<!-- ////////////////////////////////////// -->
```

```
<!-- // LOGIN script language="javascript" -->
```

```

function GETRequest(strURL,localUsername){
    var xmlhttp;
    if(window.XMLHttpRequest){ // For Mozilla, Safari, ... -----
        var xmlhttp = new XMLHttpRequest();
    }
    else if(window.ActiveXObject){ // For Internet Explorer
        var xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.open("GET", strURL, true);
    xmlhttp.setRequestHeader("Content-Type", 'application/x-www-form-urlencoded');
    xmlhttp.onreadystatechange = function(){
        if (xmlhttp.readyState == 4){
            updatepageOLD(xmlhttp.responseText,localUsername);
            delete xmlhttpRequest;
            xmlhttpRequest = null;
        }
    }
    xmlhttp.send(null);
}

```

```

var myTileLayer;
var myTileLayerHidden = false;

```

```

function updatepageOLD(str,username2show){
    <!-- document.getElementById('DB_name_check_query').innerHTML += str; -->
    if(str == "yes"){
        myTileLayerHidden =true;
        window.myTileLayer.hide();
    }
}

```

```

        map.removeOverlay(myTileLayer); //auto kanei ti douleia - to afairo mia kai kali giati me to hide otan allazes to zoom ksanarxotane
        //apla afino to myTileLayerHidden mipos xrieastei na ilopoiso kati se logout...
        alert("Welcome User. Please go to tab 3!");
        window.tempUsername = window.document.fl.username.value;

        document.getElementById('messageNotRegistered').innerHTML = "Greetings " + window.tempUsername + ". Enjoy your search!" ;
        document.getElementById('LOG_Button').innerHTML = '<input name="a1" value="Logout" onclick="call_logout()" type="button">';
        QUERY_VISIBLE_POINTS();

    }else{
        alert("Invalid Login! Please try again!");
        window.tempUsername = null;
    }
}

function call_login(){
    var username = window.document.fl.username.value;
    var password = window.document.fl.password.value;
    //var urlPOST = "loginPOST.php?username="+ username + "&password="+password ;
    var urlGET = "loginGET.php?username="+ username + "&password="+password ;

    GETRequest(urlGET,password);
}

function call_logout(){
    alert("Bye Bye dear User:" + window.tempUsername);
    document.getElementById('messageNotRegistered').innerHTML = '<strong style="color: red;"> "Sorry but you have not registered yet...Page will be unavailable"</strong>';
    document.getElementById('LOG_Button').innerHTML = '<input name="a1" value="Login" onclick="call_login()" type="button">'; // to antikathisto oste na mpei o epomenos
    window.tempUsername = null;
    myTileLayerHidden = false;
    map.addOverlay(myTileLayer);
    QUERY_VISIBLE_POINTS();
    window.document.fl.password.value = "";
    window.document.fl.username.value = "";
}

<!-- script LOGIN END -->
<!-- //////////////////////////////////////// -->

<!-- //////////////////////////////////////// -->
<!-- // UPDATE POINTS in DB script language="javascript" -->

function pausecomp(millis)
{
    var date = new Date();
    var curDate = null;

    do { curDate = new Date(); }
    while(curDate-date < millis);
}

function getServerScriptforPointInsertion(mydataSource)

```

```

{
    var XMLHttpRequestObject = false;
    var mytarget = 0;
    if (window.XMLHttpRequest) {
        XMLHttpRequestObject = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        XMLHttpRequestObject = new ActiveXObject("Microsoft.XMLHttp");
    }
    if (XMLHttpRequestObject) {
        XMLHttpRequestObject.open("GET", mydataSource, true);
        XMLHttpRequestObject.setRequestHeader('Content-type', 'application/x-www-form-urlencoded');
        XMLHttpRequestObject.onreadystatechange = function()
        {
            if (XMLHttpRequestObject.readyState == 4 && XMLHttpRequestObject.status == 200) {
                //alert("check1="+mydataSource+" kai "+XMLHttpRequestObject.readyState);

                mytarget = XMLHttpRequestObject.responseText;
                //alert("check2 response= "+mytarget);

                delete XMLHttpRequestObject;
                XMLHttpRequestObject = null;
                return mytarget;
            }
            // alert("check3="+mydataSource+XMLHttpRequestObject.readyState);
        }
        XMLHttpRequestObject.send(null);
    }
}

```

```

function getServerScriptforPoints(mydataSource)
{
    var XMLHttpRequestObject = false;
    var mytarget = 0;
    if (window.XMLHttpRequest) {
        XMLHttpRequestObject = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        XMLHttpRequestObject = new ActiveXObject("Microsoft.XMLHttp");
    }
    if (XMLHttpRequestObject) {
        XMLHttpRequestObject.open("GET", mydataSource, true);
        XMLHttpRequestObject.setRequestHeader('Content-type', 'application/x-www-form-urlencoded');
        XMLHttpRequestObject.onreadystatechange = function()
        {
            if (XMLHttpRequestObject.readyState == 4 && XMLHttpRequestObject.status == 200) {
                //alert("check1="+mydataSource+" kai "+XMLHttpRequestObject.readyState);

                mytarget = XMLHttpRequestObject.responseText;
                // alert("check2 text response= "+mytarget);
                // alert("check2 xml response= "+XMLHttpRequestObject.responseText);
            }
        }
    }
}

```

```
// to akoloutho katevazei to xml arxeio me titlo file.xml kai to anoigei gia na vrei ta simeia
```

```
//readMap("file.xml");---auto to eixa otan douleua me to arxeio --prosoxi den itan multitasking logo tou oti i leitourgia sto server stirizotan se arxeio.diorthothike 18/5/2008  
readMap(mytarget);
```

```
delete XMLHttpRequestObject;  
XMLHttpRequestObject = null;
```

```
}  
// alert("check3="+mydataSource+XMLHttpRequestObject.readyState);
```

```
}  
XMLHttpRequestObject.send(null);
```

```
}
```

```
}
```

```
<!-- to proigoumeno diko mou -->
```

```
<!-- // UPDATE POINTS in DB -->
```

```
<!-- //////////////////////////////////////// -->
```

```
<!-- // ta akoloutha energopoioun ta pedia eisagogis timon gia ta filtrarismata timon ton pedion timis, orofou, tetragonikon, parking kai buffers -->
```

```
function enableTxtPrice() {
```

```
    //to onoma tis sinartisis einai ligo paraplanitiko kanonika eprepe na legetai Revert (giati apla allazei tin timi)
```

```
    //alert(document.getElementById("filterPrice").getAttribute("disabled") );
```

```
    //to disabled alloi to anaferoun os property toy span kai oxi tou input box...
```

```
    //se emena leitourgei perierga...?? sto span den ekane tipota. Mesa sto input apla fainetai san diakoptis...aneksartitos timis
```

```
    //gia auto einai ligaki perierges oi sinartiseis mou -diladi elegxo to checkbox kai allazo to innerHTML tou span object --me neo input text (douleuei pantos)
```

```
    var dumVarFilter = document.getElementById("filterPrice");
```

```
    var dumVarCheckBox = document.getElementById("txtBoxPrice");
```

```
    if (dumVarFilter.getAttribute("disabled")) {dumVarCheckBox.innerHTML = '<input id="filterPrice" name="filterPrice" size="10" value="100000" type="text" >&euro!';}
```

```
    else {dumVarCheckBox.innerHTML = '<input id="filterPrice" name="filterPrice" size="10" value="100000" type="text" disabled="true">&euro!';}
```

```
}
```

```
function enableTxtOrofos() {
```

```
    //to onoma tis sinartisis einai ligo paraplanitiko kanonika eprepe na legetai Revert (giati apla allazei tin timi)
```

```
    var dumVarFilter = document.getElementById("filterOrofos");
```

```
    var dumVarCheckBox = document.getElementById("txtBoxOrofos");
```

```
    if (dumVarFilter.getAttribute("disabled")) {dumVarCheckBox.innerHTML = '<input id="filterOrofos" name="filterOrofos" size="2" value="2" type="text">';}
```

```
    else {dumVarCheckBox.innerHTML = '<input id="filterOrofos" name="filterOrofos" size="2" value="2" type="text" disabled="true">';}
```

```
}
```

```
function enableTxtTetragonika() {
```

```
    //to onoma tis sinartisis einai ligo paraplanitiko kanonika eprepe na legetai Revert (giati apla allazei tin timi)
```

```
    var dumVarFilter = document.getElementById("filterTetragonika");
```

```
    var dumVarCheckBox = document.getElementById("txtBoxTetragonika");
```

```

    if (dumVarFilter.getAttribute("disabled")) {dumVarCheckBox.innerHTML = '<input id="filterTetragonika" name="filterTetragonika" size="4" value="90" type="text" >';}
    else {dumVarCheckBox.innerHTML= '<input id="filterTetragonika" name="filterTetragonika" size="4" value="90" type="text" disabled="true" >';}
}
function enableTxtParking() {
    //to onoma tis sinartisis einai ligo paraplanitiko kanonika eprepe na legetai Revert (giati apla allazei tin timi)
    var dumVarFilter = document.getElementById("filterParking");
    var dumVarCheckBox = document.getElementById("txtBoxParking");

    if (dumVarFilter.getAttribute("disabled")) {dumVarCheckBox.innerHTML = '<input id="filterParking" name="filterParking" size="3" value="YES" type="text" >';}
    else {dumVarCheckBox.innerHTML= '<input id="filterParking" name="filterParking" size="3" value="YES" type="text" disabled="true" >';}
}
function enableTxtBuffer() {
    //to onoma tis sinartisis einai ligo paraplanitiko kanonika eprepe na legetai Revert (giati apla allazei tin timi)
    var dumVarFilter = document.getElementById("filterBuffer");
    var dumVarCheckBox = document.getElementById("txtBoxBuffer");

    if (dumVarFilter.getAttribute("disabled")) {dumVarCheckBox.innerHTML = '<input type="text" id="filterBuffer" name="Buf_Distance" size="7" maxlength="7" value="5000" meters />';}
    else {dumVarCheckBox.innerHTML= '<input type="text" id="filterBuffer" name="Buf_Distance" size="7" maxlength="7" value="5000" meters disabled="true" />';}
}
function Removefilter() {

    <!-- // Dimming ton koumpion -->
    document.forms["MESITIS_RADIO"].reset();//SOS

    var dumVarCheckBox = document.getElementById("txtBoxPrice");
    dumVarCheckBox.innerHTML = '<input id="filterPrice" name="filterPrice" size="10" value="100000" type="text" disabled="true">&euro;';

    var dumVarCheckBox = document.getElementById("txtBoxOrofos");
    dumVarCheckBox.innerHTML = '<input id="filterOrofos" name="filterOrofos" size="2" value="2" type="text" disabled="true">';

    var dumVarCheckBox = document.getElementById("txtBoxTetragonika");
    dumVarCheckBox.innerHTML = '<input id="filterTetragonika" name="filterTetragonika" size="4" value="90" type="text" disabled="true">';

    var dumVarCheckBox = document.getElementById("txtBoxParking");
    dumVarCheckBox.innerHTML = '<input id="filterParking" name="filterParking" size="3" value="YES" type="text" disabled="true">';

    var dumVarCheckBox = document.getElementById("txtBoxBuffer");
    dumVarCheckBox.innerHTML = '<input type="text" id="filterBuffer" name="Buf_Distance" size="7" maxlength="7" value="5000" meters disabled="true"/>';

    document.getElementById("Remove_filter_button").innerHTML = '<INPUT name="Remove Filters" TYPE="button" value="Remove Filters" onClick="Removefilter(window.tempUsername);"
disabled="true" />';
    window.GLOBALurlGET_SQLfilters = ""; //midenismos

    QUERY_VISIBLE_POINTS(" apo to remove filters"); // gia na ananeoso to xarti
}
function filter(xristis) {

```

```

var dumVarFilterPrice    = document.getElementById("filterPrice");
var dumVarFilterOrofos  = document.getElementById("filterOrofos");
var dumVarFilterTetragonika = document.getElementById("filterTetragonika");
var dumVarFilterParking = document.getElementById("filterParking");
var dumVarFilterBuffer  = document.getElementById("filterBuffer");
//var dumVarFilterMetro  = document.getElementById("filterMetro");

var dumVarFilterPriceValue    = document.getElementById("filterPrice").value;
var dumVarFilterOrofosValue  = document.getElementById("filterOrofos").value;
var dumVarFilterTetragonikaValue = document.getElementById("filterTetragonika").value;
var dumVarFilterParkingValue  = document.getElementById("filterParking").value;
var dumVarFilterBufferValue   = document.getElementById("filterBuffer").value;

//alert("filter...entered fprice2=" + dumVarFilterPriceValue);
window.GLOBALurlGET_SQLfilters = ""; //midenismos
alert(window.GLOBALurlGET_SQLfilters+"...before");
if (xristis == null) { alert("Sorry but do login first");}
else
{
    if (!(dumVarFilterPrice.getAttribute("disabled"))) {window.GLOBALurlGET_SQLfilters += "&MAX_PRICE=" + dumVarFilterPriceValue;}
    if (!(dumVarFilterOrofos.getAttribute("disabled"))) {window.GLOBALurlGET_SQLfilters += "&MAX_OROFOS=" + dumVarFilterOrofosValue;}
    if (!(dumVarFilterTetragonika.getAttribute("disabled"))) {window.GLOBALurlGET_SQLfilters += "&MIN_TETRAGONIKA=" + dumVarFilterTetragonikaValue;}
    if (!(dumVarFilterParking.getAttribute("disabled"))) {window.GLOBALurlGET_SQLfilters += "&FILTER_PARKING=" + dumVarFilterParkingValue;}
    if (!(dumVarFilterBuffer.getAttribute("disabled"))) {window.GLOBALurlGET_SQLfilters += "&BUFFER_IN_METERS=" + dumVarFilterBufferValue ;}
    //if (document.forms["MESITIS_RADIO"].ChckBoxFilters[5].checked == true ) {window.GLOBALurlGET_SQLfilters += "&METRO_BUFFER=" + '1' ;}
    if ( dumVarFilterBufferValue % 2 == 1) {window.GLOBALurlGET_SQLfilters += "&METRO_BUFFER=" + '1' ;}

    alert(window.GLOBALurlGET_SQLfilters+"...after");
    //to akoloutho gia energopiso to koumpi tis diagrafis ton filtron
    document.getElementById("Remove_filter_button").innerHTML = '<INPUT name="Remove Filters" TYPE="button" value="Remove Filters" onClick="Removefilter(window.tempUsername);'
};

    QUERY_VISIBLE_POINTS(" apo to apply filters"); // gia na ananeoso to xarti

}

//telos kai tou else apo ton elegxo tou xristis==null

<!------->
<!-- // ta akoloutha einai gia to sxediasmo kiklou giro apo ta simeia endiaferontos -->
/** Utility Functions **/
// rand - Generates random number from 1 to n, inclusive
function rand ( n )
{
    return ( Math.floor ( Math.random ( ) * n + 1 ) );
}

// byte2hex - Takes number n from 0-255 and converts to hexadecimal string e.g. 'AA'
// Courtesy Jim Bumgardner of krazydad.com
function byte2Hex(n)
{

```

```

var nybHexString = "0123456789ABCDEF";
return String(nybHexString.substr((n >> 4) & 0x0F,1)) + nybHexString.substr(n & 0x0F,1);
}

// RGB2Color - Takes 3 hexadecimal string color components and concatenates into standard HTML format
// Courtesy Jim Bumgardner of crazydad.com
function RGB2Color(r,g,b)
{
return '#' + byte2Hex(r) + byte2Hex(g) + byte2Hex(b);
}

function draw_circle(clickedPoint){
//alert("draw circle inside No1 "+clickedPoint.lat() + " " +clickedPoint.lng());
var polyPoints = Array();
var mapNormalProj = G_NORMAL_MAP.getProjection();
var mapZoom = window.map.getZoom();
var clickedPixel = mapNormalProj.fromLatLngToPixel(clickedPoint, mapZoom);

var polySmallRadius = (128 / (Math.PI*640000)) * Math.pow(2,window.map.getZoom()) * document.getElementById("filterBuffer").value ; //200;//Math.abs(clickedPixel.y -
clickedPixelDistanceAway.y)*20;
var polyColor = RGB2Color(113,255,20);

// circle mode
var polyNumSides = 20;
var polySideLength = 18;
for (var a = 0; a<(polyNumSides+1); a++) {
var aRad = polySideLength*a*(Math.PI/180);
var polyRadius = polySmallRadius;
var pixelX = clickedPixel.x + polyRadius * Math.cos(aRad);
var pixelY = clickedPixel.y + polyRadius * Math.sin(aRad);
var polyPixel = new google.maps.Point(pixelX,pixelY);
var polyPoint = mapNormalProj.fromPixelToLatLng(polyPixel,mapZoom);
polyPoints.push(polyPoint);
}
//alert("draw circle inside No2"+polyPoint);
var polygon = new google.maps.Polygon(polyPoints,"#000000",2,.5,polyColor,.5);
window.map.addOverlay(polygon);
window.greenCircles.push(polygon);
//alert("draw circle inside No3"+window.map.addOverlay(polygon));
}

<!-- telos me ton sxediasmo kiklou giro apo ta simeia endiaferontos -->
<!------- -->

<!-- ////////////////////////////////////// -->
<!-- GOOGLE MAPs START -->

<!-- SOS XRISIMOPOIO TO AJAX framework tou GOOOGLE ara prepei na foroso ti version 2 opos perigrafetai sto
http://code.google.com/apis/maps/documentation/index.html#AJAX_Loader -->

```

```

<!-- script type="text/javascript" -->
google.load("maps", "2");
google.load("search", "1");
google.load("feeds", "1");
/**
 * Use google.load() to load the AJAX Feeds API
 * Use google.setOnLoadCallback() to call LoadSlideShow once the page loads
 */

var map ;
var markerManager ;
var htmls = [];
var labels = [];
var lastmarker;
var lastmarkerPermanent;
var iwform;
var MASTER_dragging_pointX = null; //xrisi stin drag tou inputmarker --apothikeusi tis telikis thesis oste na enimerothei katallila kai i form
var MASTER_dragging_pointY = null; //xrisi stin drag tou inputmarker --apothikeusi tis telikis thesis oste na enimerothei katallila kai i form

var Cancelform;
var UploadForm;
// var MAXID; den tha to xrisimopoiiso telika -opoios vazei fotos tha prepei prota na kanei upload fotos se default dir + submit. Anthropos tha ta vazei sto sosto dir.
var NUMofinputTABSPROCESSED = 0;

// arrays to hold copies of the markers used by the side_bar
var markers = [];

var tempUsername = null;
var globalid;

//to GLOBALurlGET_SQLfilters einai gia na apothikeuo ta special filtrarismata pou tha stelno sto QUERY_VISIBLE_POINTS
var GLOBALurlGET_SQLfilters = "";

var greenCircles = [];

```

```

<!-- NEA APO TO http://econym.googlepages.com/store.htm KAI o kodikas sto http://econym.googlepages.com/example_store.htm -->

```

```

// == creates a draggable marker with an input form ==
function createInputTabbedMarker(point,htmls,labels) {
  if (!lastmarker) { //oste na min epitrepetai na ginei eisagogi deuterou simeiou an prota den exei apothikeutei to proto
    NUMofinputTABSPROCESSED = 0 ;
    window.MASTER_dragging_pointY=point.lat(); //to MASTER_dragging_point an ginei drag o marker tha allaksei-otan klithei i iwform tha periexei panta to sosto!
    window.MASTER_dragging_pointX=point.lng(); //to MASTER_dragging_point an ginei drag o marker tha allaksei-otan klithei i iwform tha periexei panta to sosto!
    if (document.forms["MESITIS_RADIO"].MESITIS_RADIO[0].checked == true) { //INSERTION

```



```

var marker = new google.maps.Marker(point, {draggable:true, icon:G_START_ICON});
// In all scriptable browsers, the onSubmit event handler cancels the submission if
// the handler evaluates to return false.
window.iwform = 'Enter details:<br>' + '<form action="#" onSubmit="process(this,htmls,labels,window.tempUsername); return false;" >'
+ '<p>House Name:<input type="text" name="PointName" size="25" maxlength="25" value="" />'
+ '<span>Price:<input type="text" name="Price" size="7" maxlength="7" value="" &euro /></span>'
+ '<input type="hidden" name="PointLat" size="12" maxlength="12" value=' + window.MASTER_dragging_pointY + ' />'
+ '<input type="hidden" name="PointLng" size="12" maxlength="12" value=' + window.MASTER_dragging_pointX + ' /></p>'
+ '<p>Orofos:<input type="text" name="Orofos" size="2" maxlength="2" value="" />'
+ '<span>Tetragonika:<input type="text" name="Tetragonika" size="5" maxlength="5" value="" /> m^2</span>'
+ '<span>Parking:<input type="text" name="Parking" size="3" maxlength="3" value="YES" /></span></p>'
+ '<p>Comments:<input type="text" name="Comments" size="30" maxlength="30" value="" /></p>'
+ '<p>Owner Email:<input type="text" name="OwnerEmail" size="30" maxlength="30" value="" />'
+ '<span>Owner Tel Number:<input type="text" name="OwnerTelNumber" size="15" maxlength="15" value="" /></span></p>'
+ '<p>Perioxi:<input type="text" name="Perioxi" size="15" maxlength="15" value="" />'
+ '<span>Address:<input type="text" name="Address" size="30" maxlength="30" value="" /></span>'
+ '<span>Nomos Prefecture:<input type="text" name="Nomos" size="20" maxlength="20" value="" /></span>'
+ '<input type="submit" value="Submit" /></p>'
+ '</form>';
//+ '<input type="hidden" name="PointLat" size="12" maxlength="12" value=' + point.lat() + ' />'
//+ '<input type="hidden" name="PointLng" size="12" maxlength="12" value=' + point.lng() + ' /></p>'

var prosorinoUrl = "db_upload_images.php";
window.UploadForm = 'Upload Images:<br>' + '<form action="#" onSubmit=" getDataPOST(prosorinoUrl); return false;" >'
+ 'Name: <input type="text" name="name" value=' + window.tempUsername + ' />'
+ 'Email: <input type="text" name="email" value="" />'
+ 'Photo1: <input type="file" name="Photo1" value="" /><br />'
+ 'Photo2:<input type="file" name="Photo2" value="" />'
+ 'Photo3:<input type="file" name="Photo3" value="" /><br />'
+ 'Photo4:<input type="file" name="Photo4" value="" />'
+ 'Photo5:<input type="file" name="Photo5" value="" /><br />'
+ '<p><input type="submit" name="submit" value="Submit Images" /></p>'
+ '</form>';

window.Cancelform = 'CANCEL OF PROPERTY INSERTION:<br>' + '<form action="#" onSubmit="removeInputTabbedMarker(); return false;" >'
+ '<input type="submit" name="submit" value="CANCEL" />'
+ '</form>';
}
else //TOTE O XРИSTIS EXEI PATISEI TO QUERY KAI OXI TO INSERT NEW PROPERTY
{
var SpitiGreenicon = new google.maps.Icon();
SpitiGreenicon.image = "housePinGreen.gif";
SpitiGreenicon.iconSize = new google.maps.Size(30,30);
SpitiGreenicon.iconAnchor = new google.maps.Point(14,25);
SpitiGreenicon.infoWindowAnchor = new google.maps.Point(14,14);
var marker = new google.maps.Marker(point, {draggable:true, icon:SpitiGreenicon});

window.iwform = 'Enter details:<br>' + '<form action="#" onSubmit="processQ(this,htmls,labels,window.tempUsername); return false;" >'
+ '<p>House Name:<input type="text" name="PointName" size="10" maxlength="10" value="" />'
+ '<input type="hidden" name="PointLat" size="12" maxlength="12" value=' + window.MASTER_dragging_pointY + ' />'

```

```

+ '<input type="hidden" name="PointLng" size="12" maxlength="12" value=' + window.MASTER_dragging_pointX + ' /></p>'
+ '<p>Comments:<input type="text" name="Comments" size="20" maxlength="20" value="" /></p>'
+ '<p>Perioxi:<input type="text" name="Perioxi" size="15" maxlength="15" value="" />'
+ '<span>Address:<input type="text" name="Address" size="20" maxlength="20" value="" /></span>'
+ '<span>Nomos Prefecture:<input type="text" name="Nomos" size="20" maxlength="20" value="" /></span>'
+ '<input type="submit" value="Submit" /></p>'
+ '</form>';

window.Cancelform = 'CANCEL OF PROPERTY INSERTION:<br>' + '<form action="#" onsubmit="removeInputTabbedMarker(); return false;" >'
+ '<input type="submit" name="submit" value="CANCEL" />'

+ '</form>';

}

var tabs = [];

// --- ADDLISTENER GIA EMFANISI PARATHIROU EISAGOGIS ---
google.maps.Event.addListener(marker, "click", function(overlay , point) {
window.lastmarker = marker;
tabs=[];//oste an patisei poles fores to infowindow na min megalonoun ta tabs
// adjust the width so that the info window is large enough for this many tabs
if (htmls.length > 2) { htmls[0] = '<div style="width:'+(htmls.length+2)*88+'px">' + htmls[0] + '</div>';}

tabs.push(new google.maps.InfoWindowTab("Input",window.iwform)); //i formitsa einai must
if (document.forms["MESITIS_RADIO"].MESITIS_RADIO[0].checked == true) { tabs.push(new google.maps.InfoWindowTab("Photos",window.UploadForm)); } //i formitsa einai must
gia upload mono gia neo insert
tabs.push(new google.maps.InfoWindowTab("CANCEL",window.Cancelform)); //i formitsa einai must
// for (var i=0; i<htmls.length; i++) { tabs.push(new google.maps.InfoWindowTab(labels[i],htmls[i]));}
marker.openInfoWindowTabsHtml(tabs);

// marker.openInfoWindowHtml(iwform);
});

// --- ADDLISTENER GIA EMFANISI PARATHIROU ANAIRESIS -- DIAGRAFIS SIMEIOU ---
google.maps.Event.addListener(marker,"singlerightclick",function(overlay,point){
if (!overlay) {
alert("....diagrafi");
removeInputTabbedMarker();

}
});

google.maps.Event.addListener(marker,"dragend", function() {
// prepei na allakso tis sintetagmenes tou simeiou PROSOXI - apothikeyONTAI oi arxikes sti vasi!! -TO window.iwform den mporo na to kaleso den einai enerogpoimeno
// orizontas voithitiki metavliti gia na apothikeyo tin allagi kai auti na rotai i iwform oute pali epekse giati i forma orizetai prin treksei o listener ara exepei tin palia timi
// ta paratao kai tha kano to marker NON-DRAGGABLE ----SOS PROSOXI TROPOPOISA TIS SINARTISEIS PROCESS kai PROCESSQ oste na pernoun to sostes sintetagmenes apo ti
global variable kai oxi apo tin idia ti forma.. oxi poli
// epistimoniko afou oi mises values pernane mesa apo ti forma kai mono sintetagmenes anagkazomai na tis peraso mesa apo global alla ti na kano...
var markerPoint =marker.getPoint();

```

```

        window.MASTER_dragging_pointY =markerPoint.lat(); //prepei na enimerothei kai to point sti db giati i apothikeuetai sti vasi.
        window.MASTER_dragging_pointX =markerPoint.lng(); //prepei na enimerothei kai to point sti db giati i apothikeuetai sti vasi.

    });

    window.map.addOverlay(marker);
    //markerManager.addMarkers(marker,8,17); distixos den paizei kala...??den emfanizei markers
    //markerManager.refresh();
    window.lastmarker = marker;

    return marker;
    } //tou if (!lastmarker)
    else
    {
        alert("Sorry please Save Or Cancel the previous pending marker and then proceed with a new one...");
    }
} //tou createInputTabbedMarker

// == creates a "normal" marker
function createTabbedMarker(point,iwform,htmls,labels,icon,booldrawcircle) {

    var marker = new google.maps.Marker(point,icon);
    google.maps.Event.addListener(marker,"click", function() {
        // adjust the width so that the info window is large enough for this many tabs
        if (htmls.length > 2) { htmls[0] = '<div style="width:'+(htmls.length+1)*88+'px">' + htmls[0] + '</div>';}
        var tabs = [];
        tabs.push(new google.maps.InfoWindowTab("Stoixeia spitiou",iwform)); //i formitsa einai must
        for (var i=0; i<htmls.length; i++) { tabs.push(new google.maps.InfoWindowTab(labels[i],htmls[i]));}
        marker.openInfoWindowTabsHtml(tabs);
        // window.lastmarker = marker; OXI auto giati pleon den tin kalei pia i createInputTabbedMarker -ta kanei meso tis QUERY_VISIBLE_POINTS stin process tis monimes emfaniseis!
        window.lastmarkerPermanent = marker;
        window.globalid = document.getElementById("PROPERTY_ID").innerHTML; //to sozo sto div oste i sinartisi diagrafis na zitisi svisimo me vasi id kai oxi xoriki anazitisi

        //alert(globalid);
        document.getElementById("messageMARKERclicked").innerHTML = globalid;
    });

    window.map.addOverlay(marker);
    //markerManager.addMarkers(marker,8,17); distixos den paizei kala...??den emfanizei markers
        //markerManager.refresh();

    if (booldrawcircle == true) { //alert("zografise to");
        if (!document.getElementById("filterBuffer").getAttribute("disabled")) { draw_circle(point);}
    }
    else { //alert("pali malakia");
    }

    markers.push(marker); // to prostheto sto array --auto trexei toso kata tin eisagogi neou oso kai kata to diavasma meta apo refresh

```

```

return marker;
}

function uploadPhotos(prosorinoUrl) { //mallon den tha to xrisimopoisio katholou
    NUMofinputTABSprocessed = NUMofinputTABSprocessed + 1;
    getDataPOST(prosorinoUrl);
    if (NUMofinputTABSprocessed == 2) {
        // == remove the input marker and replace it with a completed marker
        window.map.closeInfoWindow();
        // anti gia to parakato
        QUERY_VISIBLE_POINTS("");

        //var marker = createTabbedMarker(window.lastmarker.getPoint(),form>window.htmls,labels);
        //google.maps.Event.trigger(marker,"click"); //me auto automata sikono to parathiro me ti formitsa oste na prosthesei pithanos fotos...

        map.removeOverlay(lastmarker);
        window.lastmarker = null; //oste na epistrepso to sosimo epomenou marker
        NUMofinputTABSprocessed = 0 ;
        }
        else { //an den exei dosei stoixeia ... tote tou to afino anoikto oste na to kanei.
            alert("you have not set the values of the property yet..");
        } //telos tou if else
    }
}

function removeInputTabbedMarker() {
    var markerPoint =lastmarker.getPoint();

    alert("telos..o marker ksanavale!" +markerPoint.lat());
    window.map.removeOverlay(lastmarker);
    window.lastmarker = null;
    NUMofinputTABSprocessed = 0 ;
}

function removeMarker(tempUsername) {
    var markerPoint = lastmarkerPermanent.getPoint();
    google.maps.Event.trigger(lastmarkerPermanent, "click");
    var temporaryID = document.getElementById("PROPERTY_ID").innerHTML;
    alert("temporaryID= " +temporaryID);

    alert("papala..o marker ksanavale!" + markerPoint.lat() + ", id=" + temporaryID);
    window.map.removeOverlay(lastmarkerPermanent);
    var url ="db_delete_location2.php?Id=" + temporaryID + "&Username=" + tempUsername;
    alert(url);
    getServerScriptforPointInsertion(url);
    window.lastmarkerPermanent = null;
    document.getElementById("messageMARKERclicked").innerHTML = null; //apla na min emfanizetai sto div
}

function removeBaseMarker(tempUsername) {
    var markerPoint = lastmarkerPermanent.getPoint();

```

```

google.maps.Event.trigger(lastmarkerPermanent, "click");
var temporaryID = document.getElementById("PROPERTY_ID").innerHTML;
alert("temporaryID= " +temporaryID);

alert("telos.o marker ksanavale!" + markerPoint.lat() + ", id=" + temporaryID);
window_map.removeOverlay(lastmarkerPermanent);
var url ="db_delete_Baselocation2.php?Id=" + temporaryID + "&Username=" + tempUsername;
alert(url);
getServerScriptforPointInsertion(url);
window.lastmarkerPermanent = null;
document.getElementById("messageMARKERclicked").innerHTML = null; //apla na min emfanizetai sto div
}

function process(form, htms,labels,tempUsername) {
    NUMofinputTABSProcessed = NUMofinputTABSProcessed + 1;

    // == obtain the data
    // var id = form.PointID.value;
    var name = form.PointName.value;
    var lat = window.MASTER_dragging_pointY; // form.PointLat.value;
    var lng = window.MASTER_dragging_pointX; // form.PointLng.value;
    var Price = form.Price.value;
    var Orofos = form.Orofos.value;
    var Tetragonika = form.Tetragonika.value;
    var Parking = form.Parking.value;
    var Comments = form.Comments.value;
    var OwnerEmail = form.OwnerEmail.value;
    var OwnerTelNumber = form.OwnerTelNumber.value;
    var Address = form.Address.value;
    var Perioxi = form.Perioxi.value;
    var Nomos = form.Nomos.value;

    var url = "db_insert_marker2.php?PointName=" +name+ "&PointLat=" +lat+ "&PointLng=" +lng+ "&Price=" +Price
        + "&Orofos=" +Orofos+ "&Tetragonika=" +Tetragonika+ "&Comments=" +Comments+ "&OwnerEmail=" +OwnerEmail
        + "&OwnerTelNumber=" +OwnerTelNumber+ "&Address=" +Address+ "&Perioxi=" +Perioxi+ "&Nomos=" +Nomos+ "&tempUsername=" + tempUsername;
        alert("trying to store...the new marker"+url);
        alert("tempUsername:"+tempUsername);

    // ===== send the data to the server
    // ===== NOTE - I can't actually do this because Googlepages doesn't do server scripts
    // ===== In your real page, remove the "/" from the beginning of the next line

    //google.maps.DownloadUrl(url, function(doc) { });
    //alert(url + getServerScriptforPointInsertion(url));//den epistrefei tipota ara den eixe noima
    //alert("..stored new:" + getServerScriptforPointInsertion(url));
    getServerScriptforPointInsertion(url);

```

```

                // if (NUMofinputTABSprocessed == 2) {
                // == remove the input marker and replace it with a completed marker
window.map.closeInfoWindow();
// anti gia to parakato

//var marker = createTabbedMarker(window.lastmarker.getPoint(),form,window.htmls,labels);
//google.maps.Event.trigger(marker,"click"); //me auto automata sikono to parathiro me ti formitsa oste na prosthesei pithanos fotos...

map.removeOverlay(lastmarker);
window.lastmarker = null; //oste na epistrepso to sosimo epomenou marker

QUERY_VISIBLE_POINTS("");
NUMofinputTABSprocessed = 0 ;
                // }
                // else {/an den exei epeksergastei fotos... tote tou to afino anoikto oste na to kanei.
                // alert("you have not uploaded photos yet..");
// } //telos tou if else
}

// ////////////////////////////////////////
// TO AKOLOUTHO EINAI GIA APOTHIKEUSI TOU SIMEIOU POU THELEIS NA TREKSEIS TO QUERY APOSTASIS //
// ////////////////////////////////////////
function processQ(form, htmls,labels,tempUsername) {
    NUMofinputTABSprocessed = NUMofinputTABSprocessed + 1;

    // == obtain the data
// var id = form.PointID.value;
var name = form.PointName.value;
var lat = window.MASTER_dragging_pointY; // form.PointLat.value;
var lng = window.MASTER_dragging_pointX; // form.PointLng.value;
var Comments = form.Comments.value;
var Address = form.Address.value;
var Perioxi = form.Perioxi.value;
var Nomos = form.Nomos.value;

var url = "db_insert_QueryPoints.php?PointName=" +name+ "&PointLat=" +lat+ "&PointLng=" +lng+ "&Comments=" +Comments
        + "&Address=" +Address+ "&Perioxi=" +Perioxi+ "&Nomos=" +Nomos+ "&tempUsername=" + tempUsername;
        alert("trying to store...the new marker, by"+ tempUsername);

// ===== send the data to the server
getServerScriptforPointInsertion(url);

                // == remove the input marker and replace it with a completed marker
window.map.closeInfoWindow();
map.removeOverlay(lastmarker);
window.lastmarker = null; //oste na epistrepso to sosimo epomenou marker

QUERY_VISIBLE_POINTS("");
NUMofinputTABSprocessed = 0 ;
}

```

```

<!-- TELOS -->
function QUERY_VISIBLE_POINTS(screen_message) {

    for( var i=0;i<window.greenCircles.length; i++){
        window.map.removeOverlay(window.greenCircles[i]);
    }

    // alert("ektelesi tou QUERY_VISIBLE_POINTS fun apo to "+ screen_message);
    // ENIMEROSI GIA NEA SIMEIA
    var parathiro_xarti = map.getBounds();
    var notioditiko_simeio = parathiro_xarti.getSouthWest();
    var vorioanatoliko_simeio = parathiro_xarti.getNorthEast();
    var SW_lat = notioditiko_simeio.lat();
    var SW_lng = notioditiko_simeio.lng();
    var NE_lat = vorioanatoliko_simeio.lat();
    var NE_lng = vorioanatoliko_simeio.lng();

    var dummyURL ;
    if (!(window.tempUsername == null)) {dummyURL = "db_query_Select_Locations_Extent.php?SW_lat=" + SW_lat + "&SW_lng=" + SW_lng + "&NE_lat=" + NE_lat + "&NE_lng=" + NE_lng +
"&Username=" + window.tempUsername;}
    else {dummyURL = "db_query_Select_Locations_Extent.php?SW_lat=" + SW_lat + "&SW_lng=" + SW_lng + "&NE_lat=" + NE_lat + "&NE_lng=" + NE_lng; }

    // alert("GLOBALurlGET_SQLfilters="+window.GLOBALurlGET_SQLfilters);
    if (!window.GLOBALurlGET_SQLfilters == "") {
        dummyURL = dummyURL + window.GLOBALurlGET_SQLfilters; //stin periptosi pou exei dosei filtra
    }

    // alert("dummyURL =" + dummyURL);
    // auto tha dimiourgisei to arxeio file.xml sto server me ta simeia pou epestrafisa
    getServerScriptforPoints(dummyURL); //esteila to searcharea gia na epistrepsei ta simeia pou empiptoun se auto

    // alert("wait 10secs"+pausecomp(10000)); Epeidi mporei na ipirxe kathisterisi sti vasi logo
    // xorikon ipologismou evala tin klisi tis readmap ENTOS tis getserver oste na kaleitai mono otan
    // teleiose i paragogi tou arxeiou file.xml oste na min exo provlimata sigxronismou...
    // i readmap kaleitai pleon entos tis getServerScriptforPoints kai gia tin eisagogi simeion eftiaksa
    // mia pio apli getServerScriptforPointInsertion

}

// //////////////////////////////////////
// A function to read the data kodikas proerxetai apo http://econym.googlepages.com/example_map11.htm
// eno i theoría tou paradeigmatos vrisketai sto http://econym.googlepages.com/basic11.htm
//
//
// genika odigies gia to pos na formareis ta xml arxeia fainontai edo http://econym.googlepages.com/basic3.htm

function readMap(localurl) {

    // OLA ta sxolia pou emfanizontai os //a einai apo tin epoxi pou douleua me to arxeio file.xml

```

```

//a var Maprequest = google.maps.XmlHttpRequest.create();
//a Maprequest.open("GET", localurl, true);
//a Maprequest.onreadystatechange = function() {
//a   if (Maprequest.readyState == 4 && Maprequest.status == 200) {

//a alert("READMAP__checkPoint1 = __"+localurl + " "+Maprequest.readyState+ Maprequest.responseText);
//a var xmlDoc = google.maps.Xml.parse(Maprequest.responseText); //epestrepse to dom object apo tin xml apantisi
var xmlDoc;
var parser;
// alert(navigator.appName);
if (navigator.appName == "Microsoft Internet Explorer") {

    xmlDoc=new ActiveXObject("Microsoft.XMLDOM");
        xmlDoc.async="false";
        xmlDoc.loadXML(localurl);

        }
    else
    {
        if (navigator.appName == "Netscape") { //piane ton Mozilla - to akoloutho to vrika apo to http://www.w3schools.com/Dom/dom_parser.asp

            parser=new DOMParser();
                xmlDoc=parser.parseFromString(localurl,"text/xml"); //epestrepse to dom object apo tin xml apantisi

                } //telos gia Mozilla
            } //telos tou parsing tou xml

// alert("META TO PARSING: "+xmlDoc);
// obtain the array of markers and loop through it
var AvailableSpitiemarkers = [];
AvailableSpitiemarkers = xmlDoc.documentElement.getElementsByTagName("marker");
    // AvailableSpitiemarkers = xmlDoc.getElementsByTagName("marker");
// var MAXIDarray = xmlDoc.documentElement.getElementsByTagName("MAXID"); //to ebkala giati ithele kai alli douleia. den tha to xrisimopiso
// MAXID=MAXIDarray[0].getAttribute("value");

// hide the info window, otherwise it still stays open where the removed marker used to be
map.getInfoWindow().hide();

// map.clearOverlays(); den mporo na to kano giati tha sviso kai to aspro kanavo tou not registered!
for (var i = 0; i < markers.length; i++) {

    // alert(" Markers.length="+ markers.length +" Removed "+ i);
    map.removeOverlay(markers[i]);
}

// empty the arrays
markers = [];
htmls = [];
ii = 0;

// alert ( "POSOI MARKERS vrethikan= " + AvailableSpitiemarkers.length + " " +markers.length);

```



```

document.getElementById('message7').innerHTML = AvailableSpitiemarkers.length;

//starting to populate the PINAKAS_APOTELESMATON --PROSOXI oxi apeuthias sto document.getElementById("PINAKAS_APOTELESMATON").innerHTML --prosethete mono tou ta tags
ton kleisimatou!!
var perioxomena_pinaka_HTML = "<table border=1>";
perioxomena_pinaka_HTML += "<tr> <th> a/a </th><th> ID </th><th> Name </th><th> Address </th><th> Orofos </th><th> m^2 </th><th> Parking </th><th> Price </th><th> Email </th><th>
Telephone Number </th><th> County </th><th> Prefecture </th></tr>";

for (var i = 0; i < AvailableSpitiemarkers.length; i++) {
  // obtain the attribues of each marker
  var neolat = parseFloat(AvailableSpitiemarkers[i].getAttribute("lat"));
  var neolng = parseFloat(AvailableSpitiemarkers[i].getAttribute("lng"));
  // alert( "marker me lat= "+ neolat + " kai lng= "+neolng);
  var point = new google.maps.LatLng(neolat,neolng);
  var id = AvailableSpitiemarkers[i].getAttribute("id");
  var pointdata = AvailableSpitiemarkers[i].getAttribute("data");
  var pointname = AvailableSpitiemarkers[i].getAttribute("name");
  var orofos = parseFloat(AvailableSpitiemarkers[i].getAttribute("orofos"));
  var tetragonika = parseFloat(AvailableSpitiemarkers[i].getAttribute("tetragonika"));
  var parking = AvailableSpitiemarkers[i].getAttribute("parking");
  var price = parseFloat(AvailableSpitiemarkers[i].getAttribute("price"));
  var imagedire = AvailableSpitiemarkers[i].getAttribute("imagedire");
  var email = AvailableSpitiemarkers[i].getAttribute("email");
  var telephonenumber = AvailableSpitiemarkers[i].getAttribute("telephonenumber");
  var perioxi = AvailableSpitiemarkers[i].getAttribute("perioxi");
  var nomos = AvailableSpitiemarkers[i].getAttribute("nomos");
  var address = AvailableSpitiemarkers[i].getAttribute("address");

var neaForma = '<FONT size=1pts> Available House details:<br>' + '<form>'
  + '<p><strong>House ID Number: </strong><span id="PROPERTY_ID">' + id + '</span> </p>'
  + '<p><strong>Address: </strong>' + address + ' ' + pointname + ' </p>'
  + '<p><strong>Point Latitude: </strong>' + neolat + '&deg</p>'
  + '<p><strong>Point Longitude: </strong>' + neolng + '&deg</p>'
  + '<p><strong>Price: </strong>' + price + '&euro </p>'
  + '<p><strong>Orofos: </strong>' + orofos + ' </p>'
  + '<p><strong>Tetragonika: </strong>' + tetragonika + ' </p>'
  + '<p><strong>Comments: </strong>' + pointdata + ' </p>'
  + '<p><strong>Owner Email: </strong>' + email + ' </p>'
  + '<p><strong>Owner Tel Number: </strong>' + telephonenumber + ' </p>'
  + '<p><strong>Perioxi: </strong>' + perioxi + ' </p>'
  + '<p><strong>Nomos Prefecture: </strong>' + nomos + ' </p>'
  + '<p><strong>Parking: </strong>' + parking + ' </p>'
  + '</form></FONT>';

var dummystring = 'Some stuff to display in the<br>Second Info Window with an image<br>'
  + '<br>'
  + ''
  + '<br>'

```

```

+ '';

var marker;
window.globalid = id; //prosoxi to id den paramenei ekτος tis sinartisis otan ginei trigger to click...thelei psaksimo...mesoveziki lisi to globalid--to theto kai sto onevent click
var dummystringCANCELform = 'REMOVAL OF PROPERTY:<br>' + '<form action="#" onsubmit="#" removeMarker(window.tempUsername); return false; ">'
+ '<input type="submit" name="submit" value="CANCEL" />'
+ '</form>';
// Whenever you make calls from HTML to Javascript, e.g. from the "onclick" of a button, the called Javascript executes in global context. This means that any functions or variables referenced in
the "onclick" string must be global.
// to parapano apo to http://econym.googlepages.com/scope.htm

var SpitiRedicon = new google.maps.Icon();
SpitiRedicon.image = "housePinRed.gif";
SpitiRedicon.iconSize = new google.maps.Size(30,30);
SpitiRedicon.iconAnchor = new google.maps.Point(14,25);
SpitiRedicon.infoWindowAnchor = new google.maps.Point(14,14);

marker = createTabbedMarker(point,neaForma,[ dummystring , dummystringCANCELform],["Photos","DELETE"],SpitiRedicon,false);
// map.addOverlay(marker); DEN XREIAZETAI GIATI TO KANO MESA STIN createTabbedMarker

map.getInfoWindow().hide();

//populating the "PINAKAS_APOTELESMATON"
perioxomena_pinaka_HTML += "<tr><td> "+i+" </td><td> "+id+" </td><td> "+pointname+" </td><td> "+address+" </td><td> "+orofos+" </td><td> "+tetragonika+" </td><td> "+parking+"
</td><td> "+price+" </td><td> "+email+" </td><td> "+telephonenum+" </td><td> "+perioxi+" </td><td> "+nomos+" </td></tr>";
} // tou for

//FINISHING to populate the PINAKAS_APOTELESMATON
perioxomena_pinaka_HTML += "</table>";
document.getElementById("PINAKAS_APOTELESMATON").innerHTML = perioxomena_pinaka_HTML
//alert(document.getElementById("PINAKAS_APOTELESMATON").innerHTML);

// TORA THA PREPEI NA FEROU KAI TA SIMEIA ENDIATERONTOS TOU XRISTI POU EIXE APOTHIKEUSEU PALIA
var AvailableBasemarkers = [];
AvailableBasemarkers = xmlDoc.documentElement.getElementsByTagName("Basemarkers");
document.getElementById("GreenHousesNo").innerHTML = AvailableBasemarkers.length;
for (var i = 0; i < AvailableBasemarkers.length; i++) {
// obtain the attributes of each marker
var neolat = parseFloat(AvailableBasemarkers[i].getAttribute("lat"));
var neolng = parseFloat(AvailableBasemarkers[i].getAttribute("lng"));
var point = new google.maps.LatLng(neolat,neolng);
var id = AvailableBasemarkers[i].getAttribute("id");
var pointdata = AvailableBasemarkers[i].getAttribute("data");
var pointname = AvailableBasemarkers[i].getAttribute("name");
var perioxi = AvailableBasemarkers[i].getAttribute("perioxi");
var nomos = AvailableBasemarkers[i].getAttribute("nomos");
var address = AvailableBasemarkers[i].getAttribute("address");

var neaFormaBase = '<FONT size=1pts> Point of Interest details:<br>' + '<form>'
+ '<p><strong>House ID Number: </strong><span id="PROPERTY_ID">' + id + ' </span> </p>'

```

```

+ '<p><strong>Address: </strong>' + address + ' ' + pointname + ' </p>'
+ '<p><strong>Point Latitude: </strong>' + neolat + '&deg</p>'
+ '<p><strong>Point Longitude: </strong>' + neolng + '&deg</p>'
+ '<p><strong>Comments: </strong>' + pointdata + ' </p>'
+ '<p><strong>Perioxi: </strong>' + perioxi + ' </p>'
+ '<p><strong>Nomos Prefecture: </strong>' + nomos + ' </p>'
+ '</form></FONT>';

```

```

var SpitiGreenicon = new google.maps.Icon();
SpitiGreenicon.image = "housePinGreen.gif";
SpitiGreenicon.iconSize = new google.maps.Size(30,30);
SpitiGreenicon.iconAnchor = new google.maps.Point(14,14);
SpitiGreenicon.infoWindowAnchor = new google.maps.Point(14,14);

```

```

var MetroStationIcon = new google.maps.Icon();
MetroStationIcon.image = "METROICON_SQ.ICO";
MetroStationIcon.iconSize = new google.maps.Size(20,20);
MetroStationIcon.iconAnchor = new google.maps.Point(14,14);
MetroStationIcon.infoWindowAnchor = new google.maps.Point(14,14);

```

```

var Basemarker;
window.globalid = id; //prosoxi to id den paramenei ekto tis sinartisis otan ginei trigger to click...thelei psaksimo...mesoveziki lisi to globalid--to theto kai sto onevent click
var dummysstringCANCELform = 'REMOVAL OF PROPERTY:<br>' + '<form action="#" onsubmit="removeBaseMarker(window.tempUsername); return false; " >'
+ '<input type="submit" name="submit" value="CANCEL" />'

```

```

+ '</form>';

```

// Whenever you make calls from HTML to Javascript, e.g. from the "onclick" of a button, the called Javascript executes in global context. This means that any functions or variables referenced in the "onclick" string must be global.

```

// to parapano apo to http://economy.googlepages.com/scope.htm

```

```

if (pointdata == "METRO") {
    if (document.getElementById("filterBuffer").value % 2 == 1) {
        Basemarker = createTabbedMarker(point,neaFormaBase,[],[],MetroStationIcon,true);
    }
    else { Basemarker = createTabbedMarker(point,neaFormaBase,[],[],MetroStationIcon,false);}
}
else
{ //alert(pointdata);
  Basemarker = createTabbedMarker(point,neaFormaBase,[ dummysstringCANCELform],["DELETE"],SpitiGreenicon,true);
  // map.addOverlay(Basemarker); DEN XREIAZETAI GIATI TO KANO MESA STIN createTabbedMarker
}

map.getInfoWindow().hide();
} // tou for

```

```

//a    delete Maprequest;
//a    Maprequest = null;

//a    } // tou if == 4

//a    } //tou readystatechange
//a    Maprequest.send(null);
}

<!-- Call this function when the page has been loaded -->
function initialize() {
    <!-- Gia kapoio logo otan kano refresh paramenoun stin prosorini mnimi ton objects ton formon oi palies times -->
    <!-- anagkazomai na ta svino epi toutou giati an meinoun ta checkboxes ton filtron anamena tote leitourgoun anapoda -->
    document.forms["MESITIS_RADIO"].reset();//SOS

    <!-- me zoom 6 kentrareis sxetika kala ellada athina sto 38.0 23.8 me zoom sto 11 -->
    window.map = new google.maps.Map2(document.getElementById("map"));
    map.setCenter(new google.maps.LatLng(38.0, 23.8), 10); // to zoom paei apo 0 (olokliri i gi eos 17)

    map.addControl(new google.maps.LargeMapControl());
    var scalePos = new google.maps.ControlPosition(G_ANCHOR_BOTTOM_RIGHT, new google.maps.Size(25,20));
    //epeidi iparxei allilokalipsi kai kaliptei to searchbar to
    //lisi perigrafetai sto http://groups.google.com/group/Google-Maps-API/browse_thread/thread/3583c3f1ef9c0812
    map.addControl(new google.maps.ScaleControl(), scalePos);
    map.addControl(new google.maps.MapTypeControl());
    var mapTypeControl = new google.maps.MapTypeControl();
    var topRight = new google.maps.ControlPosition(G_ANCHOR_TOP_RIGHT, new google.maps.Size(10,10));
    map.addControl(mapTypeControl);
    // markerManager = new google.maps.MarkerManager(map);distixos den paizei kala...??den emfanizei markers-to exo akirosei

    // Create our "tiny" marker icon

    var blueIcon = new google.maps.Icon(G_DEFAULT_ICON);
    blueIcon.image = "http://www.google.com/intl/en_us/mapfiles/ms/micons/blue-dot.png";

    // Set up our google.maps.MarkerOptions object
    markerOptions = { icon:blueIcon };

    // Add 10 markers to the map at random locations
    var bounds = map.getBounds();
    var southWest = bounds.getSouthWest();
    var northEast = bounds.getNorthEast();
    var lngSpan = northEast.lng() - southWest.lng();
    var latSpan = northEast.lat() - southWest.lat();

    <!-- enimerosi gia to neo kentro tou xarti (sintentagmenes) -->
    var center = map.getCenter();
    var center2string = center.toString();

```

```

document.getElementById("message2").innerHTML = center2string.substring(center2string.search(",")+2 ,center2string.search(",")+6 );
document.getElementById("message3").innerHTML = center2string.substring(1,6);
document.getElementById("message5").innerHTML= map.getZoom();

<!-- prosthiki search bar sto xarti -->
map.enableGoogleBar();

<!-- sinexis parakolouthisi me listener -->
google.maps.Event.addListener(map, "dragend",
    function() {
        var center = map.getCenter();
        var center2string = center.toString();
        //message 2 einai to lat kai to message 3 to lon
        document.getElementById("message2").innerHTML = center2string.substring(center2string.search(",")+2 ,center2string.search(",")+7 ); //lat
        document.getElementById("message3").innerHTML = center2string.substring(1,6); //lon
        QUERY_VISIBLE_POINTS("dragend");
    }
);
google.maps.Event.addListener(map, "zoomend",
    function(oldLevel, newlevel) {
        document.getElementById("message5").innerHTML = map.getZoom();
        QUERY_VISIBLE_POINTS("zoomend");
    }
);

// map.enableDoubleClickZoom();
map.enableContinuousZoom();
map.enableScrollWheelZoom();
// draw_circle(map.getCenter()); // ok paizei

// == Listen for map click and add an input marker
google.maps.Event.addListener(map,"click",function(overlay,point){
    if (!overlay) {
        if (document.forms["MESITIS_RADIO"].MESITIS_RADIO[0].checked == true) { //INSERTION
            //alert(document.forms["MESITIS_RADIO"].MESITIS_RADIO[0].checked);
            //alert(document.forms["MESITIS_RADIO"].MESITIS_RADIO[1].checked);
            createInputTabbedMarker(point, ["Tab 1 contents","Tab 2 contents"],["Photos","CANCEL"]);
            QUERY_VISIBLE_POINTS("");
        }
        else //QUERY
        {
            createInputTabbedMarker(point, [],[]);
            QUERY_VISIBLE_POINTS("");
        }
    }
});

google.maps.Event.addListener(map,"rtgclick", function() {

```

```

        alert("...tha ginei mellontika epilogi gia svisimo...");
    });

    <!-- Emfanisi enos layer me 20% transparency oste na min mporo na do ta apotelesmata mexri na kano register -->
    <!-- to paradeigma vrisketai sto code.google.com/apis/maps/documentation/overlays.html -->
    if (google.maps.BrowserIsCompatible()) {

        // Set up the copyright information
        // Each image used should indicate its copyright permissions

        var myCopyright = new google.maps.CopyrightCollection("© ");
        myCopyright.addCopyright(new google.maps.Copyright('Demo',
            new GLatLngBounds(new google.maps.LatLng(-90,-180), new google.maps.LatLng(90,180)),0,'©2007 Google'));

        // Create the tile layer overlay and
        // implement the three abstract methods
        var theTileLayer = new google.maps.TileLayer(myCopyright);
        theTileLayer.getTileUrl = function() { return "/tile_crosshairs_white_bg.bmp"; };
        theTileLayer.isPng = function() { return false; };
        theTileLayer.getOpacity = function() { return 0.8; };

        // alert(" myTileLayerHidden EINAI =" + myTileLayerHidden);
        if(myTileLayerHidden == false) {
            myTileLayer = new google.maps.TileLayerOverlay(theTileLayer);
            map.addOverlay(myTileLayer);
        }

    }

    <!-- marker manipulation O protos kentrikos -->
    // var marker0 = new google.maps.Marker(center, {draggable: true});
    // google.maps.Event.addListener(marker0, "dragstart", function() {
    //     map.closeInfoWindow();
    // });
    // google.maps.Event.addListener(marker0, "click", function() {
    //     marker0.openInfoWindowHtml("Just checking...");
    // });
    // google.maps.Event.addListener(marker0, "dragend", function() {
    //     marker0.openInfoWindowHtml("Just bouncing along...");
    // });
    // map.addOverlay(marker0);

    <!-- GIA DIAVASMA SIMEION -SPITION POU VRISKONTAI APOTHIKEUMENA STI VASI -->
    // apo to paradeigma...http://econym.googlepages.com/basic11.htm
    var side_bar_html = "";

```

```

var gmarkers = [];
var htms = [];
var ii = 0;

// getServerScriptforPoints("db_query_locations2.php");
//alert("gia zoomare tora...ligo pio diskolo");
//map.setZoom(13);
// readMap("file.xml");
QUERY_VISIBLE_POINTS("initialization");

<!-- to akoloutho einai apokleistikai gia to slideshow - ARXI -->
var searchControl = new google.search.SearchControl();
searchControl.addSearcher(new google.search.WebSearch());
searchControl.addSearcher(new google.search.NewsSearch());
searchControl.draw(document.getElementById("searchcontrol"));

var samples = "http://www.google.com/uds/solutions/slideshow/sample.rss";
var options = {
  displayTime: 2000,
  transistionTime: 600,
  scaleImages : true,
  linkTarget : google.feeds.LINK_TARGET_BLANK
};
new GFslideShow(samples, "slideshow", options);
<!-- to proigoumeno einai apokleistikai gia to slideshow - TELOS -->

<!-- to akoloutho einai apokleistika gia to autoComplete ton Dimon (diavazo pinaka vasis gia na voithiso sosti eisagogi stoixeiiion - ARXI -->
getServerScriptforPointInsertion("db_retrieve_Dimoi.php");
var Dimoirequest = google.maps.XmlHttpRequest.create();
Dimoirequest.open("GET", "Dimoi.xml", true);
Dimoirequest.onreadystatechange = function() {
  if (Dimoirequest.readyState == 4 && Dimoirequest.status == 200) {
    // alert("11111111111111111111111111111111");
    var xmlDimoiDoc = google.maps.Xml.parse(Dimoirequest.responseText); //epestrepse to dom object apo tin xml apantisi
    // alert("11111111111111111111111111111111" + Dimoirequest.responseText);
    // alert("22222222222222222222222222222222" + " " + xmlDimoiDoc);
    // obtain the array of Dimoi and loop through it
    var DimoiArray = [];
    DimoiArray = xmlDimoiDoc.documentElement.getElementsByTagName("marker");//? DEN DOULEUEI????????????????
    // alert("DimoiArray=" + DimoiArray.length);
    var debugString;
    for (var i = 0; i < DimoiArray.length; i++) {
      window.myArray1.push(DimoiArray[i].getAttribute("dimosName"));
    }
  }
}

```

```

        debugString += DimoiArray[i].getAttribute("dimosName");
    }
    // alert(window.myArray1.length + " " + debugString);

    delete Dimoirequest;
    Dimoirequest = null;
} //tou if==4

} //tou onreadystatechange toy DimoiRequest

Dimoirequest.send(null);
<!-- to proigoumeno einai apokleistikai gia to autoComplete ton Dimon - TELOS -->

};
//google.setOnLoadCallback(initialize); --DEN xreiazetai giati kalo to body onload initialize

```

</script>

```

<!-- GOOGLE MAPS END          -->
<!-- //////////////////////////////////////////////////////////////////// -->

```

```

<!-- //////////////////////////////////////////////////////////////////// -->
<!-- // AUTOCOMplete -->
<!--CSS file (default YUI Sam Skin) -->
<link type="text/css" rel="stylesheet" href="http://yui.yahooapis.com/2.4.1/build/autocomplete/assets/skins/sam/autocomplete.css">

```

```

<!-- Dependencies -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.4.1/build/yahoo-dom-event/yahoo-dom-event.js"></script>

```

```

<!-- OPTIONAL: Get (required only if using Script Node DataSource) -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.4.1/build/get/get-beta-min.js"></script>

```

```

<!-- OPTIONAL: Connection (required only if using XHR DataSource) -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.4.1/build/connection/connection-min.js"></script>

```

```

<!-- OPTIONAL: Animation (required only if enabling animation) -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.4.1/build/animation/animation-min.js"></script>

```

```

<!-- OPTIONAL: JSON (enables JSON validation) -->
<script type="text/javascript" src="http://yui.yahooapis.com/2.4.1/build/json/json-beta-min.js"></script>

```

```

<!-- Source file -->

```



```

<script type="text/javascript" src="http://yui.yahooapis.com/2.4.1/build/autocomplete/autocomplete-min.js"></script>

<script type="text/javascript">
// A JavaScript Array DataSource
var myArray1 = ["ano_kolopetinita", "canapitsa", "bartholomio", "boultepsi", "coritsa", "cocinogouli", "capandriti"];
var myDataSource = new YAHOO.widget.DS_JSArray(myArray1);

// A JavaScript Function DataSource
var myFunction = function() {
var myArray2 = ["d", "e", "f"];
return myArray2.reverse();
}
var myDataSource2 = new YAHOO.widget.DS_JSFunction(myFunction);

// An XHR DataSource
var myServer = "/someScript.php";
var mySchema = ["ResultItem", "KeyDataField"];
var myDataSource3 = new YAHOO.widget.DS_XHR(myServer, mySchema);

// A Script Node DataSource
var myProxylessServer = "http://proxylessserverrequest.com?param=key";
var mySchema2 = ["ResultItem", "KeyDataField"];
var myDataSource4 = new YAHOO.widget.DS_ScriptNode(myServer, mySchema);
// -----

</script>

<!-- // AUTOCOMPLETE -->
<!-- //////////////////////////////////////////////////////////////////// -->

</head>

<body class="yui-skin-sam" onload="initialize()" onunload="google.maps.Unload()"> <!-- to yui skin gia ta yui -->

<h1> <center> E-MESITIS </center> </h1>
<div class="exampleIntro">
</div>
<!--BEGIN SOURCE CODE FOR EXAMPLE ===== -->
<div id="demo" class="yui-navset yui-navset-top">
  <ul class="yui-nav">

    <li title="" class=""><a href="#tab1"><em>LOGIN page</em></a></li>
    <li title="active" class="selected"><a href="#tab3"><em>GIS - Maps</em></a></li>

  </ul>
<div class="yui-content">

```

```

<div style="display: none;" id="tab1"><p>
    <div id="DB_name_check_query" width =100 height=100>DB_name_check_query (default Donald, OConnell-OR- Douglas, Grant)=</div>
    <!-- // LOGIN START -->
    <p><center>
        <form name="f1" onsubmit="return call_login();">
        <table bgcolor="#ccccff" border="0" cellpadding="3" cellspacing="1" width="287">
            <tbody><tr>
                <td colspan="2" align="left" width="275"><b><font color="#000080" size="5">Login</font></b></td>
            </tr>
            <tr>
                <td align="right" width="81"><b><font color="#000080">UserName:</font></b></td>
                <td width="184"><input name="username" id="user" size="20" value="" type="text"></td>
            </tr>
            <tr>
                <td align="right" width="81"><b><font color="#000080">Password:</font></b></td>
                <td width="184"><input name="password" size="20" value="" type="password"></td>
            </tr>
            <tr>
                <td colspan="2" align="center" width="275"><div id="LOG_Button"> <input name="a1" value="Login" onclick="call_login()" type="button"> </div></td>
            </tr>
        </tbody></table>
        </form>
    </center>
    <!-- // LOGIN END -->
</div>

<div style="display: block;" id="tab3">
    <table border=2>
    <tr> <!-- 1i grammi -->
        <td rowspan="2">
            <!-- 1i stili 1is grammis -->
            <div id="map" style="width: 900px; height: 520px"></div>
        </td>
        <td style="color: #4444ff;"> <!-- 2i stili 1is grammis -->
            <div id="side_bar"></div>
            <div></div>
            <span id="message1"><strong> LAT and LON: </strong></span><span id="message2"> LAT </span><span> and </span><span id="message3"> LON
        </td>
    </tr>
    </table>
</div>

```

```
<span id="message4"><strong> Zoom= </strong></span><span id="message5"> Zoom_level </span>
</div>
<div>
<span id="message6"><strong> Points Recovered from Oracle DB: </strong></span><span id="message7"> __ </span>
<div id="GreenHousesNoName"><strong> Points of Interest Recovered from Oracle DB: </strong></span><span id="GreenHousesNo"> __ </div>
</div>
<span id="messageNotRegistered"><strong style="color: red;"> "Sorry but you have not registered yet...Page will be unavailable"</strong></span>
<!-- <br /> <input type="button" name="go to tab 1" value="gototab1" onclick="makeTabactive(1)" /><br />
<input type="button" name="delete record" value="delete" onclick="makeTabactive(1)" />
<a href="" onclick="makeTabactive(1)" >To login go here </a> -->
<div id="messageMARKERclicked"></div>
</td>
</tr>
<tr> <!-- 2i grammi -->
<td style="color: #4444ff;"> <- MAP showing the search area of EMESITIS <br /> please refine your search<br />
<form NAME="MESITIS_RADIO" action="#" onsubmit="filter(window.tempUsername); return false;" >
<table cellpadding="2" align="center" style="color: #4444ff;">
<tr >
<td align="center">
<INPUT TYPE="radio" NAME="MESITIS_RADIO" VALUE="INSERT" >INSERT House for Sale <br />
<INPUT TYPE="radio" NAME="MESITIS_RADIO" VALUE="FILTERING_RESULTS" CHECKED>Insert Point of Interest for
current User
</td>
</tr>
</table>
<div> </div><div> </div>
<table cellpadding="2" align="center" style="color: #4444ff;">
<tr >
<td align="center"><FONT size=1pts> Filtering Available Houses - Options:<br>
<p><input type="checkbox" name=ChkBoxFilters" value="chkboxPrice" onclick="enableTxtPrice();" /> <strong>Max Price: </strong> <span
id="txtBoxPrice" > <input id="filterPrice" name="filterPrice" size="10" value="100000" type="text" disabled="true">&euro </span></p>
<p><input type="checkbox" name=ChkBoxFilters" value="chkboxOrofos" onclick="enableTxtOrofos();" /> <strong>Max Orofos: </strong> <span
id="txtBoxOrofos" > <input id="filterOrofos" name="filterOrofos" size="2" value="2" type="text" disabled="true"></span></p>
<p><input type="checkbox" name=ChkBoxFilters" value="chkboxTetragonika" onclick="enableTxtTetragonika);" /> <strong>Min Tetragonika:
</strong><span id="txtBoxTetragonika" > <input id="filterTetragonika" name="filterTetragonika" size="4" value="90" type="text" disabled="true" ></span></p>
<p><input type="checkbox" name=ChkBoxFilters" value="chkboxParking" onclick="enableTxtParking);" /> <strong>Parking: </strong> <span
id="txtBoxParking" > <input id="filterParking" name="filterParking" size="3" value="YES" type="text" disabled="true" ></span></p>
</div> </div>
<p><input type="checkbox" name=ChkBoxFilters" value="chkboxBufferDistance" onclick="enableTxtBuffer);" /> <strong>BUFFER DISTANCE:
</strong><span id="txtBoxBuffer" > <input type="text" id="filterBuffer" name="Buf_Distance" size="7" maxlength="7" value="5000" disabled="true" /></span>meters</p>
<p><input type="checkbox" name=ChkBoxFilters" value="chkboxMetroBuffer" /> <strong>METRO stations included in filter </strong> <span
id="txtMetroBuffer" > </span></p>
<p><center> <INPUT name="Apply Filters" TYPE="button" value="Apply Filters" onClic="filter(window.tempUsername);" />
<span id="Remove_filter_button"> <INPUT name="Remove Filters" TYPE="button" value="Remove Filters"
onClic="Removefilter(window.tempUsername);" disabled="true" /> </span>
</center></p>
<!-- <p><center> <button name="Apply_Filters" value="" onClic="filter(window.tempUsername)"> Apply Filters </button> </center></p> -->
<!-- Apo to JAVascript Bible selida 550 - i diafora apo to input submit einai oti to button exei ton titlo metaksis tou begin end tag kai den stelnei ta stoixeia sto
server -->

</FONT>
```

```

                </td>
            </tr>

        </table>

    </form>

    <!-- /////////////////////////////////////////////////////////////////// -->
    <!-- // AUTOCOMPLETE
    <p><div id="myAutoComplete" class="yui-ac"> PLEASE SPECIFY AREA OF INTEREST (autocomplete box)
        <input size="20" id="myInput" class="yui-ac-input" autocomplete="off" type="text">
        <div id="myContainer" class="yui-ac-container"><div class="yui-ac-content" style="display: none;"><div class="yui-ac-hd" style="display: none;"><div class="yui-ac-bd"><ul><li
style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li><li style="display: none;"></li></ul></div><div class="yui-ac-ft" style="display: none;"></div></div>
        </div></div></div>
        <script type="text/javascript">
        var myAutoComp = new YAHOO.widget.AutoComplete("myInput", "myContainer", myDataSource);
        </script>
    <!-- // AUTOCOMPLETE -->
    <!-- /////////////////////////////////////////////////////////////////// -->
</td>

<tr> <!-- 3i grammi ADEIA -->

</tr>

<tr> <!-- 4i grammi ADEIA -->

</tr>
</table>
<div> </div><div> </div><div> </div>
<center> <div id="PINAKAS_APOTELESMATON"></div> </center><!-- Dynamic content of table -->
<div id="searchcontrol"></div>
<div id="slideshow" class="gss">Loading...</div>
<div id="dummyholderDimon"></div>

</div> <!-- //tou tritou tab -->
</div> <!--tou yui content -->
</div>
<script>
(function() {
    var tabView = new YAHOO.widget.TabView('demo');

    YAHOO.log("The example has finished loading; as you interact with it, you'll see log messages appearing here.", "info", "example");
})();
</script>

<!--END SOURCE CODE FOR EXAMPLE ===== -->

<p align="center">Copyright Mike Paigniannis &copy;2007 <br /> <a href="http://users.myqnet.gr/%7Empai"> Personnal Webpage</a> </p>
<body background="CRYSTAL.JPG">

```

```
<!-- http://localhost/kalo/ -->
```

```
</body>
```

```
</html>
```



## ***Βιβλιογραφία***

- [AJAX&PHP]** AJAX and PHP , Building Responsive Web Applications , Cristian Darie Bogdan Brinzarea, Filip Cherecheș-Toșa, Mihai Bucica, PACKT PUBLISHING, 2006
- [AJAXDUM]** Ajax FOR DUMmIES by Steve Holzner, Wiley Publishing 2006
- [AJAXGAR1]** <http://www.adaptivepath.com/publications/essays/archives/000385.php>
- [AJAXGAR2]** [http://www.adaptivepath.com/images/publications/essays/ajax-fig2\\_small.png](http://www.adaptivepath.com/images/publications/essays/ajax-fig2_small.png)
- [AJAXGIS WEB SERVICES]** Integrating AJAX Approach into GIS Visualization Web Services Ahmet Sayar, Marlon Pierce1 and Geoffrey Fox Indiana University asayar, mpierce, gcf}@cs.indiana.edu NASA's Earth-Sun System Technology Office and the National Science Foundation's National Middleware initiative. Δυστυχώς στο άρθρο δεν αναφερόταν ημερομηνία – ήταν μεταγενέστερο του 2005.
- [COSMOONE]** <http://www.cosmo-one.gr/papers.htm> 7/2008
- [GGLMAP]** Google maps web site: <http://maps.google.com/> και το Google Maps API documentation: <http://www.google.com/apis/maps/documentation/>, 10/7/2008
- [GOOGLEMAPS]** <http://code.google.com/apis/ajaxsearch/>
- [IEEESP204]** IEEE SPECTRUM 2-2004 “The Web Within the Web” Enrique Castro-Leon
- [IEEESP707]** IEEE SPECTRUM 7-2007 SPECIAL REPORT MEGACITIES
- [JAVABIBLE]** JavaScript® Bible, Gold Edition Danny Goodman, Εκδόσεις Hungry Minds 2001
- [JAVADUM]** JavaScript FOR DUMmIES, 4TH EDITION, Emily Vander Veer Wiley Publishing 2005
- [JAVAIIDIOT]** Complete Idiot's Guide to JavaScript, Second Edition, Εκδόσεις QUE

<b>[RAMGOOGL1]</b>	“§12 Πτήσεις με το Google ” RAM Ιούλιος 2006 “Το απόλυτο εγχειρίδιο του Google”
<b>[RAMGOOGL2]</b>	“§13 Google Maps mashups ;-)” RAM Ιούλιος 2006 “Το απόλυτο εγχειρίδιο του Google”
<b>[TIPSHOME]</b>	Tips & Traps When Buying A Home, Robert Irwin , Mcgraw Hill-3Rd Ed-2004
<b>[WIKI08]</b>	<a href="http://en.wikipedia.org/wiki/Property">http://en.wikipedia.org/wiki/Property</a> 7/2008
<b>[WK HTTP08]</b>	<a href="http://en.wikipedia.org/wiki/HTTP">http://en.wikipedia.org/wiki/HTTP</a> 7/2008
<b>[WK PHP08]</b>	<a href="http://en.wikipedia.org/wiki/PHP">http://en.wikipedia.org/wiki/PHP</a> 7/2008
<b>[WK SOAP08]</b>	<a href="http://en.wikipedia.org/wiki/SOAP">http://en.wikipedia.org/wiki/SOAP</a> 7/2008
<b>[WK XMLHTTP08]</b>	<a href="http://en.wikipedia.org/wiki/XMLHttpRequest">http://en.wikipedia.org/wiki/XMLHttpRequest</a> 7/2008
<b>[WKJAVA]</b>	<a href="http://en.wikipedia.org/wiki/JAVASCRIPT">http://en.wikipedia.org/wiki/JAVASCRIPT</a> 7/2008
<b>[ZAMΠ08]</b>	“Real Estate Αξία, Εκτιμήσεις, Ανάπτυξη, Επενδύσεις, Διαχείριση”, Παν. Ζεντέλης, 2001
<b>[MΠΕΖ]</b>	Γεωγραφικά Μεταδεδομένα: Θέματα Χρήσης και Αξιοποίησης στο Διαδίκτυο, μεταπτυχιακή εργασία Εργασία Β. Μπεζάτη, 12-2007
<b>[ΣΙΟΛ07]</b>	Παραδόσεις κου Σιόλα 2007-2008 ΕΜΠ – Μεταπτυχιακό Γεωπληροφορικής – Ανάλυση Οικιστικών Συστημάτων
<b>[ΣΟΕ08]</b>	<a href="http://www.soe.gr/Greek/Default.htm">http://www.soe.gr/Greek/Default.htm</a> 7/2008 και από διάλεξη από τον Πρόεδρο του Σώματος Ορκωτών Εκτιμητών κου Χαράλαμπου Χαραλαμπόπουλου στις 11/9/2007 στα πλαίσια του μαθήματος του Μεταπτυχιακού του ΕΜΠ «Γεωπληροφορική» “Real Estate”