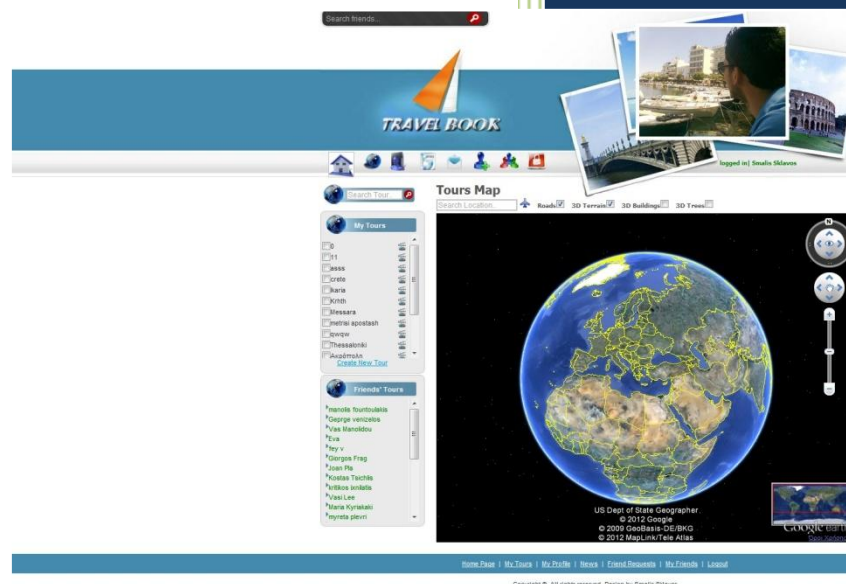


2012

Κοινωνικό δίκτυο ανασκόπησης ταξιδιών



Σκλάβος Σμάλης
Διπλωματική Εργασία
25/5/2012



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

« ΚΟΙΝΩΝΙΚΟ ΔΙΚΤΥΟ ΑΝΑΣΚΟΠΗΣΗΣ ΤΑΞΙΔΙΩΝ ΒΑΣΙΣΜΕΝΟ ΣΤΟ GOOGLEEARH »

Σκλάβος Σμάλης

Επιτροπή:

Δεληγιαννάκης Αντώνιος (Επιβλέπων)

Χριστοδουλάκης Σταύρος

Γαροφαλάκης Μίνως

Ευχαριστίες

Μου δίνεται η ευκαιρία, με την περάτωση της παρούσης διπλωματικής εργασίας να ευχαριστήσω όλα εκείνα τα πρόσωπα που με τίμησαν με τη στήριξή τους.

Πρωτίστως θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου, κ. Δεληγιαννάκης Αντώνιο, που στήριξε την προσπάθειά μου σε δύσκολες στιγμές, με καθοδήγησε και μου αφιέρωσε αρκετό από τον πολύτιμο χρόνο του, ώστε να καταφέρουμε να φτάσουμε ως εδώ.

Επίσης θερμά, θα ήθελα να ευχαριστήσω την κ. Πολυξένη Αράπη, για την βοήθεια και καθοδήγησή της στην προσπάθεια εγκατάστασης της εφαρμογής.

Δεν πρέπει να παραλείψω να ευχαριστήσω τους καθηγητές κ. Γαροφαλάκης Μίνω και κ. Χριστοδουλάκη Σταύρο για την συμμετοχή τους ως μέλη της εξεταστικής επιτροπής.

Τέλος, οφείλω ένα μεγάλο ευχαριστώ, και θέλω να εκφράσω την αγάπη και την ευγνωμοσύνη μου, στην οικογένεια μου, που στηρίζει την κάθε μου επιλογή και είναι πάντα δίπλα μου.

Περίληψη

Στη συγκεκριμένη διπλωματική εργασία γίνεται υλοποίηση μίας εφαρμογής κοινωνικής δικτύωσης, με όνομα “**Travel Book**”. Οι χρήστες της θα έχουν τη δυνατότητα να ανταλλάσσουν πληροφορίες που έχουν να κάνουν κυρίως με ταξιδιωτικές εμπειρίες. Η πληροφορία αυτή θα αποτυπώνεται τόσο χρονικά όσο και τοπικά με μεγάλη ακρίβεια χάρη στη χρήση του Google Earth, πάνω στο οποίο ο εκάστοτε χρήστης μπορεί να περιγράψει με τον καλύτερο δυνατό τρόπο την πορεία που ακολουθήθηκε κατά τη διάρκεια των ταξιδιών του και τις εμπειρίες τις οποίες βίωσε.

Η χρήση του Google Earth στην εφαρμογή μας ξεκινάει με την ενσωμάτωση του plugin το οποίο μας παρέχει δωρεάν η Google. Εκτός από το plugin η Google μας παρέχει και ένα JavaScript API στο οποίο ενσωματώνονται όλες οι λειτουργίες που ο χρήστης μπορεί να κάνει πάνω στον χάρτη, σε μορφή JavaScript συναρτήσεων, όπως σχεδιασμός markers και γραμμών, ακριβώς όπως στο Google Maps API, με τη διαφορά ότι εδώ είναι δυνατή η 3D αναπαράσταση.

Στην συγκεκριμένη εφαρμογή ο χρήστης μπορεί να σχεδιάσει ένα ταξίδι ακριβώς όπως το βίωσε πάνω στον χάρτη του Google Earth. Μπορεί να οριστεί η σειρά επισκεψιμότητας του κάθε μέρους και να περιγραφεί η εμπειρία την οποία βίωσε ο χρήστης αποτυπωμένη σε φωτογραφίες, εικονίδια περιγραφικά της δραστηριότητας του, όπως πχ ποδήλατο για ποδηλασία, και αφηγήσεις οι οποίες είναι ενσωματωμένες σε αναδυόμενα balloons. Στην περίπτωση των εικόνων η εφαρμογή μας έχει την δυνατότητα να αναγνωρίζει το που τραβήχτηκε εξάγοντας, αν υπάρχει την πληροφορία των μεταδεδομένων για τις συντεταγμένες από gps, και να τοποθετείται αυτόματα στο σημείο αυτό.

Η ανασκόπηση του ταξιδιού μπορεί να γίνει είτε με απλή απεικόνιση των σημείων στο χάρτη, ενωμένα με μία γραμμή μεταξύ τους, ώστε να φαίνεται η πορεία που ακολουθήθηκε, είτε με μορφή αναπαραγωγής ενός αφηγηματικού βίντεο. Αυτό γίνεται με τη βοήθεια της λειτουργίας touring του Google Earth, όπου οδηγούμαστε αυτόματα από σημείο σε σημείο, του οποίου προβάλλεται ταυτόχρονα η περιγραφή με τη χρήση των αναδυόμενων παραθύρων πληροφοριών. Η εφαρμογή παρέχει επίσης την δυνατότητα στους χρήστες της να συνδεθούν μεταξύ τους. Έτσι οι χρήστες που συνδέονται μεταξύ τους, μπορούν να κάνουν ανασκόπηση, εκτός των προσωπικών τους ταξιδιών, και των ταξιδιών των φίλων τους.

Οι χρήστες μπορούν να επιτελέσουν επίσης, όλες εκείνες τις λειτουργίες που σχετίζονται με το χαρακτήρα κοινωνικής δικτύωσης της εφαρμογής. Έτσι μπορούν να αναζητήσουν φίλους και να συνδεθούν με αυτούς, αποστέλλοντας αιτήματα φιλίας. Ακόμα μπορούν να επικοινωνήσουν μεταξύ τους αποστέλλοντας άμεσα μηνύματα. Τέλος μπορούν ενημερώνονται για τα πεπραγμένα των φίλων τους, λαμβάνοντας ενημερώσεις, για τις δραστηριότητες τους στην εφαρμογή, ενώ

μπορούν να δουν τις βασικές πληροφορίες των άλλων, μέσω του προφίλ που διατηρεί ο κάθε χρήστης.

Για την υλοποίηση του interface της εφαρμογής ακολουθήθηκαν οι βασικές αρχές σχεδίασης οι οποίες έχουν να κάνουν με την πλήρη ενημέρωση και καθοδήγηση του χρήστη, την ομαδοποίηση της λειτουργικότητας, την ευκολία πλοήγησης, την επιλογή χρωμάτων, τη συνέπεια, τη χρήση της ενεργής οντότητας και τέλος το χειρισμό των λαθών που μπορεί να προκύψουν από την αλληλεπίδραση του χρήστη με την εφαρμογή.

Όσον αφορά το σχεδιασμό του συστήματος, επιλέχτηκε το μοτίβο σχεδιασμού BCED (Boundary, Control, Entity, Database Interface – Layers). Η λογική της εφαρμογής σχεδιάστηκε με βάση το μοντέλο MVC pattern (Model–View–Controller), το οποίο καταφέρνει την ενοποίηση των τεχνολογιών Java Servlet και Java Server Pages με αποτέλεσμα τον αγαστό συνδυασμό των πλεονεκτημάτων της κάθε τεχνολογίας.

Για την υλοποίηση του τμήματος της εφαρμογής το οποίο είναι ορατό στο χρήστη (client side) χρησιμοποιήθηκαν εκτεταμένα οι τεχνολογίες HTML, DHTML, JavaScript και Ajax. Τα χαμηλότερα επίπεδα της εφαρμογής, δηλαδή αυτά που είναι υπεύθυνα για την εκτέλεση των αιτημάτων του χρήστη, τη διαχείριση και την αποθήκευση δεδομένων υλοποιούνται με τη χρήση της γλώσσας προγραμματισμού Java. Για την διαχείριση και υλοποίηση της βάσης δεδομένων χρησιμοποιείται η γλώσσα MySQL. Όσον αφορά τον τρόπο σύνδεσης της βάσης δεδομένων με την εφαρμογή, αυτό γίνεται με τη χρήση του κατάλληλου JDBC driver που είναι ένας οδηγός που έχει γραφτεί σε Java.

Το μεγαλύτερο μέρος της εφαρμογής αναπτύχθηκε στην πλατφόρμα ανάπτυξης δικτυακών εφαρμογών, NetBeans 6.8. Για την ανάπτυξη και διαχείριση της βάσης δεδομένων χρησιμοποιήθηκε ο MySQL server, η έκδοση του οποίου είναι η 5.1.44. Τέλος χρησιμοποιήθηκε ο Apache Server 6.1, για την προσομοίωση της εφαρμογής, ανεβάζοντας την αρχικά σε τοπικό δίκτυο, και κατόπιν σε server του πολυτεχνείου Κρήτης.

Περιεχόμενα

Ευχαριστίες	4
Περίληψη	5
Περιεχόμενα.....	7
Ευρετήριο Εικόνων	10
1. Εισαγωγή	13
1.1 Στόχος της διπλωματικής	13
1.2 Σύντομη περιγραφή λειτουργιών και δυνατοτήτων.	13
1.3 . Web Mapping Services.....	14
1.4. Mashups.....	15
1.5.Διάρθρωση επόμενων κεφαλαίων.	16
2. Google Earth	17
2.1 Γενική αναφορά στο Google Earth.....	17
2.2 Πως χρησιμοποιήθηκε το Google Earth στην εφαρμογή μας	18
2.2.1 Ενσωμάτωση Google Earth Στην εφαρμογή μας.	19
2.2.2 Βασικές εντολές του Google Earth που χρησιμοποιήθηκαν για δημιουργία και επεξεργασία ταξιδιών.	21
2.2.3 Βασικές εντολές του Google Earth που χρησιμοποιήθηκαν για την ανασκόπηση των ταξιδιών.....	28
3.Αρχιτεκτονική Συστήματος και τεχνολογίες που χρησιμοποιήθηκαν.	36
3.1 HTML και DHTML	36
3.2 JavaScript.....	37
3.2.1 Εισαγωγή στη JavaScript.....	37
3.2.2 Χαρακτηριστικά της JavaScript	37
3.2.3 Η χρήση της JavaScript και Ajax στην εφαρμογή μας	38
3.3 Java	39
3.3.1 Εισαγωγή στη Java	39
3.3.2 Τα χαρακτηριστικά της Java.....	39
3.4 Java Servlets.....	40
3.4.1 Τι είναι τα Java Servlets	40
3.4.2 Κύκλος ζωής ενός Java Servlet	41
3.5 Java Servlet Pages (JSPs).....	42
3.5.1 Η Ανάγκη για δημιουργία των JSPs	42
3.5.2 Προτυποποίηση σελίδας (page templating), η βάση της τεχνικής JSP	42
3.5.3 Η λειτουργία της JSP τεχνολογίας και η διαφορά από τα Java Servlets ...	43
3.5.4 Η χρήση της JSP τεχνολογίας στην εφαρμογή μας.	43

3.6 Αρχιτεκτονική συστήματος –Χρήση BCED και MVC	45
3.6.1 Boundary Layer	45
3.6.2 Control Layer- Χρήση MVC	45
3.6.3 Entity Layer	46
3.6.4 Database Interface Layer.	47
3.6.5 Περιγραφή βάσης δεδομένων-Σχεσιακό σχήμα.	48
3.6.6 Διαγραμματική απεικόνιση δόμησης επιπέδων της εφαρμογής (BCED) .	50
3.7 Προγραμματιστικά εργαλεία της εφαρμογής	52
3.7.1 NetBeans 6.8.....	52
3.7.2 Apache Tomcat 6.0.....	52
3.7.3 MySQL Server 5.1 - JDBC.....	53
4.Αρχές Σχεδίασης της εφαρμογής	54
4.1 Περιγραφή βασικής δομής σελίδων	54
4.2 Ενημέρωση χρήστη	56
4.3 Καθοδήγηση Χρήστη	57
4.4 Ομαδοποίηση Λειτουργιών	58
4.5 Ευκολία πλοήγησης.....	58
4.6 Επιλογή Χρωμάτων	59
4.7 Συνέπεια.....	61
4.8 Χρήση ενεργής οντότητας	62
4.9 Χειρισμός λαθών.	62
5.Λειτουργικότητας της εφαρμογής	63
5.1 Βασική Ιδέα	63
5.2 Εκτεταμένη ανάλυση απαιτήσεων.....	63
5.2.1 Εγγραφή - Είσοδος - Έξοδος στο σύστημα.....	63
5.2.2 Διαχείριση προφίλ	63
5.2.3 Αναζήτηση χρηστών και δικτύωση	64
5.2.4 Διαχείριση φίλων χρήστη	64
5.2.5 Ενημέρωση για τα γενόμενα από τους φίλους του χρήστη	64
5.2.6 Δημιουργία ταξιδιού – Γεγονότος	65
5.2.7 Συνολική διαχείριση ταξιδιών χρήστη	65
5.2.8 Επεξεργασία ταξιδιού	66
5.2.9 Ανασκόπηση προσωπικών ταξιδιών και ταξιδιών φίλων.....	66
5.2.10 Αναζήτηση ταξιδιών.....	66
5.3 Γραφική διεπαφή	67
5.3.1 Σελίδα καλωσορίσματος.....	67
5.3.2 Σελίδα εγγραφής χρήστη στο σύστημα	69

5.3.3 Homepage – Σελίδα ανασκόπησης των ταξιδιών.....	72
5.3.4 My Tours – Σελίδα διαχείρισης των ταξιδιών του χρήστη	85
5.3.5 Create New Tour –Δημιουργία νέου ταξιδιού.....	86
5.3.6 Edit Tour –Επεξεργασία ταξιδιού	94
5.3.7 My Profile –Σελίδα προσωπικού προφίλ.....	96
5.3.8 Friend’s Profile –Σελίδα προσωπικού προφίλ φίλου	99
5.3.9 My Friend Requests –Σελίδα διαχείρισης αιτημάτων φιλίας	101
5.3.10 My Friends –Σελίδα διαχείρισης φίλων.	103
5.3.11 My Messages –Inbox.....	106
5.3.12 Read Message- Σελίδα ανάγνωσης μηνύματος	107
5.3.13 New Message- Σελίδα σύνταξης μηνύματος.....	109
5.3.14 Sent Messages- Σελίδα απεσταλμένων μηνυμάτων.	111
6. Βιβλιογραφία	112

Ευρετήριο Εικόνων

Εικόνα 1. Παράδειγμα λειτουργίας ενός τυπικού GIS.....	15
Εικόνα 2. Χρήση του Navigation Control και του Overview map.	20
Εικόνα 3. Εικονίδια που ο χρήστης μπορεί να χρησιμοποιήσει για τη δημιουργία ταξιδιού.....	22
Εικόνα 4. Παράδειγμα δημιουργίας marker.	23
Εικόνα 5. Παράδειγμα χρήσης balloon (Feature Balloon).....	25
Εικόνα 6. Παράδειγμα HTML String Balloon.	26
Εικόνα 7. Παράδειγμα γεωεντοπισμού Αθήνας.....	28
Εικόνα 8. Παράδειγμα χρήσης της parseKml για δημιουργία ενός placemark.....	30
Εικόνα 9. Αναπαράσταση ενός ταξιδιού με χρήση Kml.....	30
Εικόνα 10. Αναπαραγωγή ταξιδιού καρέ καρέ.....	34
Εικόνα 11. Μπάρα ελέγχου αναπαραγωγής tour. Buttons έλεγχου ταχύτητας, stop, play και reset.....	35
Εικόνα 12. Παράδειγμα Html κώδικα (αριστερά) και του αποτελέσμάτος του (δεξιά).....	37
Εικόνα 13. Σχεδιάγραμμα λειτουργίας Ajax.....	38
Εικόνα 14. Λογότυπο Java	39
Εικόνα 15. Διαγραμματική αναπαράσταση της αρχιτεκτονικής ενός Servlet. (1): Αποστολή του αιτήματος (http request) από τον από τον client στον Server. (2): Προώθηση του αιτήματος σε ένα servlet. (3): Δημιουργία των κατάλληλων αντικειμένων από το servlet. (4): Επικοινωνία με βάση δεδομένων. (5): Αποστολή αντικειμένων πίσω στον container. (6): Δημιουργία του Http Response.	41
Εικόνα 16. Κυκλος ζωής ενός Servlet.....	42
Εικόνα 17. Παράδειγμα Jsp Κώδικα για δημιουργία του menu ταξιδιών. Μέσα σε <%> φαίνεται ο κώδικας java. Δύο τμήματα Java κώδικα περικλείονται σε κόκκινο πλαίσιο. Τα <%> περιβάλλονται από HTML κώδικα.....	44
Εικόνα 18. Αποτέλεσμα χρήσης του παραπάνω jsp κώδικα για σχηματισμό μενού ταξιδιών.	44
Εικόνα 19. Διαγραμματική απεικόνιση του MVC.....	46
Εικόνα 20. Class Diagram της εφαρμογής μας.....	47
Εικόνα 21. Διάγραμμα απεικόνισης του ρόλου των Dao.Business Object: Ο client προς τα δεδομένα. Το αντικείμενο που πρέπει να έχει πρόσβαση στην βάση. DataAccessObject: Το επίπεδο υλοποίησης της πρόσβαση στα δεδομένα. Data Source: Η βάση δεδομένων. Transfer Object: Χρησιμοποιείται για τη μεταφορά δεδομένων, δηλαδή φέρει όλα τα στοιχεία που ανακτήσαμε από τη βάση.	48
Εικόνα 22. ER που περιγράφει τη βάση δεδομένων της εφαρμογής μας.....	49
Εικόνα 23. Διάγραμμα δόμησης επιπέδων BCED.....	51
Εικόνα 24. Λογότυπο NetBeans.....	52
Εικόνα 25. Λογότυπο Apache Tomcat	52
Εικόνα 26. Λογότυπο MySQL	53
Εικόνα 27. Βασικό template.	55
Εικόνα 28. Live View του template της εφαρμογής.....	55
Εικόνα 29. Παράδειγμα ενημέρωσης χρήστη.....	57
Εικόνα 30. Ενημέρωση χρήστη με αναδυόμενα μηνύματα.	57

Εικόνα 31. Πίνακας ορθής διαχείρισης χρωμάτων.	59
Εικόνα 32. Παράδειγμα λανθασμένης εισαγωγής. Χρήση κόκκινου χρώματος για να τονιστεί το λάθος.	60
Εικόνα 33. Παράδειγμα συνδέσμων. Χρήση γαλάζιου χρώματος.	61
Εικόνα 34. Σελίδα Καλωσορίσματος.	67
Εικόνα 35. Παράδειγμα λανθασμένης εισαγωγής στοιχείων.	68
Εικόνα 36. Σελίδα εγγραφής χρήστη.	70
Εικόνα 37. Παράδειγμα συμπλήρωσης πεδίων.	71
Εικόνα 38. Αρχική σελίδα εφαρμογής, όπως φαίνεται αμέσως μετά το login,	72
Εικόνα 39. Αρχική σελίδα αφού κάνουμε zoom σε αυτή.	73
Εικόνα 40. Παράδειγμα αναζήτησης φίλων.	74
Εικόνα 41. Εικόνα προφίλ και μυνημα πραγματοποίησης login.	75
Εικόνα 42. Το μενού όταν το ποντίκι διέλθει πάνω από το εικονίδιο του my profile.	75
Εικόνα 43. Το μενού όταν υπάρχουν ειδοποιήσεις για νέα και αιτήματα φιλίας.	75
Εικόνα 44. Τα μόνου ταξιδιών στο αριστερό τμήμα της σελίδας.	76
Εικόνα 45. Παράδειγμα αναζήτησης ταξιδιού.	77
Εικόνα 46. Παράδειγμα απεικόνισης των ταξιδιών κάποιου φίλου από το μενού ταξιδιών φίλων.	78
Εικόνα 47. Παράδειγμα ταυτόχρονης απεικόνισης δύο ταξιδιών με απλή ανασκόπηση. Με λευκό χρώμα εικονίζεται το ταξίδι Κρήτη ενώ με κόκκινο το ταξίδι Messara.	79
Εικόνα 48. Παράδειγμα αναπαραγωγής live tour.	80
Εικόνα 49. Καρέ καρε αναπαραγωγή ταξιδιού Κρήτη.	81
Εικόνα 50. Παράδειγμα παράθυρου πληροφοριών για απλό placemark.	82
Εικόνα 51. Παράδειγμα παράθυρου πληροφοριών για Photo placemark.	83
Εικόνα 52. Παράδειγμα 3D απεικονίσεων.	84
Εικόνα 53. Σελίδα My Tours.	85
Εικόνα 54. Confirmation box. Ζητάει από το χρήστη να επιβεβαιώσει την επιθυμία του για διαγραφή του ταξιδιού.	86
Εικόνα 55. Σελίδα συμπλήρωσης βασικών στοιχείων κατά τη δημιουργία ταξιδιού.	87
Εικόνα 56. Σελίδα δημιουργίας σταθμών ταξιδιού.	88
Εικόνα 57. Toolkit για τη δημιουργία ταξιδιού.	89
Εικόνα 58. Παράδειγμα photo upload.	90
Εικόνα 59. Παράδειγμα δημιουργίας placemark.	91
Εικόνα 60. Αποτέλεσμα δημιουργίας placemark.	92
Εικόνα 61. Επεξεργασία βασικών πληροφοριών ταξιδιού της ενότητας Basic Tour Info.	93
Εικόνα 62. Παράδειγμα αλλαγής προτεραιότητας των placemarks, απο την ενότητα Arrange priority.	94
Εικόνα 63. Edit Tour: Σελίδα επεξεργασίας ταξιδιού.	95
Εικόνα 64. My profile. Σελίδα προσωπικού προφίλ.	96
Εικόνα 65. Παράδειγμα εμφάνισης φίλων χρήστη.	100
Εικόνα 66. Διαφοροποίηση εικονιδίων όταν το προφίλ ανήκει σε μη φίλο. Εμφάνιση εικονιδίου αποστολής αιτήματος φιλίας.	101
Εικόνα 67. Ειδοποίηση αιτήματος φιλίας.	101
Εικόνα 68. Σελίδα διαχείρισης αιτημάτων φιλίας.	102
Εικόνα 69. Παράδειγμα απαντήσεων σε αιτήματα φιλίας.	103
Εικόνα 70. Σελίδα διαχείρισης φίλων χρηστή.	104

Εικόνα 71. Παράδειγμα αναζήτησης φίλου.....	105
Εικόνα 72.Ενημέρωση ότι έχουμε ένα προσωπικό μήνυμα.....	106
Εικόνα 73.Σελίδα μηνυμάτων. Inbox.....	106
Εικόνα 74.Σελίδα ανάγνωσης προσωπικών μηνυμάτων.....	108
Εικόνα 75.Σελίδα δημιουργίας νέου μηνύματος.....	109
Εικόνα 76.Παράδειγμα αναζήτησης παραλήπτη.....	110
Εικόνα 77. Σελίδα απεσταλμένων μηνυμάτων.....	111

1. Εισαγωγή

1.1 Στόχος της διπλωματικής

Ραγδαία είναι η ανάπτυξη των κοινωνικών δικτύων τη σημερινή εποχή, με μπροστάρηδες στην προσπάθεια αυτή, γιγαντωμένα σε ισχύ, λόγω των εκατομμυρίων θαυμαστών τους, δίκτυα όπως το facebook και το twitter. Η ευρύτερη οικογένεια τέτοιου είδους εφαρμογών έχουν ως κοινό στόχο την δικτύωση χρηστών με αποτέλεσμα την ανταλλαγή κάθε είδους πληροφορίας η οποία μπορεί να διαφοροποιείται από απλές σκέψεις έως εικόνες και video από την προσωπική ζωή του κάθε χρήστη.

Στόχος της διπλωματικής αυτής εργασίας είναι η δημιουργία μίας πρότυπης εφαρμογής ανασκόπησης διακοπών. Αυτή ευελπιστούμε να αποτελέσει ένα ακόμα λίθο στην προσπάθεια για διασύνδεση των χρηστών, επιτρέποντας τους να μοιραστούν εμπειρίες που αφορούν κυρίως ταξίδια και διακοπές. Για να γίνει αυτό, επιστρατεύτηκε το πολυεργαλείο της εταιρείας Google, το Google Earth. Έχοντας σύμμαχο τη τρισδιάστατη δορυφορική χαρτογράφηση της γης που μας παρέχεται, μπορούμε να αναπαραστήσουμε την πορεία που ακολουθήσαμε κατά τη διάρκεια π.χ. των διακοπών μας με απόλυτη γλαφυρότητα και ακρίβεια, τόσο χωρικά όσο και χρονικά .

1.2 Σύντομη περιγραφή λειτουργιών και δυνατοτήτων.

Τα πρώτα βήματα στο Travel Book γίνονται κατά την εγγραφή του χρήστη και τη δημιουργία ενός προφίλ που συνεπάγεται την είσοδο του στη δικτυακή κοινότητα. Το προφίλ αυτό απαρτίζεται από προσωπικά στοιχεία που το κάθε μέλος μπορεί όποτε επιθυμεί να διαχειριστεί και να αλλάξει.

Οι χρήστες της εφαρμογής αυτής μπορούν πλέον να δημιουργήσουν ένα ταξίδι ή μία ιστορία με τα εργαλεία τα οποία τους παρέχονται. Η περιγραφή του ταξιδιού μπορεί να απαρτίζεται από εικόνες με τα σχόλια τους, μέχρι άπλα σημεία πάνω στο χάρτη του Google earth τα οποία συνοδεύονται από μια σύντομη περιγραφή-αφήγηση. Πρέπει να σημειωθεί στο σημείο αυτό, ότι στην περίπτωση των εικόνων υπάρχει η δυνατότητα αυτόματου εντοπισμού της τοποθεσίας που τραβήχτηκε και τοποθέτησή της σε αυτήν. Αυτό βέβαια απαιτεί η φωτογραφική μηχανή με την οποία τραβήχτηκε να έχει gps, ώστε τα γεωγραφικά δεδομένα να αποθηκευθούν στα μεταδεδομένα της φωτογραφίας. Μεταδεδομένα είναι επιπλέον πληροφορίες που υπάρχουν σε ένα αρχείο πχ εικόνας.

Εφόσον ολοκληρωθεί η όλη σχεδίαση του ταξιδιού τότε παρέχεται η δυνατότητα να αναπαρασταθούν τα σημεία που δημιουργήθηκαν πάνω στο χάρτη και να οδηγηθούμε σε όποιο από αυτά επιθυμούμε, βλέποντας ταυτόχρονα την εικόνα, το σημείο και την περιγραφή που το συνοδεύει στην ακριβή γεωγραφική τοποθεσία όπου διαδραματίστηκε το καθένα γεγονός.

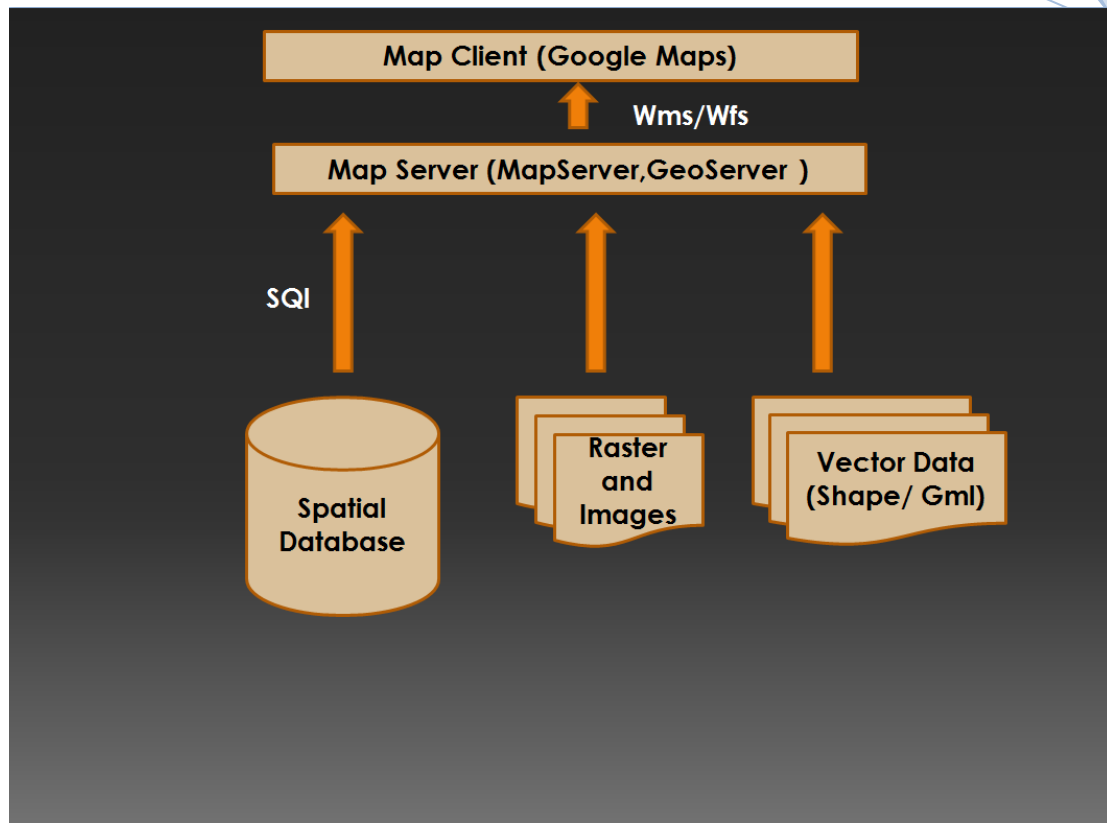
Η εμπειρία της αναπαράστασης του ταξιδιού γίνεται ακόμα πιο συναρπαστική εάν ο χρήστης επιλέξει να δει το την αναπαράσταση του ταξιδιού με μορφή ζωντανού ταξιδιού (live tour). Στην περίπτωση αυτή, γίνεται αυτόματη μεταφορά από σημείο σε σημείο με τη σωστή χρονολογική σειρά. Έτσι στην ουσία αναβιώνουμε το ταξίδι το οποίο περιγράφεται, επισκεπτόμενοι τις τοποθεσίες με τη ίδια χρονική σειρά που έγινε στην πραγματικότητα.

Για να μπορέσουν οι χρήστες να μοιραστούν τις εμπειρίες τους αυτές με φίλους τους, επίσης μέλη της δικτυακής κοινότητας της εφαρμογής, έχουν την δυνατότητα να τους αναζητήσουν με βάσει το ονοματεπώνυμο ή το email τους και να δικτυωθούν με αυτούς εάν το εγκρίνουν και οι δύο. Έτσι η δυνατότητες αναπαράστασης που περιγράφονται παραπάνω θα είναι πλέον δυνατές και για τα ταξίδια-περιγραφές των άλλων.

Τέλος, από την εφαρμογή δε θα μπορούσε να λείπει το στοιχείο της επικοινωνίας. Οι χρήστες μπορούν να συνομιλούν μεταξύ τους, ανταλλάσσοντας μηνύματα μέσω του inbox που έχει υλοποιηθεί.

1.3 . Web Mapping Services

Η εφαρμογή μας όπως είπαμε στηρίζεται στη χρήση της διαδικτυακής υπηρεσίας χαρτογράφησης (Web Mapping Service) την οποία μας παρέχει η Google, δηλαδή το Google Earth. Σύμφωνα με το Διεθνές Πρότυπο, ένα Web Mapping Service παράγει χάρτες με δυναμικά χωρικά δεδομένα από γεωγραφικές πληροφορίες, στα οποία έχει γίνει πρώτα γεωαναφορά (georeference). Η γεωαναφορά είναι διαδικασία η οποία καθορίζει την ύπαρξη ενός αντικειμένου στον φυσικό χώρο, δηλαδή την καταχώρηση της θέσης σύμφωνα με μία χαρτογραφική προβολή ή γεωγραφικές συντεταγμένες. Με τον όρο χάρτης εννοούμε την απεικόνιση των γεωγραφικών δεδομένων σε ένα, κατάλληλο για προβολή σε οθόνη υπολογιστή, αρχείο εικόνας. Το διαδίκτυο ως μέσο διάδοσης των χαρτών έδωσε σημαντική ώθηση στη χαρτογραφία, αφού προσέδωσε σημαντικά πλεονεκτήματα όπως το να έχουμε χάρτες σε πραγματικό χρόνο (real time), με φθηνότερη διάδοση, συχνές και ταυτόχρονα οικονομικές ενημερώσεις δεδομένων και λογισμικού, εξατομικευμένο περιεχόμενο χαρτών με διαφορετικές πηγές δεδομένων και την ανταλλαγή γεωγραφικών πληροφοριών. Χαρακτηριστικό γνώρισμα όσων ιστοσελίδων χρησιμοποιούν μια Web Mapping Service είναι ότι το περιεχόμενο της συμπληρώνεται από λειτουργίες χαρτογραφίας όπως για παράδειγμα η πλοήγηση (zoom and panning), αναζητήσεις, υπολογισμοί διαδρομών και σημείων ενδιαφέροντος. Συνεπώς η ιστοσελίδα δεν περιορίζεται στην απλή απεικόνιση ενός χάρτη, αλλά έχει το χαρακτήρα ενός προηγμένου διαδραστικού χάρτη με επιπρόσθετα εργαλεία. Η αρχιτεκτονική ενός τυπικού Web GIS εικονίζεται παρακάτω.



Εικόνα 1. Παράδειγμα λειτουργίας ενός τυπικού GIS

Σήμερα υπάρχουν πολλές υπηρεσίες web mapping στο διαδίκτυο. Μερικές από αυτές είναι:

- Google Maps (maps.google.com)
- Google Earth (earth.google.com)
- Bing Maps (www.bing.com/maps)
- Map Quest (www.mapquest.com)
- Nokia Maps (maps.nokia.com)
- Map24 (www.map24.com)

1.4. Mashups

Η εφαρμογή μας μπορούμε να πούμε ότι ανήκει στην ευρύτερη κατηγορία των mashups και πιο συγκεκριμένα στα map mashups. Ο όρος προέρχεται από τη μουσική και περιγράφει τη δημιουργία ενός τραγουδιού από την ανάμειξη πολλών άλλων. Όταν αναφερόμαστε σε ιστοσελίδες με τον όρο mashups εννοούμε εκείνες που «συνυφαίνουν» πληροφορίες και κώδικες από διαφορετικές πηγές, δημιουργώντας έτσι νέες διαδικτυακές υπηρεσίες. Σύμφωνα με τους Pierce, Dobson και Phillips (2009) σε ένα πραγματικό mashup τα αρχικά δεδομένα δεν φιλοξενούνται στον ίδιο υπολογιστή με τη σελίδα του, όπως συμβαίνει και με την εφαρμογή μας, όπου το μεγαλύτερο μέρος της πληροφορίας που αφορά γεωγραφικά δεδομένα είναι αποθηκευμένο όχι στον υπολογιστή μας αλλά στο server της Google. Οι πλειοψηφία των Mashups αφορά υπηρεσίες δικτυακής χαρτογράφησης. Σε αυτή την περίπτωση,

τα λεγόμενα map mashups αποτελούν μία ειδική κατηγορία δυναμικών χαρτών που σε συνεργασία με ένα ή περισσότερα API (Application Programming Interfaces), χωρίς να απαιτούνται ιδιαίτερες προγραμματιστικές γνώσεις παράγουν μία νέα και διαφορετική υπηρεσία ή έναν τροποποιημένο όπως ο χρήστης θέλει χάρτη, ο οποίος θα παρουσιαστεί σε ένα web mapping service, όπως το Google Earth, συνδυάζοντας χωρικές πληροφορίες σε αυτό. Η εφαρμογή αυτή που προκύπτει μπορεί να αφορά από μία πολύ απλή μέχρι μία εκτεταμένη εφαρμογή που καλύπτει μεγάλες γεωγραφικές περιοχές.

1.5.Διάρθρωση επόμενων κεφαλαίων.

Μέχρι στιγμής έχουμε δει σε γενικές γραμμές τι υλοποιείται στην συγκεκριμένη εφαρμογή ενώ έχουμε αναφερθεί στην ένταξη της εφαρμογής μας στις κατηγορίες των δικτύων κοινωνικής δικτύωσης αλλά και σε αυτή των mashups.

Στο επόμενο κεφάλαιο, δηλαδή το **κεφάλαιο 2**, θα αναφερθούμε στην υπηρεσία χαρτογράφησης μέσω διαδικτύου του Google earth, θα κάνουμε μία σύντομη ιστορική αναδρομή στο πώς ξεκίνησε το Google earth, ενώ θα αναλύσουμε διεξοδικά το πώς εγκαθίσταται και χρησιμοποιείται στην εφαρμογή. Επίσης θα εξετάσουμε ταυτόχρονα τις συναρτήσεις του Google earth api που μας είναι χρήσιμες.

Ακολούθως, στο **κεφάλαιο 3** θα αναφερθούμε στην αρχιτεκτονική του BCED με την οποία οικοδομείται η εφαρμογή μας, ενώ θα εξετάσουμε και όλες εκείνες τις τεχνολογίες που χρησιμοποιούνται για να έχουμε το επιθυμητό αποτέλεσμα. Στο κεφάλαιο αυτό θα γίνει επίσης μία σύντομη αναφορά στις γλώσσες προγραμματισμού που χρησιμοποιήθηκαν, όπως επίσης και στα εργαλεία ανάπτυξης της εφαρμογής.

Στο **κεφάλαιο 4** που ακολουθεί αναπτύσσεται το ζήτημα των αρχών και των κανόνων εκείνων που ακολουθήθηκαν για την ορθή σχεδίαση του user interface της εφαρμογής.

Τέλος, στο **κεφάλαιο 5**, θα γίνει μία εκτενής ανάλυση της λειτουργικότητας της εφαρμογής, χρησιμοποιώντας τα αντίστοιχα screenshots τα οποία περιγράφουν όλες εκείνες τις ενεργείες που μπορεί να κάνει ο χρήστης της εφαρμογής μας.

Στο **κεφάλαιο 6** γίνεται απλή αναφορά στη βιβλιογραφία και τους συνδέσμους που χρησιμοποιήθηκαν για τη συλλογή των απαραίτητων πληροφοριών.

2. Google Earth

Η ενότητα αυτή είναι αφιερωμένη στο βασικό εργαλείο πάνω στο οποίο στηρίζεται η λειτουργία της εφαρμογής μας, με σκοπό να καταφέρει να αναπαραστήσει γεωγραφικά τα γεγονότα που ο κάθε χρήστης επιθυμεί. Εδώ αναλύεται η πορεία του Google Earth από την γέννηση του, μέχρι σήμερα, οι δυνατότητες αναπαράστασης τις οποίες μας παρέχει και πως αυτές προκύπτουν. Τέλος γίνεται εκτενής αναφορά στον τρόπο με τον οποίο χρησιμοποιείται το εργαλείο αυτό και συνάμα αναλύονται οι βασικές συναρτήσεις και λειτουργίες που χρησιμοποιούνται.

2.1 Γενική αναφορά στο Google Earth

Το Google Earth είναι μία εικονική υδρόγειος-χάρτης και κατά συνέπεια ένα πρόγραμμα παροχής γεωγραφικών δεδομένων, το οποίο αρχικά ονομάζονταν Earth Viewer 3D. Δημιουργός του ήταν η εταιρεία keyhole, inc η οποία ήταν χρηματοδοτούμενη από την Αμερικάνικη CIA. Η keyhole εξαγοράστηκε από την Google το 2004. Το πρόγραμμα αυτό χαρτογραφεί την Γή κάνοντας χρήση της υπέρθεσης (Superimposition) φωτογραφιών οι οποίες έχουν αποκτηθεί από δορυφορική λήψη, από αεροφωτογραφίες, και από το GIS 3D, ένα σύστημα επίσης γεωγραφικών πληροφοριών σχεδιασμένο να χειρίζεται και να προβάλλει όλων των ειδών τα γεωγραφικά δεδομένα. Οι εκδόσεις στις οποίες ήταν αρχικά διαθέσιμο το Google Earth, ήταν το Google Earth, το οποίο ήταν δωρεάν αλλά με περιορισμένη λειτουργικότητα, το Google Earth Plus, το οποίο περιείχε κάποια επιπλέον χαρακτηριστικά και το Google Earth Pro (\$399 το χρόνο) του οποίου η χρήση ήταν εμπορική.

Η επανέκδοση του προϊόντος έγινε το 2005 και πλέον είναι διαθέσιμο για χρήση σε υπολογιστές που υποστηρίζουν Windows 2000 ή νεότερη έκδοση, MAC OS X 10.3.9 ή νεότερη έκδοση, και Kernel Linux 2.9 ή νεότερο. Το Google Earth είναι επίσης διαθέσιμο και ως plug-in για browsers για δικτυακή χρήση, από το Μάιο του 2008. Ακόμα, τον Οκτώβριο του 2008 έγινε διαθέσιμο και για το λειτουργικό του iPhone και είναι διαθέσιμο και για τους χρήστες του Android. Τέλος η Google προσέθεσε τις εικόνες στις οποίες στηρίζεται η ανανεωμένη έκδοση του keyhole (Google Earth) στο δικό της χαρτογραφικό λογισμικό, το Google Maps. Η έκδοση του Google Earth τον Ιούνιο του 2005 στο κοινό, προκάλεσε τεράστια αύξηση των μέσων που χρησιμοποιούν την εικονική γη, οδηγώντας το ενδιαφέρον του κοινού σε τεχνολογίες και εφαρμογές που αφορούν γεωγραφικά δεδομένα.

Πλέον, πολλά μέρη στην επιφάνεια της Γής, όπως κτήρια και βουνά είναι διαθέσιμα σε μορφή 3D. Το Google Earth χρησιμοποιεί DEM (digital elevation model) δεδομένα, τα οποία έχουν συλλεχτεί από το SRTM (Shuttle Radar Topography Mission) της NASA. Αυτό έχει ως αποτέλεσμα να μπορούμε να δούμε όλη τη Γή σε 3 διαστάσεις. Από τον Νοέμβριο του 2006, η 3D όψεις πολλών βουνών όπως το Everest έχει βελτιωθεί από την χρήση των συμπληρωματικών DEM δεδομένων για να καλυφτούν τα κενά της κάλυψης του SRTM.

Πολλοί άνθρωποι σήμερα χρησιμοποιούν τις εφαρμογές γεωγραφικού περιεχομένου για να προσθέσουν τα δικά τους δεδομένα κάνοντας τα διαθέσιμα μέσω πολλών πηγών όπως το BBS (Bulletin Board Systems) ή blogs. Το Google Earth είναι σε θέση να αναπαραστήσει όλα τα είδη των εικόνων που είναι προσαρμοσμένες στην επιφάνεια της γης ενώ ταυτόχρονα αποτελεί μη δικτυακή εφαρμογή χαρτογράφησης. Ο χειρισμός 3D δεδομένων υποστηρίζεται από το Google Earth με τη βοήθεια της χρήσης KML (Keyhole Markup Language) .

Το Google Earth μπορεί εύκολα να αναπαραστήσει σε μορφή 3D κτήρια και κατασκευές τα οποία μπορεί να προέρχονται από καταχωρήσεις χρηστών χρησιμοποιώντας το Sketch Up, ένα πρόγραμμα σχεδιασμού 3D μοντέλων. Έτσι ενώ στις αρχικές εκδόσεις (πριν την 4) του Google Earth τα 3D κτήρια είναι περιορισμένα σε λίγες μόνο πόλεις , και είχαν φτωχή απόδοση χωρίς καθόλου καλή υφή, πλέον πολλά κτήρια και κατασκευές είναι διαθέσιμα με λεπτομερή 3D ανάλυση και από διάφορα σημεία του πλανήτη. Έτσι σήμερα πολλές πόλεις παρουσιάζουν μερική ή πλήρη κάλυψη σε 3D αναπαράσταση [1] [2].

2.2 Πως χρησιμοποιήθηκε το Google Earth στην εφαρμογή μας

Η εφαρμογή μας στηρίζεται στην χρήση του Google Earth. Αυτό γίνεται με την ενσωμάτωση του Google Earth Api, που η εταιρεία Google παρέχει δωρεάν. Με τη χρήση του είναι δυνατή η εκμετάλλευση των υπηρεσιών του Google Earth σε οποιαδήποτε ιστοσελίδα. Η υλοποίηση του Api έχει γίνει με τη χρήση της γλώσσας προγραμματισμού JavaScript. Γενικά όσον αφορά την λειτουργία του, αυτό που συμβαίνει είναι καθώς ο χρήστης περιπλανιέται πάνω στο χάρτη, νέες εικόνες να κατεβαίνουν από τον κεντρικό server της Google και να φορτώνονται στο χάρτη. Το plugin που χρειάζεται για να χρησιμοποιήσουμε το Google Earth api, και κατά συνέπεια η εφαρμογή μας, είναι συμβατή με τις εξής πλατφόρμες:

- **Microsoft Windows (XP, Vista και Win 7)**
 - Google Chrome 5.0+
 - Internet Explorer 7.0+
 - Firefox 3.0+
 - Flock 1.0+
- **Apple Mac OS X 10.5 and higher (Intel)**
 - Google Chrome 5.0+
 - Safari 3.1+
 - Firefox 3.0+

2.2.1 Ενσωμάτωση Google Earth Στην εφαρμογή μας.

Για την ενσωμάτωση του Google Earth Plugin στην ιστοσελίδα μας έγιναν οι ακόλουθες ενέργειες [3] :

1. Προσθέτουμε στο τμήμα <head> που βρίσκεται στην αρχή της HTML σελίδας κομμάτι κώδικα που ακολουθεί και δηλώνει ένα μοναδικό κλειδί. Το κλειδιά αυτό το έχουμε αποκτήσει από την Google, προκειμένου να έχουμε την άδεια για χρήση του Google Earth Plugin. Η απόκτηση του κλειδιού γίνεται με δήλωση του url της ιστοσελίδας μας.

```
<script type="text/javascript"
src="https://www.google.com/jsapi?key=AIzaSyCFkxDkv56W3l04SkPbh9C-
K33NypsOWw4"></script>
```

2. Δημιουργία ενός στοιχείου DIV όπου θα τοποθετηθεί το plugin με την εντολή:

```
<div id="map3d" style="height: 400px; width: 600px;"></div>
```

3. Δημιουργία των συναρτήσεων που αρχικοποιούν το plugin.

Αυτές είναι:

```
function init() {
    google.earth.createInstance('map3d', initCB, failureCB);
}
function initCB(instance) {
    ge = instance;
    ge.getWindow().setVisibility(true);
}
function failureCB(errorCode) {
}
```

Αυτές μας λένε ότι στο DIV στο οποίο το στιγμιότυπο του Google earth θα φορτωθεί, εάν είναι αυτό που ορίζεται στην πρώτη περίπτωση, δηλαδή το “map3d”. Εάν είναι επιτυχής η φόρτωση, τότε καλείται η initCB διαφορετικά η failureCB. Η initCB στην περίπτωση μας αυτό που κάνει είναι αρχικά να κάνει ορατό το παράθυρο του Google earth μέσα στο Div του map3d, κάνοντας χρήση της συνάρτησης:

```
ge.getWindow().setVisibility(true);
```

Στη συνέχεια, φορτώνονται τα διάφορα στοιχεία που θέλουμε να είναι ορατά πάνω στο χάρτη, τα λεγόμενα επίπεδα, και αφορούν τα σύνορα των χωρών, ονόματα των οδών και το τρισδιάστατο εδαφικό ανάγλυφο. Ο έλεγχος των επιπέδων γίνεται και μέσα από το interface της εφαρμογής όπου ο χρήστης επιλέγει τι θέλει να φαίνεται

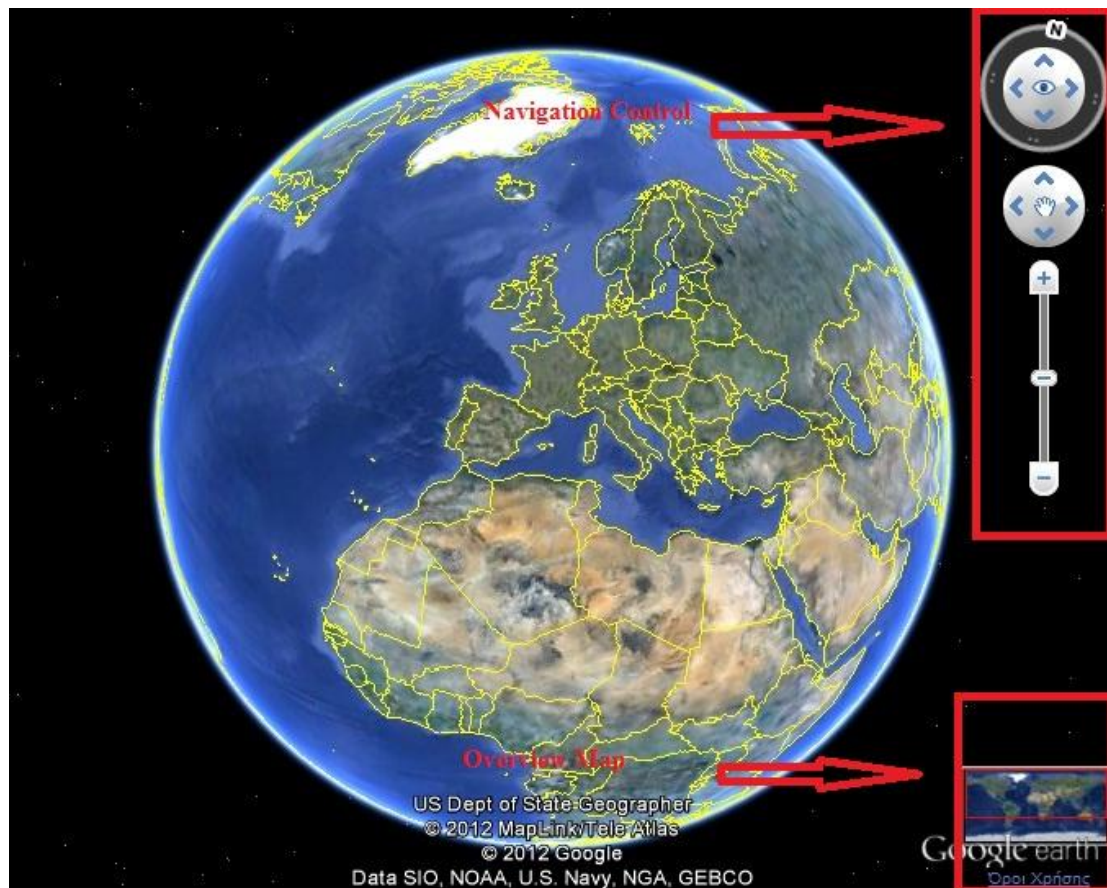
κάθε φορά με τη χρήση checkboxes. Οι εντολές με τις οποίες αρχικοποιούνται τα επίπεδα είναι:

```
ge.getLayerRoot().enableLayerById(ge. LAYER_BORDERS, true);  
ge.getLayerRoot().enableLayerById(ge. LAYER_BUILDINGS, false);  
ge.getLayerRoot().enableLayerById(ge. LAYER_ROADS, true);  
ge.getLayerRoot().enableLayerById(ge. LAYER_TERRAIN, true);  
ge.getLayerRoot().enableLayerById(ge. LAYER_TREES, false);
```

Να σημειωθεί ότι όπου υπάρχει true σημαίνει ότι η αντίστοιχη λειτουργία που υποδηλώνει και το όνομά πχ `LAYER_BORDERS` είναι ενεργοποιημένη και σημαίνει ενεργοποίηση της εμφάνισης των χωρικών συνόρων.

Τέλος χρησιμοποιείται η εντολή ενεργοποίησης της μπάρας πλοήγησης που βοηθάει στην αλλαγή οπτικής γωνίας, στα δεξιά του χάρτη και η εντολή εμφάνισης μίας μικρογραφίας του χάρτη όλου του κόσμου, ώστε σε περίπτωση μεγάλου zoom ο χρήστης να ξέρει που βρίσκεται. Οι εντολές και το αποτέλεσμα τους πάνω στο χάρτη παρατίθενται παρακάτω:

```
ge.getNavigationControl().setVisibility(ge.VISIBILITY_SHOW);  
ge.getOptions().setOverviewMapVisibility(true);
```



Εικόνα 2. Χρήση του Navigation Control και του Overview map.

4. Κλήση των συναρτήσεων αρχικοποίησης που προαναφέρθηκαν. Αυτό γίνεται εφόσον ολοκληρωθεί η φόρτωση του plugin με τη συνάρτηση η οποία καλεί την `init`, που στην ουσία κάνει όλες τις αρχικοποιήσεις:

```
google.setOnLoadCallback(init);
```

2.2.2 Βασικές εντολές του Google Earth που χρησιμοποιήθηκαν για δημιουργία και επεξεργασία ταξιδιών.

Εφόσον ολοκληρώθηκε ομαλά η εγκατάσταση του plugin στην εφαρμογή μας και οι διάφορες αρχικοποιήσεις, ο χρήστης είναι πλέον έτοιμος εφόσον το επιλέξει να δημιουργήσει το δικό του ταξίδι. Σε κάθε ταξίδι ο χρήστης θα περιγράφει τα διάφορα γεγονότα και εμπειρίες που βίωσε χρησιμοποιώντας είτε `markers` με εικονίδια, περιγραφικά της εκάστοτε δραστηριότητας, είτε φωτογραφίες. Για να επιτελέσουμε αυτές τις λειτουργίες απαραίτητη είναι η χρήση των συναρτήσεων του Google Earth για δημιουργία `placemarks` και `balloons` τα οποία αναδύονται όταν κάνουμε `click` σε ένα `placemark`. Επίσης σημαντική είναι η χρήση συναρτήσεων δημιουργίας `events` πάνω στο χάρτη που θα μας βοηθήσουν στο χειρισμό των `placemarks` όπως είναι για παράδειγμα το σύρσιμο ενός `marker` με το ποντίκι και η μετακίνηση του σε άλλη θέση.

Έτσι για τη δημιουργία `placemarks` οι βασικές εντολές, τις οποίες στην εφαρμογή μας ενσωματώνουμε στις συναρτήσεις JavaScript `createPlacemarkDrag()` και `uploadedImageAjax()` παρατίθενται παρακάτω. Οι συναρτήσεις αυτές χειρίζονται αντίστοιχα την δημιουργία απλού `marker`, ο οποίος έχει ένα συγκεκριμένο εικονίδιο, και την δημιουργία `marker` για τις φωτογραφίες που ο χρήστης ανεβάζει και είναι [4]:

```
// Create the placemark.
var placemark = ge.createPlacemark("");
placemark.setName("placemark");

// Define a custom icon.
var icon = ge.createIcon("");
icon.setHref('http://maps.google.com/mapfiles/kml/paddle/red-circle.png');
var style = ge.createStyle(""); //create a new style
style.setIconStyle().setIcon(icon); //apply the icon to the style
placemark.setStyleSelector(style); //apply the style to the placemark

// Set the placemark's location.
var point = ge.createPoint("");
point.setLatitude(12.345);
point.setLongitude(54.321);
placemark.setGeometry(point);

// Add the placemark to Earth.
ge.getFeatures().appendChild(placemark);
```

Όπως φαίνεται από τον παραπάνω κώδικα, αρχικά δημιουργούμε το Google Earth `Placemark` στο οποίο με την `setName("placemark")` δίνουμε το όνομα 'placemark'. Έπειτα ορίζουμε το εικονίδιο το οποίο το `placemark` θα έχει πάνω στο χάρτη στα

πλαίσια ορισμού του style του. Τα εικονίδια μπορούμε να τα κατεβάσουμε απευθείας από τον server του Google Maps ορίζοντας την κατάλληλη διεύθυνση με την εντολή **icon.setHref('διεύθυνση εικονιδίου');**. Τα εικονίδια που ο χρήστης μπορεί να χρησιμοποιήσει στην εφαρμογή είναι αυτά που εικονίζονται παρακάτω.



Εικόνα 3. Εικονίδια που ο χρήστης μπορεί να χρησιμοποιήσει για τη δημιουργία ταξιδιού.

Τέλος, για να ολοκληρώσουμε τη δημιουργία του placemark, πρέπει να ορίσουμε τις συντεταγμένες του στο χάρτη. Οι συντεταγμένες για απλούς markers και εικόνες ορίζονται ως το κέντρο εστίασης του χάρτη όπως φαίνεται εκείνη τη στιγμή. Ιδιαιτερότητα παρουσιάζουν οι εικόνες που περιέχουν μεταδεδομένα (metadata) με γεωγραφική πληροφορία. Αυτό συμβαίνει όταν η συσκευή από την οποία τραβήχτηκε η φωτογραφία έχει ενσωματωμένο και ενεργοποιημένο κατά τη λήψη Gps. Στην περίπτωση αυτή, κατά το upload, εξάγουμε με java κώδικα τα δεδομένα αυτά και τοποθετούμε την φωτογραφία στο γεωγραφικό σημείο ακριβώς όπου τραβήχτηκε διευκολύνοντας σημαντικά το χρήστη. Ένα παράδειγμα δημιουργίας marker με όνομα “my marker” και με εικονίδιο την ποδηλασία είναι το παρακάτω εικονιζόμενο.



Εικόνα 4. Παράδειγμα δημιουργίας marker.

Για να μπορέσουμε να αποτυπώσουμε πλήρως την πληροφορία που το κάθε marker φέρει, είναι αναγκαία η χρήση **balloons**[5]. Τα balloons είναι παράθυρα πληροφοριών τα οποία προβάλλονται πάνω στο Google Earth Plugin και προαιρετικά μπορεί να σχετίζονται με ένα στοιχείο όπως π.χ. ένα placemark ή ένα πολύγωνο στοιχείων. Το περιεχόμενό τους μπορεί να αποτελείται από HTML κώδικα, CSS και JavaScript κώδικα. Η μορφή τους μπορεί να είναι όπως αυτή εικονίζεται στην εικόνα 4. Υπάρχουν τριών ειδών balloons ανάλογα με το περιεχόμενό τους. Αυτά είναι τα:

1. **Feature Balloons.** Είναι το βασικό είδος balloon του οποίου τα περιεχόμενα είναι scrubed. Αυτό σημαίνει ότι οποιοσδήποτε κώδικας JavaScript, CSS, <iframe>, <embed> (πχ ενσωματωμένο video), <object> αφαιρείται αυτόματα. Τα περιεχόμενα του είναι αυτά που υπάρχουν στο <description> tag και είναι άμεσα συνδεδεμένο με ένα στοιχείο του χάρτη, όπως ένα placemark. Ενδεικτικά ο κώδικας που υλοποιεί ένα feature balloon για ένα συγκεκριμένο placemark είναι ο εξής:

```
var balloon = ge.createFeatureBalloon('');  
balloon.setMaxWidth(300);  
balloon.setFeature(placemark);  
ge.setBalloon(balloon);
```

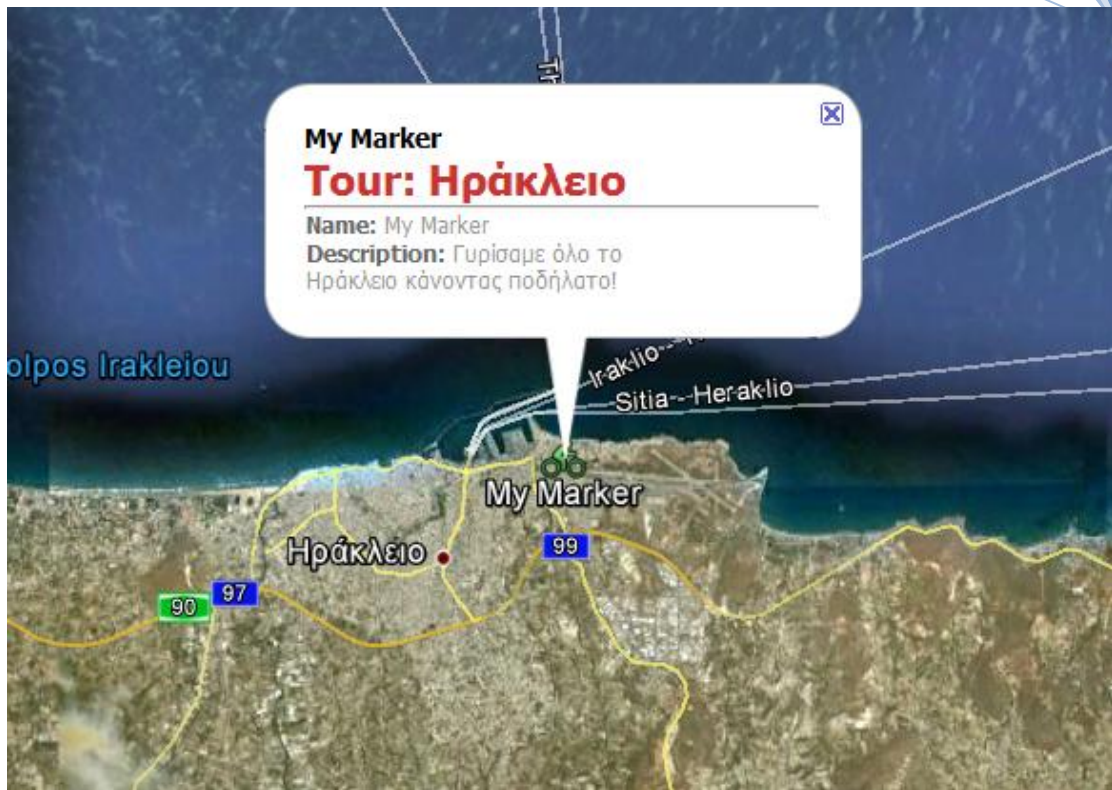
2. **HTML string balloons.** Μπορούν να περιέχουν HTML, CSS και JavaScript κώδικα. Είναι δυνατόν να μην συνδέονται με κάποιο στοιχείο του χάρτη αλλά να προβάλλονται στο κέντρο της τρέχουσας όψης. Ενδεικτικά ένα παράδειγμα κώδικα που υλοποιεί ένα HTML String balloon για ένα συγκεκριμένο placemark είναι ο εξής:

```
var balloon = ge.createHtmlStringBalloon('');
balloon.setFeature(placemark); // optional
balloon.setMaxWidth(300);
// Google logo.
balloon.setContentString(
  '<br>'
+ '<font size=20>Earth Plugin</font><br><font size=-2>sample
info '
+ 'window</font>');
ge.setBalloon(balloon);
```

3. **HTML DIV balloons.** Μπορούν να περιέχουν HTML και CSS και JavaScript κώδικα. Το περιεχόμενό τους συμπεριλαμβάνεται με σε ένα στοιχείο Div. Ενδεικτικά ένα παράδειγμα κώδικα που υλοποιεί ένα HTML DIV balloon για ένα συγκεκριμένο placemark είναι ο εξής:

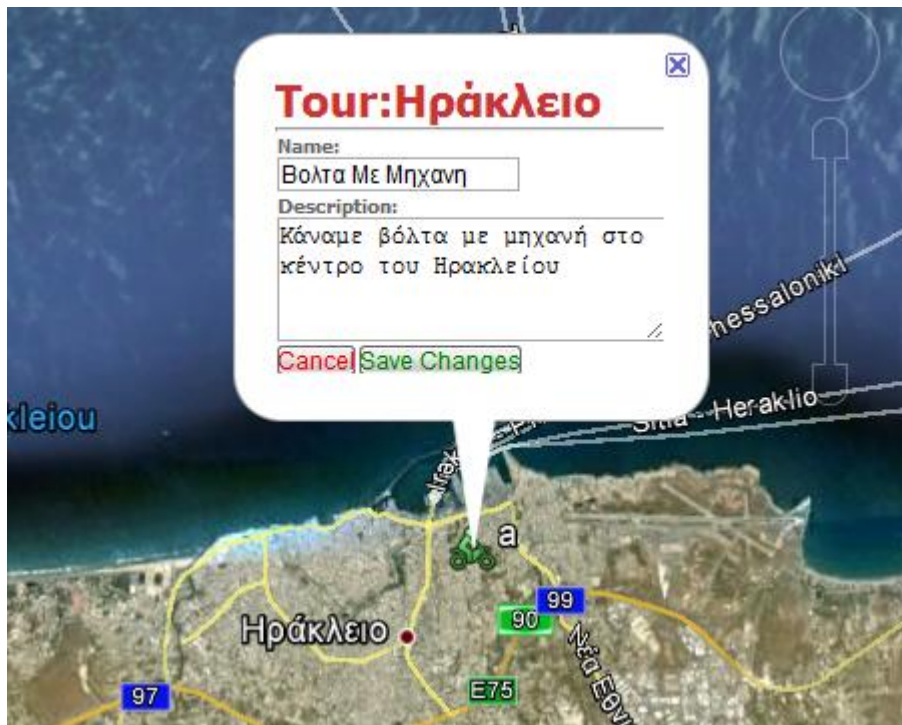
```
var balloon = ge.createHtmlDivBalloon('');
balloon.setFeature(placemark); // optional
balloon.setMaxWidth(800);

// create the <div>
var div = document.createElement('DIV');
div.innerHTML =
  '<br>'
+ '<a href="http://www.google.com/googlegulp/">Google
Gulp</a>';
balloon.setContentDiv(div);
ge.setBalloon(balloon);
```

Εικόνα 5. Παράδειγμα χρήσης balloon (Feature Balloon).

Στην εφαρμογή μας χρησιμοποιούνται κυρίως τα Feature Balloons στην περίπτωση απεικόνισης ταξιδιών, όπου δηλαδή δε χρειάζεται να περιέχουν κώδικα JavaScript κτλ. Τέτοιο είναι και αυτό που εικονίζεται στην εικόνα. Ωστόσο στην περίπτωση δημιουργίας και επεξεργασίας ταξιδιών, όπου η πληροφορία που περιέχουν τα balloons είναι επεξεργάσιμη, η χρήση κώδικα είναι απαραίτητη. Έτσι χρησιμοποιούνται και HTML String Balloons. Ένα παράδειγμα εικονίζεται στην εικόνα 5 που ακολουθεί.



Εικόνα 6. Παράδειγμα HTML String Balloon.

Ακόμα, μία βασική λειτουργικότητα που μας παρέχεται από το Google earth api, είναι η δυνατότητα ορισμού **Events**[6] τα οποία ορίζουν μία σειρά από εντολές-ενέργειες που πρόκειται να εκτελεστούν για συγκεκριμένες ενέργειες του χρήστη πάνω στο χάρτη. Οι ενέργειες αυτές που πυροδοτούν τα events μπορεί να είναι για παράδειγμα το κλικ πάνω σε ένα συγκεκριμένο αντικείμενο του χάρτη, όπως marker ή το σύρσιμο του marker πάνω στο χάρτη. Όπως γίνεται αντιληπτό, ο ορισμός events στην εφαρμογή μας είναι υπερπολύτιμος, αφού μας επιτρέπουν να χειριστούμε τα markers μας κατά τη διαδικασία δημιουργίας ενός ταξιδιού, μετακινώντας τα πάνω στο χάρτη, σύροντας τα με το ποντίκι ή ανοίγοντας το παράθυρο πληροφοριών τους, κάνοντας απλά κλικ πάνω τους. Τα events που το Google earth api παρέχει, μπορούν να χρησιμοποιηθούν με την συνάρτηση `google.earth.addEventListener`. Η συνάρτηση αυτή λαμβάνει τρία υποχρεωτικά ορίσματα και ένα τέταρτο προαιρετικό. Αυτά είναι: Το αντικείμενο πάνω στο οποίο τοποθετούμε τον listener πχ ένα placemark, το event το οποίο συνδέεται με τον listener, τη JavaScript συνάρτηση που θέλουμε να εκτελεστεί όταν συμβεί το event και τέλος προαιρετικά true ή false (το default είναι false) για το αν επιθυμούμε την ύπαρξη capture. Το τελευταίο δε μας απασχολεί στην εφαρμογή μας. Άρα η συνάρτησή μας έχει την ακόλουθη μορφή:

```
google.earth.addEventListener(placemark10, 'click', doSomething);
```

Τα κυριότερα events τα οποία χρησιμοποιούμε στην εφαρμογή μας είναι το onclick, onmouseover και onmousemove. Στην εφαρμογή μας η χρήση event όπως είπαμε είναι απαραίτητη για το άνοιγμα ενός balloon που συνδέεται με ένα στοιχείο κάνοντας κλικ. Ο κώδικας που απαιτείται για να το ορίσουμε αυτό είναι ο ακόλουθος:

```

google.earth.addEventListener(placemark, 'click', function(event) {
  // Prevent the default balloon from popping up.
  event.preventDefault();

  var balloon = ge.createHtmlStringBalloon("");
  balloon.setFeature(placemark); // optional
  balloon.setContentString(
    '<object width="200" height="150"><param name="movie" '
    + 'value="http://www.youtube.com/v/6mrG_bsqC6k&hl=en&fs=1"/>'
    + '<param name="allowFullScreen" value="true"/>'
    + '<embed src="http://www.youtube.com/v/6mrG_bsqC6k&hl=en&fs=1" '
    + 'type="application/x-shockwave-flash" allowfullscreen="true" '
    + 'width="200" height="150"></embed></object>');

  ge.setBalloon(balloon);
});

```

Τέλος, για τη διευκόλυνση εντοπισμού μίας τοποθεσίας, κατά τη διάρκεια της δημιουργίας ενός ταξιδιού κάνουμε χρήση της δυνατότητας **γεωεντοπισμού (Geocoding)** [7], η οποία μας παρέχεται, όχι από το Google earth api αλλά από το Google maps api. Για φορτώσουμε τις λειτουργίες του Google maps api χρησιμοποιούμε την εντολή:

```
google.load("maps", "2.xx")
```

Έπειτα, μπορούμε να αναζητήσουμε μία τοποθεσία γράφοντας το όνομά της και παίρνοντας τις συντεταγμένες της. Έτσι πχ για να ψάξουμε την τοποθεσία “Athens” χρειάζομαστε τον παρακάτω κώδικα, ο οποίος ορίζει το lookAt του χάρτη, δηλαδή την όψη (που και πως κοιτάζουμε) του χάρτη, στην περιοχή που αναζητούμε αν αυτή έχει βρεθεί:

```

var geocodeLocation = "Athens";
var geocoder = new google.maps.ClientGeocoder();
geocoder.getLatLng(geocodeLocation, function(point) {
  if (point) {
    var lookAt = ge.createLookAt('');
    lookAt.set(point.y, point.x, 10,
      ge.ALTITUDE_RELATIVE_TO_GROUND,
      0, 0, 1000000);
    ge.getView().setAbstractView(lookAt);
  }
});

```

Ενα παράδειγμα γεωεντοπισμού εικονίζεται παρακάτω στην εικόνα 7.



Εικόνα 7. Παράδειγμα γεωεντοπισμού Αθήνας.

2.2.3 Βασικές εντολές του Google Earth που χρησιμοποιήθηκαν για την ανασκόπηση των ταξιδιών.

Όπως αναφέρθηκε προηγουμένως, ο χρήστης της εφαρμογής μπορούν να κάνουν ανασκόπηση είτε των δικών τους ταξιδιών είτε των φίλων τους, με απλή προβολή των markers του ταξιδιού και με μορφή αυτόματης μεταβίβασης από σημείο σε σημείο (touring). Και οι δύο αυτές λειτουργίες πραγματοποιούνται με τη δυναμική δημιουργία **Kml** και προβολή του στο χάρτη.

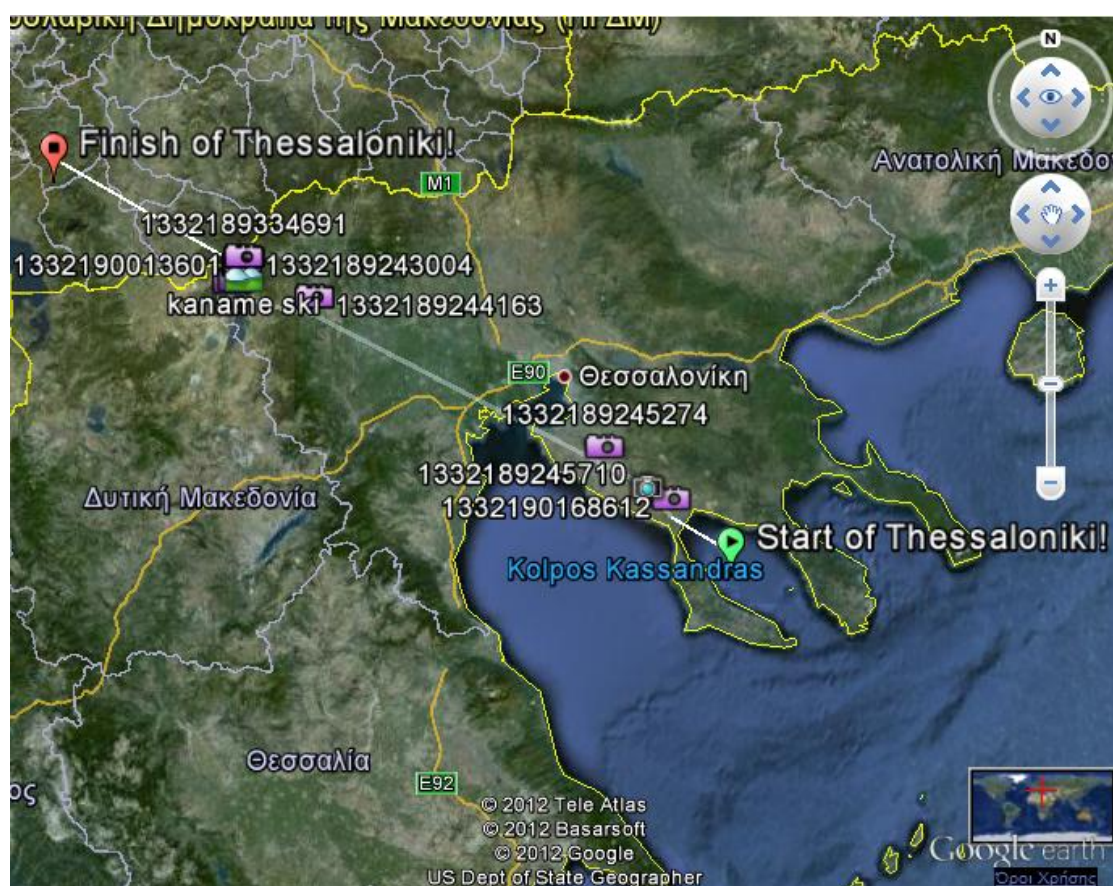
Το Kml είναι γλώσσα σήμανσης για την προβολή γεωγραφικών δεδομένων. Το Google Earth Plugin μπορεί να εισάγει kmls με τρεις τρόπους για να προβάλλει διάφορα χαρακτηριστικά και ταξίδια πάνω στο χάρτη. Οι συναρτήσεις του Google Earth για την εισαγωγή kml είναι οι **KmlNetworkLink**, η **fetchKml** και η **parseKml**. Οι πρώτες δύο φορτώνουν το Kml από ένα συγκεκριμένο URL, ενώ η **parseKml** παίρνει σαν όρισα το ίδιο το kml σε μορφή string και επιστρέφει ένα αντικείμενο της μορφής **KmlFeature**. Στην εφαρμογή μας χρησιμοποιείται η μόνο η **parseKml**. Αυτό γιατί τόσο για την αναπαράσταση όσο και για το **touring** το kml παράγεται δυναμικά από διατρέχοντας όλα τα στοιχεία από τα οποία αποτελείται (stations του ταξιδιού). Παρακάτω φαίνεται ο τρόπος με τον οποίο δημιουργούμε ένα placemark χρησιμοποιώντας την **parseKml** [8]:

```
var kmlString = "  
  + '<?xml version=\"1.0\" encoding=\"UTF-8\"?>'  
  + '<kml xmlns=\"http://www.opengis.net/kml/2.2\">'  
  
  + '<Document>'  
  + ' <Camera>'  
  + '   <longitude>-122.444633</longitude>'  
  + '   <latitude>37.801899</latitude>'  
  + '   <altitude>139.629438</altitude>'  
  + '   <heading>-70.0</heading>'  
  + '   <tilt>75</tilt>'  
  + ' </Camera>'  
  
  + ' <Placemark>'  
  + '   <name>Placemark from KML string</name>'  
  + '   <Point>'  
  + '     <coordinates>-122.448425,37.802907,0</coordinates>'  
  + '   </Point>'  
  + ' </Placemark>'  
  
  + '</Document>'  
  + '</kml>';  
  
var kmlObject = ge.parseKml(kmlString); // ΧΡΗΣΗ ΤΗΣ ParseKml  
ge.getFeatures().appendChild(kmlObject);
```

Το αποτέλεσμα του κώδικα αυτού φαίνεται στην εικόνα 8. Η camera μεταφέρεται όπως ορίζουν τα στοιχεία που βρίσκονται στο camera tag (longitude, latitude, altitude, heading, tilt). Στο σημείο αυτό έχει τοποθετηθεί το placemark με τις συντεταγμένες του να ορίζονται από το coordinates tag και το όνομά του από το name tag. Ένα ποιο σύνθετο παράδειγμα kml από την εφαρμογή μας που αναπαριστά ένα ταξίδι στην Θεσσαλονίκη εικονίζεται στην εικόνα 8.



Εικόνα 8. Παράδειγμα χρήσης της parseKml για δημιουργία ενός placemark



Εικόνα 9. Αναπαράσταση ενός ταξιδιού με χρήση Kml.

Με το Google Earth Plugin, όπως είπαμε προηγουμένως μπορούμε να δημιουργήσουμε και να αναπαραγάγουμε ταξίδια με KML, επιτρέποντας στους χρήστες να αλληλεπιδρούν με το περιβάλλον της Γης, καθώς παρακολουθούν την ελεγχόμενη από κώδικα, αναπαραγωγή του ταξιδιού τους. Για να γίνει αυτό, πρώτα χρησιμοποιούμε την `parseKml` για να φορτώσουμε τα διάφορα τμήματα (markers, πολύγωνα, κτλ.) από τα οποία αποτελείται το ταξίδι. Η `fetch` είπαμε ότι μας επιστρέφει ένα object, το οποίο στην περίπτωση μας είναι όλα τα τμήματα που συνθέτουν το ταξίδι. Το object αυτό, το περνάμε σαν όρισμα στο **GeTourPlayer** που είναι ο χειριστής στην ουσία της αναπαραγωγής των tours. Για να φορτώσουμε το ταξίδι η να το αφαιρέσουμε από τον player χρησιμοποιούμε αντίστοιχα τις εντολές [9]:

```
ge.getTourPlayer().setTour(tour);  
ge.getTourPlayer().setTour(null);
```

Για να είναι δυνατός ο χειρισμός του tour το Google Earth plugin μας παρέχει τις εξής συναρτήσεις ελέγχου:

- **ge.getTourPlayer().play():** Αναπαράγει το τρέχων ταξίδι.
- **ge.getTourPlayer().pause():** Σταματάει την αναπαραγωγή του τρέχοντος ταξιδιού.
- **ge.getTourPlayer().reset():** Σταματάει και ξεκινάει από την αρχή το τρέχων ταξίδι.
- **ge.getTourPlayer().setCurrentTime(time):** Ορίζει την αναπαραγωγή να ξεκινάει από ένα συγκεκριμένο χρονικό σημείο σε δευτερόλεπτα.

Ο κώδικας για την αναπαραγωγή ενός απλού tour του οποίου το kml φορτώνεται από ένα URL είναι:

```
<script type="text/javascript">  
var ge;  
google.load("earth", "1");  
  
function init() {  
    google.earth.createInstance('map3d', initCB, failureCB);  
}  
  
function initCB(instance) {  
    ge = instance;  
    ge.getWindow().setVisibility(true);  
    ge.getNavigationControl().setVisibility(ge.VISIBILITY_SHOW);  
  
    var href = 'http://developers.google.com/kml/documentation/kmlfiles/bounce_example.kml';  
    google.earth.fetchKml(ge, href, kmlFinishedLoading);  
}  
  
function kmlFinishedLoading(object) {  
    ge.getTourPlayer().setTour(object);  
    ge.getTourPlayer().play();  
}
```

```
function failureCB(errorCode) {  
}  
  
google.setOnLoadCallback(init);  
</script>
```

Στην περίπτωση που το kml δεν είναι απλό, δηλαδή περιέχει περισσότερα από ένα ταξίδια, η το tour είναι ενσωματωμένο σε ένα container όπως ο <Document> ή ο <Folder>, κάτι το οποίο συμβαίνει στην εφαρμογή μας, πρέπει χειροποίητα να αναζητήσουμε το <gx:Tour> (tour object) μέσα στο αρχείο μας. Αυτό γίνεται διατρέχοντας το DOM, δηλαδή το σύνολο των στοιχείων του αρχείου εμφωλευμένα και μη, έως ότου βρούμε το tour object. Απο γίνεται με τη συνάρτηση WalkKmlDom η οποία μας επιστρέφει όπως φαίνεται παρακάτω το tour object αν το βρει.

```
// Walk through the KML to find the tour object;  
// assign to variable 'tour'.  
walkKmlDom(fetchedKml, function() {  
  if (this.getType() == 'KmlTour') {  
    tour = this;  
    return false;  
  }  
});  
}
```

Ο ολοκληρωμένος κώδικας αναπαραγωγής ενός πιο σύνθετου ταξιδιού είναι:

```
<script type="text/javascript">  
  
var ge;  
var tour;  
google.load("earth", "1");  
  
function init() {  
  google.earth.createInstance('map3d', initCB, failureCB);  
}  
  
function initCB(instance) {  
  ge = instance;  
  ge.getWindow().setVisibility(true);  
  ge.getNavigationControl().setVisibility(ge.VISIBILITY_SHOW);  
  
  var href = 'http://developers.google.com/kml/documentation/kmlfiles/complete_tour_example.kml';  
  google.earth.fetchKml(ge, href, fetchCallback);  
}  
  
function fetchCallback(fetchedKml) {  
  // Alert if no KML was found at the specified URL.  
  if (!fetchedKml) {  
    setTimeout(function() {  
      alert('Bad or null KML');  
    }, 0);  
  }  
}
```



```
return;
}

// Add the fetched KML into this Earth instance.
ge.getFeatures().appendChild(fetchedKml);

// Walk through the KML to find the tour object;
// assign to variable 'tour'.
walkKmlDom(fetchedKml, function() {
  if (this.getType() == 'KmlTour') {
    tour = this;
    return false;
  }
});

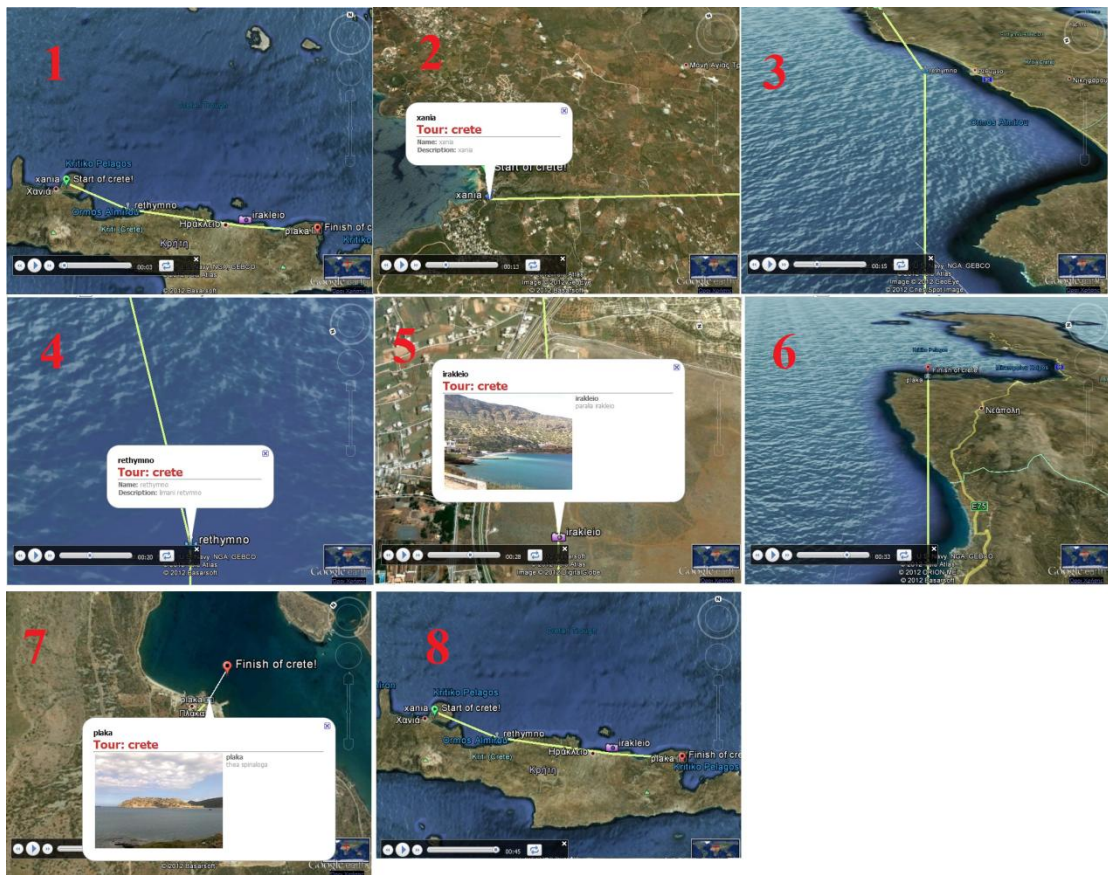
function failureCB(errorCode) {
}

google.setOnLoadCallback(init);

</script>
```

Με το kml εκτός του ότι ορίζουμε ποια είναι τα σημεία (stations) ενός ταξιδιού, ορίζουμε και το τρόπο με τον οποίο θα γίνεται η επίσκεψη κάθε σημείου κατά την αναπαραγωγή. Για παράδειγμα μπορούμε να ορίσουμε ότι κατά την μετάβαση στο σημείο «placemark» θα μας πάρει 3 δεύτερα, θα βλέπουμε το σημείο από τα 2000 μέτρα ύψος (alt) και υπό γωνία 35 μοιρών(tilt). Οι εντολές που μπορούμε να χρησιμοποιήσουμε για την σύνταξη ενός kml και συνεπώς ενός kml για touring ορίζονται από το Kml Reference [10]. Δεσπόζουσα θέση στη σύνταξη των kml της εφαρμογής μας έχουν οι εντολές για τα animated updates, δηλαδή η `<gx:AnimatedUpdate>`. Animated Updates είναι αλλαγές που συμβαίνουν πάνω στο χάρτη κατά τη διάρκεια αναπαραγωγής ενός ταξιδιού, χωρίς να επηρεάζονται τα στοιχεία του DOM. Ένα παράδειγμα είναι να βλέπουμε το εικονίδιο ενός placemark να μεγαλώνει για κάποια δευτερόλεπτα, καθώς το επισκεπτόμαστε. Τα δευτερόλεπτα που κρατάει η αλλαγή που προκαλούν τα animated updates ορίζονται από την επίσης σημαντική εντολή `<gx:duration>`.

Για την εφαρμογή μας, η αναπαραγωγή ενός ταξιδιού γίνεται αφού υπολογίσουμε το πολύγωνο, το οποίο σχηματίζουν τα διάφορα στοιχεία από τα οποία αποτελείται το ταξίδι μας, δηλαδή τους σταθμούς του ταξιδιού, μεταφερόμαστε πάνω στη Γή με τρόπο τέτοιο (κατάλληλες συντεταγμένες και ύψος) ώστε να είναι ορατό όλο το πολύγωνο. Έπειτα μεταβαίνουμε από σημείο σε σημείο, και κάνοντας χρήση των animated updates ανοίγουμε για κάποιο χρονικό διάστημα το παράθυρο πληροφοριών (balloon) για κάθε marker. Μετά το κλείνουμε. Ο τρόπος, δηλαδή η ταχύτητα, το ύψος και η θέση της κάμερας για κάθε μετάβαση είναι ελεγχόμενη με βάση την απόσταση που έχουν τα διαδοχικά σημεία. Αυτό γίνεται μέχρι να οδηγηθούμε στο τέλος της διαδρομής, όπου το ταξίδι ολοκληρώνεται με μία πανοραμική θέα όλου του πολυγώνου των placemarks όπως στην αρχή. Ένα παράδειγμα αναπαραγωγής tour για 4 σημεία φαίνεται καρέ καρέ στην εικόνα 9.



Εικόνα 10. Αναπαραγωγή ταξιδιού καρέ καρέ.

Ο έλεγχος που περιγράφηκε ότι κάνουν οι συναρτήσεις ελέγχου `ge.getTourPlayer().play()`, `ge.getTourPlayer().pause()`, `ge.getTourPlayer().reset()` και `ge.getTourPlayer().setCurrentTime(time)`, γίνεται μέσω του interface ελέγχου του tour με τη βοήθεια της μαύρης μπάρας που βλέπουμε ότι εμφανίζεται στο κάτω, αριστερό μέρος των εικόνων της εικόνας 9. Μεγεθυμμένη φαίνεται στην εικόνα 10 μαζί με τις λειτουργίες ελέγχου αναπαραγωγής που εξυπηρετεί.



Εικόνα 11. Μπάρα ελέγχου αναπαραγωγής tour. Buttons έλεγχου ταχύτητας, stop, play και reset.

3.Αρχιτεκτονική Συστήματος και τεχνολογίες που χρησιμοποιήθηκαν.

Στο προηγούμενο κεφάλαιο αναφερθήκαμε διεξοδικά στην χρήση της τεχνολογίας που μας παρέχεται από το Google Earth Api, ενώ κάναμε μία πρώτη αναφορά και στη γλώσσα JavaScript. Στο Κεφάλαιο αυτό θα δούμε αναλυτικά τις υπόλοιπες σημαντικές τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής συμπεριλαμβανομένης μιας ποιο διεξοδικής ανάλυσης για την JavaScript. Για πρώτη φορά θα αναφερθούμε στην χρήση της Html, της Java, των Java Servlets, στη χρήση των DAOs για την σύνδεση με τη βάση δεδομένων, καθώς επίσης και στα μοντέλα BCED και MVC ,πάνω στα οποία στηρίζεται η αρχιτεκτονική του συστήματός μας.

3.1 HTML και DHTML

Η γλώσσα HTML (HyperText Markup Language) είναι η κυρίαρχη γλώσσα σήμανσης (markup) για τις τεχνολογίες Web. Η συντριπτική πλειοψηφία της διακίνησης πληροφορίας στο Διαδίκτυο αφορά σε κάποιας μορφής περιεχόμενο ή έγγραφο HTML, που βρίσκεται ή παράγεται σε ένα περιβάλλον διακομιστή, και το οποίο επιστρέφεται σε εφαρμογές πελάτη (client) μέσω της εξυπηρέτησης ενός αιτήματος. Οι εφαρμογές πελάτη που αναλαμβάνουν να εμφανίσουν το περιεχόμενο HTML σε μια μορφή χρήσιμη για τον τελικό αποδέκτη αποκαλούνται προγράμματα περιήγησης (browsers).

Ένα έγγραφο HTML αποτελείται κυρίως από ετικέτες (tags) , οι οποίες αποτελούν ονόματα στοιχείων που έχουν την μορφή <όνομα>. Τα tags διακρίνονται σε δομικά στοιχεία (όπως το <p> για τη δήλωση μίας παραγράφου), σε στοιχεία εμφάνισης (όπως το το οποίο μετατρέπει σε bold μορφή το κείμενο που περικλείει), σε στοιχεία διασύνδεσης τα οποία μας παραπέμπουν σε άλλες σελίδες (όπως το <a>). Κάθε ετικέτα μπορεί να συνοδεύεται από attributes τα οποία συνήθως αποτελούν ρυθμιστές της εμφάνισης ενός αντικειμένου, καθορίζοντας πχ το χρώμα, το μήκος το πλάτος κτλ.

Η ανάπτυξη βοηθητικών τεχνολογιών της HTML και ο συνδυασμός της με αυτούς μας οδήγησε στην δυναμική πλέον HTML, την DHTML (Dynamic HTML). Οι τεχνολογίες αυτές, οι οποίες χρησιμοποιούνται κατά κόρον στην εφαρμογή μας είναι η τεχνολογία CSS, που καθορίζει τις ιδιότητες εμφάνισης των στοιχείων HTML, το μοντέλο Δυναμικών Αντικειμένων DOM (Dynamic Object Model), που μοντελοποιεί το περιεχόμενο ενός HTML εγγράφου σε μια ιεραρχική δένδρική δομή και η τεχνολογία javascript, η οποία αποτελεί μία ισχυρή scripting γλώσσα, που θα αναφερθούμε παρακάτω.


```
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

My First Heading

My first paragraph.

Εικόνα 12. Παράδειγμα Html κώδικα (αριστερά) και του αποτελέσματός του (δεξιά).

3.2 JavaScript

3.2.1 Εισαγωγή στη JavaScript

Η γλώσσα προγραμματισμού JavaScript αποτελεί μια καθαρά «client side τεχνολογία» και ανήκει στην κατηγορία των scripting γλωσσών. Ο όρος «client side» αναφέρεται, στην προκειμένη περίπτωση, στο γεγονός ότι η JavaScript εκτελείται και αφορά αποκλειστικά στο περιβάλλον ενός προγράμματος πελάτη, δηλαδή ενός περιηγητή, ενώ δεν έχει πρόσβαση σε πόρους στο περιβάλλον διακομιστή (Server). Ο όρος «scripting» συνήθως αναφέρεται σε μια κατηγορία γλωσσών προγραμματισμού «ειδικού σκοπού» που έχουν απλό συντακτικό και περιορισμένες δυνατότητες. Οι γλώσσες αυτές προορίζονται να χρησιμοποιηθούν για συγκεκριμένου τύπου εφαρμογές και να καλύψουν ειδικές ανάγκες. Στην περίπτωση της JavaScript, η ανάγκη που καλύπτεται είναι η ο εμπλουτισμός μιας Web σελίδας με στοιχεία διαδραστικότητας. Πρέπει να τονιστεί ότι οι διαφορές τις με τη Java είναι πάρα πολλές και δε πρέπει να συγχέονται. Και οι δύο έχουν στηριχτεί στη C++ άλλα η JavaScript, όπως και η πλειοψηφία των scripting γλωσσών δεν μεταγλωττίζεται (compiled), αλλά ερμηνεύεται (Interpreted) κατά την εκτέλεσή της από ειδικό πρόγραμμα Interpreted στο περιβάλλον του περιηγητή (browser) [11].

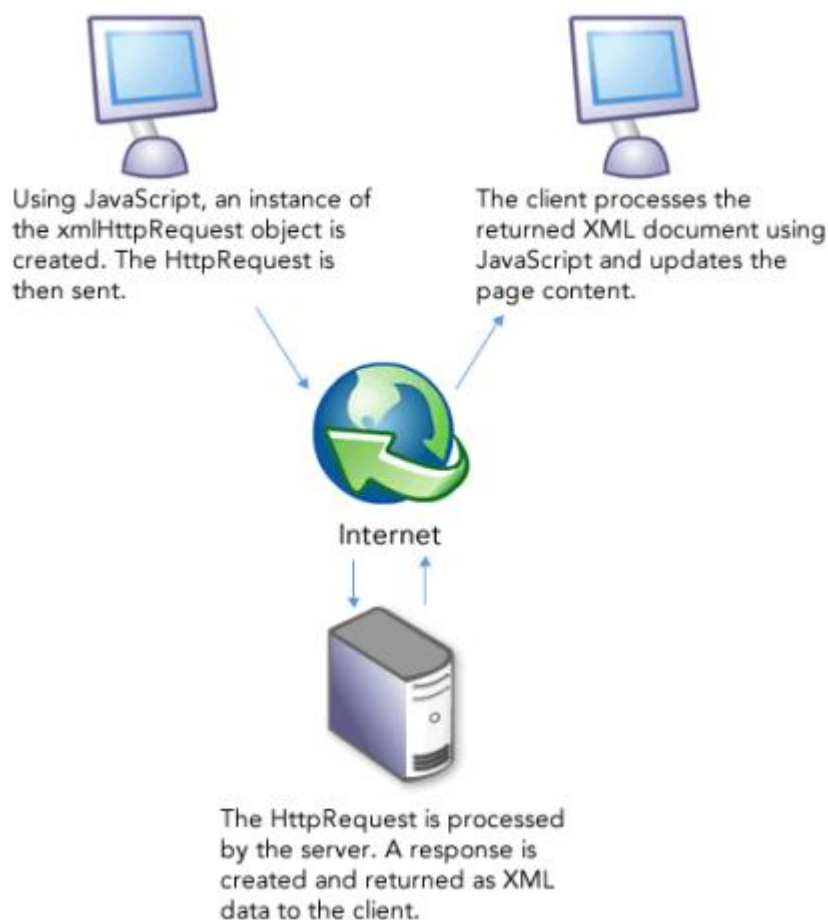
3.2.2 Χαρακτηριστικά της JavaScript

Πλεονεκτήματα της JavaScript θεωρούνται η **ταχύτητα**, αφού εκτελείται απευθείας στο περιβάλλον του διακομιστή, η **ευκολία εκμάθησης** και η **ευελιξία** αφού συνεργάζεται πολύ καλά με την πλειοψηφία των τεχνολογιών Web. Ωστόσο, το γεγονός ότι ο κώδικας εκτελείται στην πλευρά του χρήστη αφήνει σημαντικά περιθώρια για κακή χρήση από κακόβουλους παράγοντες. Αυτό αντιμετωπίζεται εν μέρει πρώτον, με τη πρόσβαση μόνο σε αντικείμενα σχετιζόμενα με το Web και όχι με τους υπόλοιπους πόρους του μηχανήματος στο οποίο εκτελείται και δεύτερον, με

την απαγόρευση της εκτέλεση κάποιου κώδικα σε μια σελίδα παρά μόνο αν ο κώδικας και η σελίδα ανήκουν στην ίδια εφαρμογή.

3.2.3 Η χρήση της JavaScript και Ajax στην εφαρμογή μας .

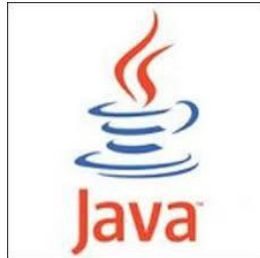
Όσον αφορά την εφαρμογή μας, η JavaScript χρησιμοποιείται σε πάρα πολλά σημεία για τον έλεγχο της συμπλήρωσης των στοιχείων των φορμών, για την εκμετάλλευση των δυνατοτήτων του Google Earth Api, όπως πχ δημιουργία placemarks, για γραφικές αλλαγές της εφαρμογής ανάλογα με τις επιλογές του χρήστη. Τέλος σημαντική είναι η συμβολή της JavaScript στην υλοποίηση **Ajax (Asynchronous JavaScript and XML)** μεθόδων. Η Ajax τεχνολογία μας επιτρέπει την ασύγχρονη επικοινωνία του χρήστη με το server. Αυτό σημαίνει ότι μπορούμε να ελέγξουμε κάποια δεδομένα που υπάρχουν στην βάση δεδομένων μας, να φορτώσουμε μία εικόνα ή να προκαλέσουμε αλλαγές στην σελίδα μας, χωρίς να χρειαστεί να την ξαναφορτώσουμε. Με άλλα λόγια επιτρέπεται η ανταλλαγή δεδομένων με το server χωρίς να χρειαστεί να ξαναφορτωθεί ολόκληρη η σελίδα. Η AJAX χρησιμοποιεί έναν HTML και CSS για την σήμανση της δομής και της εμφάνισης. Η χρήση της JavaScript σε συνδυασμό με το αντικείμενο XMLHttpRequest έρχεται να καλύψει τον χρόνο που κάνει μια σελίδα για να φορτώσει (page loading). Δηλαδή, με την χρήση της τεχνολογίας αυτής, δεν υφίσταται page loading, παρά μόνο φόρτωση συγκεκριμένης πληροφορίας (partial loading) [12].



Εικόνα 13. Σχεδιάγραμμα λειτουργίας Ajax

3.3 Java

3.3.1 Εισαγωγή στη Java



Εικόνα 14 Λογότυπο Java

Η γλώσσα προγραμματισμού *Java* είναι ένα πρόσφατο προϊόν της *Sun Microsystems Inc.*. Ξεκίνησε σαν μέρος ενός μεγαλύτερου σχεδίου που αφορούσε την ανάπτυξη λογισμικού για καταναλωτικά ηλεκτρονικά. Πρόκειται για μικρές, αξιόπιστες, φορητές, πραγματικού χρόνου συσκευές που στην αρχή βασιζόντουσαν στην C++. Αρκετά προβλήματα όμως παρουσιάστηκαν και η γλώσσα C++ δεν μπόρεσε να εφαρμοστεί τελικά. Χρειάστηκε να αναπτυχθεί μία νέα γλώσσα, η Java. Η Java, στην τελική της μορφή, βρήκε περαιτέρω εφαρμογή στην επίλυση μερικών προβλημάτων του σημερινού προγραμματισμού, όπως animation, την αλληλεπίδραση πραγματικού χρόνου (*real-time interaction*) και την εξερεύνηση του Web (*Web browsing*).

3.3.2 Τα χαρακτηριστικά της Java

Η Java είναι η γλώσσα πάνω στην οποία στηρίζεται το μεγαλύτερο μέρος της εφαρμογής μας. Η Java είναι μια αντικειμενοστραφής γλώσσα ανοιχτού κώδικα. Το συντακτικό της βασίζεται στο συντακτικό της C++. Την χρησιμοποιούμε για την υλοποίηση τμημάτων (κλάσεις) που αφορούν τον έλεγχο της λειτουργικότητας όπως επίσης και για τη δυναμική παραγωγή των ιστοσελίδων μας. Το πώς γίνεται αυτό θα αναλυθεί παρακάτω με την αναφορά των τεχνολογιών των Java Servlets και JSP.

Κύρια χαρακτηριστικά της Java είναι τα εξής:

- Η μεταγλώττιση δεν γίνεται σε εκτελέσιμο native κώδικα αλλά σε ενδιάμεσο κώδικα που έχει την ονομασία bytecode. Ο bytecode εκτελείται μέσω interpreter ή με just-in-time compilation από το runtime μέρος της Java το Java Virtual Machine. Αυτό έχει ως μειονέκτημα την απώλεια ταχύτητας καθώς δε γίνεται απευθείας εκτέλεση του κώδικα αλλά πλεονέκτημα το ότι οποιαδήποτε υλοποίηση JVM σε οποιαδήποτε αρχιτεκτονική μπορεί να τρέξει κάποιο bytecode αρχείο. Δηλαδή

επιτρέπει το ίδιο το πρόγραμμα να μπορεί να εκτελεστεί σε πολλά διαφορετικά λειτουργικά συστήματα.

- Η Java παρέχει την δυνατότητα να εκτελεστεί κώδικας Java από απομακρυσμένες πηγές με ασφάλεια.
- Η γλώσσα έχει σχεδιαστεί για να προσφέρει ευκολία και μεγαλύτερη παραγωγικότητα στον προγραμματιστή.
- Η Java είναι γενικά εύκολη στη χρήση χρησιμοποιώντας τα καλά στοιχεία άλλων αντικειμενοστραφών γλωσσών προγραμματισμού και επίσης προσπαθεί να κάνει πιο εύκολη ιδιαίτερα την ανάπτυξη διαδικτυακών εφαρμογών.
- Η Java εκμεταλλεύεται την ύπαρξη του virtual machine ώστε να προσφέρει πιο εξελιγμένες runtime υπηρεσίες πχ το reflection.
- Η Java προσφέρει βιβλιοθήκες για μεγάλο πλήθος αναγκών, ως μέρος του Java platform.

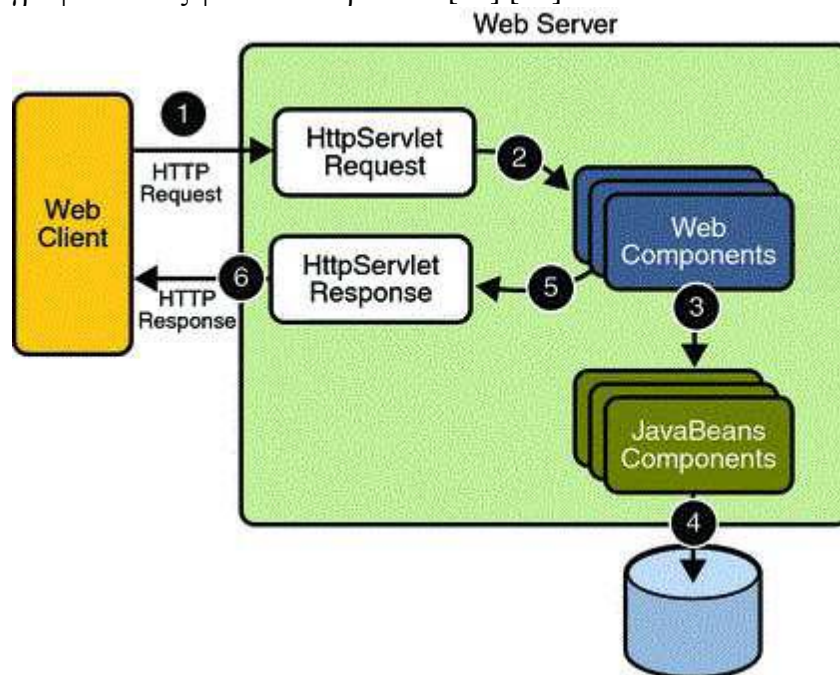
Κατ' αρχήν, χωρίς η Java να αποτελεί κάποιο πρότυπο για διαδικτυακές εφαρμογές αποτελεί ίσως την πιο δημοφιλή λύση. Μεγάλο μέρος από ανάλογα προγράμματα χρησιμοποιούν αυτήν τη γλώσσα. Σε πρώτη φάση η επιλογή της "πιο διαδεδομένης" γλώσσας προγραμματισμού γίνεται κυρίως για λόγους οικονομικούς. Είναι πιο εύκολο να βρεθούν προγραμματιστές που θα γνωρίζουν τη Java ώστε να προγραμματίσουν και εν συνεχεία να συντηρήσουν κάποια εφαρμογή. Η διάδοση της Java στην ανάπτυξη ανάλογων εφαρμογών εξασφαλίζει ότι θα υπάρχει ήδη εμπειρία και τεχνικές και σε τελική φάση ακόμα και χρησιμοποιήσιμος έτοιμος κώδικας που θα βοηθήσουν στην ανάπτυξη του προγράμματος. Σε μια διπλωματική άσκηση δεν τίθενται παράγοντες οικονομικοί ή εύρεσης προγραμματιστών, όμως σε αντιστοιχία με το τι θα γινόταν κατά την ανάπτυξη μιας εμπορικής εφαρμογής πρέπει η διάδοση της Java να ληφθεί υπόψη. Πέραν της διάδοσης της σε ανάλογες εφαρμογές, άλλα χαρακτηριστικά της Java για τα οποία έγινε η επιλογή της είναι η ευκολία στη χρήση και η αυξημένη παραγωγικότητα που παρέχει η Java μέσω χαρακτηριστικών της γλώσσας (όπως το αυτόματο garbage collection) και των βιβλιοθηκών της. Πρέπει να σημειωθεί ότι στην εργασία χρησιμοποιήθηκε η 1.6 έκδοση της Java [13].

3.4 Java Servlets

3.4.1 Τι είναι τα Java Servlets

Τα Servlets είναι μια τεχνολογία που μας παρέχει η Java για την επέκταση και ενίσχυση των διαδικτυακών εφαρμογών. Χρησιμοποιείται για τη λήψη και την αποστολή απαντήσεων σε αιτήματα (requests) από Web clients, κάνοντας κυρίως χρήση του πρωτοκόλλου HTTP. Τα Servlets είναι μικρά τμήματα κώδικα που τρέχουν στην πλευρά του web server (server sided). Στην εφαρμογή μας η τεχνολογία των Servlets χρησιμοποιείται στην πλευρά του server, για να απαντά στα αιτήματα

που υποβάλλει ο χρήστης (client). Έτσι, για παράδειγμα κάθε φορά που έχουμε εγγραφή νέου χρήστη (register), ένα servlet είναι υπεύθυνο να αποκριθεί σε αυτό το αίτημα, κάνοντας εγγραφή των στοιχείων του χρήστη στη βάση δεδομένων και να τον οδηγήσει στην αρχική σελίδα. Η λειτουργία των Servlets μπορεί να αναπαρασταθεί γραφικά όπως φαίνεται παρακάτω[14].[15]



Εικόνα 15. Διαγραμματική αναπαράσταση της αρχιτεκτονικής ενός Servlet. (1): Αποστολή του αιτήματος (http request) από τον client στον Server. (2): Προώθηση του αιτήματος σε ένα servlet. (3): Δημιουργία των κατάλληλων αντικειμένων από το servlet. (4): Επικοινωνία με βάση δεδομένων. (5): Αποστολή αντικειμένων πίσω στον container. (6): Δημιουργία του Http Response.

3.4.2 Κύκλος ζωής ενός Java Servlet

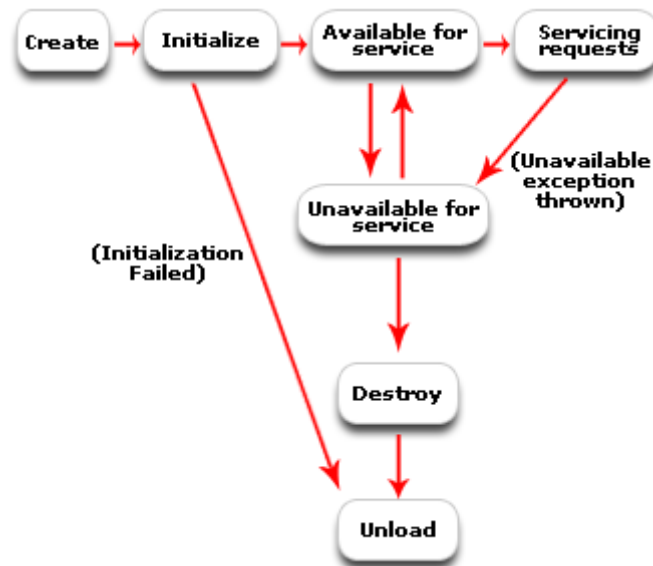
Ο κύκλος ζωής για κάθε servlet είναι συγκεκριμένος και αποτελείται από τα εξής στάδια τα οποία εικονίζεται και γραφικά στην εικόνα που ακολουθεί [16]:

1. Η κλάση servlet φορτώνεται στον web container κατά την εκκίνηση του.
2. Ο container καλεί τον υπεύθυνο constructor για τη δημιουργία ενός αντικειμένου servlet.
3. Ο container καλεί την συνάρτηση init() του αντικειμένου που δημιουργήθηκε. Αυτή αρχικοποιεί το αντικείμενο και πρέπει να κληθεί πριν αυτό αρχίσει να εξυπηρετεί αιτήματα. Σε όλη την διάρκεια ζωής του αντικειμένου η init() καλείται μόνο μια φορά.
4. Μετά την αρχικοποίηση το αντικείμενο μπορεί πλέον να εξυπηρετεί αιτήματα. Κάθε αίτημα εξυπηρετείται σε δικό του ξεχωριστό νήμα. Για κάθε αίτημα που φτάνει εξυπηρετείται σε δικό του ξεχωριστό νήμα.

Για κάθε αίτημα που φτάνει στον server ο web container καλεί την συνάρτηση service του αντικειμένου. Αυτή ανάλογα με το είδος του αιτήματος που δέχεται, το

παραπέμπει σε μία κατάλληλη μέθοδο για να το χειριστεί (action). Οι μέθοδοι αυτοί υλοποιούνται από τον προγραμματιστή. Αν κάποια μέθοδος δεν έχει υλοποιηθεί επιστρέφεται μήνυμα λάθους ως απάντηση.

5. Τέλος, ο container καλεί την συνάρτηση `destroy()`. Αυτή θέτει το αντικείμενο εκτός υπηρεσίας και κάνει την αποδέσμευσή του. Η `destroy()`, καλείται μία μόνο φορά κατά τη διάρκεια του κύκλου ζωής ενός servlet.



Εικόνα 16.Κυκλος ζωής ενός Servlet.

3.5 Java Servlet Pages (JSPs)

3.5.1 Η Ανάγκη για δημιουργία των JSPs

Η τεχνολογία Java Server Pages δημιουργήθηκε για να **διευκολύνει την ανάπτυξη εφαρμογών σε περιβάλλον Java** και αποτέλεσε, στην πραγματικότητα, μια επέκταση της τεχνολογίας Java Servlets. Το νέο στοιχείο αυτής της τεχνολογίας, το οποίο είχε εμφανιστεί ήδη στις ανταγωνιστικές τεχνολογίες PHP και ASP, είναι η κάλυψη αφενός της ανάγκης **διαχωρισμού του στατικού περιεχομένου μιας Web σελίδας από το δυναμικά παραγόμενο περιεχόμενο**, και αφετέρου η καλύτερη **διάκριση ανάμεσα στα HTML στοιχεία (markup) που αφορούν στην εμφάνιση μιας σελίδας και στα κομμάτια κώδικα (scriptlet) που αφορούν στη λειτουργικότητα και λογική του συστήματος**. Τα scriptlets είναι ο κώδικας Java ο οποίος συνήθως συμπεριλαμβάνεται μέσα στα tags `<% %>`.

3.5.2 Προτυποποίηση σελίδας (page templating), η βάση της τεχνικής JSP

Η τεχνική πάνω στην οποία βασίζεται η τεχνολογία είναι αυτής της **προτυποποίησης σελίδας (page templating)**, η οποία υποστηρίζει το διαχωρισμό της σχεδίασης μιας σελίδας (πρότυπο) από την λογική υλοποίησης. Έτσι, ο προγραμματιστής

αναπτύσσει μόνο τις απαραίτητες μεθόδους που υλοποιούν τη λογική της εφαρμογής, και ο διαχειριστής περιεχομένου σχεδιάζει τη σελίδα χωρίς να χρειάζεται καμιά γνώση προγραμματισμού, φροντίζοντας να εισάγει στα κατάλληλα σημεία της σελίδας κλήσεις προς αυτές τις μεθόδους, που τυπικά παράγουν και επιστρέφουν δυναμικά HTML περιεχόμενο.

3.5.3 Η λειτουργία της JSP τεχνολογίας και η διαφορά από τα Java Servlets

Σε τεχνικό επίπεδο, για την «εκτέλεση» μιας σελίδας JSP ισχύουν όλα όσα ισχύουν για ένα Servlet και περιγράφηκαν παραπάνω, όπως π.χ. ο κύκλος ζωής. Αυτό συμβαίνει επειδή ως βάση της τεχνολογίας παραμένει η εκτέλεση ενός Servlet στο περιβάλλον του Servlet Container. Συγκεκριμένα, μια σελίδα JSP μεταγλωττίζεται από έναν ειδικό «JSP μεταγλωττιστή» (compiler) και αποθηκεύεται στο περιβάλλον σαν ένα εκτελέσιμο αρχείο μιας κλάσης Servlet. Κατόπιν αυτού, το περιβάλλον μεταχειρίζεται με τον ίδιο τρόπο τα αιτήματα που αφορούν είτε JSP σελίδες είτε «καθαρά» Servlets, εκτελώντας δηλαδή την αντίστοιχη μέθοδο service() του εκτελέσιμου αρχείου και στις δύο περιπτώσεις. Η μόνη διαφορά είναι ότι **το εκτελέσιμο αρχείο δημιουργείται δυναμικά στην περίπτωση μιας JSP σελίδας**, δηλαδή την πρώτη φορά που ο διακομιστής λαμβάνει ένα αίτημα γι' αυτή τη σελίδα. Εάν η σελίδα τροποποιηθεί, τότε το περιβάλλον εντοπίζει την αλλαγή και ξαναδημιουργεί το αντίστοιχο αρχείο. Αντίθετα, στην περίπτωση ενός «καθαρού» Servlet δεν συμβαίνει κάτι τέτοιο, καθώς η Servlet κλάση έχει ήδη μεταγλωττιστεί και προστεθεί στην εφαρμογή από τον προγραμματιστή [17]. Τέλος, πρέπει να αναφερθεί, ότι στα Servlets υπήρχε δυσκολία στη δημιουργία Html κώδικα, παρόλο που υπήρχε το δυναμικό στοιχείο, καθώς ο κώδικας της HTML παράγεται μέσα από τις κλήσεις της συνάρτησης println(), κάτι το οποίο λύνεται με τη χρήση των JSPs.

3.5.4 Η χρήση της JSP τεχνολογίας στην εφαρμογή μας.

Χρησιμοποιώντας την τεχνολογία Jsp στην εφαρμογή μας μπορούμε να γράφουμε HTML κώδικα και ταυτόχρονα να ενσωματώσουμε κώδικα java ανάμεσα στον HTML. Αυτό γίνεται προκειμένου να κάνουμε το περιεχόμενο της σελίδας μας δυναμικό και το καταφέρνουμε συμπεριλαμβάνοντας τον Java κώδικα ανάμεσα στα ειδικά tags, κυρίως <% %>. Ένα χαρακτηριστικό παράδειγμα χρήσης της Jsp είναι η δυναμική εμφάνιση των ταξιδιών του χρήστη. Διατρέχοντας με κώδικα Java παίρνουμε τα ταξίδια του χρήστη. Χρησιμοποιώντας το όνομά τους δημιουργούμε το μενού από το οποίο επιλέγει ο χρήστης ποιο θέλει να εμφανίσει στο χάρτη. Ο κώδικας Jsp ο οποίος κάνει αυτό το επιθυμητό πάντρεμα HTML και Java εικονίζεται παρακάτω, όπως επίσης και το αποτέλεσμα του.

```

<div class="title">My Tours</div>
<ul class="contries" id="countries" style="overflow-x: hidden;" >
  <%= session.getAttribute("mytriplist") != null ?
    LinkedList trips = (LinkedList) session.getAttribute("mytriplist");

    for (int i = 0; i < trips.size(); i++) {
      TripTransfer aTrip = new TripTransfer();
      aTrip = (TripTransfer) trips.get(i);
      int tripId = aTrip.getTripID();%>

  <li onmouseover="" style="margin-left: -8px; padding-left: 0px" >
    <table border="0" >
      <tr >
        <td align="left" width="135px" >
          <table><tr>
            <td width="0px">
              <a id="loadA<%=tripId%>" style="visibility: hidden; display: none">
                </a>
              <input type="checkbox" align="left" name="mytour"
                onclick="buffer(<%=tripId%>);" id="<%=tripId%>" />
            </td>
            <td width="115px">
              <a id="tripName<%=tripId%>"
                onclick="document.getElementById('<%=tripId%>').checked=true;buffer(<%=tripId%>);
                return false;"><%=aTrip.getName() %> </a>
            </td>
          </tr>
        </table>
      </td>
      <td align="right" >
        <a id="play<%=tripId%>" href="#" onclick=" buffer2(<%=tripId%>);" >
          <img id="playIcon<%=tripId%>"
            width="20" height="20" src="views/images/play.png" > </a>
        </td>
      </tr>
    </table>
  </li>
  <%= %></ul>
<p class="more" style="left:8px"><input type="text" onkeyup="searchMyToursAjax(this.value)"
  name="searchtour" value="" placeholder="Search tour" size="15" /></p>
<a href="Controller?action=ShowNewTourPage" class="more">Create New Tour</a>

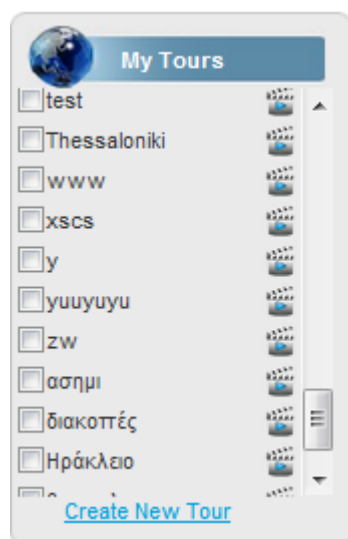
</div>

```

JAVA ΚΩΔΙΚΑΣ

JAVA ΚΩΔΙΚΑΣ

Εικόνα 17. Παράδειγμα Jsp Κώδικα για δημιουργία του menu ταξιδιών. Μέσα σε `<%%>` φαίνεται ο κώδικας java. Δύο τμήματα Java κώδικα περικλείονται σε κόκκινο πλαίσιο. Τα `<%%>` περιβάλλονται από HTML κώδικα.



Εικόνα 18. Αποτέλεσμα χρήσης του παραπάνω jsp κώδικα για σχηματισμό μενού ταξιδιών.

3.6 Αρχιτεκτονική συστήματος –Χρήση BCED και MVC

Το μοντέλο σχεδιασμού (design approach) που εφαρμόσαμε είναι το BCED, το οποίο ορίζει 4 πακέτα κλάσεων-επίπεδα: **Boundary**, **Control**, **Entity** και **Database_Interface**.

3.6.1 Boundary Layer

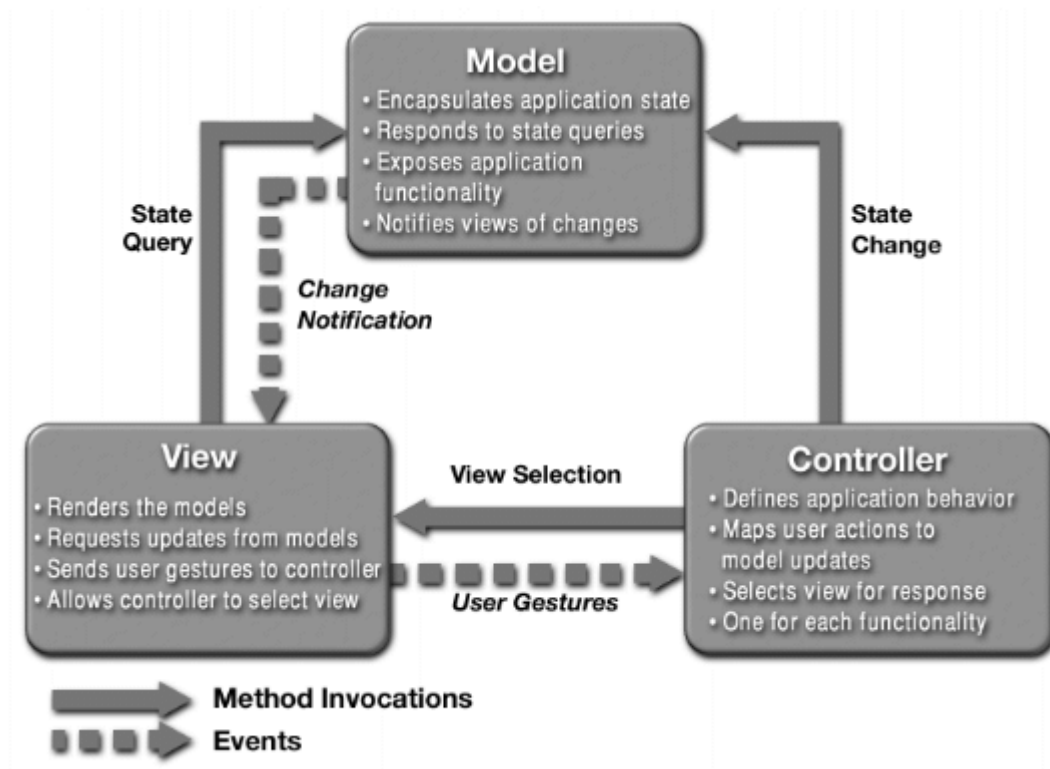
1. Οι **Boundary** κλάσεις είναι υπεύθυνες για την επικοινωνία με το χρήστη έτσι ώστε να γίνεται δυνατή η εισαγωγή ή παρουσίαση των δεδομένων. Εκτελούν λειτουργίες όπως η ανταπόκριση στην είσοδο του χρήστη, χειρισμός γεγονότων αλληλεπίδρασης με το χρήστη, χειρισμό συνθηκών λαθών της εισόδου του χρήστη κλπ. Ανήκουν στο κομμάτι του User Interface.

3.6.2 Control Layer- Χρήση MVC

2. Οι **Control** κλάσεις υλοποιούν το business logic της εφαρμογής. Ενθυλακώνουν κανόνες απόφασης ανάλογα με τις ενέργειες του χρήστη και παίζουν σημαντικό ρόλο στην επικοινωνία του συστήματος με το χρήστη. Επικοινωνούν με τις Boundary και Entity κλάσεις. Οι Control κλάσεις ανήκουν στο κομμάτι της παρουσίασης (Presentation Tier).

Σε αυτό το επίπεδο χρησιμοποιούμε το **MVC (Model-View-Controller) design pattern** [18]. Αυτό μας επιτρέπει να διαχωρίζουμε την κύρια λειτουργικότητα από το user interface, καθώς επίσης να χρησιμοποιούμε πολλαπλά interface για τα ίδια δεδομένα πχ (διαφορετική απεικόνιση για κινητά). Επίσης σημαντικό πλεονέκτημα της χρήσης του MVC είναι το γεγονός ότι μπορούμε πολύ εύκολα μελλοντικά να αλλάξουμε το επίπεδο δεδομένων(πχ αλλαγή βάσης δεδομένων) αφού τα επίπεδά του είναι ανεξάρτητα και αυτόνομα. Τα στοιχεία του MVC αναλύονται ως εξής:

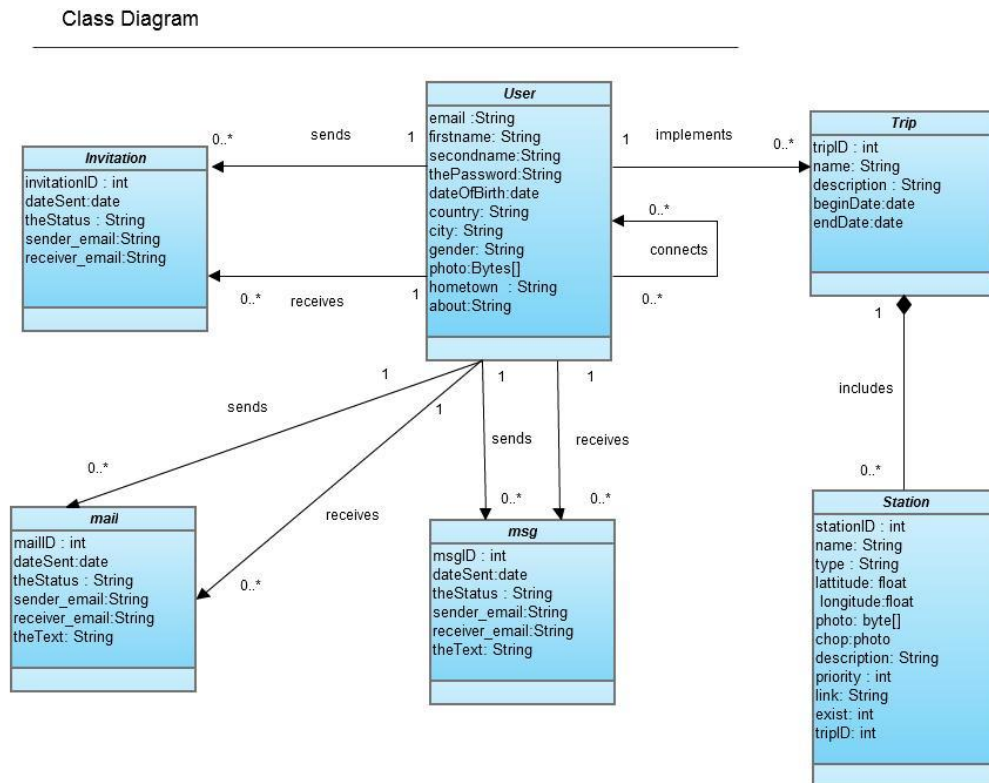
- **Model:** Αναπαριστά δεδομένα και τους κανόνες οι οποίοι διέπουν την πρόσβαση σε αυτά, καθώς και την ενημέρωσή τους. Είναι υπεύθυνο για την υλοποίηση του business logic της εφαρμογής και υλοποιείται με Java Beans. Στην ουσία το μοντέλο (*Model*) αναπαριστά τα δεδομένα της εφαρμογής.
- **View:** Παρουσιάζει τα περιεχόμενα ενός model. Ορίζει ακριβώς πώς τα στοιχεία του model πρέπει να παρουσιαστούν. Εάν η πληροφορία του model αλλάξει, το view πρέπει να ενημερώσει την παρουσίασή του όπως απαιτείται. Υλοποιείται με JSPs, οποία αναλύσαμε παραπάνω. Με άλλα λόγια η προβολή (*View*) παράγει μια παρουσίαση των δεδομένων του μοντέλου.
- **Controller:** Μεταφράζει την αλληλεπίδραση του χρήστη με το view σε ενέργειες που θα εκτελέσει το model. Ο ελεγκτής (*Controller*) διαχειρίζεται και κατευθύνει τις αιτήσεις που κάνει ο χρήστης. Η υλοποίηση του μπορεί να γίνει με ένα Servlet κάτι το οποίο κάνουμε και στην εφαρμογή μας.



Εικόνα 19. Διαγραμματική απεικόνιση του MVC.

3.6.3 Entity Layer

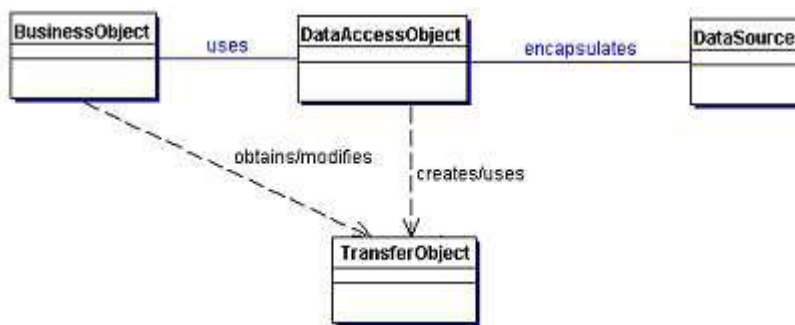
3. Οι **Entity** κλάσεις περιγράφουν αντικείμενα που αναπαριστούν τις κύριες οντότητες της εφαρμογής μας. Επικοινωνούν με τις Boundary και Control κλάσεις στα δύο παραπάνω επίπεδα και με τις Database_Interface κλάσεις στο παρακάτω επίπεδο. Ανήκουν στο **Business Logic** κομμάτι της αρχιτεκτονικής. Το Class Diagram που περιγράφει τις κλάσεις του entity layer της εφαρμογής μας είναι το παρακάτω και έχει προκύψει με βάσει τις απαιτήσεις του συστήματός μας.



Εικόνα 20. Class Diagram της εφαρμογής μας.

3.6.4 Database Interface Layer.

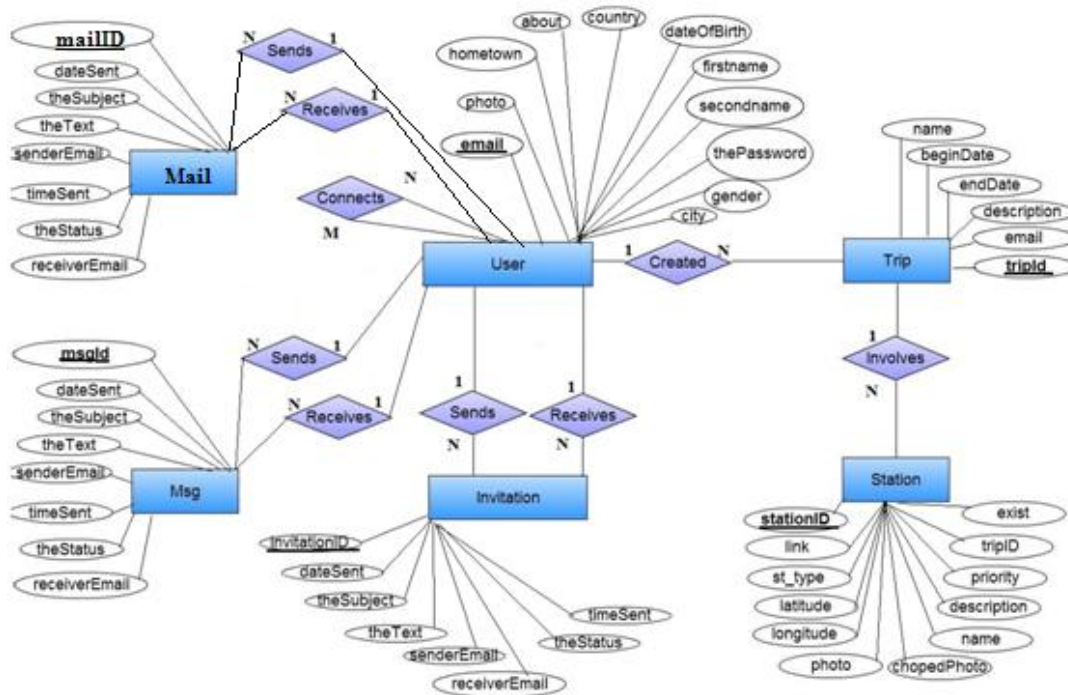
Το επίπεδο αυτό υλοποιείται από τις κλάσεις, οι οποίες είναι υπεύθυνες για την αντιστοίχιση των entity κλάσεων με τους αντίστοιχους πίνακες της βάσης δεδομένων. Οι κλάσεις αυτές επικοινωνούν με τις entity κλάσεις και με τη βάση δεδομένων. Το επίπεδο αυτό υλοποιείται με την χρήση των Data Access Objects (DAO) [19]. Τα Daos διευκολύνουν την πρόσβαση στην πηγή δεδομένων (data source), δηλαδή τη βάση δεδομένων για την περίπτωσή μας. Στην ουσία είναι κλάσεις Java οι οποίες παρέχουν όλους τους απαραίτητους μηχανισμούς για την πρόσβαση στην πηγή, δηλαδή συναρτήσεις ανάκτησης και εγγραφής στη βάση δεδομένων. Η χρήση των Dao εκτός του ότι εξασφαλίζει την απόκρυψη από τους clients, των λεπτομερών υλοποίησης της πηγής, διαχωρίζει το business logic της εφαρμογής από το persistent logic. Έτσι μελλοντική αλλαγή στην βάση δεδομένων δεν θα επηρεάσει τους clients πουθενά. Όπως εικονίζεται και παρακάτω, όπου φαίνεται ο ρόλος των Daos στο σύστημα μας, διαμεσολαβούν ανάμεσα στο business logic και το persistent. Δηλαδή τη βάση δεδομένων.



Εικόνα 21. Διάγραμμα απεικόνισης του ρόλου των Dao. Business Object: Ο client προς τα δεδομένα. Το αντικείμενο που πρέπει να έχει πρόσβαση στην βάση. DataAccessObject: Το επίπεδο υλοποίησης της πρόσβαση στα δεδομένα. Data Source: Η βάση δεδομένων. Transfer Object: Χρησιμοποιείται για τη μεταφορά δεδομένων, δηλαδή φέρει όλα τα στοιχεία που ανακτήσαμε από τη βάση.

Τα Daos πρέπει να πούμε ότι επικοινωνούν με τη βάση δεδομένων χρησιμοποιώντας μία συνάρτηση που με τη σειρά της καλεί διάφορα stored procedures που είναι συνάρτησης ανάκτησης για μία βάση δεδομένων. Η οργάνωση του Database Interdace Layer γίνεται ως εξής: Υπάρχει ένα DOFactory class, το οποίο κατασκευάζει αντικείμενα τύπου (όνομα βάσης)DAOFactory για διάφορες βάσεις δεδομένων. Στην περίπτωση μας ,χρησιμοποιούμε MySQL. Το MySQLDAOFactory class συνδέεται με τη βάση και φτιάχνει αντικείμενα τύπου MySQLUserDAO ,MySQLTripDao κτλ. Οι μέθοδοι που υλοποιούν βρίσκονται συγκεντρωμένοι στα αντίστοιχα interfaces.

3.6.5 Περιγραφή βάσης δεδομένων-Σχεσιακό σχήμα.



Εικόνα 22. ER που περιγράφει τη βάση δεδομένων της εφαρμογής μας.

Η βάση δεδομένων που υλοποιήθηκε για την υποστήριξη της εφαρμογής μας περιγράφεται από το παραπάνω Entity Relationship Diagram (ER) το λεγόμενο διάγραμμα σχέσεων οντοτήτων. Στο διάγραμμα αυτό με μπλε τετράγωνα παρουσιάζονται οι διάφορες οντότητες. Αυτές όπως φαίνεται είναι:

- Η οντότητα **User** που αντιστοιχεί στους χρήστες της εφαρμογής.
- Η οντότητα **Trip** που αντιστοιχεί τα ταξίδια που δημιουργούν οι χρήστες.
- Η οντότητα **Station** που αντιστοιχεί στους σταθμούς, το σύνολο των οποίων υλοποιούν ένα ταξίδι.
- Η οντότητα **Invitation** που αντιστοιχεί στα αιτήματα φιλίας που κάνει ένα χρήστης σε κάποιον άλλο.
- Η οντότητα **Msg** που αντιστοιχεί στα ενημερωτικά μηνύματα που αποστέλλονται στους χρήστες και αφορούν τις κινήσεις των φίλων τους.
- Η οντότητα **Mail** που αντιστοιχεί στα μηνύματα που ανταλλάσσουν οι χρήστες για να επικοινωνήσουν.

Η κάθε οντότητα περιγράφεται από τα attributes της, τα οποία είναι τα χαρακτηριστικά της. Αυτά αναγράφονται μέσα στα μπαλόνια που συνδέονται με τις οντότητες.

Οι σχέσεις που συνδέουν τις διάφορες οντότητες μεταξύ τους εικονίζονται με μορφή ρόμβους, και μαρτυρούν την αλληλεπίδραση η οποία υπάρχει. Για παράδειγμα η σχέση created μας λέει ότι 1 χρήστης μπορεί να δημιουργήσει πολλά ταξίδια (N), ενώ ένα συγκεκριμένο ταξίδι μπορεί να ανήκει σε ένα μόνον χρήστη.

Εξετάζοντας τις διάφορες σχέσεις μεταξύ των οντοτήτων υλοποιούμε το παρακάτω σχεσιακό σχήμα, όπου παρουσιάζονται οι πίνακες που προκύπτει ότι απαιτούνται για

την υλοποίηση της βάσης δεδομένων. Τα **κύρια κλειδιά (primary keys)** είναι υπογραμμισμένα, ενώ τα **ξένα κλειδιά (foreign keys)** επισημαίνονται με κόκκινο χρώμα. Στην περίπτωση που το ξένο κλειδί (στην Connects) είναι ταυτόχρονα και κλειδί, τότε και υπογραμμίζουμε και επισημαίνουμε με κόκκινο χρώμα.

Σχεσιακό Σχήμα:

User(email, photo, hometown, about, country, dateOfBirth, firstname, secondname, thePassword, gender, city)

Trip(tripID, name, beginDate, endDate, description, email)

Station(stationID, link, st_type, latitude, longitude, photo, choppedPhoto, name, description, priority, exists, tripID)

Invitation(invitationID, dateSent, theSubject, theText, theStatus, timeSent, senderEmail, receiverEmail)

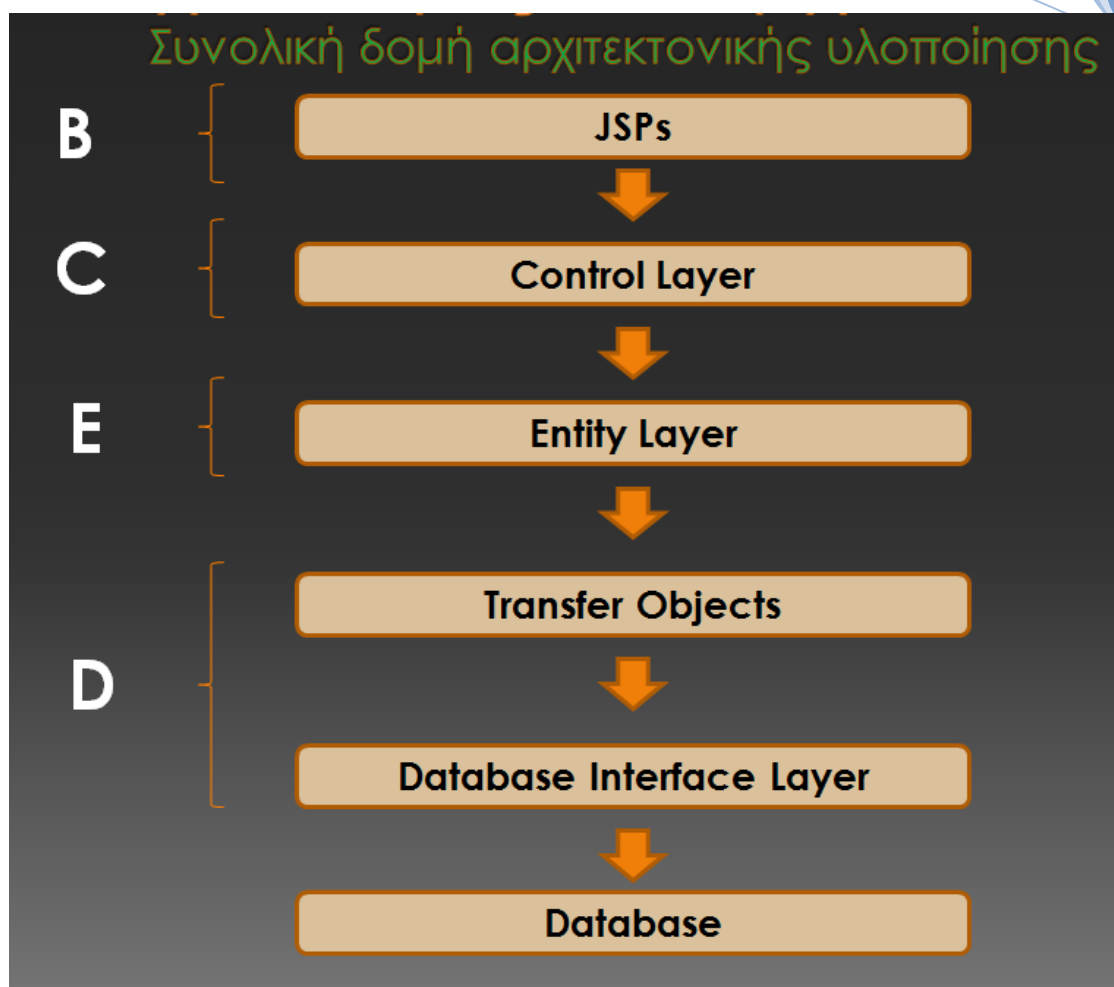
Msg(msgID, dateSent, theSubject, theText, theStatus, timeSent, senderEmail, receiverEmail)

Mail(mailID, dateSent, theSubject, theText, theStatus, theSentStatus, timeSent, senderEmail, receiverEmail)

Connects(email, connectedWithEmail, dateOfJoin)

3.6.6 Διαγραμματική απεικόνιση δόμησης επιπέδων της εφαρμογής (BCED)

Η διαγραμματική απεικόνιση των επιπέδων του BCED μοντέλου που ακολουθήσαμε στην εφαρμογή μας, όπως αυτά αναλύθηκαν παραπάνω έχει την εξής μορφή:



Εικόνα 23. Διάγραμμα δόμησης επιπέδων BCED.

3.7 Προγραμματιστικά εργαλεία της εφαρμογής

3.7.1 NetBeans 6.8



Εικόνα 24. Λογότυπο NetBeans

Η εφαρμογή μας υλοποιήθηκε το περιβάλλον NetBeans. Το NetBeans αποτελεί μια ανοιχτή πλατφόρμα ανάπτυξης λογισμικού σε γλώσσες Java, JavaScript, PHP, Python, Ruby, Groovy, C και C++. Επίσης αποτελεί και ένα ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment-IDE) και προσφέρει εργαλεία εύκολα στη χρήση. Τα εργαλεία αυτά καλύπτουν όλα τα στάδια: από την υλοποίηση και τη μεταγλώττιση έως την εφαρμογή του λογισμικού που παράχθηκε. Για την παρούσα εφαρμογή έγινε χρήση της έκδοσης NetBeans IDE 6.8. Το NetBeans 6.8 αποτελεί το βασικό περιβάλλον ανάπτυξης προγραμματιστικού κώδικα και πολύπλοκων εφαρμογών. Οι εφαρμογές που μπορούν να αναπτυχθούν στο περιβάλλον ανάπτυξης NetBeans περιλαμβάνουν από απλές Java εφαρμογές επιπέδου κλάσεων και κονσόλας μέχρι πολύπλοκες διαδικτυακές και επιχειρηματικές εφαρμογές με γραφικό περιβάλλον και διαδικτυακές υπηρεσίες. Επομένως γίνεται κατανοητό ότι αποτελεί το περιβάλλον στο οποίο στηρίχθηκε η ανάπτυξη, η υλοποίηση, η ενορχήστρωση, η εκτέλεση και η αποθήκευση/δημοσίευση των διαδικτυακών υπηρεσιών και των σύνθετων επιχειρηματικών διαδικασιών [20].

3.7.2 Apache Tomcat 6.0



Εικόνα 25. Λογότυπο Apache Tomcat

Ο Apache Tomcat είναι ένας εξυπηρετητής εφαρμογών (Application Server) που αναπτύχθηκε από την Apache Software Foundation (ASF). Ο Tomcat, υλοποιεί τις προδιαγραφές των servlets και JavaServer Pages (JSP) που έχει θέσει η Sun Microsystems προσφέροντας ένα περιβάλλον για εκτέλεση Java κώδικα σε

συνεργασία με ένα web server. Παρέχει εργαλεία στο γραφικό του περιβάλλον για ρύθμιση και διαχείριση αλλά το ίδιο μπορεί επίσης να γίνει από αρχεία ρυθμίσεων (configuration files) που είναι συνήθως σε μορφή XML. Το tomcat περιέχει επίσης ενσωματωμένο έναν HTTP server. Στην εργασία χρησιμοποιήθηκε η 6.0.20 έκδοση του Apache Tomcat, η οποία ικανοποιεί τις προδιαγραφές των Servlet και JSP [21] .

3.7.3 MySQL Server 5.1 - JDBC



Εικόνα 26.Λογότυπο MySQL

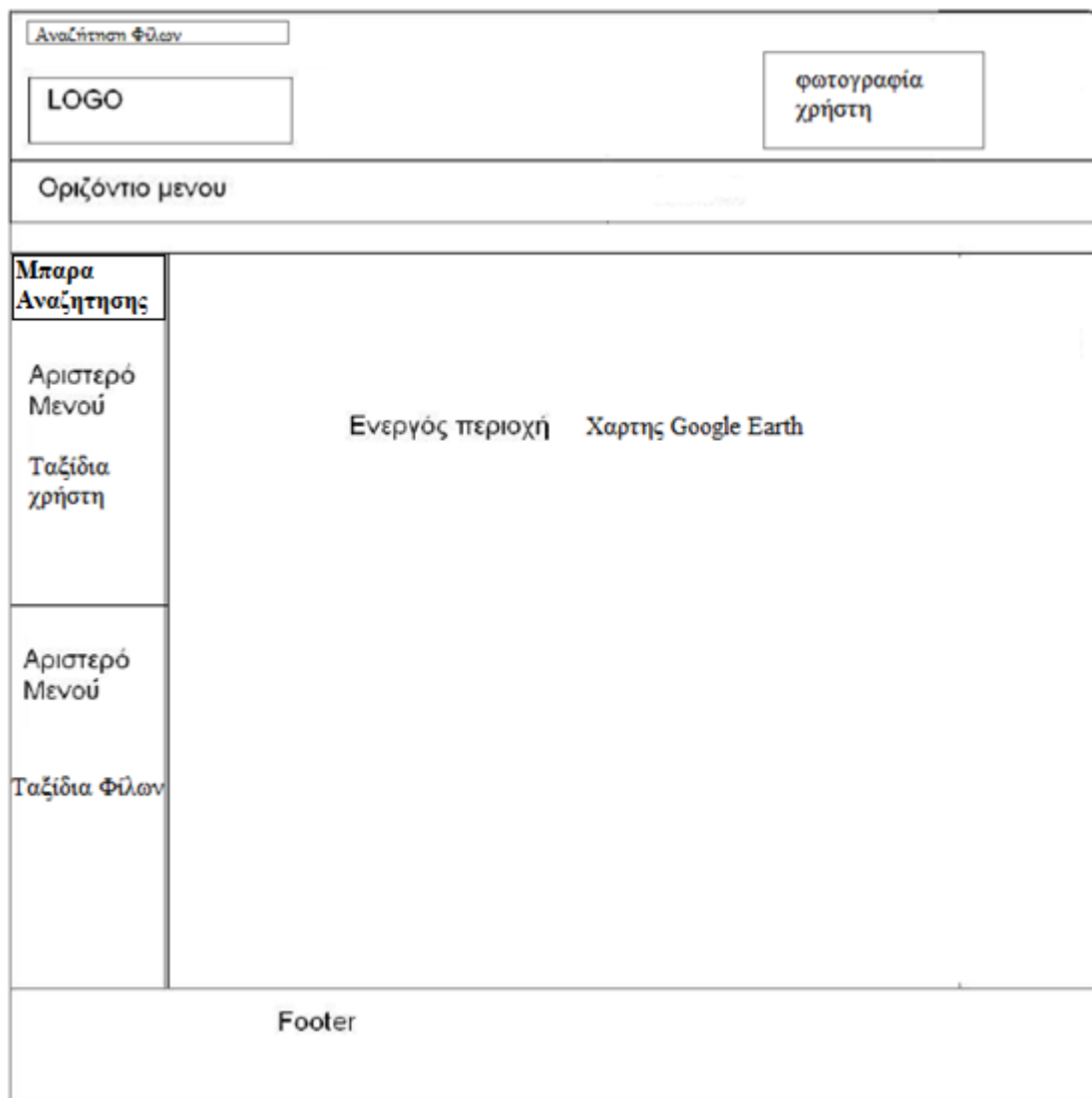
Για την αποθήκευση των δεδομένων, χρησιμοποιήθηκε όπως αναφέρθηκε και προηγουμένως βάση δεδομένων τύπου MySQL. Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Το εργαλείο το οποίο χρησιμοποιήθηκε για το χειρισμό της βάσης ήταν ο MySQL Server. Η έκδοση που χρησιμοποιήθηκε ήταν η 5.1. Για την επικοινωνία της βάσης με την εφαρμογή μας χρησιμοποιήθηκε ο JDBC Driver (MySQL Connector). Αυτός είναι ένας οδηγός σύνδεσης και επικοινωνίας βάσεων δεδομένων με εφαρμογές γραμμένες σε Java και είναι επίσης γραμμένος σε Java [22].

4. Αρχές Σχεδίασης της εφαρμογής

Η σωστή σχεδίαση του interface μίας εφαρμογής αποτελεί βασική προϋπόθεση ώστε αυτή να θεωρείται επιτυχημένη. Σημαντικό είναι να μπορεί να συνδυάζει ελκυστικότητα και ευχρηστία. Να παροτρύνει και όχι να αποθαρρύνει το χρήστη να την χρησιμοποιήσει. Για να επιτευχτεί αυτό κρίνεται αναγκαίο να ακολουθήσουμε κάποιες βασικές αρχές σχεδίασης. Εξίσου σημαντική είναι και η δομή των σελίδων ώστε να δίνεται η αίσθηση της αρτιότητας αλλά και να διευκολύνεται ο χρήστης κατά το χειρισμό [23].

4.1 Περιγραφή βασικής δομής σελίδων

Το βασικό template της εφαρμογής μας αποτελείται από εννέα επιμέρους τμήματα, όπως φαίνεται και στην παρακάτω εικόνα.



Εικόνα 27,Βασικό template.



Εικόνα 28.Live View του template της εφαρμογής.

Στην Κορυφή του template όπως παρατηρούμε υπάρχει μία μικρή οριζόντια περιοχή, η οποία είναι μπάρα αναζήτησης φίλων. Αμέσως παρακάτω από αυτήν υπάρχει το logo της εφαρμογής. Στην ευθεία του logo και δεξιά από αυτό βλέπουμε τον χώρο τοποθέτησης της φωτογραφίας του χρήστη όταν αυτός έχει εισέλθει στο σύστημα..

Αμέσως παρακάτω από την περιοχή της εικόνας του χρήστη και του logo υπάρχει η οριζόντια περιοχή για το βασικό μενού της εφαρμογής μας. Το μενού αυτό αποτελείται από εικονίδια τα οποία παραπέμπουν σε διάφορες βασικές λειτουργίες που παρέχονται στο χρήστη. Αυτά είναι:

Το **Homepage**, που μας οδηγεί στην αρχική σελίδα.

Το **My profile**, που οδηγεί στην σελίδα του προφίλ του χρηστή.

Το **My tours**, για τη σελίδα διαχείρισης των ταξιδιών του χρήστη.

Το **My Messages**, για τη σελίδα διαχείρισης των ταξιδιών του χρήστη.

Το **News**, όπου ο χρήστης μπορεί να ενημερωθεί για τα νέα που τον αφορούν, σε σχέση με την αποδοχή αιτημάτων φιλίας από φίλους ή τη δημιουργία ταξιδιών από αυτούς.

Το **Friend Requests**, όπου ο χρήστης ελέγχει αν έχει αιτήματα φιλίας.

Το **My Friends**, όπου παραπέμπει στην διαχείριση των φίλων.

Και τέλος το **Logout**, όπου ο χρήστης μπορεί να κάνει αποσύνδεση από το λογαριασμό του.

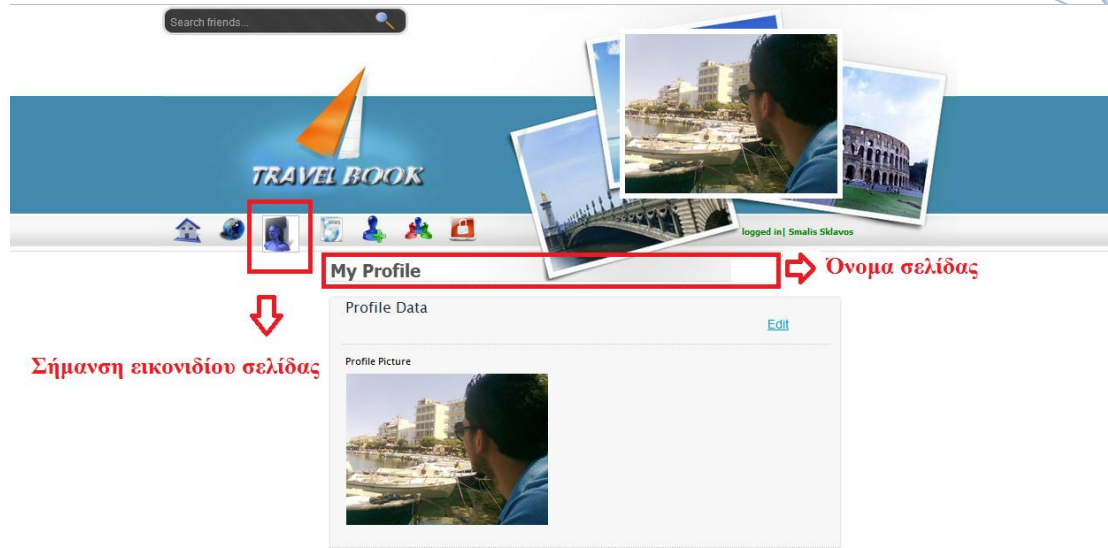
Όσον αφορά το αριστερό μέρος της σελίδας, εκεί υπάρχει το μενού των ταξιδιών. Το μενού αυτό χωρίζεται σε τρία επιμέρους μενού. Το ένα είναι αυτό που προκύπτει έπειτα από αναζήτηση του ονόματος ενός ταξιδιού με όλα τα πιθανά αποτελέσματα. Τα ταξίδια που έχουν βρεθεί συμφωνούν με την εισαγωγή μας στην μπάρα αναζήτησης και μπορεί να είναι είτε προσωπικά, είτε φίλων δεύτερο μενού είναι αυτό των προσωπικών ταξιδιών του χρήστη και το άλλο αυτό των ταξιδιών των φίλων του χρήστη. Τα ταξίδια που υπάρχουν στις περιοχές αυτές ο χρήστης μπορεί να επιλέξει αν θέλει να τα δει σε μορφή απλής απεικόνισης, ή σε μορφή live tour. Για τα ταξίδια των φίλων πρέπει να επισημάνουμε ότι είναι ομαδοποιημένα ανάλογα με το χρήστη-φίλο στον όποιον ανήκουν.

Η ενεργός περιοχή, δηλαδή το κεντρικό τμήμα της σελίδας, καλύπτεται από το χάρτη του Google Earth. Εκεί πάνω γίνονται όλες οι απεικονίσεις το ταξιδιών. Πάνω ακριβώς από το χάρτη υπάρχουν κάποιες επιλογές για το χειρισμό του χάρτη και των ρυθμίσεων προβολής, όπως για παράδειγμα ενεργοποίηση των 3D όψεων για κτήρια ή δέντρα.

Τελειώνοντας με τη διάταξη του template, πρέπει να αναφέρουμε την ύπαρξη του οριζόντιου τμήματος στο κάτω μέρος της σελίδας το οποίο ονομάζεται footer. Είναι το τμήμα με το οποίο τελειώνει η κάθε σελίδα και περιέχει κάποιους χρήσιμους συνδέσμους, όπως και το copyright.

4.2 Ενημέρωση χρήστη

Στη σχεδιάσή μας λάβαμε υπόψη την ανάγκη που έχει ο χρήστης να γνωρίζει ανά πάσα στιγμή τη θέση στην οποία βρίσκεται μέσα στο σύστημα. Μία τακτική που ακολουθείται για να επιτευχθεί αυτό είναι στην περίπτωση του κεντρικού οριζόντιου menu να φωτίζεται με λευκό background το εικονίδιο της επιλεγμένης σελίδας. Αυτό βέβαια ισχύει μόνο για τις σελίδες που ορίζονται από το menu. Ένα παράδειγμα εικονίζεται παρακάτω, όπου ο χρήστης έχει επιλέξει να δει τη σελίδα του προσωπικού του profile με το αντίστοιχο εικονίδιο να είναι φωτισμένο. Στα πλαίσια της ενημέρωσης του χρήστη, σε όλες τις σελίδες υπάρχει ένα τίτλος πάνω από την ενεργό περιοχή που λέει στο χρήστη σε ποια σελίδα βρίσκεται. Αυτό φαίνεται επίσης στην εικόνα που ακολουθεί. Έτσι μπορούμε να πούμε ότι η εφαρμογή μας ακολουθεί τον κανόνα ότι ο χρήστης κάθε φορά πρέπει να μπορεί να δει τη θέση του στην εφαρμογή για να μην χάνει την αυτοπεποίθησή του (Nielsen #1). Ο χρήστης ενημερώνεται επίσης κατά τη συμπλήρωση των διάφορων φορμών. Όταν εστιάσει σε κάποιο πεδίο εάν κρίνεται αναγκαίο αναδύεται ένα ενημερωτικό μήνυμα δίπλα από το πεδίο. Για παράδειγμα στην περίπτωση του πεδίου του password, ο χρήστης ενημερώνεται ότι πρέπει να χρησιμοποιήσει το λιγότερο έξι χαρακτήρες. Τέλος, ο χρήστης ενημερώνεται για την επιτυχία ή μη, έκβαση των ενεργειών του με τα κατάλληλα μηνύματα.



Εικόνα 29. Παράδειγμα ενημέρωσης χρήστη.

Εικόνα 30. Ενημέρωση χρήστη με αναδυόμενα μηνύματα.

4.3 Καθοδήγηση Χρήστη

Το User Interface που σχεδιάσαμε καθοδηγεί το χρήστη στην πραγματοποίηση διαφόρων ενεργειών. Ο χρήστης διευκολύνεται στην πλοήγηση στις σελίδες μέσω κουμπιών και οδηγιών που αποτελούν ταυτόχρονα links προς τις επόμενες σελίδες. Για παράδειγμα, στη δημιουργία νέου ταξιδιού αρχικά ο χρήστης οδηγείται στη σελίδα συμπλήρωσης των βασικών στοιχείων του ταξιδιού (όνομα, ημερομηνίες, περιγραφή). Στη συνέχεια, εφόσον ο χρήστης ολοκλήρωσε τη συμπλήρωση, το σύστημα τον οδηγεί στην σελίδα δημιουργίας πάνω στο χάρτη του ταξιδιού. Εφόσον ο χρήστης θεωρεί ότι ολοκλήρωσε την δουλειά του, τότε αποθηκεύοντας το ταξίδι οδηγείται ξανά στην αρχική σελίδα, όπου μπορεί πλέον να δει το ταξίδι που έφτιαξε.

4.4 Ομαδοποίηση Λειτουργιών

Στην παρακάτω εικόνα φαίνεται πως τηρούνται οι απαιτήσεις για ομαδοποιημένη λειτουργικότητα στην εφαρμογή μας, η οποία αποσκοπεί στην διευκόλυνση του χρήστη.



Η σελίδα που εικονίζεται είναι η αρχική. Όπως βλέπουμε, στα κόκκινα πλαίσια υπάρχουν ομαδοποιημένες ενέργειες που μπορεί να κάνει ο χρήστης. Στο πάνω μέρος βρίσκονται οι βασικές λειτουργίες για την πλοήγηση στην εφαρμογή, το χειρισμό του προφίλ και τη διαχείριση των συνδέσεων με άλλους χρήστες. Στο αριστερό μέρος φαίνεται ότι είναι ομαδοποιημένα τα ταξίδια του χρήστη και τα ταξίδια των φίλων του χρήστη. Στο τετράγωνο των λειτουργιών του χάρτη βρίσκονται οι επιλογές εμφάνισης για το χάρτη των ταξιδιών (3D κτήρια, δέντρα κτλ). Με αυτόν τον τρόπο, οι διάφορες λειτουργίες που μπορεί να εκτελέσει ένας χρήστης εμφανίζονται ομαδοποιημένα, κάτι που βοηθάει στο να μην μπερδεύεται ο χρήστης και να διατηρείται μία τάξη στις σελίδες της εφαρμογής μας.

5.5 Ευκολία πλοήγησης

Το User Interface σχεδιάστηκε με γνώμονα την εύκολη πλοήγηση στις σελίδες από έναν χρήστη της εφαρμογής, λαμβάνοντας υπόψη ότι η εφαρμογή μπορεί να χρησιμοποιείται από χρήστες οι οποίοι δεν είναι απόλυτα εξοικειωμένοι με το διαδίκτυο. Για παράδειγμα, οι διάφορες λειτουργίες που μπορεί να εκτελέσει ένας χρήστης και αποτελούν πολλές φορές **links** προς άλλες σελίδες, έχουν διαφορετικό χρώμα από το υπόλοιπο κείμενο, ενώ συχνά χρησιμοποιούνται **εικονίδια** (π.χ. στο μενού στο πάνω μέρος της σελίδας), έτσι ώστε ο χρήστης να μπορεί να τις ξεχωρίσει. Υπάρχουν ωστόσο, λειτουργίες οι οποίες δεν μπορούν να εκτελεστούν από οποιοδήποτε χρήστη. Για παράδειγμα, ένας χρήστης δεν μπορεί να καταχωρήσει ένα ταξίδι, εάν δεν έχουν συμπληρωθεί τα απαραίτητα πεδία στη φόρμα καταχώρησης της ταξιδιού. Σε αυτήν την περίπτωση, εμφανίζεται **μήνυμα λάθους** στον χρήστη, ο

οποίος καλείται να συμπληρώσει σωστά τη φόρμα, σύμφωνα με τις οδηγίες που του δίνονται.

4.6 Επιλογή Χρωμάτων

Η επιλογή χρωμάτων έγινε έτσι ώστε το interface της εφαρμογής να είναι φιλικό ,ευχάριστο για το χρήστη και ταυτόχρονα επαγγελματικό, αποδοτικό, και να διευκολύνει την πλοήγηση του χρήστη. Σύμφωνα και με τον παρακάτω πίνακα, για την επιλογή χρωμάτων και background επιλέχθηκαν τα βασικά χρώματα: μπλε, άσπρο, μαύρο τα οποία εκτός του ότι είναι ευχάριστα και ξεκούραστα για το μάτι, παρουσιάζουν αντίθεση μεταξύ τους . Αυτό έχει ως αποτέλεσμα τα τμήματα που περιγράφουν να είναι ευδιάκριτα μεταξύ τους .

User Interface Design Principles & Guidelines Screen Layout And Design (Mayhew)								
<u>Color : Experimental Results</u> Background for text displays								
Foreground	Black	Blue	Green	Cyan	Red	Magenta	Brown	White
Black				v		v		v
Blue								v
Green	v	v		v				
Cyan	v							
White	v							
Bold Green	v							
Bold Cyan	v	v		v	v	v		
Bold Magenta	v		v					
Yellow	v	v		v		v		
Bold White	v		v	v				

Εικόνα 31. Πίνακας ορθής διαχείρισης χρωμάτων.

Ωστόσο, σε κάποιες περιπτώσεις στην εφαρμογή μας, χρειάζεται να τονίσουμε ότι κάποιες ενέργειες του χρήστη είναι αποτυχημένες ή οδηγούν σε αποτυχία, ενώ άλλες οδηγούν σε επιτυχία εκτέλεσης της ζητούμενης λειτουργίας. Χρησιμοποιούνται το

κόκκινο και το πράσινο χρώμα αντίστοιχα, όπως παρατηρούμε στην παρακάτω εικόνα:

Search friends...

TRAVEL BOOK

logged in | Smalis Sklavos

Create Tour

Basic Tour Info Form
You should complete this form to create a new tour.
The fields with * are mandatory.

Tour Info

Tour Name *
Athens
You already have a tour with the same name.

Begin Date *
January / 1 / 2010

End Date *
January / 1 / 2010

Description

[Reset](#) [Save](#)

Home Page | My Tours | My Profile | News | Friend Requests | My Friends | Logout

Copyright ©. All rights reserved. Design by Smalis Sklavos

Εικόνα 32. Παράδειγμα λανθασμένης εισαγωγής. Χρήση κόκκινου χρώματος για να τονιστεί το λάθος.

Επίσης χρώματα που έχουν κάποια σημασιολογία είναι τα χρώματα των συνδέσμων (links). Όλοι οι σύνδεσμοι έχουν το ίδιο γαλάζιο χρώμα ώστε ο χρήστης να τους αναγνωρίζει ευκολότερα και γρηγορότερα. Ένα παράδειγμα link εικονίζεται παρακάτω.

Search friends...

TRAVEL BOOK

logged in! Smalis Sklavos

My Tours

Name	Begin Date	End Date		
0	2010-01-01	2010-01-01		X
11	2010-01-01	2010-01-01		X
1221212	2010-01-01	2010-01-01		X
ass	2010-01-01	2010-01-01		X
Athens	2010-01-01	2010-01-01		X
crete	2010-01-01	2010-01-01		X
Ikaria	2010-01-01	2010-01-01		X
Kithi	2010-01-01	2010-01-01		X
Messara	2010-01-01	2010-01-01		X
metrisi apostash	2010-01-01	2010-01-01		X
qwqw	2010-01-01	2010-01-01		X
Thessaloniki	2012-03-10	2012-09-15		X
wawaw sasas	2010-01-01	2010-01-01		X
Ακρόπολη	2010-01-01	2010-01-01		X
ασημι	2010-01-01	2010-01-01		X
Ηράκλειο	2010-01-01	2010-01-01		X
Θεσσαλονικη	2010-01-01	2010-01-01		X
χρος	2010-01-01	2010-01-01		X

Create New Tour!

Page: 1

Σύνδεσμος με γαλάζιο χρώμα

Εικόνα 33. Παράδειγμα συνδέσμων. Χρήση γαλάζιου χρώματος.

4.7 Συνέπεια

Το User Interface της εφαρμογής μας σχεδιάστηκε έτσι ώστε να τηρεί της αρχές που αφορούν τη συνέπεια. Χαρακτηριστικό παράδειγμα είναι το κύριο μενού για τις ενέργειες που μπορεί να πραγματοποιήσει ένας χρήστης, που διατηρείται στο πάνω μέρος της σελίδας, ανεξάρτητα από τη σελίδα στην οποία βρίσκεται ο χρήστης.

Επίσης, ένα άλλο παράδειγμα συνέπειας στην εφαρμογή μας είναι το γεγονός ότι οι σύνδεσμοι (links) στην ενεργή περιοχή επισημαίνονται σε όλες τις σελίδες με γαλάζιο χρώμα, ενώ όταν ο δείκτης του ποντικιού είναι πάνω σε αυτές, υπογραμμίζονται.

Τα κουμπιά τα οποία χρησιμοποιούνται για την ίδια λειτουργία διατηρούν τη θέση, το χρώμα, το όνομά τους και γενικά όλα τους τα χαρακτηριστικά σε όλες τις σελίδες του συστήματος. Έτσι π.χ τα κουμπιά αποθήκευσης είναι με πράσινο χρώμα, και

βρίσκονται πάντα κάτω δεξιά, ενώ αυτά που σημαίνουν ακύρωση βρίσκονται κάτω αριστερά.

Τέλος, τα χρώματα της εφαρμογής διατηρούνται φυσικά σταθερά και δεν αλλάζουν από σελίδα σε σελίδα.

4.8 Χρήση ενεργής οντότητας

Το user interface που υλοποιήσαμε χρησιμοποιεί όσο πιο αποδοτικά γίνεται την **ενεργή περιοχή** της οθόνης. Λόγω του ότι το μενού στα αριστερά της οθόνης διατηρείται σταθερό, ο κενός χώρος που δημιουργείται αποτελεί την ενεργή περιοχή της οθόνης μας, η οποία κάθε φορά απεικονίζει διαφορετικά δεδομένα με βάση τις πολλές διαφορετικές λειτουργίες της εφαρμογής. Για κάθε ξεχωριστή λειτουργία, τα αντίστοιχα δεδομένα (πίνακες, φόρμες, λίστες κλπ.) παρουσιάζονται στην ενεργή περιοχή της οθόνης.

Αυτό λοιπόν εξασφαλίζει ότι τα παραπάνω δεδομένα θα παρουσιάζονται μόνο στην ενεργή περιοχή της οθόνης και πουθενά αλλού. Δε γίνεται δηλαδή να κάποια δεδομένα να παρουσιαστούν, για παράδειγμα, στο κάτω μέρος της σελίδας και άλλα στο αριστερό μέρος.

4.9 Χειρισμός λαθών

Κατά τη σχεδίαση του User Interface, λάβαμε υπόψη τις περιπτώσεις όπου ένας χρήστης μπορεί να κάνει λάθος, οπότε κατέστη αναγκαίο να χειριστούμε ή να προβλέψουμε αυτά τα λάθη. Για παράδειγμα, η πρόβλεψη των λαθών πραγματοποιείται με τη χρησιμοποίηση radio buttons όταν για παράδειγμα, θέλουμε ο χρήστης να δηλώσει το φύλο του σε μία φόρμα. Επίσης, χρησιμοποιούμε κάποιες default λίστες, όταν π.χ ο χρήστης θέλει να επιλέξει ημερομηνία ή χώρα. Δεν απαιτούμε δηλαδή από το χρήστη να εισάγει εκείνος το όνομα της κατηγορίας στην οποία θέλει να κάνει αναζήτηση, αλλά αντίθετα, του παρέχουμε μία έτοιμη λίστα με τις υπάρχουσες κατηγορίες. Ο χειρισμός των λαθών γίνεται, επίσης, χρησιμοποιώντας τη γλώσσα JavaScript για κάποιες ενέργειες του χρήστη, έτσι ώστε να τον ενημερώσουμε για το λάθος που έκανε και να τον παραπέμψουμε στη σωστή ενέργεια. Για παράδειγμα, κατά την επεξεργασία ενός ταξιδιού, αν ο χρήστης εισάγει ένα μη αποδεκτό όνομα, τότε εμφανίζεται μηνύματα με κόκκινα γράμματα επισημαίνοντας το λάθος. Επίσης γίνεται χρήση pop up παραθύρων, που τον βοηθούν να καταλάβει άμεσα που βρίσκεται το λάθος και να το διορθώσει.

5. Λειτουργικότητας της εφαρμογής

5.1 Βασική Ιδέα

Βασική ιδέα για την ανάπτυξη της εφαρμογής μας, ήταν η δημιουργία ενός συστήματος κοινωνικού χαρακτήρα (social network), για την ανταλλαγή ταξιδιωτικών εμπειριών, χρησιμοποιώντας το χάρτη του Google Earth. Οι χρήστες της εφαρμογής θα μπορούν να συνδέονται μεταξύ τους, να περιγράψουν τα ταξίδια τους με ακρίβεια και γλαφυρότητα, λόγω των 3D απεικονίσεων, που προσφέρει το Google Earth. Ωστόσο, η εφαρμογή δεν περιορίζεται στην αναπαράσταση μονάχα ταξιδιών. Ο χρήστης μπορεί να αναπαραστήσει κάθε είδους γεγονός, όπως πχ ιστορικά γεγονότα, και να τα χρησιμοποιήσει για εκπαιδευτικό σκοπό.

5.2 Εκτεταμένη ανάλυση απαιτήσεων

Μία εφαρμογή για να μπορεί να ανταπεξέλθει στις απαιτήσεις ενός συστήματος όπως αυτό που χονδρικά περιγράφεται παραπάνω, θα πρέπει να μπορεί να ικανοποιεί κάποιες βασικές λειτουργίες, οι οποίες αναπτύσσονται στις παρακάτω παραγράφους.

5.2.1 Εγγραφή - Είσοδος - Έξοδος στο σύστημα

Χρήση της συγκεκριμένης εφαρμογής, εφόσον ανήκει στην κατηγορία των κοινωνικών, μπορεί να γίνει μόνο από χρήστες που διατηρούν ένα προφίλ. Συνεπώς, για να έχει κάποιος χρήστης δικαίωμα χρήσης της εφαρμογής πρέπει πρώτα να κάνει εγγραφή. Κατά την εγγραφή, είναι απαραίτητη η συμπλήρωση κάποιων πεδίων προσωπικών δεδομένων που αποτελούν μέλη μιας φόρμας προσωπικών δεδομένων. Στα πεδία αυτά συμπεριλαμβάνεται η φωτογραφία προφίλ του χρήστη, το ονοματεπώνυμο, η πόλη του κτλ. Εφόσον ολοκληρωθεί η διαδικασία εγγραφής και τα στοιχεία είναι έγκυρα χρήστης θα μπορεί πλέον να εισέρχεται όποτε θέλει στο σύστημα, εισάγοντας κάθε φορά το σωστό προσωπικό κωδικό και username. Σε περίπτωση που ο χρήστης ξεχάσει το κωδικό του, όπως πολλές φορές συμβαίνει, θα υπάρχει δυνατότητα ανάκτησης του με αποστολή στο email του χρήστη.

5.2.2 Διαχείριση προφίλ

Σημαντικό είναι ο χρήστης να μπορεί να δει ολοκληρωμένο το προφίλ που διατηρεί, και να αλλάξει ανά πάσα στιγμή οποιοδήποτε στοιχείο επιθυμεί. Για παράδειγμα, αν μετακομίσει σε άλλη πόλη θα έχει την δυνατότητα να ανανεώσει το πεδίο city. Από τη συγκεκριμένη ενότητα θα μπορεί να διαχειρίζεται και τα στοιχεία που αφορούν το λογαριασμό του (account). Έτσι μπορεί να αλλάξει το username και το password του όποτε θελήσει, εισάγοντας τα νέα τα οποία πληρούν βέβαια τις προδιαγραφές εγκυρότητας (μοναδικότητα username, αριθμός χαρακτήρων password κτλ). Συνεπώς θεωρείται απαραίτητη η ύπαρξη ενότητας διαχείρισης προφίλ.

5.2.3 Αναζήτηση χρηστών και δικτύωση

Εφόσον η εφαρμογή αποτελεί ένα κοινωνικό δίκτυο, ένα από τα σημαντικότερα κομμάτια της είναι η δικτύωση των χρηστών. Η διαδικασία της δικτύωσης μπορούμε να πούμε ότι χωρίζεται σε τρία επιμέρους σκέλη. Αυτά είναι:

- **Αναζήτηση Φίλων:** Η διαδικασία δικτύωσης ξεκινάει από την αναζήτηση του προσώπου με το οποίο επιθυμούμε να συνδεθούμε. Έτσι σε ειδική μπάρα για αναζήτηση φίλων, ο χρήστης εισάγει κάποια από τα στοιχεία του χρήστη που επιθυμεί να βρεί. Ταυτόχρονα, για λόγους ταχύτητας και ευκολίας θα πρέπει να εμφανίζονται οι προτάσεις χρηστών που συμφωνούν με αυτήν την εισαγωγή. Η αναζήτηση πρέπει να μπορεί επίσης να εξυπηρετεί και την γρήγορη εύρεση κάποιου από τους χρήστες με τον οποίο ήδη υπάρχει σύνδεση.
- **Αποστολή αιτήματος φιλίας:** Για να γίνει η σύνδεση δύο χρηστών απαραίτητη προϋπόθεση είναι να συμφωνούν και οι δύο. Εφόσον ο χρήστης που αναζητείται βρεθεί, τότε μέσα από τη σελίδα του προφίλ του, μπορεί να γίνει έκφραση της επιθυμίας για σύνδεση μαζί του. Η επιλογή για αποστολή αιτήματος φιλίας θα πρέπει να είναι ορατή μόνο στην περίπτωση που οι χρήστες δεν συνδέονται ήδη.
- **Αποδοχή/Απόρριψη αιτήματος φιλίας:** Από τη στιγμή που ένα αίτημα φιλίας σταλεί, ο χρήστης που το στέλνει αναμένει την θετική ή αρνητική απάντηση. Ο χρήστης που δέχεται το αίτημα, πρέπει να ενημερώνεται για αυτό από ειδική ενότητα για τα εισερχόμενα αιτήματα φιλίας και να επιλέγει αν επιθυμεί ή όχι αυτή τη σύνδεση.

5.2.4 Διαχείριση φίλων χρήστη

Ο αριθμός των φίλων που ένας χρήστης μπορεί να έχει πρέπει να είναι απεριόριστος. Συνεπώς υπάρχει ανάγκη για εύκολη αναζήτηση, προβολή και διαχείρισή τους. Για να γίνει αυτό η εφαρμογή θα επιτρέπει στο χρήστη να δει όλους τους φίλους με τους οποίους συνδέεται. Η φωτογραφία προφίλ και το ονοματεπώνυμο είναι αρκετά για να προσδιορίσουν την ταυτότητα του χρήστη φίλου. Ωστόσο, είναι δυνατόν να περιοριστούν τα αποτελέσματα και να μην εμφανίζονται όλοι οι φίλοι, αναγκάζοντας το χρήστη σε χρονοβόρα αναζήτηση. Πληκτρολογώντας στην μπάρα αναζήτησης τα αποτελέσματα ελαττώνονται, παρουσιάζοντας μόνο όσα ταιριάζουν. Εφόσον ο χρήστης που αναζητείται βρεθεί, είναι δυνατή η επιλογή προβολής του προφίλ του και η επιλογή αποσύνδεσης από αυτόν, διαγράφοντας τον από τη λίστα φίλων.

5.2.5 Ενημέρωση για τα γενόμενα από τους φίλους του χρήστη

Για να διασφαλιστεί η διατήρηση της επαφής μεταξύ των συνδεδεμένων χρηστών, και η παρακολούθηση των όποιων ενεργειών κάνει ένας χρήστης και αφορούν και τους φίλους του, σημαντική είναι η ενημέρωση του. Η ενημέρωση θα γίνεται από μία

ειδική ενότητα που θα αφορά τα νέα και θα τα οργανώνει σε μη αναγνωσμένα και αναγνωσμένα. Όταν κάτι νέο προκύπτει, ο χρήστης πρέπει να ενημερώνεται, για να το δει από ένα αναδυόμενο μήνυμα (notification), το οποίο είναι ο αριθμός των μη αναγνωσμένων. Γεγονότα, τα οποία θα αφορά αυτή η ενημέρωση, είναι τα εξής:

- **Αποδοχή αιτήματος φιλίας από άλλο χρήστη:** Κάποιος χρήστης στον οποίο έχουμε πρωτίτερα πραγματοποιήσει αίτημα φιλίας αποδέχεται την πρόταση μας. Έτσι ενημερωνόμαστε ότι πλέον ο χρήστης αυτός απάντησε θετικά και θεωρείτε πλέον φίλος μας.
- **Δημιουργία νέου ταξιδιού από φίλο.** Κάποιος φίλος μας, δημιούργησε ένα νέο ταξίδι το οποίο περιγράφει κάποια εμπειρία τα και ενημερωνόμαστε για αυτό ώστε να το δούμε αν το επιθυμούμε.
- **Επεξεργασία από φίλο σε κάποιο από τα ταξίδια του.** Όπως και στη δημιουργία ταξιδιού, ενημερωνόμαστε ότι κάποιος φίλος μας ανανέωσε κάποιο από τα ταξίδια του, ώστε να δούμε αν θέλουμε αυτή την αλλαγή.
- **Αλλαγή στο προφίλ φίλου.** Ενημερωνόμαστε ότι κάποιος φίλος μας άλλαξε τα δεδομένα από το προφίλ του.

5.2.6 Δημιουργία ταξιδιού – Γεγονότος

Αφού η εφαρμογή μας αφορά την ανασκόπηση γεγονότων και ταξιδιών, μεγάλη σημασία έχει η διαδικασία με την οποία ο χρήστης θα πραγματοποιήσει όσο ευκολότερα γίνεται την δημιουργία και την περιγραφή του πάνω στο χάρτη. Η υλοποίηση ενός ταξιδιού θα πραγματοποιείται σε δύο στάδια, αυτό της γενικής περιγραφής, και αυτό τη περιγραφής με τη χρήση του χάρτη.

- **Γενική περιγραφή του ταξιδιού–γεγονότος:** Σε αυτό το στάδιο θα δηλώνεται το όνομα του ταξιδιού, η χρονική περίοδος την οποία αφορά, και ένας σχολιασμός του.
- **Περιγραφή του γεγονότος πάνω στο χάρτη:** Το στάδιο αυτό είναι ποιο σύνθετο. Ο χρήστης θα πρέπει να μπορεί να δημιουργήσει ένα ταξίδι, δομικά στοιχεία του οποίου θα είναι οι σταθμοί. Οι σταθμοί θα περιγράφουν ένα μεμονωμένο γεγονός που έλαβε χώρα σε μία συγκεκριμένη τοποθεσία. Περιγραφικά του γεγονότος είναι εκτός από την τοποθεσία, το όνομα, ίσως μία φωτογραφία και μία σύντομη περιγραφή του τι συνέβη. Όλα αυτά τα στοιχεία ο χρήστης θα μπορεί άμεσα να τα δει και να τα επεξεργαστεί. Έτσι πρέπει να είναι ενσωματωμένα σε ένα παράθυρο πληροφοριών (balloon) που θα αναδύεται όταν ο χρήστης κάνει κλικ πάνω στο χάρτη και το σημείο αυτό. Σημαντικό για τη δόμηση ενός ταξιδιού είναι η αυτόματη τοποθέτηση όταν αυτό είναι εφικτό των φωτογραφιών στο χάρτη, ειδικά στην περίπτωση που έχουμε ανέβασμα πολλών εικόνων ταυτόχρονα. Το θέμα αυτό λύνεται με την εκμετάλλευση των δεδομένων gps που κάποιες φωτογραφίες μπορεί να φέρουν. Τέλος για να ολοκληρωθεί η δημιουργία ενός ταξιδιού πρέπει να ορίζεται η σειρά επισκεψιμότητας του κάθε σταθμού–σημείου.

5.2.7 Συνολική διαχείριση ταξιδιών χρήστη

Ο χρήστης της εφαρμογής μπορεί να υλοποιήσει όσα ταξίδια θέλει, τα οποία πρέπει να είναι σε θέση να διαχειρίζεται. Στα πλαίσια της διαχείρισης συμπεριλαμβάνεται η επιλογή της διαγραφής ενός ταξιδιού και η επιλογή της επεξεργασίας ενός ταξιδιού. Ο χρήστης θα πρέπει να μπορεί να βεβαιώνεται ότι το ταξίδι που θέλει να επιλέξει είναι το σωστό, έχοντας τη δυνατότητα να το προβάλει αν το επιθυμεί.

5.2.8 Επεξεργασία ταξιδιού

Στην περίπτωση που ο χρήστης επιλέξει να επεξεργαστεί ένα ταξίδι τότε πρέπει να του παρέχονται οι ίδιες δυνατότητες που έχει κατά τη δημιουργία. Έτσι εφόσον προβληθεί το σύνολο των σταθμών του ταξιδιού θα υπάρχει η δυνατότητα δημιουργίας νέων ή η αλλαγή σε κάποια από τα στοιχεία των παλιών.

5.2.9 Ανασκόπηση προσωπικών ταξιδιών και ταξιδιών φίλων

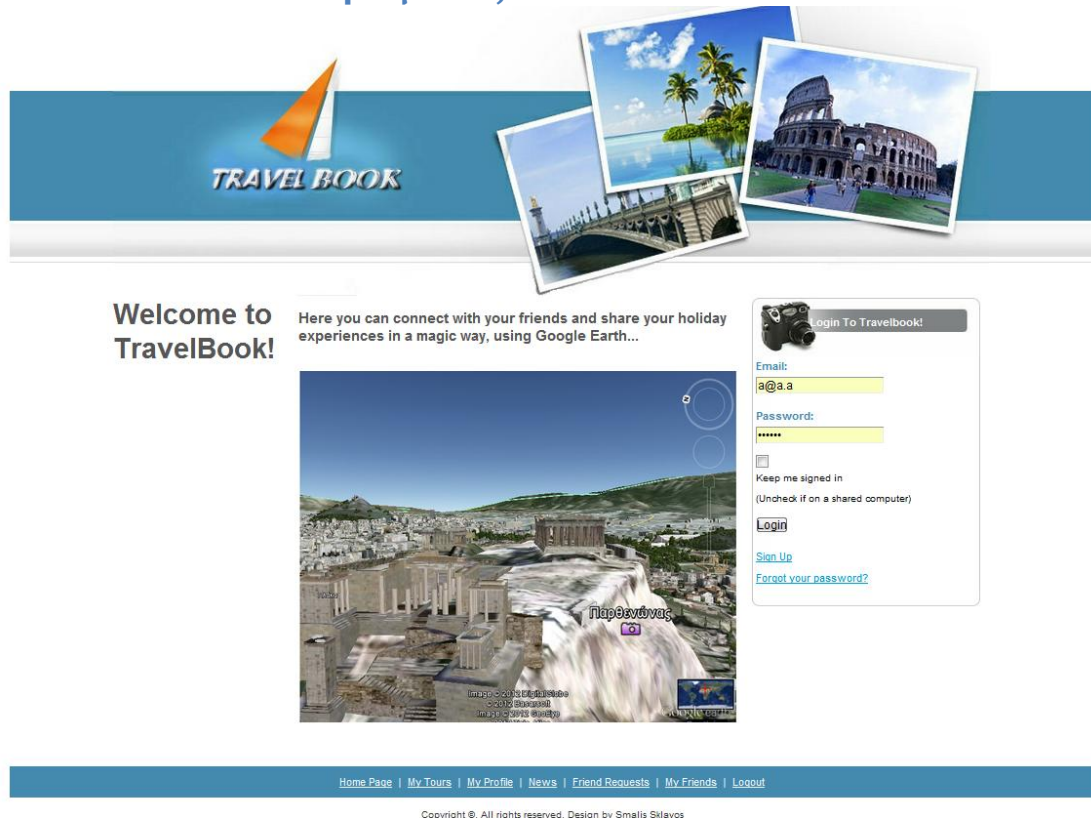
Όταν ένα ταξίδι δημιουργηθεί ο χρήστης της εφαρμογής θα μπορεί να κάνει ανασκόπηση με δύο τρόπους. Ο ένας είναι με **απλή προβολή** του συνόλου των σταθμών που συγκροτούν ένα ταξίδι πάνω στο χάρτη. Για να μπορεί να υπάρχει συνέχεια μεταξύ των σταθμών, πρέπει να είναι ενωμένοι μεταξύ του. Έτσι ο χρήστης θα μπορεί να έχει πλήρη εικόνα για την πορεία που ακολουθήθηκε, ενώ θα μπορεί να δει κάθε σταθμό μεμονωμένα, επιλέγοντας την προβολή, μέσω των balloons της πληροφορίας που φέρει. Το ίδιο συμβαίνει και στην περίπτωση που το ταξίδι ανήκει σε κάποιον φίλο. Σημαντικό είναι, η εφαρμογή να μπορεί να προσφέρει την ταυτόχρονη προβολή ταξιδιών, ούτως ώστε να είναι δυνατή η σύγκριση των πορειών που ακολουθήθηκαν. Ο έτερος τρόπος ανασκόπησης ταξιδιού είναι με την μορφή **live tour**. Αυτό σημαίνει την αυτόματη μετάβαση της κάμερας από σημείο σε σημείο, ακολουθώντας την πορεία, η οποία έχει οριστεί με βάσει την προτεραιότητα των σταθμών. Στην περίπτωση του live tour θα μπορεί να προβληθεί ένα live tour τη φορά, αλλά αυτό δεν αποκλείει να έχουν φορτωθεί και άλλα ταξίδια, με απλή προβολή, ώστε να είναι δυνατή η σύγκριση.

5.2.10 Αναζήτηση ταξιδιών

Η εφαρμογή θα παρέχει τη δυνατότητα αναζήτησης ενός ταξιδιού με κριτήριο το όνομά του. Για την επιβεβαίωση ότι το αποτέλεσμα είναι αυτό που αναζητείται πρέπει να φαίνεται και σε ποιόν χρήστη ανήκει, καθώς κάλλιστα δύο χρήστες μπορεί να έχουν επισπευτεί πχ την Αθήνα και να έχουν δώσει το ίδιο όνομα στο ταξίδι τους. Η δυνατότητα αναζήτησης ενός ταξιδιού είναι σημαντική για τον αυτονόητο λόγο, που είναι η διευκόλυνση του χρήστη να βρει το ταξίδι που θέλει να προβάλει. Ωστόσο μπορεί να προσφέρει επιπλέον πλεονεκτήματα όπως το να δει κάποιος ποιοι φίλοι του έχουν επισπευτεί το ίδιο μέρος με αυτόν, τότε κτλ.

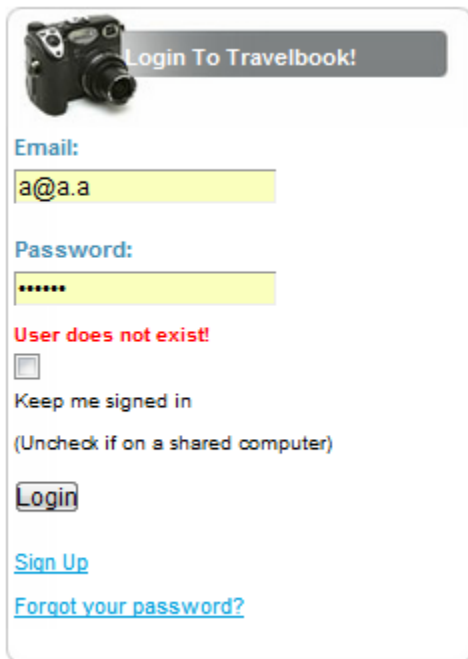
5.3 Γραφική διεπαφή

5.3.1 Σελίδα καλωσορίσματος



Εικόνα 34.Σελίδα Καλωσορίσματος

Η παραπάνω σελίδα είναι αυτή που εμφανίζεται όταν κάποιος θελήσει να επισκευτεί την εφαρμογή μας για πρώτη φορά. Είναι η σελίδα του καλωσορίσματος και έχει σκοπό να κινήσει το ενδιαφέρον του χρήστη ώστε να χρησιμοποιήσει την εφαρμογή. Όπως βλέπουμε, στην κορυφή υπάρχει το λογότυπο με φωτογραφίες κάποιων δημοφιλών ταξιδιωτικών προορισμών, οι οποίοι δίνουν από την αρχή στο χρήστη να καταλάβει τον ταξιδιωτικό χαρακτήρα της εφαρμογής. Στο Μεσαίο τμήμα της σελίδας και κοιτώντας από αριστερά προς τα δεξιά υπάρχουν: Το καλωσόρισμα στην εφαρμογή, μία ελάχιστη περιγραφή των δυνατοτήτων της εφαρμογής, στη μέση, και η φόρμα όπου δίνονται οι δυνατότητες στο χρήστη για να πραγματοποιήσει είσοδο (login), εγγραφή στο σύστημα, και ανάκτηση του κωδικού στην περίπτωση που τον έχει ξεχάσει. Εάν κάποιος χρήστης επιχειρήσει να κάνει εισαγωγή με στοιχεία τα οποία δεν υπάρχουν(π.χ λάθος username η password) τότε εμφανίζεται μήνυμα λάθους, όπως φαίνεται παρακάτω:



The image shows a login form for 'Travelbook'. At the top left is a camera icon. To its right is a dark grey button with the text 'Login To Travelbook!'. Below the icon, the form has two input fields: 'Email:' with the value 'a@a.a' and 'Password:' with masked characters '*****'. Below these fields is a red error message 'User does not exist!'. Underneath the error message is a checkbox that is currently unchecked, followed by the text 'Keep me signed in' and '(Uncheck if on a shared computer)'. At the bottom of the form is a 'Login' button, and below that are two links: 'Sign Up' and 'Forgot your password?'.

Login To Travelbook!

Email:
a@a.a

Password:

User does not exist!

☐

Keep me signed in
(Uncheck if on a shared computer)



Login

[Sign Up](#)

[Forgot your password?](#)

Εικόνα 35. Παράδειγμα λανθασμένης εισαγωγής στοιχείων

5.3.2 Σελίδα εγγραφής χρήστη στο σύστημα



Sign up

Registration Form

You should complete this form to sign up.
The fields with * are mandatory.

Profile Picture
Select your profile picture.

Upload Picture

Account Settings

Username (email) *

Password *

Password Confirmation *

Basic Information

Name *

Surname

Birthday
January / 1 / 1900

Sex
☒ Male
☐ Female

Country
United States

Current City

Hometown

About Me

Home Page | My Tours | My Profile | News | Friend Requests | My Friends | Logout

Copyright ©. All rights reserved. Design by Smalis Sklavos

Εικόνα 36.Σελίδα εγγραφής χρήστη

Η εικονιζόμενη σελίδα αποτελεί τη σελίδα εγγραφής χρήστη. Εμπεριέχει μία φόρμα (γαλάζιο χρώμα) με τα προσωπικά στοιχεία του χρήστη προς εγγραφή. Αυτά είναι χωρισμένα σε 3 ενότητες και κάθε ενότητα αποτελείται από κάποια επιμέρους στοιχεία. Οι ενότητες αυτές είναι:

- **Profile Picture:** Είναι η ενότητα φωτογραφίας χρήστη. Ο χρήστης καλείται να ανεβάσει μία φωτογραφία για το προφίλ του.
- **Account Settings:** Είναι η ενότητα στην οποία εμπεριέχονται το username και το password. Όπως λέει και το όνομά της συμπεριλαμβάνει τα στοιχεία του λογαριασμού χρήση τα οποία θα χρησιμοποιούνται για την ταυτοποίηση και την εισαγωγή στο σύστημα.
- **Basic Information:** Ο χρήστης καλείται να συμπληρώσει κάποιες βασικές πληροφορίες για αυτόν.


Για τα στοιχεία των παραπάνω ενότητων υποχρεωτικά είναι αυτά με τον κόκκινο αστερίσκο και είναι το username , το password, η επιβεβαίωση του password και το όνομα χρήστη.

Αξίζει να προσέξουμε την παρακάτω εικόνα για να καταλάβουμε πώς λειτουργεί το ανέβασμα εικόνων, η ειδοποίηση χρήστη για λάθος, και η καθοδήγηση χρήση.

You should complete this form to sign up.
The fields with * are mandatory.

Profile Picture
Select your profile picture.

Upload Picture



Upload (only .jpg)

Account Settings

Username (email) *
ssklavos@yahoo.gr **Already exists!**

Your username must be a valid email. Eg. ssklavos@yahoo.gr

Password *

Password Confirmation *

Basic Information

Name *

Surname

Εικόνα 37. Παράδειγμα συμπλήρωσης πεδίων.

Όπως βλέπουμε, όταν ολοκληρωθεί το upload μιας εικόνας, τότε αυτή με τη χρήση της τεχνικής Ajax που αναλύσαμε σε προηγούμενο κεφάλαιο εμφανίζεται στη σελίδα μας, χωρίς να χρειαστεί να φορτωθεί ξανά. Επίσης, όταν εστιάσουμε σε ένα πεδίο της φόρμας, τότε αυτό αυτόματα περιβάλλεται από ένα μπλε πλαίσιο, ενώ δεξιά από αυτό, σε ένα επίσης μπλε πλαίσιο εμφανίζονται οι οδηγίες για τη συμπλήρωση του πεδίου. Αυτό συμβαίνει παραπάνω για την περίπτωση του username όπου ο χρήστης ενημερώνεται για το πώς πρέπει να είναι αυτό. Ωστόσο, όσον αφορά το username, πρέπει να είναι μοναδικό. Για το λόγο αυτό κατά την εισαγωγή του, πάλι κάνοντας χρήση Ajax, ελέγχουμε συνεχώς την μοναδικότητά του. Αν κάποια εισαγωγή υπάρχει ήδη στη βάση δεδομένων μας, τότε ο χρήστης ενημερώνεται ότι υπάρχει σφάλμα με την κόκκινη επιγραφή “**already exists!**”. Μήνυμα λάθους εμφανίζεται επίσης όταν το password και η επιβεβαίωση δεν συμφωνούν.

5.3.3 Homepage – Σελίδα ανασκόπησης των ταξιδιών



Εικόνα 38. Αρχική σελίδα εφαρμογής, όπως φαίνεται αμέσως μετά το login,

Εφόσον ο χρήστης πραγματοποιήσει είσοδο στο σύστημα οδηγείται στην αρχική σελίδα, η οποία έχει την παραπάνω εικονιζόμενη μορφή. Τα στοιχεία από τα οποία δομείται φαίνονται καθαρότερα στην εικόνα που ακολουθεί.



Εικόνα 39. Αρχική σελίδα αφού κάνουμε zoom σε αυτή

Ξεκινώντας την ανάλυση της σελίδας από πάνω προς τα κάτω, θα αναφερθούμε πρώτα σε μία ομάδα στοιχείων τα οποία είναι κοινά για όλες τις σελίδες του αφορούν χρήστες, που έχουν κάνει login στο σύστημα. Τα στοιχεία αυτά αποτελούν κομμάτια του “header” και θα αναλυθούν μόνο σε αυτό το σημείο και όχι σε κάθε σελίδα. Αυτά είναι τα εξής:

Μπάρα αναζήτησης φίλων: Είναι η γκρι μπάρα που εικονίζεται πάνω αριστερά. Οι χρήστες μπορούν να εισάγουν το όνομα, το επώνυμο, το ονοματεπώνυμο ή το username-email του προσώπου που αναζητούν. Ταυτόχρονα με την εισαγωγή, και ενώ ο αριθμός των χαρακτήρων είναι μεγαλύτερος ή ίσος του 3 (για αποφυγή άσκοπων αναζητήσεων) το σύστημα ψάχνει στη βάση και μας προτείνει τα αποτελέσματα τα οποία ταιριάζουν. Για την ταυτόχρονη αναζήτηση αποτελεσμάτων χρησιμοποιούμε Ajax. Δηλαδή, πραγματοποιούμε ασύγχρονη αναζήτηση. Ένα

παράδειγμα του τρόπου εμφάνισης των αποτελεσμάτων εικονίζεται παρακάτω:



Εικόνα 40.Παράδειγμα αναζήτησης φίλων

Logo εφαρμογής: Το λογότυπο της εφαρμογής αποτελείται από το όνομα της μαζί με ένα καραβάκι το οποίο συνειρμικά παραπέμπει σε ταξίδια. Με κλικ πάνω αυτό οδηγούμαστε στην αρχική σελίδα.

Εικόνα προφίλ χρήστη: Η φωτογραφία του χρήστη βρίσκεται στο πάνω δεξί μέρος της οθόνης, μπροστά από τις φωτογραφίες των δημοφιλών προορισμών που αναφερθήκαμε προηγουμένως. Με κλικ σε αυτήν οδηγούμαστε στην σελίδα προσωπικού προφίλ (My Profile).

Μήνυμα ότι έχει πραγματοποιηθεί login του χρήστη: Κάτω από την φωτογραφία υπάρχει το μήνυμα με πράσινα γράμματα το οποίο ενημερώνει ότι ο εικονιζόμενος χρήστης, έχει εισέρθει (logged in: name surname).



Εικόνα 41. Εικόνα προφίλ και μνημα πραγματοποίησης login

Κεντρικό μενού της εφαρμογής: Το μενού αποτελείται από διάφορα εικονίδια σχετικά με την ενέργεια που το καθένα επιτελεί. Όταν το ποντίκι περνάει πάνω από αυτά εμφανίζεται το όνομα της λειτουργίας που εκτελεί. Τα στοιχεία του μενού, από αριστερά προς δεξιά είναι το Homepage, My tours, My profile, My News, My Messages, Friend Requests, My friends και Logout. Οι λειτουργίες και οι σελίδες που αντιπροσωπεύει το καθένα αναλύονται στη συνέχεια.

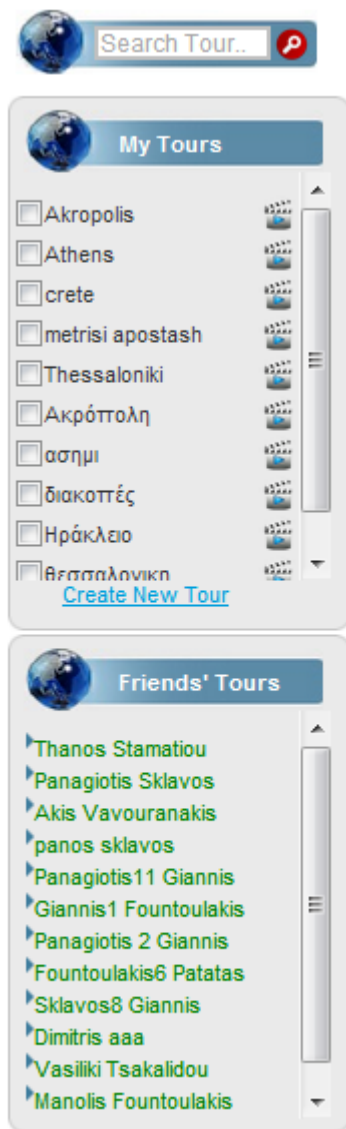


Εικόνα 42. Το μενού όταν το ποντίκι διέλθει πάνω από το εικονίδιο του my profile.



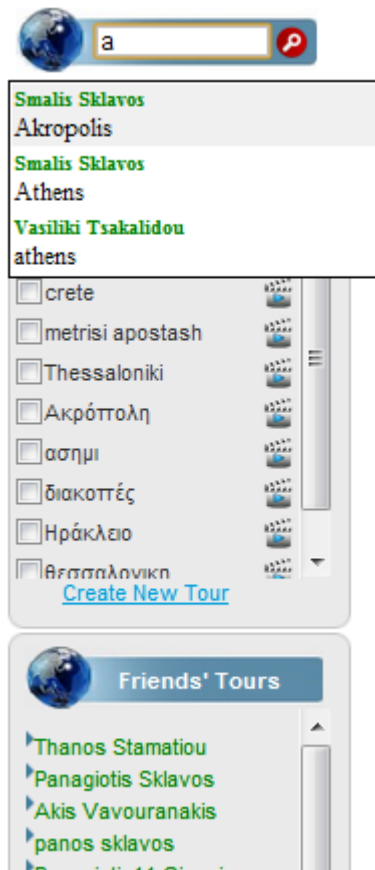
Εικόνα 43. Το μενού όταν υπάρχουν ειδοποιήσεις για νέα και αιτήματα φιλίας.

Συνεχίζοντας την ανάλυση, και φεύγοντας από το header το οποίο όπως είπαμε είναι κοινό για τις σελίδες της εφαρμογής, θα αναλύσουμε το αριστερό μέρος της αρχικής σελίδας. Το κομμάτι αυτό αποτελείται από τρία επιμέρους τμήματα τα οποία αποτελούν μενού ταξιδιών, και είναι τα εξής:



Εικόνα 44. Τα μόνον ταξιδιών στο αριστερό τμήμα της σελίδας.

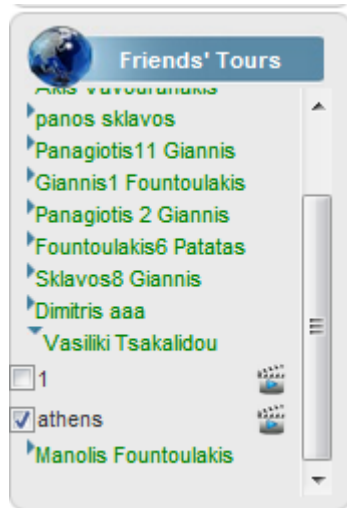
Μενού αποτελεσμάτων αναζήτησης ταξιδιών: Το μενού αυτό δεν είναι προσχεδιασμένο, αλλά δημιουργείται δυναμικά ανάλογα με την είσοδο που πληκτρολογεί ο χρήστης στην μπάρα αναζήτησης. Τα αποτελέσματα εμφανίζονται σε drop down list κάτω από αυτήν. Κατά την αναζήτηση, το σύστημα ψάχνει ανάμεσα στα προσωπικά ταξίδια του κάθε χρήστη και τα ταξίδια των φίλων του. Τα αποτελέσματα προβάλλονται ασύγχρονα με τη χρήση Ajax, όπως ακριβώς στην περίπτωση αναζήτησης φίλων. Ο χρήστης επιλέγει αν θέλει να δεί κάποιο από αυτά κάνοντας κλικ πάνω του, με αποτέλεσμα αυτό να αυτό προβάλλεται πάνω στο χάρτη. Παρακάτω εικονίζεται ένα παράδειγμα αναζήτησης, όπου όπως φαίνεται, το μενού σχεδιάζεται από το όνομα του χρήστη που ανήκει το ταξίδι και από το όνομα του ταξιδιού.



Εικόνα 45. Παράδειγμα αναζήτησης ταξιδιού

Μενού προσωπικών ταξιδιών (My Tours): Στο μενού αυτό εικονίζονται όλα τα προσωπικά ταξίδια του χρήστη. Για κάθε ταξίδι εμφανίζεται ένα checkbox πριν το όνομά του, και ένα εικονίδιο αναπαραγωγής video. Η σημασία αυτών των δύο στοιχείων είναι μεγάλη, καθώς επιτελούν τις βασικές λειτουργίες της εφαρμογής μας, δηλαδή της απλής ανασκόπησης και της ανασκόπησης με μορφή live tour video. Ο χρήστης μπορεί να επιλέξει όσα ταξίδια θέλει και να τα αναπαραστήσει πάνω στο χάρτη ταυτόχρονα. Αυτό υλοποιείται με τη βοήθεια των checkboxes η επιλογή των οποίων σημαίνει προβολή του αντίστοιχου ταξιδιού στο χάρτη. Για την αναπαραγωγή live tour ωστόσο, μπορεί να αναπαραχθεί μόνο ένα τη φορά. Στο μενού αυτό συμπεριλαμβάνεται και η επιλογή στο κάτω μέρος, για δημιουργία νέου ταξιδιού.

Μενού ταξιδιών φίλων (Friends' Tours): Το μενού αυτό μοιάζει με εκείνο των προσωπικών ταξιδιών, με τη διαφορά ότι τα ταξίδια είναι οργανωμένα σε ομάδες ανάλογα με το φίλο που ανήκουν. Το κάθε ταξίδι εμφανίζεται και εδώ με τη μορφή checkbox- όνομα- κουμπί αναπαραγωγής.



Εικόνα 46. Παράδειγμα απεικόνισης των ταξιδιών κάποιου φίλου από το μενού ταξιδιών φίλων.

Συνεχίζοντας την ανάλυση της αρχικής σελίδας οδηγούμαστε στην ενεργό περιοχή όπου υπάρχει ο χάρτης του Google Earth και ένα μενού με τις διάφορες επιλογές που αφορούν τον τρόπο προβολής των δεδομένων πάνω στο χάρτη.

Χάρτης Google Earth: Εδώ γίνεται όπως είπαμε η ανασκόπηση ταξιδιών με τον τρόπο επιλογής του χρήστη.

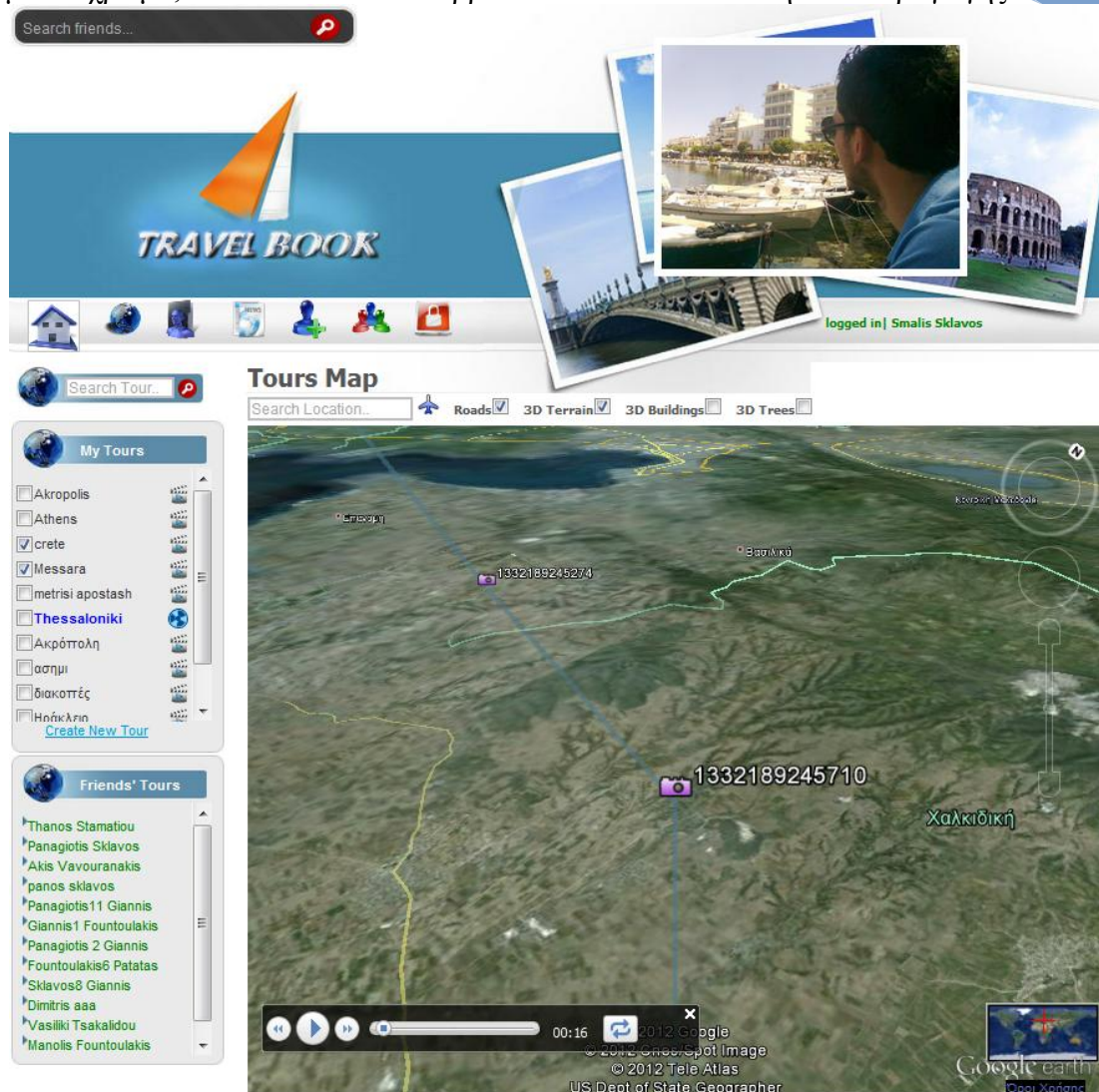
- **Με την απλή ανασκόπηση:** Μπορούμε να προβάλλουμε όπως είπαμε όσα ταξίδια θέλουμε ταυτόχρονα. Τα σημεία ενώνονται με διαφορετικό χρώμα για κάθε ταξίδι ώστε να ξεχωρίζουν. Ένα παράδειγμα προβολής δύο ταξιδιών ταυτόχρονα ακολουθεί παρακάτω για τα ταξίδια «Κρήτη» και «Messara», των οποίων έχουμε επιλέξει τα αντίστοιχα checkboxes. Εφόσον προβληθούν ο χρήστης μπορεί να αλληλεπιδράσει με το χάρτη και να δει λεπτομέρειες για όποιο από τα σημεία επιθυμεί, να κάνει zoom κτλ.



Εικόνα 47 Παράδειγμα ταυτόχρονης απεικόνισης δύο ταξιδιών με απλή ανασκόπηση. Με λευκό χρώμα εικονίζεται το ταξίδι Κρήτη ενώ με κόκκινο το ταξίδι Messara.

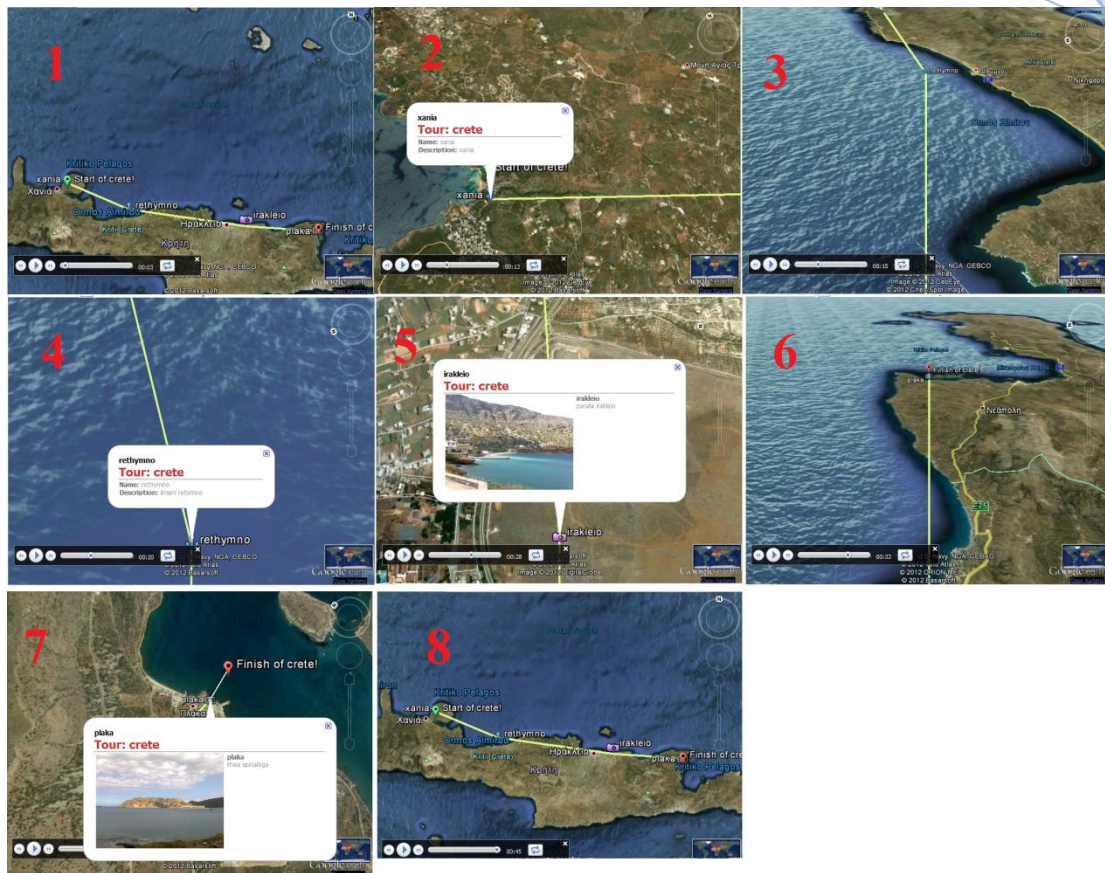
- **Με Ανασκόπηση της μορφής live Tour:** Στην περίπτωση αυτή, το μόνο που έχει να κάνει ένας χρήστης είναι να επιλέξει το εικονίδιο αναπαραγωγής για τι ταξίδι που επιθυμεί. Στη συνέχεια το σύστημα πραγματοποιεί πλοήγηση πάνω στο χάρτη από σημείο σε σημείο μόνο του. Ένα παράδειγμα τέτοιας αναπαραγωγής εικονίζεται παρακάτω. Το στιγμιότυπο που βλέπουμε είναι από την αυτόματη μετάβαση από ένα placemark σε κάποιο άλλο. Το επιλεγμένο ταξίδι είναι το «Thessaloniki» του οποίου το εικονίδιο και το

μπλε χρώμα, υποδυκνύει ότι βρίσκεται σε κατάσταση αναπαραγωγής



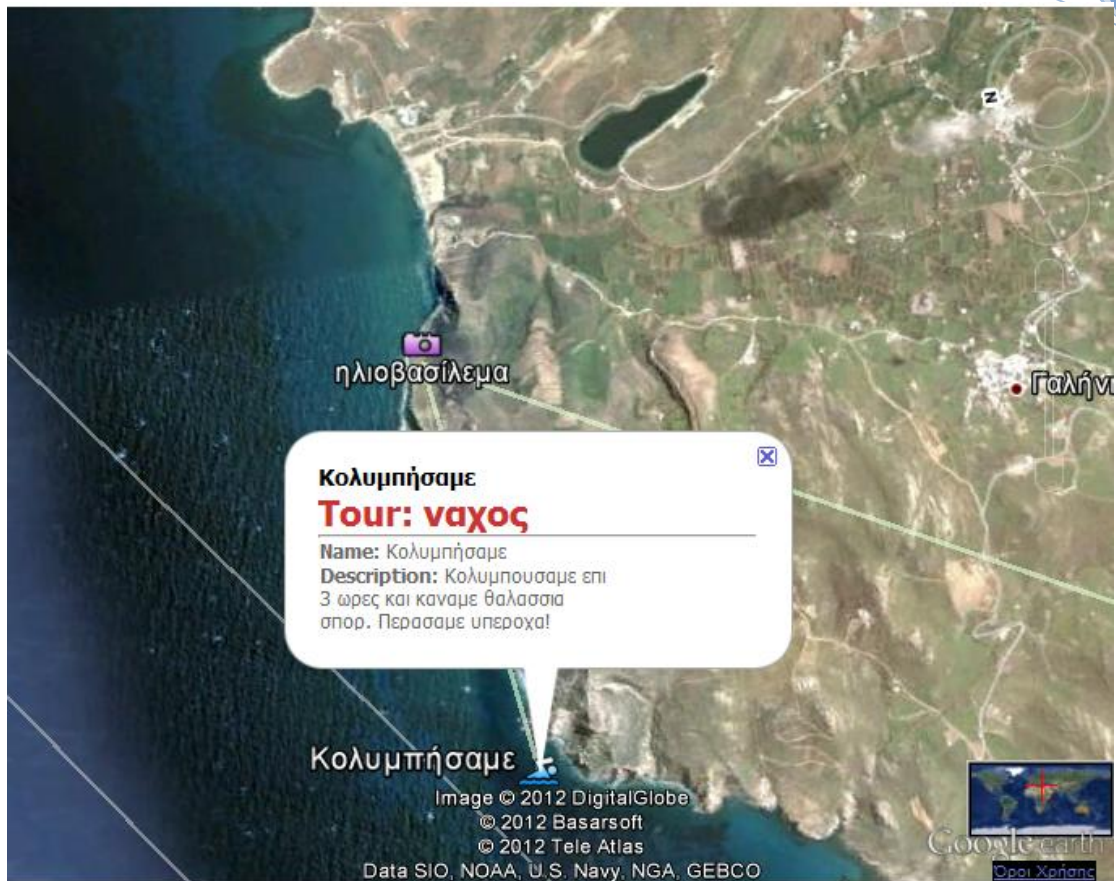
Εικόνα 48. Παράδειγμα αναπαραγωγής live tour.

Καρέ καρέ η αναπαραγωγή ενός άλλου ταξιδιού, του «Κρήτη» εικονίζεται παρακάτω.



Εικόνα 49. Καρέ καρέ αναπαραγωγή ταξιδιού Κρήτη.

Όσον αφορά τον τρόπο με τον οποίο βλέπουμε την πληροφορία που φέρουν τα placemarks, και στις 2 περιπτώσεις ανασκόπησης έχουν την ίδια μορφή. Παρακάτω βλέπουμε δύο παραδείγματα απεικόνισης για απλά placemarks και για photo placemarks, για την περίπτωση της απλής ανασκόπησης. Για την περίπτωση του live tour αυτό που μπορούμε να το διακρίνουμε στις εικόνες 2 και 5 του της παραπάνω εικόνας.



Εικόνα 50. Παράδειγμα παράθυρου πληροφοριών για απλό placemark



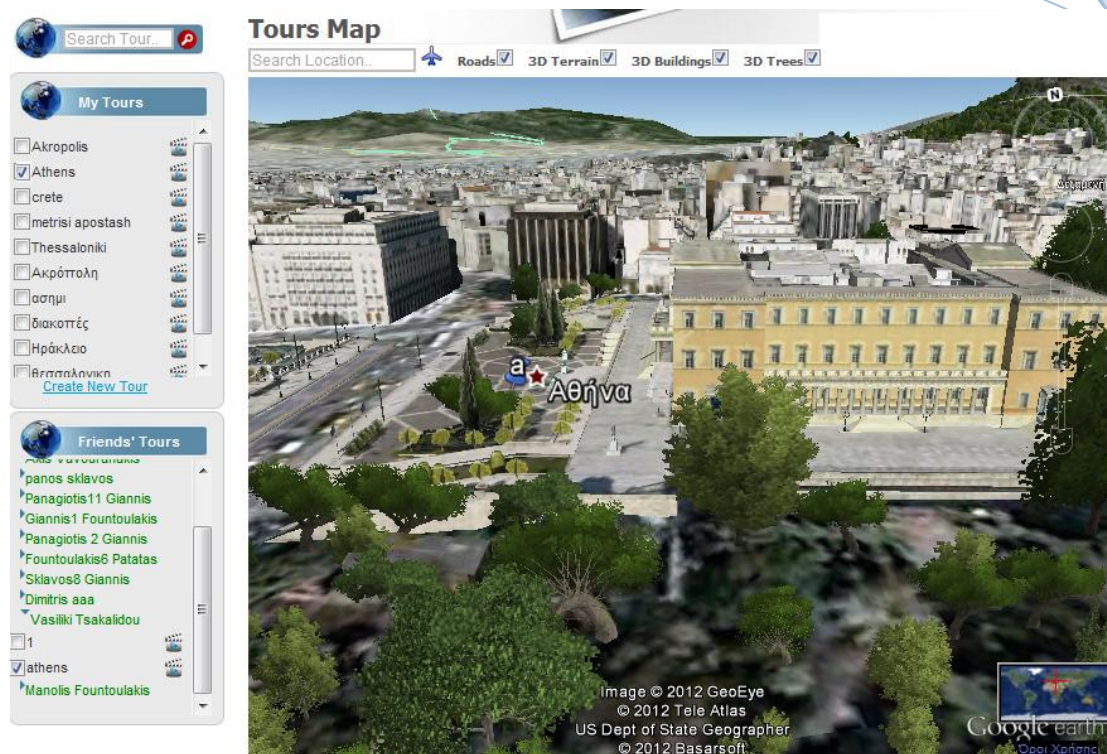
Εικόνα 51. Παράδειγμα παράθυρου πληροφοριών για Photo placemark

Μενού επιλογών χάρτη: Από το συγκεκριμένο μενού μπορούμε να επιλέξουμε αν θέλουμε να μεταβούμε σε κάποιο συγκεκριμένο σημείο με τη διαδικασία του geocoding. Επίσης μπορούμε να επιλέξουμε αν θέλουμε να είναι ορατά τα ονόματα δρόμων, το δισδιάστατο γεωγραφικό ανάγλυφο, τα τρισδιάστατα κτήρια, ή τα τρισδιάστατα δέντρα. Αυτό γίνεται τσεκάροντας το checkbox που αντιστοιχεί στην κάθε επιλογή.



Εικόνα 51. Μενού επιλογών χάρτη.

Ένα παράδειγμα απεικόνισης με ενεργοποιημένες όλες τις δυνατότητες που μας παρέχει το μενού αυτό, εικονίζεται παρακάτω:



Εικόνα 52. Παράδειγμα 3D απεικονίσεων

5.3.4 My Tours-Σελίδα διαχείρισης των ταξιδιών του χρήστη

Search friends...

TRAVEL BOOK

logged in! Smalis Sklavos

My Tours

Name	Begin Date	End Date		
<input type="checkbox"/> 0	2010-01-01	2010-01-01		
<input type="checkbox"/> 11	2010-01-01	2010-01-01		
<input type="checkbox"/> 1221212	2010-01-01	2010-01-01		
<input type="checkbox"/> ass	2010-01-01	2010-01-01		
<input type="checkbox"/> Athens	2010-01-01	2010-01-01		
<input type="checkbox"/> aste	2010-01-01	2010-01-01		
<input type="checkbox"/> Ikaria	2010-01-01	2010-01-01		
<input type="checkbox"/> Kith	2010-01-01	2010-01-01		
<input type="checkbox"/> Messara	2010-01-01	2010-01-01		
<input type="checkbox"/> metrisi apostash	2010-01-01	2010-01-01		
<input type="checkbox"/> qwqw	2010-01-01	2010-01-01		
<input type="checkbox"/> Thessaloniki	2012-03-10	2012-09-15		
<input type="checkbox"/> wawaw sasas	2010-01-01	2010-01-01		
<input type="checkbox"/> Ακρόπολη	2010-01-01	2010-01-01		
<input type="checkbox"/> ασημι	2010-01-01	2010-01-01		
<input type="checkbox"/> Ηράκλειο	2010-01-01	2010-01-01		
<input type="checkbox"/> Θεσσαλονικη	2010-01-01	2010-01-01		
<input type="checkbox"/> ναχος	2010-01-01	2010-01-01		

Create New Tour!

Page: 1

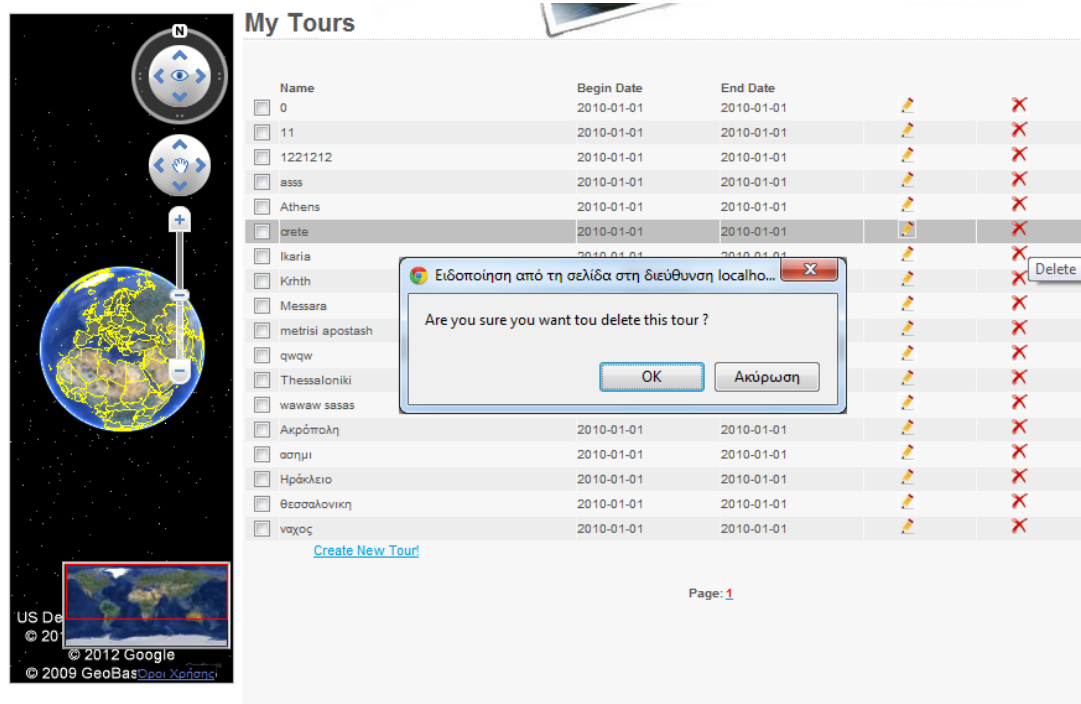
Εικόνα 53. Σελίδα My Tours

Επιλέγοντας από το κεντρικό οριζόντιο μενού το εικονίδιο του My Tours οδηγούμαστε στην σελίδα που εικονίζεται παραπάνω. Αυτή είναι η σελίδα διαχείρισης των ταξιδιών του χρήστη. Παρατηρώντας την δόμηση της σελίδας βλέπουμε ότι το επάνω μέρος παραμένει ίδιο με την προηγούμενη σελίδα, κάτι το οποίο είναι λογικό εφόσον όπως είπαμε αποτελεί το κοινό τμήμα για όλες τις σελίδες, που αφορούν εγγεγραμμένους χρήστες. Μοναδική διαφορά από την προηγούμενη είναι ότι το εικονίδιο του my tour είναι αυτό που έχει λευκό φόντο, και έχει σημειωθεί ως επιλεγμένο. Συνεχίζοντας λοιπόν και κοιτώντας παρακάτω στη σελίδα, παρατηρούμε την ύπαρξη των εξής στοιχείων:

Χάρτης Google Earth: Αποτελεί βοηθητικό χάρτη ,για την απλή προβολή ταξιδιών τα οποία ο χρήστης επιλέγει με τη βοήθεια πάλι των Checkboxes. Ο χάρτης αυτός βοηθάει το χρήστη να θυμηθεί κάποιο ταξίδι που πρόκειται να επεξεργαστεί στην περίπτωση που δεν είναι σίγουρος, χωρίς να χρειάζεται να ανατρέξει στην αρχική σελίδα.

Ενεργός περιοχή με πίνακα ταξιδιών: Όπως παρατηρούμε για κάθε ταξίδι παρέχονται τα βασικά στοιχεία του, που είναι το όνομα (name) και οι ημερομηνίες εκκίνησης και λήξης (begin Date, end date). Μπροστά από κάθε ταξίδι υπάρχουν τα

checkboxes προβολής στο χάρτη. Στις τελευταίες δύο στήλες του πίνακα ταξιδιών υπάρχουν λειτουργίες διαχείρισης. Αυτές είναι το Edit (επεξεργασία) και το Delete (Διαγραφή) για τα οποία υπάρχουν τα αντίστοιχα εικονίδια. Επιλέγοντας ο χρήστης επεξεργασία οδηγείται στην σελίδα επεξεργασίας του ταξιδιού, ενώ επιλέγοντας διαγραφή μπορεί άμεσα να διαγράψει το ταξίδι αυτό, μια ενέργεια που μπορεί να πραγματοποιηθεί και μέσα από τη σελίδα επεξεργασίας ταξιδιού. Εφόσον ο χρήστης επιλέξει διαγραφή, για να ολοκληρώσει την ενέργειά του θα πρέπει πρώτα να επιβεβαιώσει την επιθυμία του, απαντώντας θετικά στο confirmation box που αναδύεται με τη βοήθεια της JavaScript.



Εικόνα 54. Confirmation box. Ζητάει από το χρήστη να επιβεβαιώσει την επιθυμία του για διαγραφή του ταξιδιού

Επιλογή δημιουργίας νέου ταξιδιού (create New Tour): Ο χρήστης στα πλαίσια της διαχείρισης των ταξιδιών του, έχει την δυνατότητα να δημιουργήσει ένα νέο ταξίδι, οδηγούμενος στη σελίδα δημιουργίας ταξιδιού.

Ένδειξη αριθμού σελίδων: Για να είναι πρακτική η χρήση του χάρτη, και να αποτρέπονται περιττά scroll που ενδέχεται να κουράζουν το χρήστη, υπάρχει κατανομή των ταξιδιών σε σελίδες. Έτσι ανά σελίδα εμφανίζονται μόνο είκοσι ταξίδια, και ο χρήστης μπορεί να επιλέξει σε ποια σελίδα επιθυμεί να μεταβεί. Η τρέχουσα σελίδα σημαίνεται με κόκκινο χρώμα για να ξεχωρίζει από τις υπόλοιπες που έχουν το γαλάζιο χαρακτηριστικό χρώμα των συνδέσμων.

5.3.5 Create New Tour -Δημιουργία νέου ταξιδιού

Η δημιουργία ενός νέου ταξιδιού ολοκληρώνεται σε δύο βήματα. Σε πρώτη φάση, αφού επιλέξουμε τη δημιουργία ταξιδιού, οδηγούμαστε στη σελίδα συμπλήρωσης των βασικών στοιχείων του ταξιδιού. Τα στοιχεία αυτά είναι το όνομα, οι

ημερομηνίες έναρξης και ολοκλήρωσης, καθώς και μία σύντομη περιγραφή του ταξιδιού. Η σελίδα αυτή είναι η ακόλουθη:

Search friends...

TRAVEL BOOK

logged in | Smalis Sklavos

Create Tour

Basic Tour Info Form

You should complete this form to create a new tour.
The fields with * are mandatory.

Tour Info

Tour Name *
athens
The max number of characters is 17.
You already have a tour with the same name.

Begin Date *
January / 1 / 2010

End Date *
January / 1 / 2010

Description

[Reset](#) [Save](#)

[Home Page](#) | [My Tours](#) | [My Profile](#) | [News](#) | [Friend Requests](#) | [My Friends](#) | [Logout](#)

Copyright ©. All rights reserved. Design by Smalis Sklavos

Εικόνα 55 Σελίδα συμπλήρωσης βασικών στοιχείων κατά τη δημιουργία ταξιδιού.

Όπως παρατηρούμε η ενεργός περιοχή της σελίδας εμπεριέχει μία φόρμα συμπλήρωσης με το χαρακτηριστικό για τις φόρμες της εφαρμογής γαλάζιο χρώμα. Όταν εστιάζουμε σε κάποιο πεδίο, αυτό περιβάλλεται από μπλε πλαίσιο, ενώ στο δεξί μέρος εμφανίζονται οι οδηγίες συμπλήρωσης, όπως στο παραπάνω παράδειγμα. Τέλος, κάτι το οποίο φαίνεται και παραπάνω, το όνομα του ταξιδιού πρέπει να είναι μοναδικό για κάθε χρήστη. Δηλαδή να μην έχει δημιουργήσει ο χρήστης ταξίδι με ίδιο όνομα. Αυτό ελέγχεται ασύγχρονα με Ajax, κατά την εισαγωγή ονόματος και σε περίπτωση σφάλματος ενημερωνόμαστε με το κόκκινο μήνυμα λάθους που βλέπουμε.

Εφόσον η συμπλήρωση ολοκληρωθεί σωστά ο χρήστης μπορεί να επιλέξει save για να οδηγηθεί στην σελίδα δημιουργίας των σταθμών του ταξιδιού. Η σελίδα αυτή εικονίζεται παρακάτω.

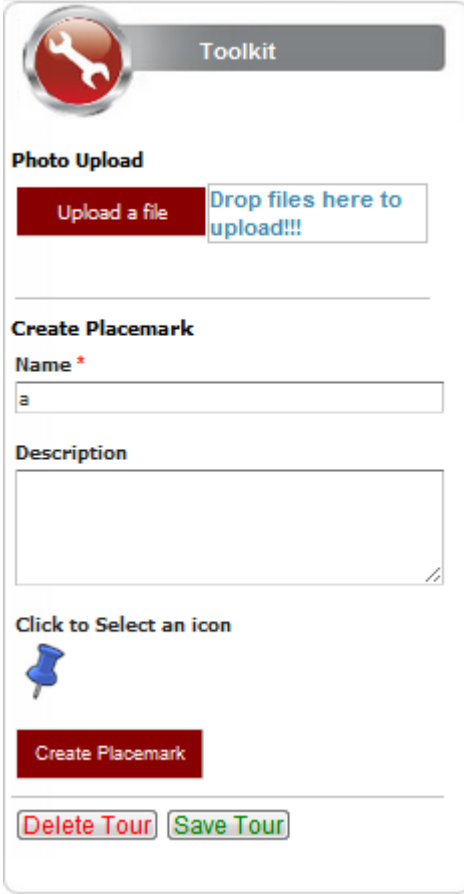


Εικόνα 56.Σελίδα δημιουργίας σταθμών ταξιδιού

Όπως παρατηρούμε το κεντρικό τμήμα της σελίδας αποτελείται από τα εξής τμήματα:

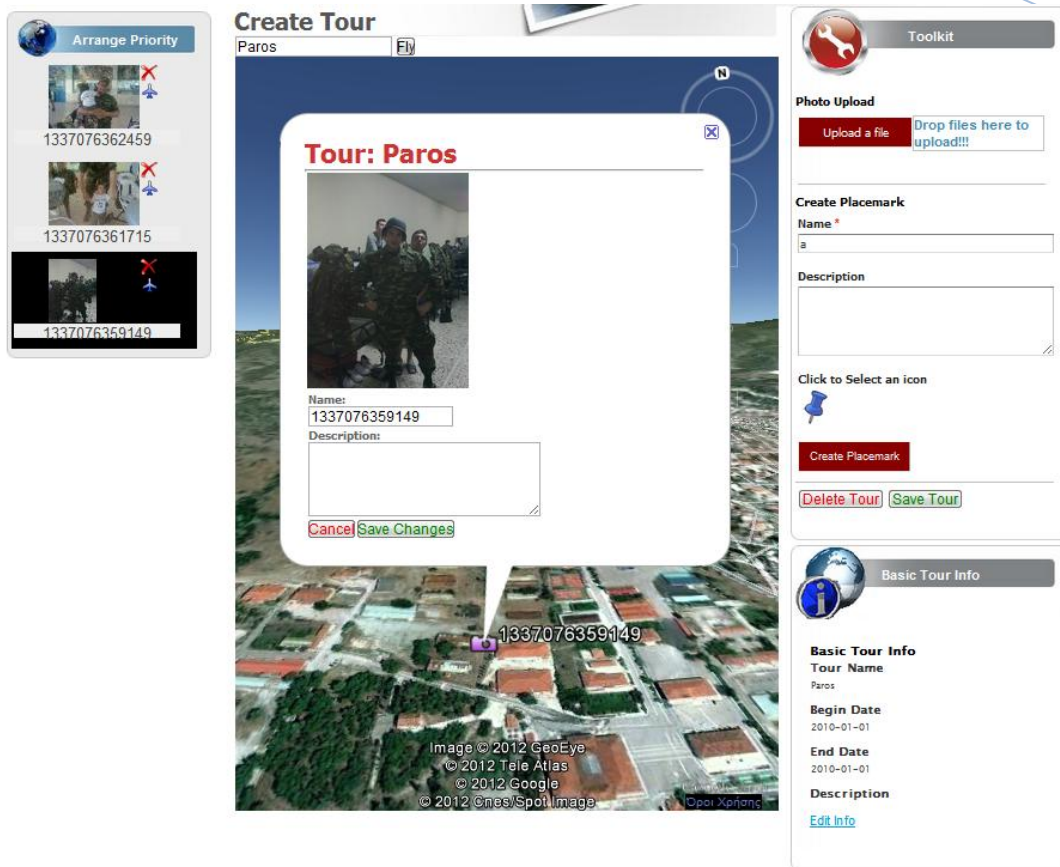
Χάρτης: Κατά την ολοκλήρωση του φορτώματος (on load) της παραπάνω σελίδας πραγματοποιείται η διαδικασία γεωεντοπισμού της θέσης που έχουμε ορίσει ως όνομα του ταξιδιού. Σε περίπτωση που βρεθεί η θέση αυτή, ο χάρτης μας οδηγεί αυτόματα εκεί, διευκολύνοντας το χρήστη να αποφύγει το περιττό ψάξιμο. Στο πάνω μέρος του χάρτη υπάρχει και μία μπάρα γεωεντοπισμού σε περίπτωση που θέλουμε να αναζητήσουμε κάποια άλλη περιοχή.

Toolkit: Στο τμήμα αυτό της σελίδας που βρίσκεται στα δεξιά υπάρχουν τα απαραίτητα εργαλεία για τη δημιουργία των σταθμών του ταξιδιού. Όπως αναφέραμε πριν, οι σταθμοί είναι δύο ειδών, photo placemarks και τα απλά placemarks. Έτσι το toolkit είναι χωρισμένο σε δύο ενότητες. Αυτές είναι το photo upload και το create placemark.



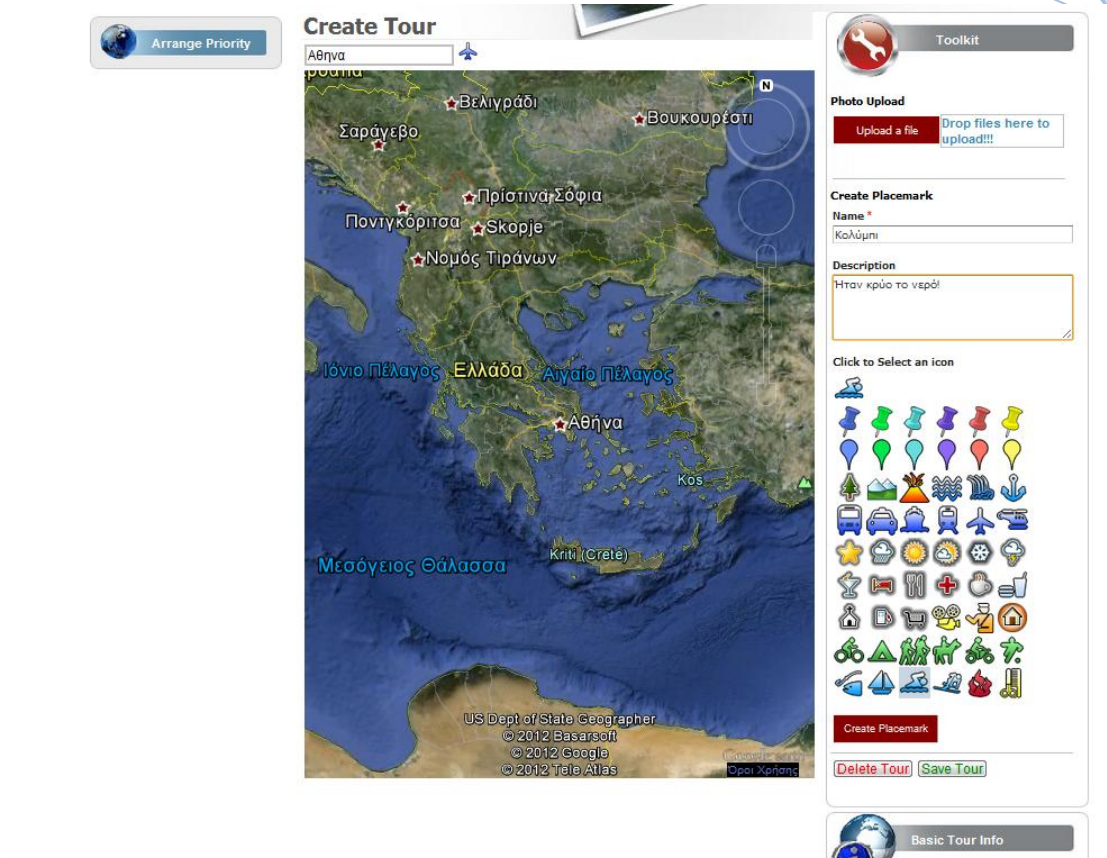
Εικόνα 57. Toolkit για τη δημιουργία ταξιδιού.

Ξεκινώντας με το **photo upload** παρατηρούμε ότι αποτελείται από ένα κόκκινο κουμπί ,κάνοντας κλικ στο οποίο, μπορούμε να ανεβάσουμε όσες φωτογραφίες θέλουμε. Φωτογραφίες μπορούμε να ανεβάσουμε και σέρνοντας τις (drag) ,στην περιοχή δίπλα από το κουμπί (drop files here to upload). Κατά το ανέβασμα των φωτογραφιών εμφανίζεται η λίστα τους κάτω από το κουμπί του upload με το ποσοστό φόρτωσης της καθεμίας. Για καθεμία που ολοκληρώνεται το ανέβασμά της σβήνεται αυτόματα από τη λίστα το όνομα της. Όταν ολοκληρωθεί το photo upload τότε όσες είχαν δεδομένα gps τοποθετούνται στο χάρτη στο σημείο ακριβώς που τραβήχτηκαν, ενώ όσες δεν είχαν τοποθετούνται στο σημείο εστίασης, εκείνη τη στιγμή ,του χάρτη. Με την ολοκλήρωση του upload μικρογραφίες των φωτογραφιών εμφανίζονται και στο αριστερό τμήμα της σελίδας, απ όπου μπορούμε να ορίσουμε τη σειρά επισκεψιμότητας (προτεραιότητα) στα σημεία των εικόνων. Αφού ανεβάσουμε μία εικόνα μπορούμε να τη δούμε πάνω στο χάρτη όπως φαίνεται στην παρακάτω εικόνα. Από το παράθυρο πληροφοριών και τη φόρμα συμπλήρωσης που εμπεριέχει μπορούμε αν θέλουμε να αλλάξουμε το όνομα της εικόνας, η να δώσουμε μία περιγραφή. Πατώντας save η cancel, σώζουμε η ακυρώνουμε τις αλλαγές της φόρμας αντίστοιχα.



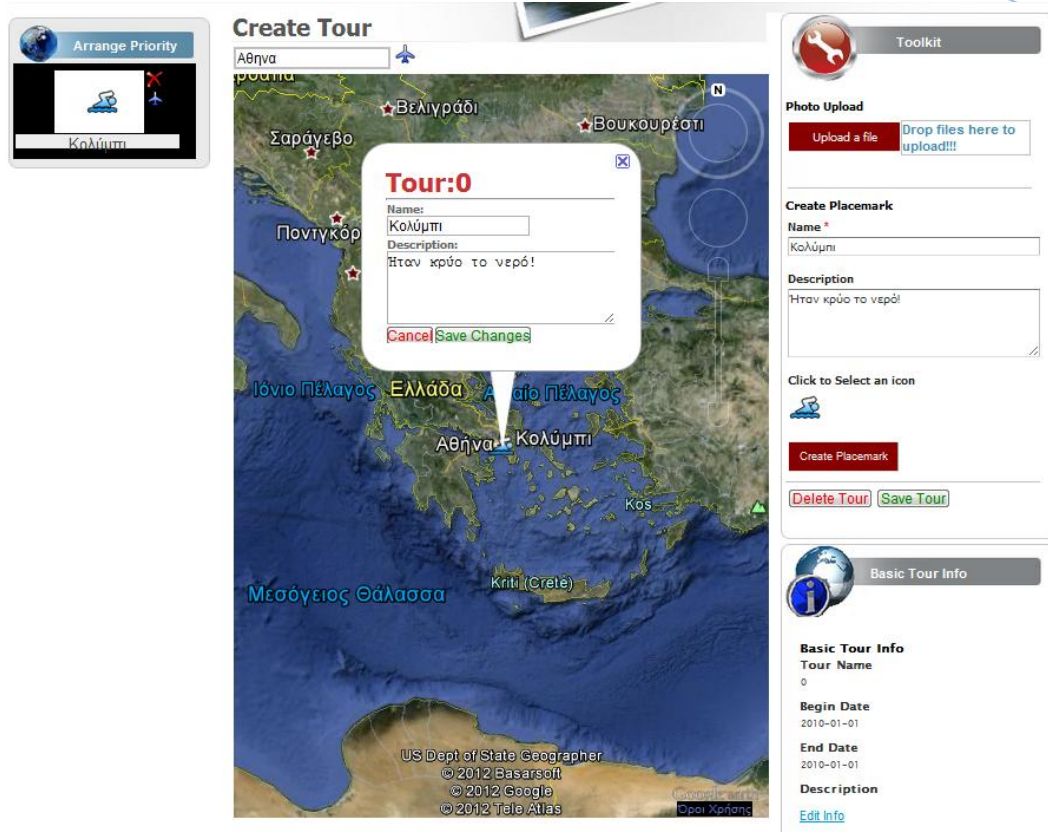
Εικόνα 58. Παράδειγμα photo upload.

Συνεχίζοντας με τη **δημιουργία placemark**, παρατηρούμε ότι αποτελείται από μία φόρμα συμπλήρωσης, με πεδία το όνομα του placemark, την περιγραφή του, και το εικονίδιο, στο οποίο κάνοντας κλίκ, ανοίγει μία λίστα επιλογής εικονιδίου. Η εικόνα του toolkit με τη λίστα αυτή ανοιγμένη κατά τη δημιουργία ενός placemark είναι η παρακάτω:



Εικόνα 59. Παράδειγμα δημιουργίας placemark

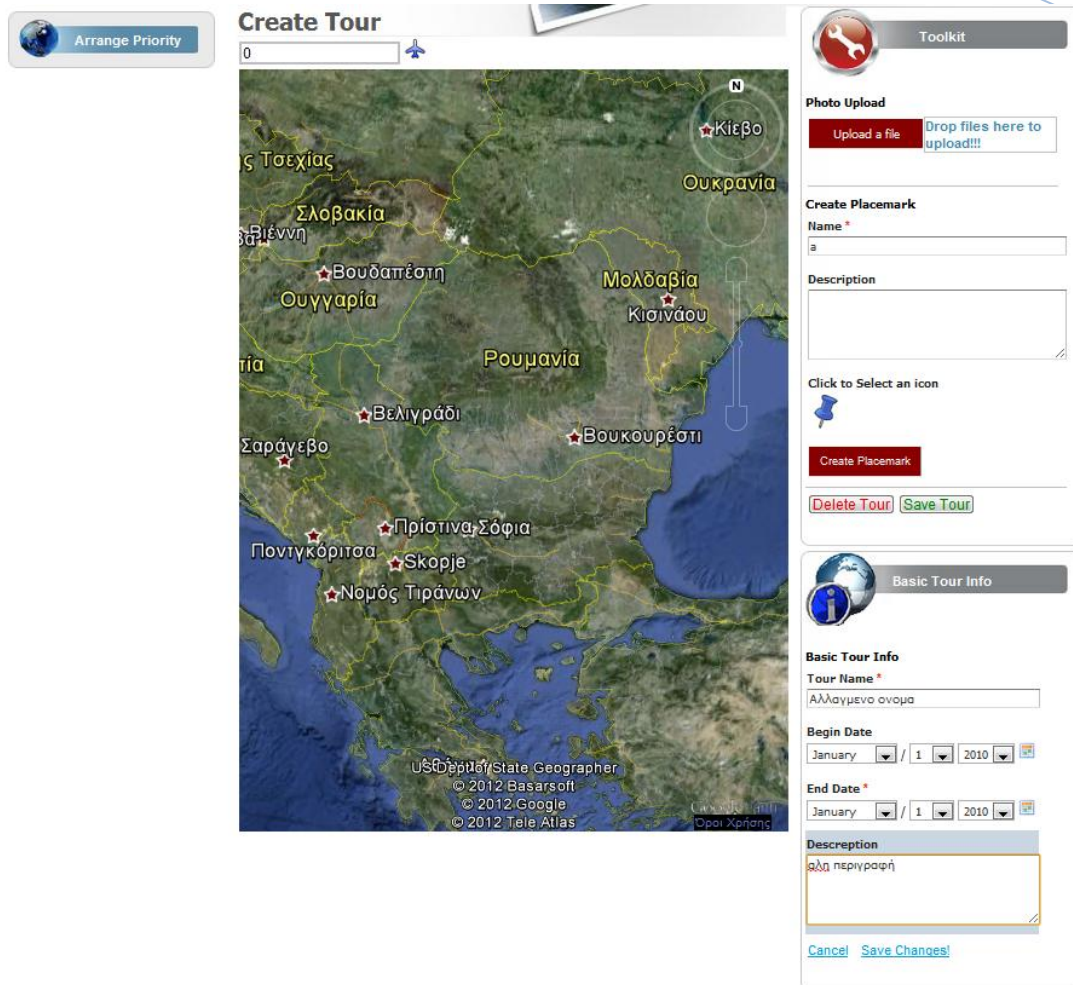
Το αποτέλεσμα από τη παραπάνω διαδικασία είναι η δημιουργία του placemark στο σημείο εστίασης του χάρτη. Όπως και στην περίπτωση των photo placemark μπορούμε να επεξεργαστούμε την πληροφορία τους. Αυτά εικονίζονται και στη παρακάτω εικόνα.



Εικόνα 60. Αποτέλεσμα δημιουργίας placemark

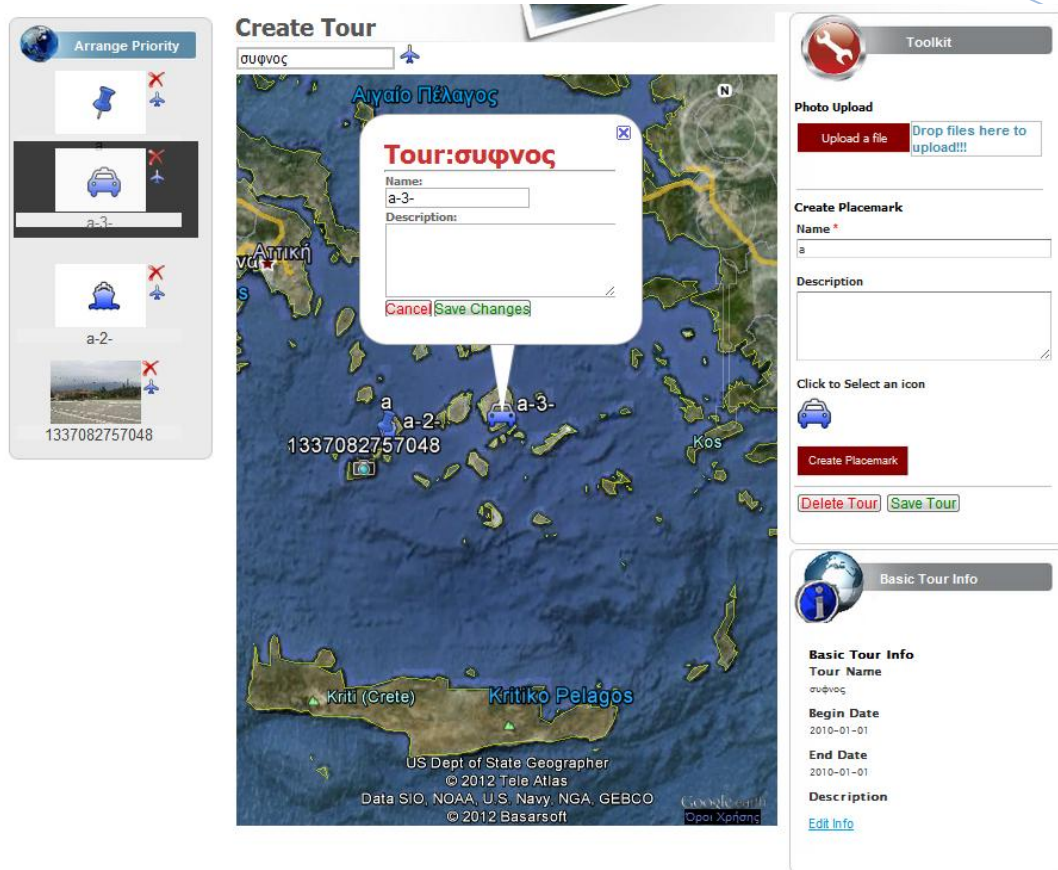
Save-Delete: Όπως μπορούμε να δούμε και στις εικόνες που έχουν προηγηθεί, στο toolkit για τη δημιουργία ταξιδιών υπάρχουν και τα κουμπιά με τα οποία μπορούμε να ολοκληρώσουμε το δημιούργημά μας, αποθηκεύοντας το, ή να το διαγράψουμε.

Basic Tour Info: Κάτω από το toolkit υπάρχουν τα βασικά στοιχεία του ταξιδιού τα οποία είχαμε συμπληρώσει στο πρώτο στάδιο δημιουργίας tour. Τα στοιχεία αυτά μπορούμε να τα επεξεργαστούμε, αν το θελήσουμε, μέσα από αυτή την ενότητα, επιλέγοντας την λειτουργία edit info. Κάνοντας το αυτό, εμφανίζεται μία φόρμα με τα τρέχοντα στοιχεία του ταξιδιού, δίνοντας μας τη δυνατότητα να τα αλλάξουμε και να σώσουμε τις αλλαγές μας πατώντας save changes. Όλη αυτή η διαδικασία γίνεται ασύγχρονα χωρίς να εγκαταλείπουμε την σελίδα. Αυτό φαίνεται παρακάτω:



Εικόνα 61. Επεξεργασία βασικών πληροφοριών ταξιδιού της ενότητας Basic Tour Info

Arrange Priority: Η ενότητα αυτή βρίσκεται στο αριστερό μέρος της σελίδας μας. Σε αυτήν τοποθετούνται μικρογραφίες των placemarks και των photo placemarks που δημιουργούνται. Η ύπαρξή της εξυπηρετεί την ανάγκη για ταξινόμηση των διαφορών σταθμών του ταξιδιού, ώστε να μπορεί να γνωρίζει το σύστημα τη σειρά με την οποία έγινε η επίσκεψη του κάθε σημείου. Η ταξινόμηση γίνεται σέρνοντας και ανακατανέμοντας τα εικονίδια (drag and drop). Να σημειωθεί ότι για να διευκολύνεται η διαδικασία αυτή, όταν κάποιο στοιχείο επιλεγεί από το χάρτη, τότε το αντίστοιχο εικονίδιο περιβάλλεται από μαύρο φόντο. Έτσι δεν ταλαιπωρούμαστε στο να το αναζητούμε ανάμεσα σε πολλές άλλες μικρογραφίες. Όταν κάνουμε hover με το ποντίκι πάνω από το κάθε εικονίδιο ο δείκτης έχει τη μορφή σταυρού. Ένα παράδειγμα χρήσης της ενότητας αυτής είναι το ακόλουθο:



Εικόνα 62. Παράδειγμα αλλαγής προτεραιότητας των placemarks, απο την ενότητα Arrange priority.

5.3.6 Edit Tour –Επεξεργασία ταξιδιού

Η σελίδα τις επεξεργασίας ταξιδιού είναι ίδια με αυτήν της δημιουργίας νέου. Η διαφορά είναι ότι όταν επιλέξουμε να επεξεργαστούμε ένα ταξίδι, τότε όλες του οι πληροφορίες φορτώνονται, δημιουργούνται τα placemarks που το αποτελούν πάνω στο χάρτη, και οι αντίστοιχες μικρογραφίες τους εμφανίζονται με τη προτεραιότητα που είχε οριστεί κατά τη δημιουργία, στην περιοχή του arrange priority. Έτσι μπορούμε να αλλάξουμε την πληροφορία των placemarks, τις βασικές πληροφορίες των ταξιδιών, όπως επίσης και να δημιουργήσουμε νέα στοιχεία. Η σελίδα του Edit tour για το ταξίδι Thessaloniki παρουσιάζεται παρακάτω:

Search friends...

logged in| Smalis Sklavos

Arrange Priority

1332190168612

1332189245710

1332189245274

1332189244163

1332189244878

1332189240870

1332189243788

1332190082894

1332190036953

1332190057161

Edit Tour: Thessaloniki

Thessaloniki

Toolkit

Photo Upload

Upload a file

Drop files here to upload!!!

Create Placemark

Name *

Description

Click to Select an icon

Create Placemark

Delete Tour

Save Tour

Basic Tour Info

Basic Tour Info

Tour Name

Thessaloniki

Begin Date

2012-03-10

End Date

2012-09-15

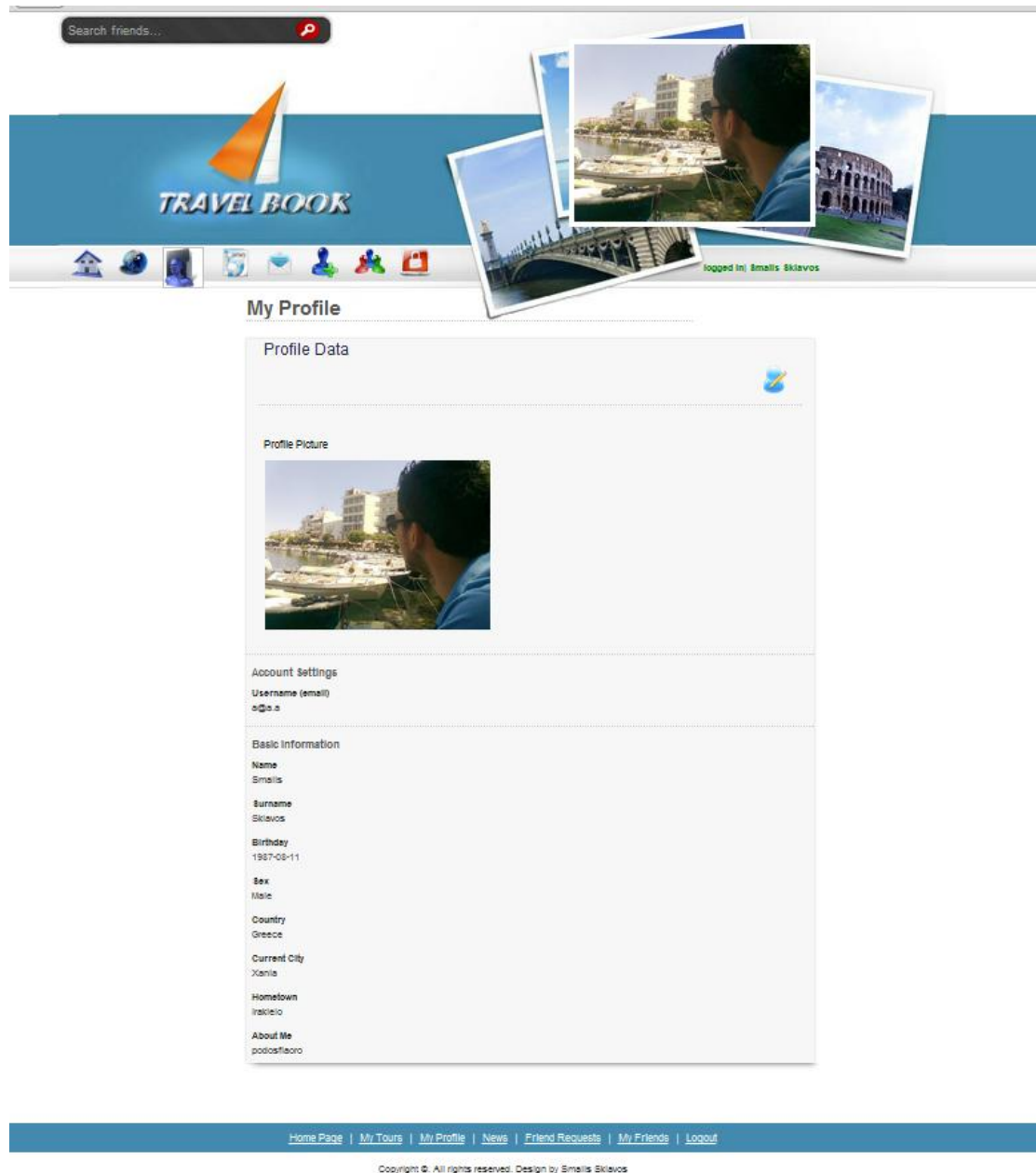
Description

Taksidi sti voreia ellada

[Edit Info](#)

Εικόνα 63. Edit Tour: Σελίδα επεξεργασίας ταξιδιού.

5.3.7 My Profile –Σελίδα προσωπικού προφίλ



Εικόνα 64. My profile. Σελίδα προσωπικού προφίλ

Επιλέγοντας από το κεντρικό οριζόντιο μενού το εικονίδιο του my profile, οδηγούμαστε στη σελίδα του προσωπικού μας προφίλ. Η σελίδα αυτή εικονίζεται παραπάνω. Εκεί ο χρήστης μπορεί να δει όλα τα στοιχεία που χαρακτηρίζουν το προφίλ του, εκτός από το password για λόγους ασφαλείας. Εάν κάποιο από τα μη

υποχρεωτικά πεδία που δεν το συμπλήρωσε ο χρήστης κατά τη διαδικασία της εγγραφής του ή έπειτα από επεξεργασία του προφίλ του, τότε αυτά δεν εμφανίζονται. Ένας χρήστης που βρίσκεται στη σελίδα του προσωπικού του προφίλ μπορεί να επιλέξει να το επεξεργαστεί, κάνοντας κλικ στο εικονίδιο που φαίνεται στο πάνω δεξί μέρος του τμήματος του προφίλ. Η σελίδα επεξεργασίας είναι η παρακάτω, και μοιάζει όπως θα δούμε με αυτήν της εγγραφής. Πρέπει να παρατηρήσουμε ότι η περιοχή του προφίλ παίρνει εδώ το χαρακτηριστικό γαλάζιο χρώμα που έχουν οι φόρμες της εφαρμογής.



Edit Profile

Profile Data Form

You should complete this form to sign up.
The fields with * are mandatory.

Profile Picture

Select your profile picture.

Upload Picture



Upload (only .jpg)

Account Settings

Username (email) *

Password *

Password Confirmation *

Basic Information

Name *

Surname

Birthday

 / /

Sex

☒ Male

☐ Female

Country

Current City

Hometown

About Me

[Reset](#)
[Save Changes](#)

5.3.8 Friend's Profile-Σελίδα προσωπικού προφίλ φίλου

Search friends...

TRAVEL BOOK

logged in! Smalls Sklavos

Kostas Tsiichlis

Profile Data

Profile Picture

Basic Information

Name
Kostas

Surname
Tsiichlis

Birthday
1986-02-06

Sex
Male

Country
Greece

Current City
Chania

Hometown
Uganda

About Me
Oe!H!

Friends

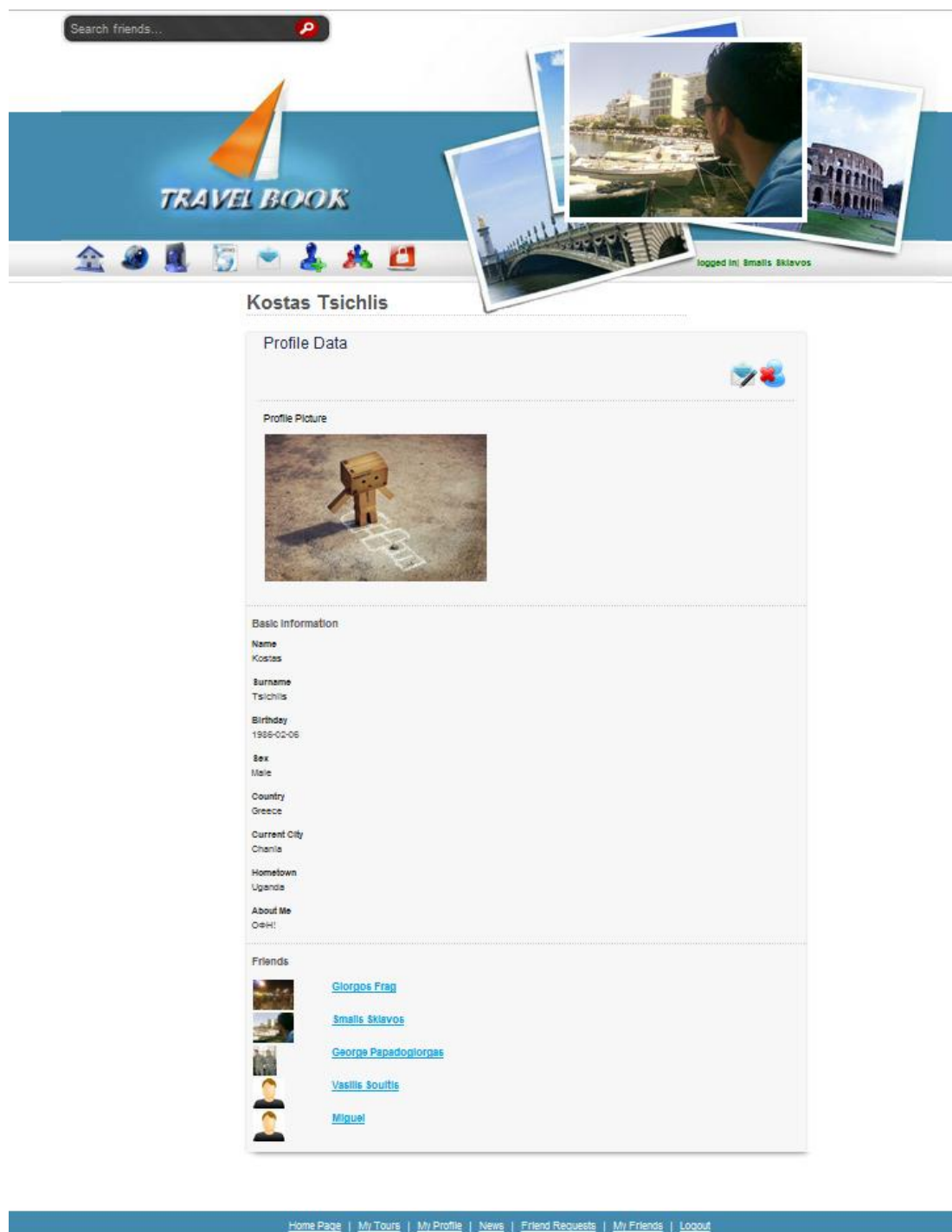
[→5 Friends](#)

Home Page | My Tours | My Profile | News | Friend Requests | My Friends | Logout

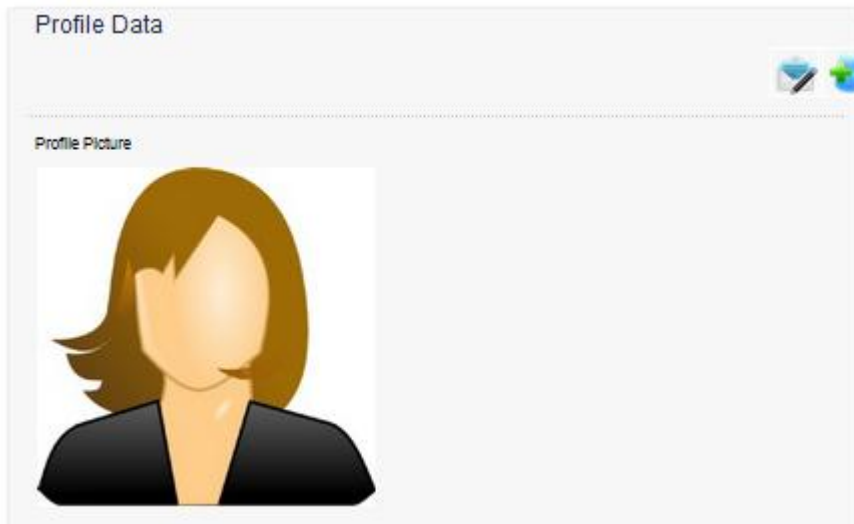
Copyright ©. All rights reserved. Design by Smalls Sklavos

Επιλέγοντας να οδηγηθούμε στην σελίδα προφίλ κάποιου χρήστη βλέπουμε την παραπάνω σελίδα. Αυτή μοιάζει με τη σελίδα προσωπικού προφίλ με τη διαφορά ότι στο πάνω δεξί μέρος της περιοχής του προφίλ υπάρχει πλέον ένα κουμπί-εικονίδιο. Αν ο χρήστης του οποίου το προφίλ βλέπουμε δεν είναι φίλος μας, υπάρχει το εικονίδιο αποστολής αιτήματος φιλίας. Αν έχουμε ήδη στείλει αίτημα φιλίας στο πρόσωπο αυτό ή είναι ήδη φίλος μας, τότε στο σημείο αυτό υπάρχει το εικονίδιο ακύρωσης φιλίας όπως παραπάνω. Πατώντας σε αυτό, είτε ακυρώνουμε το αίτημα φιλίας είτε διαγράφουμε το χρήστη από τη λίστα φίλων μας. Και στην περίπτωση που ο χρήστης είναι φίλος μας και σε αυτήν που δεν είναι, δίπλα στο εικονίδιο για τα αιτήματα φιλίας, υπάρχει και το εικονίδιο αποστολής προσωπικού μηνύματος στο χρήστη. Τέλος στο κάτω μέρος της σελίδας υπάρχει ένας σύνδεσμος με τον αριθμό φίλων του συγκεκριμένου χρήστη. Κάνοντας κλικ σε αυτό, οι φίλοι αυτοί φορτώνονται και εμφανίζονται ασύγχρονα στην ίδια σελίδα, όπως φαίνεται και στην

εικόνα που ακολουθεί. Με κλίκ σε αυτούς οδηγούμαστε στη σελίδα προφίλ τους, μια λειτουργία που βοηθάει στην ανάπτυξη δικτύου χρηστών και εύρεση νέων φίλων.



Εικόνα 65. Παράδειγμα εμφάνισης φίλων χρήστη.



Εικόνα 66. Διαφοροποίηση εικονιδίων όταν το προφίλ ανήκει σε μη φίλο. Εμφάνιση εικονιδίου αποστολής αιτήματος φιλίας.

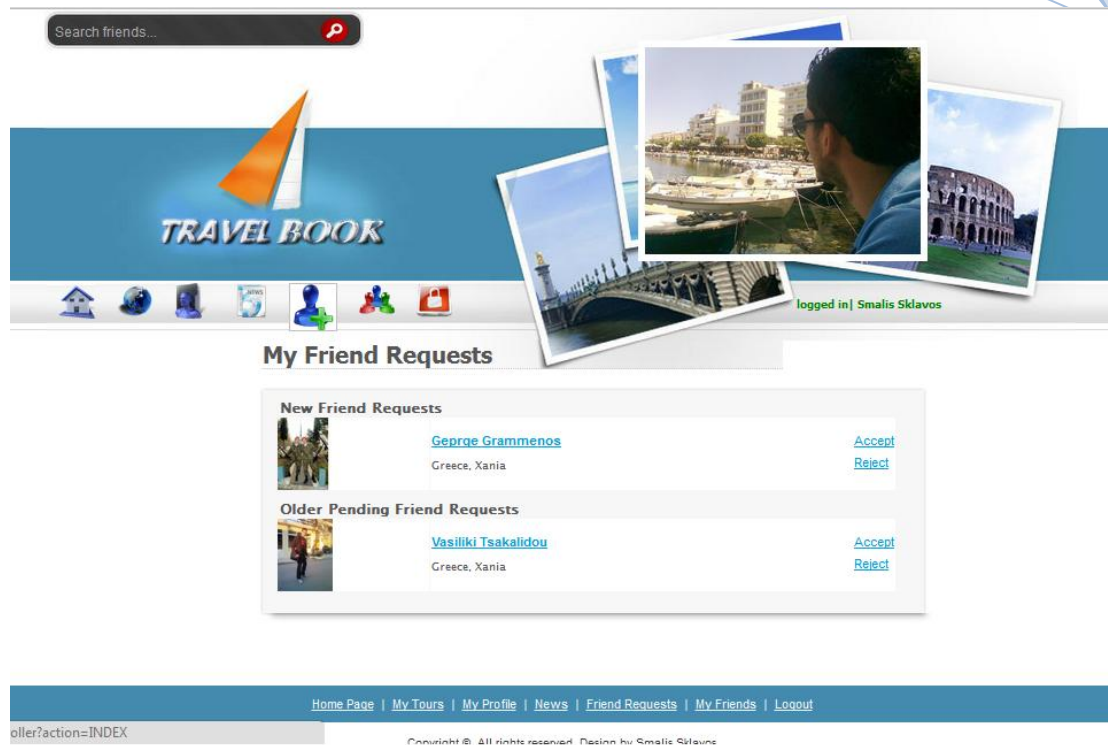
5.3.9 My Friend Requests –Σελίδα διαχείρισης αιτημάτων φιλίας.

Όταν κάποιος χρήστης επιλέξει να μας στείλει αίτημα φιλίας , τότε αυτό μας ανακοινώνεται τοποθετώντας δίπλα στο εικονίδιο του κεντρικού οριζόντιου μενού τον αριθμό των ΜΗ ΔΙΑΒΑΣΜΕΝΩΝ αιτημάτων φιλίας. Αυτό επιτυγχάνεται με το συνεχή έλεγχο (ασύγχρονο), με χρήση Ajax αν υπάρχουν αιτήματα φιλίας για τον χρήστη του συστήματος. Ένα παράδειγμα την νέας μορφής του μενού είναι το ακόλουθο, το οποίο μας ειδοποιεί ότι έχουμε ένα μη διαβασμένο αίτημα φιλίας. Να σημειωθεί ότι ο συγκεκριμένος έλεγχος γίνεται σε όλες τις σελίδες του συστήματος, και εάν ανακαλυφθεί αίτημα φιλίας, απλά εμφανίζεται ο κόκκινος αριθμός ή αλλάζει τιμή ανάλογα με το πλήθος των αιτημάτων, χωρίς να χρειάζεται να ξαναφορτωθεί η σελίδα.



Εικόνα 67.Ειδοποίηση αιτήματος φιλίας.

Η σελίδα διαχείρισης των αιτημάτων φιλίας είναι η ακόλουθη και οδηγούμαστε σε αυτή με κλικ στο αντίστοιχο εικονίδιο του κυρίου μενού.



Εικόνα 68. Σελίδα διαχείρισης αιτημάτων φιλίας

Όπως παρατηρούμε στην παραπάνω εικόνα μόλις μεταβούμε στην σελίδα διαχείρισης αιτημάτων, χάνεται η ένδειξη από το κεντρικό μενού ότι υπάρχουν μη διαβασμένα αιτήματα. Πλέον θεωρούνται διαβασμένα. Επίσης παρατηρώντας την εικόνα βλέπουμε ότι τα αιτήματα είναι οργανωμένα σε «New Friend Requests», που περιλαμβάνονται όλα τα μη αναγνωσμένα νέα μηνύματα, και «Older Pending Friend Requests» όπου περιλαμβάνονται τα ήδη αναγνωσμένα. Για κάθε αίτημα φιλίας, φαίνεται η εικόνα του χρήστη που μας το έστειλε, μερικές βασικές πληροφορίες για αυτόν, και οι σύνδεσμοι για αποδοχή ή απόρριψη του αιτήματος. Ειδοποιούμαστε για την ολοκλήρωση της απόρριψης ή αποδοχής από τα πράσινα μηνύματα “accepted” και “rejected” όπως αυτά εικονίζονται παρακάτω. Τα μηνύματα αυτά εμφανίζονται στη θέση των αντίστοιχων συνδέσμων, με την ολοκλήρωση της ασύγχρονη διαδικασίας απάντησης στο αίτημα.

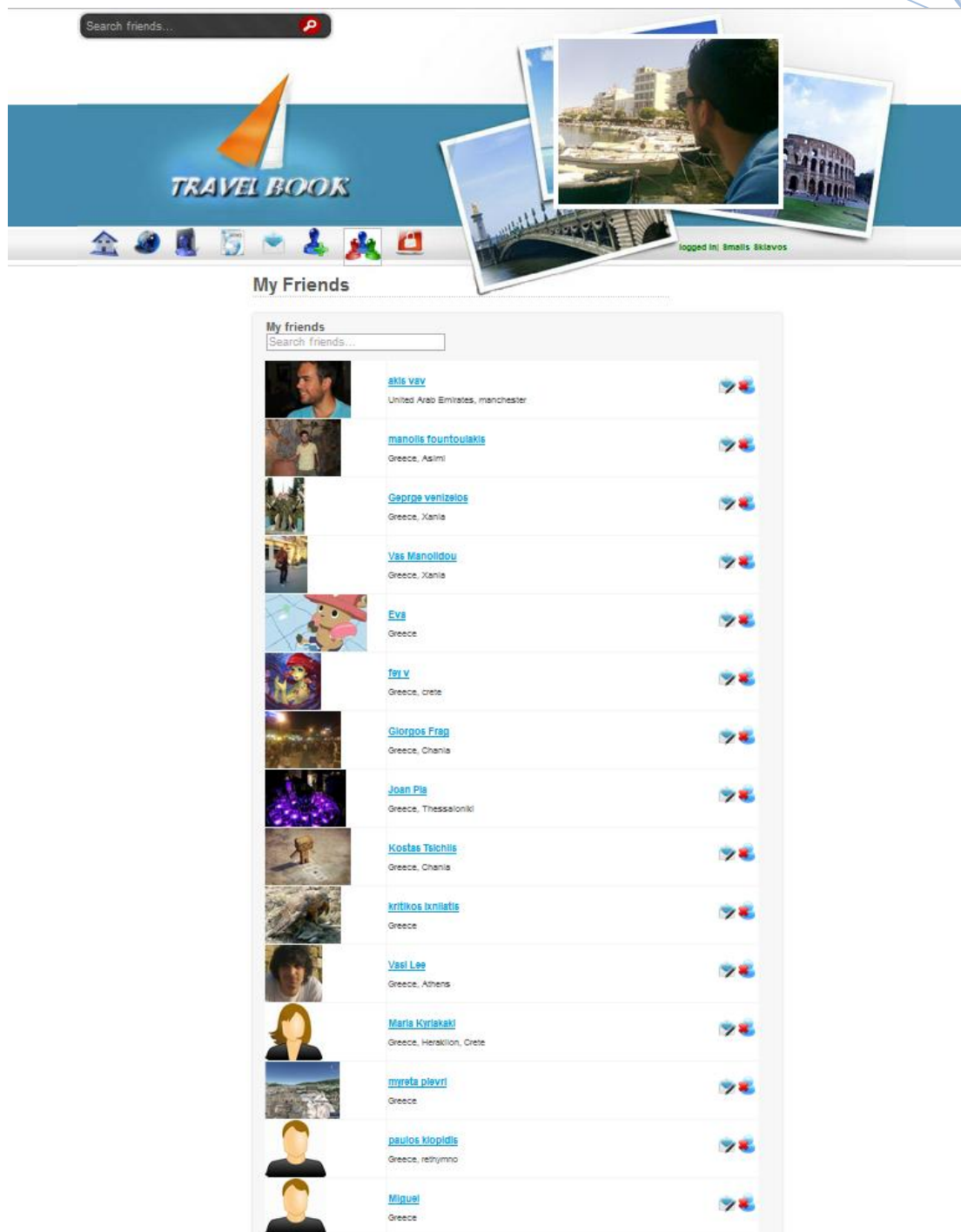
My Friend Requests



Εικόνα 69. Παράδειγμα απαντήσεων σε αιτήματα φιλίας.

5.3.10 My Friends –Σελίδα διαχείρισης φίλων.

Επιλέγοντας από το κύριο μενού το εικονίδιο My Friends οδηγούμαστε στη σελίδα διαχείρισης φίλων. Στη σελίδα αυτή εμφανίζεται το σύνολο των φίλων του χρήστη. Η σελίδα αυτή εικονίζεται παρακάτω, και όπως θα παρατηρήσουμε για κάθε χρήστη-φίλο εμφανίζονται η φωτογραφία προφίλ του στο αριστερό μέρος, κάποιες βασικές πληροφορίες για αυτόν και το εικονίδιο ακύρωσης φιλίας (Unfriend), με κλικ στο οποίο γίνεται η διαγραφή του συγκεκριμένου χρήστη από τη λίστα φίλων. Επίσης υπάρχει και το εικονίδιο αποστολής μηνύματος. Το ονοματεπώνυμο όπως βλέπουμε έχει το γαλάζιο χρώμα που υποδεικνύει ότι είναι σύνδεσμος και οδηγεί στη σελίδα προφίλ του συγκεκριμένου χρήστη.



Εικόνα 70.Σελίδα διαχείρισης φίλων χρηστή

Όσον αφορά τη διαδικασία αναζήτησης, αυτή λειτουργεί ασύγχρονα, χωρίς να χρειαστεί να φορτώνεται όλη η σελίδα κάθε φορά. Ενώ γράφουμε στην μπάρα αναζήτησης, πάνω από τη λίστα των φίλων εμφανίζονται στη λίστα μόνο τα αποτελέσματα τα οποία ταιριάζουν με την εισαγωγή μας. Ένα παράδειγμα αναζήτησης εικονίζεται παρακάτω.



Εικόνα 71. Παράδειγμα αναζήτησης φίλου

5.3.11 My Messages-Inbox.

Επιλέγοντας από το κεντρικό οριζόντιο μενού το εικονίδιο του My Messages οδηγούμαστε στην σελίδα των προσωπικών μηνυμάτων. Το εικονίδιο αυτό μας ειδοποιεί εάν έχουμε νέα εισερχόμενα μηνύματα με μία φωτεινή κόκκινη ένδειξη με τον αριθμό των μηνυμάτων.



Εικόνα 72.Ενημέρωση ότι έχουμε ένα προσωπικό μήνυμα.

Search friends...

TRAVEL BOOK

logged in| Smalis Sklavos

Inbox

Messages

From	Subject	Date
<input type="checkbox"/> Smalis Sklavos	to thema	Sat 26/05/2012 04:03 PM
<input type="checkbox"/> Eleni Fini	geiaa	Sat 26/05/2012 01:03 PM
<input type="checkbox"/> Eleni Fini	geiaa	Sat 26/05/2012 00:52 PM
<input type="checkbox"/> Eleni Fini	geiaa	Sat 26/05/2012 00:48 PM
<input type="checkbox"/> Eleni Fini	geiaa	Sat 26/05/2012 00:37 PM
<input type="checkbox"/> Eleni Fini	geiaa	Sat 26/05/2012 11:50 AM
<input type="checkbox"/> akis vav	pou saiiiiiiii!!!!	Fri 25/05/2012 08:19 PM
<input type="checkbox"/> akis vav	pou saiiiiiiii!!!!	Fri 25/05/2012 05:28 PM
<input type="checkbox"/> Smalis Sklavos		Fri 25/05/2012 01:14 AM
<input type="checkbox"/> Smalis Sklavos	aaa	Thu 24/05/2012 08:06 PM
<input type="checkbox"/> Smalis Sklavos	aa	Thu 24/05/2012 06:57 PM

Page: 1

Home Page | My Tours | My Profile | News | Friend Requests | My Friends | Logout

Copyright ©. All rights reserved. Design by Smalis Sklavos

Εικόνα 73.Σελίδα μηνυμάτων. Inbox.

Όταν ο χρήστης επιλέξει να δει τα προσωπικά του μηνύματα οδηγείται στη σελίδα του Inbox, που φαίνεται παραπάνω. Αναλύοντας τη σελίδα έχουμε:

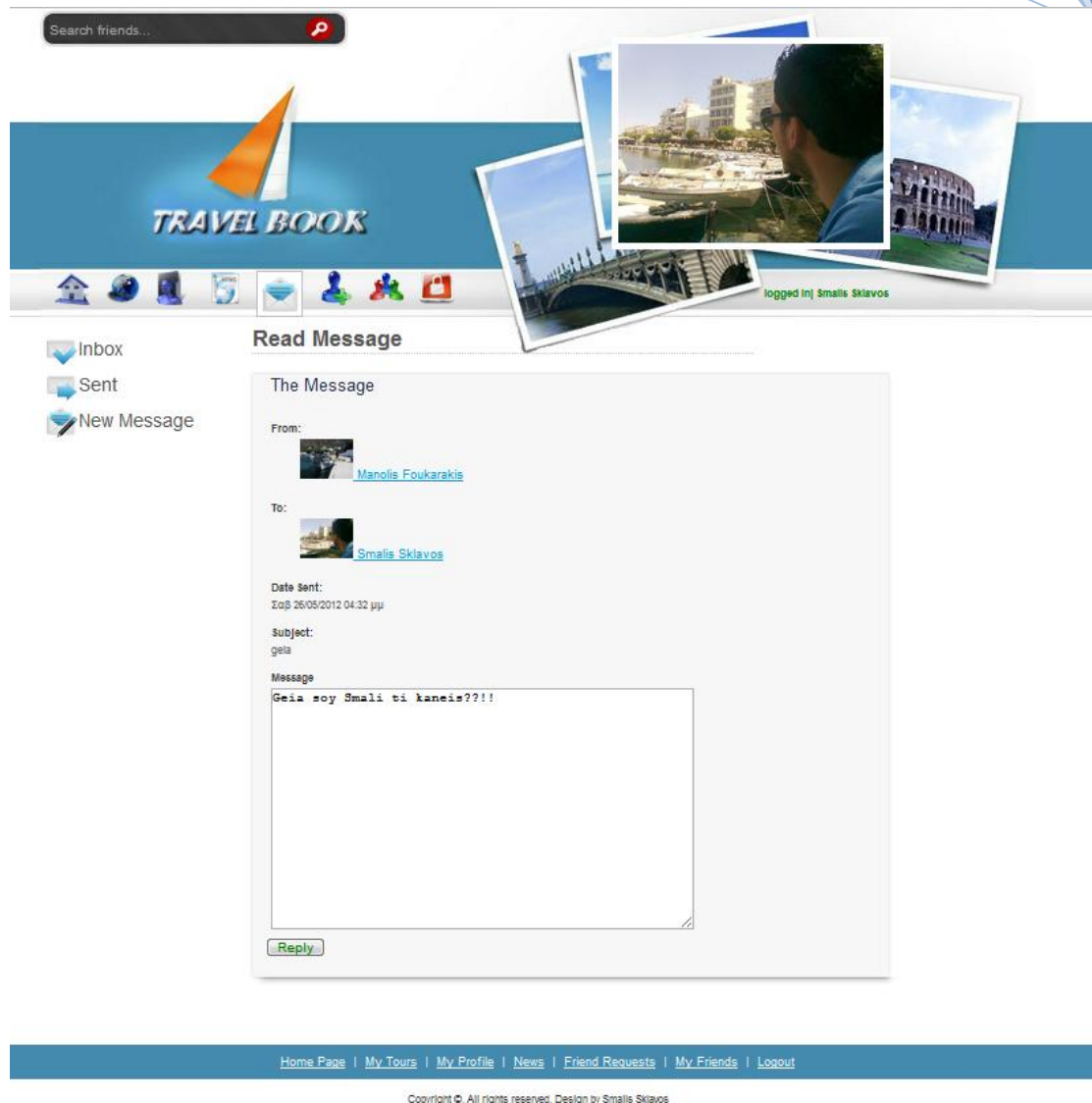
Πίνακας εισερχομένων: Εδώ βρίσκονται όλα τα εισερχόμενα μηνύματα του χρήστη. Τα νεότερα σημαίνονται με ποιο σκούρα γραφή (bold), όπως το πρώτο μήνυμα της παραπάνω εικόνας. Στον πίνακα εμφανίζεται ο αποστολέας (from), το θέμα (subject) και η ημερομηνία και ώρα αποστολής. Στο αριστερό μέρος του πίνακα υπάρχουν

checkboxes, τα οποία ο χρήστης μπορεί να επιλέξει, και ακολούθως να διαγράψει τα αντίστοιχα μηνύματα πατώντας το κουμπί της διαγραφής που βρίσκεται πάνω από τον πίνακα. Δίπλα στο κουμπί της διαγραφής υπάρχει ένα κουμπί επιλογής όλων των checkboxes. Ο μέγιστος αριθμός των μηνυμάτων ανά σελίδα είναι 30, και είναι οργανωμένα σε σελίδες.

Υπομενού σελίδων μηνυμάτων: Στο αριστερό μέρος της σελίδας, όπως βλέπουμε υπάρχει ένα μενού. Αυτό δίνει τη δυνατότητα στο χρήστη να επιλέξει, αν θέλει να οδηγηθεί στη σελίδα του inbox και να δει τα εισερχόμενα του, αν θέλει να οδηγηθεί στη σελίδα των απεσταλμένων μηνυμάτων, η εάν θέλει να δημιουργήσει ένα νέο μήνυμα. Ανάλογα με το ποια σελίδα μηνυμάτων βρίσκεται ο χρήστης, σημαίνεται με ασημί χρώμα η αντίστοιχη επιλογή του μενού.

5.3.12 Read Message- Σελίδα ανάγνωσης μηνύματος

Επιλέγοντας ο χρήστης τα μηνύματα που θέλει να δει, ανεξάρτητα με το εάν είναι εισερχόμενο ή βρίσκεται στην κατηγορία των σταλμένων, οδηγείται στη σελίδα ανάγνωσής του, που είναι η ακόλουθη.



Εικόνα 74.Σελίδα ανάγνωσης προσωπικών μηνυμάτων.

Στο αριστερό μέρος της σελίδας παρατηρούμε την ύπαρξη του υπομενού για το οποίο αναφερθήκαμε προηγουμένως. Τα νέα στοιχεία είναι τα ακόλουθα:

Στοιχεία μηνύματος: Κατά την ανάγνωση ενός μηνύματος τα στοιχεία τα οποία αναγράφονται είναι ο αποστολέας (from) και ο αποδέκτης (to). Για τα δύο αυτά πρόσωπα εμφανίζεται το ονοματεπώνυμο και η φωτογραφία τους. Στα στοιχεία συγκαταλέγονται επίσης το θέμα, και το ίδιο το κείμενο του μηνύματος.

Επιλογή απάντησης: Στο κάτω μέρος του μηνύματος δίνεται η δυνατότητα απάντησης, πατώντας στο κουμπί reply.

5.3.13 New Message-Σελίδα σύνταξης μηνύματος

Search friends...

TRAVEL BOOK

logged in | Smalls Sklavos

New Message

Create New Message

To: +

Subject:

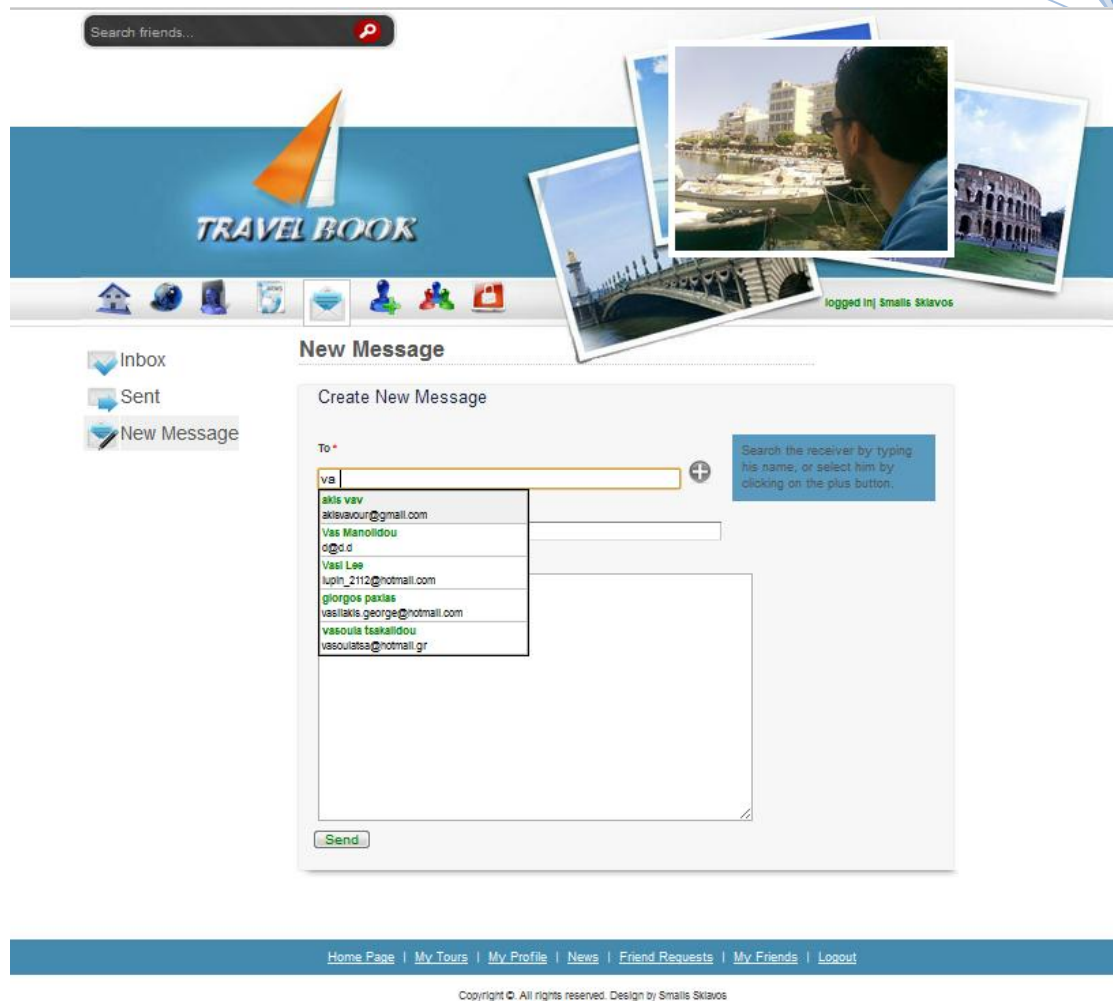
Message:

Home Page | My Tours | My Profile | News | Friend Requests | My Friends | Logout

Copyright ©. All rights reserved. Design by Smalls Sklavos

Εικόνα 75.Σελίδα δημιουργίας νέου μηνύματος

Επιλέγοντας ο χρήστης τη σύνταξη νέου μηνύματος (New Message) οδηγείται στην παραπάνω σελίδα. Τα πεδία που πρέπει να συμπληρώσει είναι το email-username του παραλήπτη (To), το θέμα του μηνύματος (Subject) και το κείμενο του μηνύματος (Message). Η συμπλήρωση του παραλήπτη μπορεί να γίνει αυτόματα, με την εμφάνιση του μενού χρηστών πατώντας στο κουμπί +, δίπλα στο πεδίο του παραλήπτη και επιλέγοντας τον επιθυμητό. Επίσης κατά την εγγραφή στο πεδίο του αποστολέα του ονόματος, του επιθέτου ή της αρχής του username, γίνεται αναζήτηση των χρηστών, φίλων του χρήστη και εμφανίζονται οι αντίστοιχες προτάσεις. Ένα παράδειγμα είναι το ακόλουθο.



Εικόνα 76. Παράδειγμα αναζήτησης παραλήπτη.

5.3.14 Sent Messages-Σελίδα απεσταλμένων μηνυμάτων.

Search friends...

TRAVEL BOOK

logged in Smalls Skiavos

Sent

Messages

To	Subject	Date
<input type="checkbox"/> Smalls Skiavos	aa	Παρ 25/05/2012 00:09 μμ
<input type="checkbox"/> akis vav	pou saillllllll!!!	Παρ 25/05/2012 11:13 πμ
<input type="checkbox"/> Smalls Skiavos		Παρ 25/05/2012 01:14 πμ
<input type="checkbox"/> Giorgos Frag	para poly wrala	Παρ 25/05/2012 01:14 πμ
<input type="checkbox"/> giorgos parlas	gela	Παρ 25/05/2012 01:12 πμ
<input type="checkbox"/> Miguel	para poly wrala	Παρ 25/05/2012 01:10 πμ
<input type="checkbox"/> Smalls Skiavos	aaa	Πεμ 24/05/2012 08:06 μμ
<input type="checkbox"/> George Papadogiorgas	PSaroukkaa	Πεμ 24/05/2012 07:57 μμ
<input type="checkbox"/> Smalls Skiavos	aa	Πεμ 24/05/2012 06:57 μμ

Page: 1

[Home Page](#) | [My Tours](#) | [My Profile](#) | [News](#) | [Friend Requests](#) | [My Friends](#) | [Logout](#)

Copyright © All rights reserved. Design by Smalls Skiavos

Εικόνα 77. Σελίδα απεσταλμένων μηνυμάτων.

Η σελίδα αυτή είναι ίδια με τη σελίδα του inbox με τη διαφορά ότι εδώ τα μηνύματα που μπορεί να δει ο χρήστης είναι όλα αυτά που αυτός έχει στείλει.

6. Βιβλιογραφία

- [1] Wikipedia, «Wikipedia,» [Ηλεκτρονικό]. Available: http://en.wikipedia.org/wiki/Google_Earth.
- [2] Wikipedia, «Wikipedia,» [Ηλεκτρονικό]. Available: http://en.wikipedia.org/wiki/Geographic_information_system.
- [3] G. E. Api, «Google Earth Api,» Google, [Ηλεκτρονικό]. Available: <https://developers.google.com/earth/documentation/index>. [Πρόσβαση 27 5 2012].
- [4] G. Earth, «Google Earth Api,» Google, [Ηλεκτρονικό]. Available: <https://developers.google.com/earth/documentation/placemarks>. [Πρόσβαση 27 5 2012].
- [5] G. E. Api, «Google Earth Api,» Google, [Ηλεκτρονικό]. Available: <https://developers.google.com/earth/documentation/balloons>. [Πρόσβαση 27 5 2012].
- [6] G. E. Api, «Google Earth Api,» Google, [Ηλεκτρονικό]. Available: <https://developers.google.com/earth/documentation/events>. [Πρόσβαση 27 5 2012].
- [7] G. M. Api, «Google Maps Api,» [Ηλεκτρονικό]. Available: <https://developers.google.com/maps/documentation/geocoding/>. [Πρόσβαση 27 5 2012].
- [8] G. E. Api, «Google Earth Api Developers Guide,» Google, [Ηλεκτρονικό]. Available: <https://developers.google.com/earth/documentation/kml>.
- [9] G. E. Api, «Google Earth Api- Developers Guide,» [Ηλεκτρονικό]. Available: <https://developers.google.com/earth/documentation/touring>.
- [10] G. E. Api, «Kml Reference,» [Ηλεκτρονικό]. Available: <https://developers.google.com/kml/documentation/kmlreference?hl=el-GR#icon>.
- [11] «Wikipedia,» [Ηλεκτρονικό]. Available: <http://el.wikipedia.org/wiki/JavaScript>.
- [12] DeveloperWorks, «DeveloperWorks,» IBM, [Ηλεκτρονικό]. Available: <http://www.ibm.com/developerworks/java/library/j-ajax1/?ca=dgrwikibAJAX4Java1>.
- [13] «Java Tutorial,» [Ηλεκτρονικό]. Available: http://papa.det.uvigo.es/~theiere/cursos/Curso_Java/characteristics.html.
- [14] Wikipedia, «Java Servlet,» [Ηλεκτρονικό]. Available: http://en.wikipedia.org/wiki/Java_Servlet.
- [15] M. H. -L. Brown, «Servlets και σελίδες διακομιστή Java,» σε *Servlets και σελίδες διακομιστή Java*.
- [16] Κ. Π. Φλώρινας, «Εισαγωγή στα Java Servlets,» [Ηλεκτρονικό]. Available:

- <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-JavaServlets.html>.
- [17] «Jsp Intro and Overview,» [Ηλεκτρονικό]. Available:
<http://courses.coreservlets.com/Course-Materials/pdf/csajsp2/09-JSP-Intro.pdf>.
- [18] Stardeveloper.com, «Combining Servlets, JSP, and JavaBeans,» [Ηλεκτρονικό]. Available:
<http://www.stardeveloper.com/articles/display.html?article=2001060501&page=1>.
- [19] Wikipedia, «Data Access Objects,» [Ηλεκτρονικό]. Available:
http://en.wikipedia.org/wiki/Data_access_object.
- [20] Netbeans, «NetBeans IDE 6.8 Release Information,» [Ηλεκτρονικό]. Available:
<http://netbeans.org/community/releases/68/>.
- [21] A. Tomcat, «Apache Tomcat,» [Ηλεκτρονικό]. Available: <http://tomcat.apache.org>.
- [22] MySQL, «MySQL 5.0,» [Ηλεκτρονικό]. Available: <http://dev.mysql.com>.
- [23] T. M. Ph.D, «The Golden Rules of User Interface Design,» [Ηλεκτρονικό]. Available:
<http://www.theomandel.com/docs/Mandel-GoldenRules.pdf>.

