



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΥΠΟΛΟΓΙΣΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης
χρήστη σε χάρτη**

ΒΑΣΙΛΕΙΟΣ ΜΑΤΡΑΓΚΟΣ
A.M. mngd22008

Εισηγητής: ΝΙΚΟΛΑΟΣ ΖΑΧΑΡΗΣ, ΚΑΘΗΓΗΤΗΣ

(Κενό φύλλο)

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΗΣ ΕΦΑΡΜΟΓΗΣ ΠΛΟΗΓΗΣΗΣ
ΧΡΗΣΤΗ ΣΕ ΧΑΡΤΗ**

**Βασίλειος Ματράγκος
Α.Μ. mngd22008**

Εισηγητής:

ΝΙΚΟΛΑΟΣ ΖΑΧΑΡΗΣ, ΚΑΘΗΓΗΤΗΣ

Εξεταστική Επιτροπή:

**ΓΕΩΡΓΙΟΣ ΠΡΕΖΕΡΑΚΟΣ, ΚΑΘΗΓΗΤΗΣ
ΑΝΤΩΝΗΣ ΜΠΟΓΡΗΣ, ΚΑΘΗΓΗΤΗΣ**

Ημερομηνία εξέτασης 20/02/2024

(Κενό φύλλο)

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ/ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος Βασίλειος Ματράγκος του Θεοδώρου, με αριθμό μητρώου mngd22008 φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της πτυχιακής/διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο Δηλών



(Κενό φύλλο)

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εργασία ολοκληρώθηκε μετά από επίμονες προσπάθειες, σε ένα ενδιαφέρον γνωστικό αντικείμενο, όπως αυτό της ανάπτυξης διαδικτυακών εφαρμογών. Την προσπάθειά μου αυτή υποστήριξε ο επιβλέπων καθηγητής μου, κύριος Νικόλαος Ζάχαρης τον οποίο θα ήθελα να ευχαριστήσω.

Ακόμα θα ήθελα να ευχαριστήσω την οικογένειά μου για τη συμπαράσταση κατά τη διάρκεια των σπουδών μου.

(Κενό φύλλο)

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία ασχολείται με τον σχεδιασμό και την ανάπτυξη διαδικτυακής εφαρμογής που επιτρέπει την πλοήγηση ενός χρήστη σε χάρτη.

Η εφαρμογή απευθύνεται σε χρηστές κινητών συσκευών που διαθέτουν GPS και σύνδεση στο διαδίκτυο. Ο χρήστης μπορεί να βλέπει το στίγμα του στον χάρτη σε πραγματικό χρόνο και να πραγματοποιεί καταγραφή της δραστηριότητάς του.

Η καταγραφή της δραστηριότητάς του χρήστη, στην παρούσα εφαρμογή χωρίζεται σε δυο κατηγορίες. Η πρώτη κατηγορία είναι η αυτόματη καταγραφή, όπου καταγράφεται η θέση του χρήστη, καθώς και άλλα δεδομένα σε πραγματικό χρόνο. Η δεύτερη κατηγορία είναι ένας συνδυασμός αυτόματης και χειροκίνητης καταγραφής, όπου ο χρήστης έχει τη δυνατότητα να καταγράψει επιπλέον δεδομένα, όπως καρδιακούς παλμούς, σημειώσεις, φωτογραφίες και βίντεο.

Τα δεδομένα της καταγραφής του χρήστη, συνοψίζονται και παρουσιάζονται με ποικίλους τρόπους, όπως προβολή σε χάρτη των διαδρομών που πραγματοποιήθηκαν, ιστορικό διαδρομών, μετρήσεις αποστάσεων, περιοχές που επισκέφτηκε ο χρήστης, γραφήματα με τους καρδιακούς παλμούς, προβολή σημειώσεων, φωτογραφιών και βίντεο.

Ο βασικός στόχος της παρούσας διπλωματικής εργασίας είναι να παρουσιαστεί η λειτουργικότητα και οι δυνατότητες που προσφέρει το Geolocation API της HTML, μέσω ενός φυλλομετρητή (browser), που σε συνδυασμό με ασύγχρονα αιτήματα Http (AJAX) της γλώσσας προγραμματισμού JavaScript, μπορεί να αποτελέσει τη βάση για μια πληθώρα εφαρμογών, όπως υπηρεσιών χαρτών και πλοήγησης, εφαρμογών υγείας, υπηρεσιών διαφήμισης, παρακολούθησης οχημάτων και άλλων εφαρμογών που εκμεταλλεύονται γεωγραφικά δεδομένα για πολλούς διαφορετικούς σκοπούς.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Ανάπτυξη Διαδικτυακών Εφαρμογών
ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Geolocation, Χάρτες, Client, Server, AJAX

“Design and development of a web application for user navigation in a map”

ABSTRACT

The present thesis deals with the design and development of a web application that allows a user to navigate on a map.

The application is aimed at users of mobile devices that have GPS and an internet connection. The user can see his position on the map in real time and record his activity.

The recording of the user's activity, in this application, is divided into two categories. The first category is automatic recording, where the user's location and other data are recorded in real time.

The second category is a combination of automatic and manual recording, where the user has the option to record additional data such as heart rate, notes, photos and videos.

User log data is summarized and presented in a variety of ways, such as a map view of routes taken, route history, distance measurements, areas visited by the user, heart rate graphs, view notes, photos and videos.

The main objective of this thesis is to present the functionality and possibilities offered by the Geolocation API of HTML, through a browser, which, in combination with asynchronous Http requests (AJAX) of the JavaScript programming language, can form the basis for a multitude of applications such as map and navigation services, health applications, advertising services, vehicle tracking and other applications that exploit geographic data for many different purposes.

SCIENTIFIC AREA: Web Application Development

KEYWORDS: Geolocation, Maps, Client, Server, AJAX

ΠΕΡΙΕΧΟΜΕΝΑ

1.	ΕΙΣΑΓΩΓΗ	14
1.1	Περιγραφή του αντικειμένου της διπλωματικής εργασίας	14
1.2	Τεχνολογίες	16
2.	ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΕΦΑΡΜΟΓΗΣ	21
2.1	Διεπαφή χρήστη	21
2.2	Δομή αρχείων	24
2.3	Δομή κώδικα	26
2.4	Περιγραφή της βάσης δεδομένων	27
2.5	Ασφάλεια	31
2.6	Υποδομές	33
3.	ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ ΕΦΑΡΜΟΓΗΣ	43
3.1	Είσοδος στην εφαρμογή	43
3.2	Εγγραφή Χρήστη	44
3.3	Σύνδεση Χρήστη	47
3.4	Μενού επιλογών κορυφής	49
3.5	Κεντρικό μενού επιλογών	51
3.6	Καταγραφή διαδρομής	52
3.7	Ιστορικό διαδρομών	57
3.8	Περιοχές που επισκέφτηκα	67
3.9	Οι παλμοί μου	70
3.10	Οι σημειώσεις μου	71
3.11	Οι φωτογραφίες μου	72
3.12	Τα video μου	73
3.13	Copyright	74
4.	ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΙΘΑΝΕΣ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ	75
4.1	Συμπεράσματα	75
4.2	Πιθανές μελλοντικές επεκτάσεις	77
	ΒΙΒΛΙΟΓΡΑΦΙΑ	82
	ΠΑΡΑΡΤΗΜΑ Α	83

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 2.1: Μοντέλο Οντοτήτων-Συσχετίσεων	30
Εικόνα 2.2: Αναπαράσταση πινάκων στο phpMyAdmin	31
Εικόνα 2.3: Το XAMPP σε λειτουργία	34
Εικόνα 2.4: phpMyAdmin	34
Εικόνα 2.5: Microsoft Azure	36
Εικόνα 2.6: Resource group	37
Εικόνα 2.7: Virtual machine	38
Εικόνα 2.8: Virtual network	39
Εικόνα 2.9: Web App	40
Εικόνα 3.1: Πρώτη οθόνη της εφαρμογής	44
Εικόνα 3.2: Οθόνη εγγραφής χρήστη	45
Εικόνα 3.2.1: Μήνυμα πρόληψης σφάλματος	46
Εικόνα 3.2.2: Μήνυμα ολοκλήρωσης εγγραφής χρήστη	47
Εικόνα 3.3: Οθόνη σύνδεσης χρήστη	48
Εικόνα 3.3.1: Μήνυμα λάθους σε καταχώρηση στοιχείων	49
Εικόνα 3.4: Μενού επιλογών κορυφής	49
Εικόνα 3.4.1: Παράθυρο διαλόγου αποσύνδεσης χρήστη	50
Εικόνα 3.5: Κεντρικό μενού επιλογών	51
Εικόνα 3.6: Μήνυμα συστήματος για την κατάσταση της τοποθεσίας	53
Εικόνα 3.6.1: Οθόνη καταγραφής διαδρομής	54
Εικόνα 3.6.2: Οθόνη καταγραφής επιπλέον δεδομένων	56
Εικόνα 3.7: Οθόνη ιστορικού διαδρομών	57
Εικόνα 3.7.1: Οθόνη της διαδρομής του χρήστη	58
Εικόνα 3.7.2: Οθόνη με αναλυτικές πληροφορίες της διαδρομής του χρήστη	59
Εικόνα 3.7.3: Παράθυρο επιβεβαίωσης διαγραφής της διαδρομής	60
Εικόνα 3.7.4: Οθόνη προβολής καρδιακών παλμών του χρήστη	61
Εικόνα 3.7.5: Οθόνη προβολής πληροφοριών καρδιακών παλμών	62
Εικόνα 3.7.6: Οθόνη προβολής σημειώσεων του χρήστη	63
Εικόνα 3.7.7: Οθόνη προβολής φωτογραφιών του χρήστη	64
Εικόνα 3.7.8: Οθόνη προβολής βίντεο του χρήστη	65
Εικόνα 3.7.9: Οθόνη αναζήτησης σε δεδομένα διαδρομών	66

Εικόνα 3.8: Οθόνη περιοχών που έχει επισκεφτεί ο χρήστης	67
Εικόνα 3.8.1: Χάρτης περιοχής που έχει επισκεφτεί ο χρήστης	68
Εικόνα 3.8.2: Προβολή σε χάρτη των περιοχών που έχει επισκεφτεί ο χρήστης ...	69
Εικόνα 3.9: Προβολή των καρδιακών παλμών του χρήστη ανά διαδρομή	70
Εικόνα 3.10: Προβολή όλων των σημειώσεων του χρήστη	71
Εικόνα 3.11: Προβολή όλων των φωτογραφιών του χρήστη	72
Εικόνα 3.12: Προβολή των όλων των βίντεο του χρήστη	73
Εικόνα 3.13: Copyright	74

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 2.1: Πίνακας users	27
Πίνακας 2.2: Πίνακας route	28
Πίνακας 2.3: Πίνακας routes	29
Πίνακας 2.4: Πίνακας notes	29

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

API Application Programming Interface

AJAX Asynchronous JavaScript And XML

HTML HyperText Markup Language

CSS Cascading Style Sheets

UI User Interface

UX User Experience

VM Virtual machine

SaaS Software as a Service

FTP File Transfer Protocol

GPS Global Positioning System

JS Java Script

IP Internet Protocol

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

Η ανάπτυξη μιας διαδικτυακής εφαρμογής για κινητές συσκευές με ενσωματωμένο GPS αποτελεί πρόκληση που παρέχει τη δυνατότητα για μια εξατομικευμένη εμπειρία από την πλευρά του χρήστη.

Οι κινητές συσκευές με GPS προσφέρουν μια πληθώρα δυνατοτήτων, όπως πλοήγηση σε τοποθεσίες, εντοπισμό ατόμων, διαφήμιση βασισμένη στη θέση του χρήστη, διαδραστικά παιχνίδια και πολλές άλλες.

Οι διαδικτυακές εφαρμογές με GPS επίσης απαιτούν σύνδεση στο διαδίκτυο για να ανταλλάσσουν δεδομένα με διακομιστές, όπως χάρτες, πληροφορίες θέσης και άλλα.

Τα δεδομένα που παρέχει το GPS, μπορούν να εκμεταλλευτούν κατάλληλα ώστε να χρησιμοποιηθούν για τον εντοπισμό της θέσης του χρήστη σε έναν χάρτη, την παρουσίαση χαρτών και την περαιτέρω εμφάνιση πληροφοριών.

Μια εύχρηστη διεπαφή, επιτρέπει στον χρήστη να αλληλεπιδρά με την εφαρμογή και να χρησιμοποιεί τις λειτουργίες του GPS με ευκολία.

Η ανάπτυξη μιας διαδικτυακής εφαρμογής για κινητές συσκευές με GPS μπορεί να προσφέρει μια σειρά από συναρπαστικές δυνατότητες με μια εξαιρετική εμπειρία χρήστη.

1.1 Περιγραφή του αντικειμένου της διπλωματικής εργασίας

Ο βασικός στόχος της παρούσας διπλωματικής εργασίας είναι να παρουσιαστεί η λειτουργικότητα και οι δυνατότητες που προσφέρει το Geolocation API της HTML, μέσω ενός φυλλομετρητή (browser), που σε συνδυασμό την τεχνική των ασύγχρονων αιτημάτων Http (AJAX) της γλώσσας προγραμματισμού JavaScript, μπορεί να αποτελέσει τη βάση για μια πληθώρα εφαρμογών, όπως υπηρεσιών χαρτών και πλοήγησης, εφαρμογών υγείας, υπηρεσιών διαφήμισης,

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

παρακολούθησης οχημάτων και πολλών άλλων εφαρμογών που εκμεταλλεύονται γεωγραφικά δεδομένα.

Η παρούσα διπλωματική εργασία ασχολείται με τον σχεδιασμό και την ανάπτυξη μιας διαδικτυακής εφαρμογής που επιτρέπει την πλοήγηση ενός χρήστη σε χάρτη, καθώς και την καταγραφή και προβολή της δραστηριότητάς του.

Η εφαρμογή προορίζεται για χρηστές κινητών συσκευών που διαθέτουν GPS και σύνδεση στο διαδίκτυο. Το βασικό μέρος της εφαρμογής είναι η προβολή της θέσης του χρήστη σε χάρτη, σε πραγματικό χρόνο και η καταγραφή της δραστηριότητάς του. Η καταγραφή της δραστηριότητάς του χρήστη, χωρίζεται σε δυο κατηγορίες.

Η πρώτη κατηγορία είναι η αυτόματη καταγραφή, που λειτουργεί σε πραγματικό χρόνο σύμφωνα με την κίνηση του χρήστη. Κατά την κίνηση του χρήστη, λαμβάνονται δεδομένα από το GPS και αποθηκεύονται απευθείας σε μια βάση δεδομένων. Τα δεδομένα αυτά περιέχουν τις συντεταγμένες του χρήστη, την ημερομηνία και την ώρα, ένα μοναδικό αναγνωριστικό ταυτοποίησης του χρήστη, καθώς και το όνομα της περιοχής που βρίσκεται ο χρήστης εκείνη τη στιγμή.

Η δεύτερη κατηγορία είναι ένας συνδυασμός αυτόματης και χειροκίνητης καταγραφής. Ο χρήστης έχει τη δυνατότητα να καταγράφει χειροκίνητα επιπλέον δεδομένα, όπως καρδιακούς παλμούς, σημειώσεις, φωτογραφίες, βίντεο και παράλληλα να καταγράφονται αυτόματα και άλλα δεδομένα που σχετίζονται με την τρέχουσα θέση του.

Διαθέτοντας αυτά τα δεδομένα, προκύπτουν επιπλέον δυνατότητες, που συνθέτουν το δεύτερο μέρος της εφαρμογής, προσφέροντας στον χρήστη μια εξατομικευμένη εμπειρία. Τα δεδομένα της καταγραφής του χρήστη παρουσιάζονται με ποικίλους τρόπους, όπως:

- Προβολή σε χάρτη των διαδρομών που πραγματοποίησε ο χρήστης.
- Ιστορικό διαδρομών του χρήστη.
- Αναζήτηση με συγκεκριμένα κριτήρια.

- Μέτρηση αποστάσεων.
- Περιοχές που επισκέφτηκε ο χρήστης.
- Γραφήματα με τους καρδιακούς παλμούς.
- Προβολή των σημειώσεων του χρήστη.
- Προβολή των φωτογραφιών του χρήστη.
- Προβολή των βίντεο του χρήστη.

1.2 Τεχνολογίες

Οι τεχνολογίες ανάπτυξης εφαρμογών διαδικτύου αποτελούν ένα σύνολο εργαλείων, γλωσσών προγραμματισμού και πλατφορμών που μας επιτρέπουν να δημιουργήσουμε δυναμικές, λειτουργικές και καινοτόμες εφαρμογές που εξυπηρετούν τις ανάγκες των χρηστών.

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκαν οι τεχνολογίες HTML, CSS, JavaScript, PHP και η βάση δεδομένων MySQL, όπως περιγράφονται αναλυτικά παρακάτω:

HTML

Η HTML (HyperText Markup Language) αποτελεί τη βάση του διαδικτύου και την γλώσσα που χρησιμοποιείται για τη δημιουργία ιστοσελίδων και εφαρμογών διαδικτύου.

Είναι ένα από τα πιο σημαντικά εργαλεία για την ανάπτυξη περιεχομένου και αποτελεί τον σκελετό κάθε ιστοσελίδας ή εφαρμογής που προβάλλεται σε έναν φυλλομετρήτη (browser).

Η HTML χρησιμοποιεί μια δομή με ετικέτες (tags) που καθορίζουν τον τρόπο παρουσίασης του περιεχομένου. Μέσω αυτών των ετικετών, μπορούμε να προσδιορίσουμε τον τίτλο της σελίδας, τον τύπο των γραμμάτων, τις εικόνες, τους συνδέσμους, τους πίνακες και πολλά άλλα στοιχεία που συνθέτουν μια ιστοσελίδα ή εφαρμογή.

CSS

Το CSS (Cascading Style Sheets) είναι μια γλώσσα που καθορίζει τον τρόπο παρουσίασης του περιεχομένου που δημιουργείται με τη χρήση HTML.

Σχεδιάστηκε προκειμένου να διαχωρίσει την εμφάνιση από το περιεχόμενο και να παρέχει μεγαλύτερη ευελιξία και έλεγχο. Χρησιμοποιείται για τον έλεγχο της εμφάνισης μιας ιστοσελίδας ή μιας διαδικτυακής εφαρμογής.

Μέσω του CSS, μπορούμε να προσδιορίσουμε τα χρώματα, τις γραμματοσειρές, τα περιθώρια, τις διαστάσεις, τις διατάξεις και πολλά άλλα στοιχεία.

JavaScript

Η JavaScript αποτελεί μια από τις πιο δυναμικές και ευέλικτες γλώσσες προγραμματισμού. Είναι η γλώσσα που δίνει "ζωή" στις ιστοσελίδες και τις εφαρμογές διαδικτύου, προσθέτοντας διαδραστικότητα και λειτουργικότητα.

Με τη JavaScript υπάρχει η δυνατότητα να δημιουργηθούν δυναμικά στοιχεία, όπως αλληλεπιδραστικές φόρμες, κινούμενα γραφικά, συστήματα περιήγησης, και πολλά άλλα. Επιτρέπει την αλληλεπίδραση με τον χρήστη, όπως η επικοινωνία μέσω παραθύρων διαλόγου, η ανταπόκριση σε κλικ και πληκτρολογήσεις, και η δυνατότητα ανάκτησης και εμφάνισης δεδομένων.

Ένα από τα κύρια πλεονεκτήματα της JavaScript είναι η δυνατότητα λειτουργίας client-side, δηλαδή στον υπολογιστή του χρήστη, χωρίς την ανάγκη για συνεχή αλληλεπίδραση με τον διακομιστή (server), ενώ είναι ευρέως υποστηριζόμενη και διαθέσιμη σε κάθε σύγχρονο φυλλομετρητή (web browser).

PHP

Η PHP είναι μια δυναμική γλώσσα προγραμματισμού, ειδικά σχεδιασμένη για τη δημιουργία διαδικτυακών εφαρμογών. Είναι ένα από τα πιο δημοφιλή και ισχυρά εργαλεία για τη διαχείριση δυναμικών ιστοσελίδων και διακομιστών.

Μία από τις βασικές δυνατότητες της PHP είναι η δημιουργία διαδικτυακών σελίδων που ανταποκρίνονται δυναμικά στις αιτήσεις των χρηστών. Με τη χρήση PHP,

μπορούμε να δημιουργήσουμε φόρμες επικοινωνίας, συστήματα σύνδεσης χρηστών και πολλές άλλες λειτουργίες που κάνουν την εμπειρία των χρηστών πιο διαδραστική και λειτουργική.

Ένα άλλο σημαντικό χαρακτηριστικό της PHP είναι η δυνατότητα σύνδεσης με βάσεις δεδομένων, όπως MySQL, PostgreSQL και πολλές άλλες. Αυτό επιτρέπει την αποθήκευση, την ανάκτηση και τη διαχείριση δεδομένων, κάτι που είναι κρίσιμο για τη λειτουργία πολύπλοκων εφαρμογών και ιστοσελίδων.

MySQL

Η MySQL είναι ένα από τα πιο δημοφιλή και ισχυρά σχεσιακά συστήματα διαχείρισης βάσεων δεδομένων (RDBMS). Είναι ανοικτού κώδικα (open-source) και προσφέρει μια αξιόπιστη, αποδοτική και εύχρηστη λύση για την αποθήκευση, τη διαχείριση και την ανάκτηση δεδομένων.

Η βάση δεδομένων αποτελεί τον πυρήνα των περισσότερων εφαρμογών και ιστοσελίδων, αποθηκεύοντας πληροφορίες όπως χρήστες, προϊόντα και πολλά άλλα.

Η MySQL είναι γνωστή για την υψηλή της απόδοση και τη δυνατότητα να χειρίζεται μεγάλους όγκους δεδομένων και πολύπλοκες επερωτήσεις. Είναι γρήγορη και ανταποκρίνεται άμεσα στις επερωτήσεις των εφαρμογών, εξασφαλίζοντας μια ομαλή εμπειρία για τους χρήστες, ενώ έχει μια ενεργή κοινότητα προγραμματιστών και χρηστών που παρέχει υποστήριξη και αναβαθμίσεις συστηματικά.

Πέραν των βασικών τεχνολογιών που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής, χρησιμοποιήθηκαν επιπλέον βοηθητικές βιβλιοθήκες, APIs και τεχνικές, οι οποίες αναλύονται παρακάτω:

JQuery

Η JQuery είναι μια βιβλιοθήκη JavaScript σχεδιασμένη να απλοποιήσει την υλοποίηση σεναρίων (scripting) στην πλευρά του πελάτη (client-side) της HTML.

Χρησιμοποιήθηκε στην εφαρμογή προκειμένου να ενσωματωθούν πρόσθετα και να απλοποιηθεί ο κώδικας JavaScript.

AJAX

Η AJAX είναι μια τεχνική της JavaScript που χρησιμοποιείται προκειμένου να δημιουργούνται ασύγχρονα αιτήματα http. Αυτό επιτρέπει σε ιστοσελίδες και εφαρμογές να αλληλεπιδρούν με περιεχόμενο, χωρίς να ανανεώσουν τη σελίδα.

Χρησιμοποιήθηκε στην εφαρμογή προκειμένου να γίνουν ασύγχρονα αιτήματα http για την αποθήκευση δεδομένων του χρήστη, στη βάση δεδομένων.

Geolocation API

Το Geolocation API, χρησιμοποιείται για την ανάκτηση των πληροφοριών γεωγραφικής τοποθεσίας από μια συσκευή.

Χρησιμοποιήθηκε στην εφαρμογή προκειμένου να λαμβάνονται οι συνταγμένες του χρήστη, ώστε να προβάλλεται η θέση του χρήστη στο χάρτη σε πραγματικό χρόνο και να καταχωρούνται τα δεδομένα αυτά στη βάση δεδομένων.

Reverse Geocoding

Το Reverse Geocoding είναι η διαδικασία του εντοπισμού γεωγραφικών στοιχείων από γεωγραφικές συντεταγμένες.

Χρησιμοποιήθηκε στην εφαρμογή προκειμένου να γίνονται ως είσοδος οι συντεταγμένες του χρήστη και να μετατρέπονται στο όνομα της περιοχής που αντιστοιχούν αυτές οι συντεταγμένες.

OpenStreetMap

Το OpenStreetMap είναι ένας χάρτης με ελεύθερη άδεια ο οποίος αναπτύσσεται από μια κοινότητα εθελοντών που συνεισφέρουν και διατηρούν δεδομένα σε όλον τον κόσμο.

Χρησιμοποιήθηκε στην εφαρμογή για την ενσωμάτωση χαρτών.

Leaflet

Το Leaflet είναι μια βιβλιοθήκη JavaScript ανοιχτού κώδικα που χρησιμοποιείται για τη δημιουργία εφαρμογών χαρτογράφησης.

Χρησιμοποιήθηκε στην εφαρμογή για την αλληλεπίδραση με τα στοιχεία του χάρτη.

Markercluster

Το Markercluster είναι ένα πρόσθετο σε Javascript που αλλάζει τον τρόπο παρουσίασης των σημείων στο χάρτη και τα ομαδοποιεί με βάση την απόστασή τους.

Χρησιμοποιήθηκε στην εφαρμογή για την ομαδοποίηση των στοιχείων του χάρτη.

Sweetalert

Το Sweetalert είναι ένα πρόσθετο σε Javascript και CSS που αλλάζει τον τρόπο παρουσίασης του κλασικού alert της JavaScript. Επεκτείνει τις δυνατότητές του και παρουσιάζεται με την ίδια μορφή σε όλους τους browsers.

Χρησιμοποιήθηκε στην εφαρμογή για την προβολή των αναδυόμενων παραθύρων.

Tooltipster

Το Tooltipster είναι ένα πρόσθετο σε Javascript, JQuery και CSS που αλλάζει τον τρόπο παρουσίασης του κλασικού title της HTML. Επεκτείνει τις δυνατότητές του και παρουσιάζεται με την ίδια μορφή σε όλους τους browsers.

Χρησιμοποιήθηκε στην εφαρμογή για την προβολή των hover.

Fancybox

Το Fancybox είναι ένα πρόσθετο σε Javascript, JQuery και CSS που αλλάζει τον τρόπο παρουσίασης του κλασικού iframe.

Χρησιμοποιήθηκε στην εφαρμογή για να ανοίγουν άλλες σελίδες μέσω iframe, στην ίδια σελίδα.

Chart.js

Το Chart.js είναι ένα πρόσθετο σε Javascript που χρησιμοποιείται για την απεικόνιση γραφημάτων.

Χρησιμοποιήθηκε στην εφαρμογή για την απεικόνιση γραφημάτων που παρουσιάζουν τους καρδιακούς παλμούς.

ΚΕΦΑΛΑΙΟ 2

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΕΦΑΡΜΟΓΗΣ

Η παρούσα εφαρμογή βασίζεται στην αρχιτεκτονική πελάτη-εξυπηρετητή (Client-Server) και ποιο συγκεκριμένα στην αρχιτεκτονική τριών επιπέδων (Three-Tier Architecture).

Η αρχιτεκτονική τριών επιπέδων είναι ένα μοντέλο σχεδιασμού που διαχωρίζει τη λειτουργικότητα μιας εφαρμογής σε τρία κυρίως επίπεδα.

Το επίπεδο παρουσίασης (Presentation Layer), είναι το ανώτερο επίπεδο και ασχολείται με την αλληλεπίδραση με τον χρήστη και την παρουσίαση των δεδομένων. Περιλαμβάνει γραφικά περιβάλλοντα και άλλα στοιχεία που αφορούν την απεικόνιση των πληροφοριών.

Το επίπεδο επιχειρησιακής λογικής (Business Logic Layer), αποτελεί το μεσαίο επίπεδο που περιλαμβάνει τη λογική της εφαρμογής και την επεξεργασία των δεδομένων. Είναι υπεύθυνο για την επιχειρησιακή λογική της εφαρμογής, όπως οι κανόνες επιχειρησιακής διαδικασίας και οι υπολογισμοί.

Το επίπεδο δεδομένων (Data Layer), αποτελεί το χαμηλότερο επίπεδο και ασχολείται με την αποθήκευση και την ανάκτηση των δεδομένων. Περιλαμβάνει τις βάσεις δεδομένων και τη διαχείριση των αποθηκευμένων δεδομένων.

Κάθε επίπεδο λειτουργεί ανεξάρτητα από τα άλλα, επιτρέποντας την ανεξάρτητη ανάπτυξη, συντήρηση και ενημέρωση κάθε επιπέδου.

2.1 Διεπαφή χρήστη

Η διεπαφή της εφαρμογής είναι το σημείο επαφής ανάμεσα στον χρήστη και την εφαρμογή. Η ποιότητα της διεπαφής χρήστη είναι κρίσιμη για την επιτυχία μιας εφαρμογής, καθώς επηρεάζει την ευχρηστία και τη γενική εμπειρία χρήσης.

Ορισμένα από τα βασικά στοιχεία μιας καλής διεπαφής χρήστη σε μια διαδικτυακή εφαρμογή είναι τα εξής:

1. Η διάταξη πρέπει να είναι σαφής και ευανάγνωστη, με κατάλληλη οργάνωση των στοιχείων. Συχνά, η διάταξη ακολουθεί αρχές όπως η περιεκτικότητα (proximity), η συνοχή (consistency), και η ιεραρχία (hierarchy).
2. Η χρήση χρωμάτων πρέπει να είναι ευανάγνωστη και ευχάριστη για το μάτι. Το στυλ της διεπαφής πρέπει να αντανakλά τη γενική ταυτότητα και τη φιλοσοφία της εφαρμογής.
3. Η πλοήγηση πρέπει να είναι εύκολη και κατανοητή. Ο χρήστης πρέπει να μπορεί να περιηγείται στην εφαρμογή χωρίς σύγχυση.
4. Η διεπαφή πρέπει να προσαρμόζεται σε διάφορες συσκευές και μεγέθη οθονών.
5. Πρέπει να υπάρχουν κατάλληλα μηνύματα σφαλμάτων και οδηγίες για τον χρήστη σε περίπτωση προβλημάτων.

Όσον αφορά τη διεπαφή της παρούσας διαδικτυακής εφαρμογής, έγινε προσπάθεια ώστε να πληρούνται τα παραπάνω βασικά στοιχεία.

Αναλυτικότερα, η διάταξη της εφαρμογής είναι σαφής και ευανάγνωστη, με κατάλληλη οργάνωση των στοιχείων. Η εφαρμογή διαθέτει μια οριζόντια μπάρα με λευκό φόντο, η οποία εμφανίζεται σταθερά στο επάνω μέρος της οθόνης και περιέχει εικονίδια συντομεύσεων για το κεντρικό μενού και την αποσύνδεση του χρήστη.

Το κεντρικό μέρος της εφαρμογής εμφανίζεται σε γκρι φόντο, ενώ η περιοχή που περιέχει δεδομένα εμφανίζεται με λευκό φόντο.

Τα κουμπιά είναι σε μπλε φόντο, μεγάλα και ευανάγνωστα, ενώ παράλληλα περιέχουν και εικονίδια ώστε να γίνονται ευκολότερο κατανοητά από τον χρήστη.

Το στυλ της διεπαφής αντανακλά τη γενική ταυτότητα και τη φιλοσοφία της εφαρμογής, προβάλλοντας χάρτες για καταγραφή της διαδρομής του χρήστη και προβολή των διαδρομών του. Οι καρδιακοί παλμοί του χρήστη προβάλλονται με μορφή γραφημάτων και οι περισσότερες σελίδες της εφαρμογής μπορούν να προβάλλονται χωρίς να γίνεται είσοδος σε νέα σελίδα. Έτσι η πλοήγηση είναι εύκολη και κατανοητή και ο χρήστης μπορεί να περιηγείται χωρίς σύγχυση.

Παρ' όλο που η παρούσα εφαρμογή απευθύνεται σε κινητές συσκευές και κυρίως κινητά τηλέφωνα smart phones, η διεπαφή είναι σχεδιασμένη έτσι ώστε να μπορεί να προσαρμόζεται σε οποιαδήποτε συσκευή και μέγεθος οθόνης (responsive design).

Επίσης η εφαρμογή παρουσιάζει κατάλληλα μηνύματα σφαλμάτων και οδηγίες σε περίπτωση προβλημάτων.

Ορισμένα από αυτά είναι τα εξής:

1. Κατά την αποσύνδεση του χρήστη γίνεται ερώτηση επιβεβαίωσης από το σύστημα.
2. Κατά τη διαγραφή μιας διαδρομής γίνεται ερώτηση επιβεβαίωσης από το σύστημα.
3. Σε περίπτωση που δεν έχουν πραγματοποιηθεί διαδρομές, εμφανίζονται ενημερωτικά μηνύματα.
4. Σε περίπτωση που δεν έχουν καταχωριστεί δεδομένα, εμφανίζονται ενημερωτικά μηνύματα.
5. Σε περίπτωση επιτυχίας ή αποτυχίας καταχώρησης δεδομένων, εμφανίζονται ενημερωτικά μηνύματα.
6. Κατά την καταγραφή μιας διαδρομής το σύστημα παρέχει αναλυτικές οδηγίες σχετικά με τη χρήση της τοποθεσίας.
7. Κατά την καταχώριση δεδομένων το σύστημα προβάλλει την κατάστασή του.

Για την υλοποίηση της διεπαφής χρήστη χρησιμοποιήθηκαν οι τεχνολογίες που αναφέρθηκαν παραπάνω και ιδιαίτερα το CSS που καθορίζει τον τρόπο παρουσίασης του περιεχομένου. Τα εικονίδια που χρησιμοποιήθηκαν είναι σε μορφή svg, τα οποία είναι διανυσματικά γραφικά και έχουν την δυνατότητα να μην παραμορφώνονται σε αλλαγές του μεγέθους και της ανάλυσης οθόνης. Τα χρώματα

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

επιλέχτηκαν με κριτήριο την αντίθεση, ενώ χρησιμοποιήθηκε παράλληλα η ίδια χρωματική παλέτα. Επίσης το κριτήριο για τα μεγέθη των κουμπιών και των στοιχείων, ήταν έτσι ώστε να είναι εύκολα επιλέξιμα.

2.2 Δομή αρχείων

Η δομή αρχείων αφορά τον τρόπο οργάνωσης των αρχείων και των φακέλων που απαρτίζουν τον κώδικα, τις εικόνες, τα στυλ, και άλλους πόρους της εφαρμογής.

Μια καλή δομή αρχείων συμβάλλει στην ευκολία συντήρησης, επέκτασης, και συνεργασίας στην ανάπτυξη της εφαρμογής.

Στην παρούσα εφαρμογή η δομή των αρχείων συνοψίζεται ως εξής:

Κεντρικός κατάλογος

Ο κεντρικός κατάλογος της εφαρμογής περιέχει αρχεία php τα οποία αντιστοιχούν σε κάθε μια από τις σελίδες της εφαρμογής. Για παράδειγμα το αρχείο register.php περιέχει τη σελίδα η οποία γίνεται η καταχώρηση των στοιχείων του χρήστη για εγγραφή στην εφαρμογή.

Επίσης ο κεντρικός κατάλογος περιέχει τους φακέλους css, img, js, lib και uploads, οι όποιοι περιέχουν υποφακέλους και αρχεία.

Με την παραπάνω οργάνωση των αρχείων και των φακέλων μπορούμε να έχουμε μια σαφή δομή ανάπτυξης της εφαρμογής που συμβάλλει στην ευκολία συντήρησης και επέκτασης.

Παρακάτω αναλύονται οι βασικοί φάκελοι της εφαρμογής:

Φάκελος css

Ο φάκελος css περιέχει αρχεία css που αφορούν την μορφοποίηση για την παρουσίαση της εφαρμογής. Τα αρχεία css είναι κοινά για όλες τις σελίδες. Αυτό σημαίνει ότι σε περίπτωση που γίνει μια αλλαγή σε κάποιο από τα αρχεία css, η αλλαγή αυτή θα εφαρμοστεί σε όλες τις σελίδες της εφαρμογής.

Φάκελος img

Ο φάκελος `img` περιέχει αρχεία `svg` που είναι τα εικονίδια που χρησιμοποιούνται από την εφαρμογή. Τα εικονίδια αυτά είναι κατά την πλειοψηφία τους μαύρου χρώματος ενώ οι προκαθορισμένες διαστάσεις τους είναι να προβάλλονται σε πλήρες μέγεθος οθόνης. Η επεξεργασία τους για τα χρώματα και τις διαστάσεις, γίνονται από τα αρχεία `css`.

Φάκελος js

Ο φάκελος `js` περιέχει όλα τα αρχεία `javascript`, καθώς και υποφακέλους που περιέχουν πρόσθετα και βιβλιοθήκες `javascript`.

Ένα από τα βασικότερα αρχεία είναι το `scripts.js` που περιέχει όλες τις συναρτήσεις και τα `events javascript` της εφαρμογής.

Επίσης άλλα εξίσου βασικά αρχεία είναι το `jquery-2.1.0.js` που περιέχει την βιβλιοθήκη `jquery` και συγκεκριμένα την έκδοση 2.1.0, το `geolocation.js` που περιέχει τον κώδικα για τον εντοπισμό, την προβολή και την καταγραφή της τοποθεσίας του χρήστη, το `chart.js` που περιέχει τον κώδικα για την παρουσίαση των γραφημάτων και το `reverse_geolocation.js` που περιέχει τον κώδικα για την εύρεση διευθύνσεων από συντεταγμένες.

Φάκελος lib

Ο φάκελος `lib` περιέχει αρχεία `php` που χρησιμοποιούνται στην εφαρμογή παραπάνω από μια φορές. Δηλαδή είναι αρχεία που περιέχουν συναρτήσεις, μεταβλητές σταθερές, γραφικά, καθώς και τμήματα σελίδων που χρησιμοποιούνται σχεδόν σε όλες τις σελίδες της εφαρμογής και ο λόγος της διαφοροποίησής τους σε σχέση με τον κεντρικό κατάλογο είναι η επαναχρησιμοποίησή τους.

Κάποια από τα βασικότερα αρχεία του φακέλου `lib` είναι το `cfg.php` που περιέχει δεδομένα ρυθμίσεων, όπως στοιχεία της εφαρμογής, `path`, σύνδεση με τη βάση δεδομένων κ.α.

Το αρχείο `functions.php` που περιέχει όλες τις συναρτήσεις της εφαρμογής.

Το αρχείο `check_login.php` που ελέγχει αν έχει γίνει είσοδος χρήση και αυθεντικοποίηση στην εφαρμογή, καθώς και άλλα αρχεία που σκοπό έχουν την επαναχρησιμοποίηση στην εφαρμογή.

Φάκελος uploads

Ο φάκελος uploads περιέχει όλα τα αρχεία που ανεβάζει ο χρήστης κατά τη διάρκεια μιας διαδρομής.

Τα αρχεία αυτά είναι αρχεία φωτογραφιών και μπορεί να έχουν κατάληξη jpg, png, jpeg και gif.

2.3 Δομή κώδικα

Η δομή του κώδικα για την παρούσα διαδικτυακή εφαρμογή επιλέχθηκε να είναι με τη μέθοδο του διαδικαστικού προγραμματισμού (Procedural Programming), όπου ο κώδικας οργανώνεται σε συναρτήσεις, οι οποίες εκτελούν συγκεκριμένες εργασίες και κάθε συνάρτηση μπορεί να καλείται όταν απαιτείται η εκτέλεσή της.

Η επιλογή αυτής της μεθόδου έγινε με βάση για τους εξής λόγους:

Μικρή Κλίμακα

Η παρούσα διαδικτυακή εφαρμογή είναι μικρής κλίμακας και έχει δημιουργηθεί για εκπαιδευτικούς σκοπούς. Δεν απαιτούνται πολλά επίπεδα αφαίρεσης και δομικές περιπλοκότητες που συχνά χρειάζονται για μεγαλύτερες εφαρμογές.

Απλότητα και Κατανοητότητα

Ο διαδικαστικός προγραμματισμός είναι πιο απλός και ευανάγνωστος σε σχέση με άλλες προγραμματιστικές προσεγγίσεις. Για μια μικρή εφαρμογή, όπου η πολυπλοκότητα μπορεί να παραμείνει χαμηλή, η απλότητα του διαδικαστικού προγραμματισμού μπορεί να επιτρέψει στον προγραμματιστή να γράφει γρήγορα και αποτελεσματικά κώδικα.

Γρήγορη Υλοποίηση

Σε περιπτώσεις όπου η ταχεία υλοποίηση της εφαρμογής είναι προτεραιότητα, ο διαδικαστικός προγραμματισμός μπορεί να είναι πιο απευθείας εμπειρογνώμον.

Ευελιξία

Σε περιπτώσεις όπου η εφαρμογή δεν προβλέπεται να εξελιχθεί σε μεγάλη κλίμακα, σε επιστημονικά έργα ή πειράματα, όπου η προτεραιότητα είναι η υλοποίηση γρήγορων ιδεών, ο διαδικαστικός προγραμματισμός μπορεί να είναι κατάλληλος.

2.4 Περιγραφή της βάσης δεδομένων

Η βάση δεδομένων που επιλέχθηκε για την υλοποίηση της παρούσας διαδικτυακής εφαρμογής είναι η MySQL.

Η MySQL είναι ανοικτού κώδικα (open-source) σχεσιακή βάση δεδομένων και προσφέρει μια αξιόπιστη, αποδοτική και εύχρηστη λύση για την αποθήκευση, τη διαχείριση και την ανάκτηση δεδομένων. Είναι γρήγορη και ανταποκρίνεται άμεσα στα ερωτήματα των εφαρμογών, εξασφαλίζοντας μια ομαλή εμπειρία για τους χρήστες. Επίσης η MySQL συνδέεται εύκολα με την PHP, και αυτό αποτελεί ένα από τα δυνατά σημεία του συνδυασμού αυτών των δύο τεχνολογιών.

Η βάση δεδομένων της εφαρμογής περιέχει τους εξής τέσσερις πίνακες notes, route, routes, users οι οποίοι αναλύονται παρακάτω:

Πίνακας users

Ο πίνακας users χρησιμοποιείται για την αποθήκευση των χρηστών και περιλαμβάνει τα παρακάτω πεδία:

Όνομα	Τύπος	Σύνθεση
Id (Πρωτεύον κλειδί) AUTO_INCREMENT	int(11)	
username	text	utf8_unicode_ci
password	text	utf8_unicode_ci
u_ip	varchar(100)	utf8_unicode_ci
last_visit	datetime	
published	tinyint(1)	

Πίνακας 2.1: Πίνακας users

Ενδεικτικά δεδομένα του πίνακα users σε μορφή SQL:

```
(14, 'vas', '3b94452ef7291bd6b348d4ec1afacc793444587c', '169.254.130.1', '2023-09-27 11:22:45', 1);
```

Πίνακας route

Ο πίνακας route χρησιμοποιείται για την αποθήκευση ενός σημείου σε μια διαδρομή του χρήστη. Τα σημεία αυτά (μοναδικές εγγραφές) συνοψίζονται με κριτήρια το route_id και το user_id και παρουσιάζονται ως μια διαδρομή του χρήστη. Ο πίνακας route περιλαμβάνει τα παρακάτω πεδία:

Όνομα	Τύπος	Σύνθεση
id (Πρωτεύον κλειδί) AUTO_INCREMENT	bigint(25)	
user_id (Ευρετήριο)	int(11)	
route_id (Ευρετήριο)	int(11)	
lat	tinytext	utf8_unicode_ci
lon	tinytext	utf8_unicode_ci
date	datetime	
loc_name	text	utf8_unicode_ci

Πίνακας 2.2: Πίνακας route

Ενδεικτικά δεδομένα του πίνακα route σε μορφή SQL:

```
(685, 14, 139, '37.9189', '22.9798', '2023-04-18 22:37:03', 'Διακλάδωση Κυράς Βρύσης, Κοινότητα Ισθμίας, Δημοτική Ενότητα Λουτρακίου - Περαχώρας, Δήμος Λουτρακίου - Περαχώρας - Αγίων Θεοδώρων, Περιφερειακή Ενότητα Κορινθίας, Περιφέρεια Πελοποννήσου, Αποκεντρωμένη Διοίκηση Πελοποννήσου, Δυτικής Ελλάδας και Ιονίου, 200 10, Ελλάς'),
```

Πίνακας routes

Ο πίνακας routes χρησιμοποιείται για την αποθήκευση των συνολικώς διαδρομών του χρήστη. Αυτός ο πίνακας είναι βοηθητικός, χρησιμεύει ώστε να μην δημιουργούνται άσκοπα νέες εγγραφές διαδρομών καθώς και για ταχύτερη προσπέλαση των εγγραφών. Περιλαμβάνει τα παρακάτω πεδία:

Όνομα	Τύπος	Σύνθεση
id (Πρωτεύον κλειδί) AUTO_INCREMENT	bigint(25)	
user_id (Ευρετήριο)	int(11)	
date	datetime	
cords_num	int(11)	

Πίνακας 2.3: Πίνακας routes

Ενδεικτικά δεδομένα του πίνακα routes σε μορφή SQL:

```
(198, 14, '2023-09-13 16:27:33', 20),
```

Πίνακας notes

Ο πίνακας notes χρησιμοποιείται για την αποθήκευση περεταίρω δεδομένων για μια διαδρομή του χρήστη, όπως καρδιακοί παλμοί, σημειώσεις, φωτογραφίες και βίντεο. Περιλαμβάνει τα παρακάτω πεδία:

Όνομα	Τύπος	Σύνθεση
id (Πρωτεύον κλειδί) AUTO_INCREMENT	bigint(25)	
user_id (Ευρετήριο)	int(11)	
route_id (Ευρετήριο)	int(11)	
lat	tinytext	utf8_unicode_ci
lon	tinytext	utf8_unicode_ci
date	datetime	
bpm	int(11)	
notes	text	utf8_unicode_ci
image	text	utf8_unicode_ci
video	text	utf8_unicode_ci

Πίνακας 2.4: Πίνακας notes

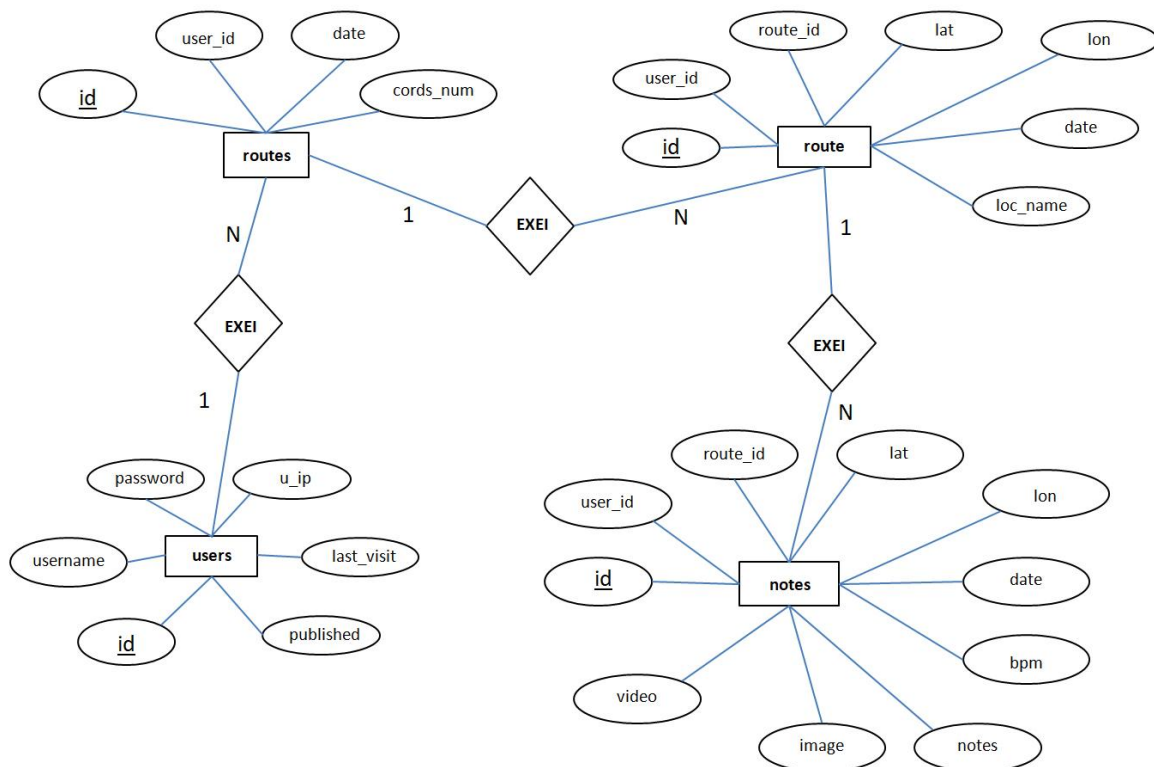
Ενδεικτικά δεδομένα του πίνακα notes σε μορφή SQL:

```
(130, 14, 198, '37.9588944', '23.6865261', '2023-09-13 16:28:29', 86, 'Ξεκίνησα ',  
'198_20230913_132829.jpg', ")
```

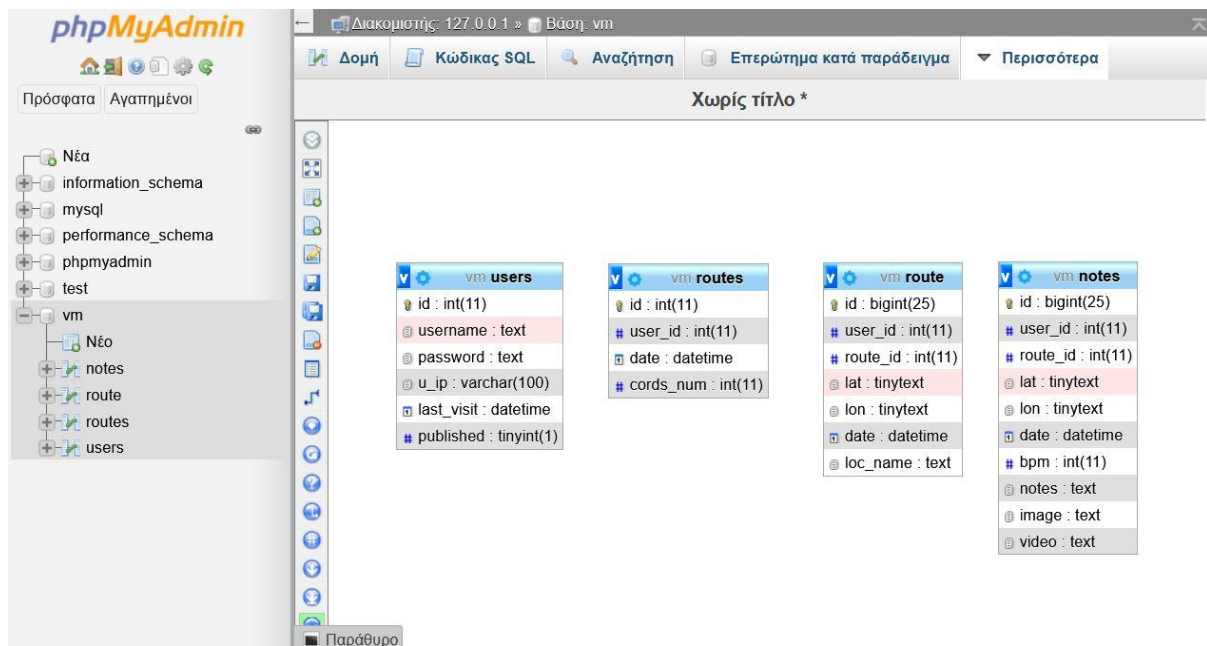
Μοντέλο Οντοτήτων-Συσχετίσεων

Το Μοντέλο Οντοτήτων-Συσχετίσεων (Entity-Relationship Model, ER Model) είναι ένα γραφικό μοντέλο που χρησιμοποιείται για τον σχεδιασμό βάσεων δεδομένων. Χρησιμοποιείται για την αναπαράσταση των σχέσεων μεταξύ διαφόρων στοιχείων σε ένα σύστημα βάσης δεδομένων.

Παρακάτω παρουσιάζεται το μοντέλο οντοτήτων συσχετίσεων.



Εικόνα 2.1: Μοντέλο Οντοτήτων-Συσχετίσεων



Εικόνα 2.2: Αναπαράσταση πινάκων στο phpMyAdmin

2.5 Ασφάλεια

Η ασφάλεια της διαδικτυακής εφαρμογής είναι ένα σημαντικό θέμα, καθώς μια επίθεση μπορεί να οδηγήσει σε μια σειρά από αρνητικές επιπλοκές, όπως δυσλειτουργία της εφαρμογής, διακοπή, απώλεια πρόσβασης, διαρροές δεδομένων και άλλα σοβαρά προβλήματα.

Παρακάτω αναλύονται κάποιες από τις μεθόδους που υλοποιήθηκαν για την ασφάλεια της παρούσας διαδικτυακής εφαρμογής.

Κρυπτογράφηση κωδικού πρόσβασης χρήστη

Κατά την εγγραφή ενός χρήστη στην εφαρμογή, ο κωδικός εισάγεται στη βάση δεδομένων κρυπτογραφημένος, χρησιμοποιώντας την συνάρτηση sha1() της php.

Η συνάρτηση sha1() είναι μια συνάρτηση κατακερματισμού (hash function) που χρησιμοποιείται για τον υπολογισμό της τιμής hash (κατακερματισμού) μιας συμβολοσειράς. Η τιμή hash είναι μια σταθερή μοναδική συμβολοσειρά που δημιουργείται από τα δεδομένα εισόδου.

Προστασία από Επιθέσεις SQL Injection και XSS

Για την αποφυγή της εκτέλεσης μη ασφαλών ερωτημάτων SQL και προστασία των εισόδων από επιθέσεις Cross-Site Scripting (XSS), χρησιμοποιήθηκε η συνάρτηση `htmlspecialchars()` σε κάθε είσοδο του χρήστη.

Η συνάρτηση `htmlspecialchars()` χρησιμοποιείται για τον περιορισμό του κινδύνου επιθέσεων XSS (Cross-Site Scripting) σε web εφαρμογές. Μετατρέπει κάποιους ειδικούς χαρακτήρες σε HTML entities, επιτρέποντας την ασφαλή εμφάνιση περιεχομένου σε μια σελίδα HTML.

Αυτό το μέτρο ασφαλείας είναι σημαντικό για την πρόληψη επιθέσεων XSS, καθώς εξασφαλίζει ότι το περιεχόμενο εμφανίζεται ως κείμενο στη σελίδα, όχι ως εκτελέσιμος κώδικας.

Επικύρωση δεδομένων (Validation)

Για την επικύρωση δεδομένων στην παρούσα εφαρμογή, χρησιμοποιήθηκε `javascript` για τον έλεγχο των δεδομένων σε επίπεδο πελάτη (client). Τα δεδομένα αυτά προέρχονται κυρίως από φόρμες καταχώρισης δεδομένων,

Επίσης χρησιμοποιήθηκε η `php` για τον έλεγχο των δεδομένων σε επίπεδο διακομιστή (server), έτσι ώστε αν «αποτύχει» ο έλεγχος των δεδομένων σε επίπεδο πελάτη (client), να μην επηρεαστεί η ασφάλεια της εφαρμογής.

Αποτροπή προβολής των αρχείων ενός φακέλου

Για την αποτροπή της προβολής των αρχείων ενός φακέλου χρησιμοποιήθηκε ένα αρχείο `index.html` που παρουσιάζει μια λευκή σελίδα.

Επικοινωνία με HTTPS

Έγινε εγκατάσταση ενός πιστοποιητικού TLS και συγκεκριμένα το Let's Encrypt.

Το HTTPS (Hypertext Transfer Protocol Secure) είναι η ασφαλής έκδοση του πρωτοκόλλου HTTP που χρησιμοποιείται για τη μετάδοση δεδομένων στον παγκόσμιο ιστό. Το HTTPS παρέχει ασφάλεια και απορρήτου στην επικοινωνία μεταξύ του πελάτη (όπως ένας web browser) και του διακομιστή.

Το HTTPS παρέχει κρυπτογράφηση, επαλήθευση ταυτότητας και ακεραιότητα των δεδομένων.

Ενημερωμένη έκδοση της γλώσσας php

Χρησιμοποιήθηκε η τελευταία ενημερωμένη έκδοση της γλώσσας προγραμματισμού php (8.0) η οποία διορθώνει προβλήματα ασφάλειας σε σχέση με παλαιότερες εκδόσεις.

2.6 Υποδομές

Οι υποδομές είναι το σύνολο των τεχνολογιών και των συστατικών που χρησιμοποιούνται για την υλοποίηση και τη λειτουργία μιας εφαρμογής.

Οι υποδομές είναι κρίσιμες για τη διασφάλιση της αποτελεσματικής λειτουργίας, απόδοσης και ασφάλειας μιας εφαρμογής.

Ορισμένα βασικά στοιχεία των υποδομών περιλαμβάνουν, εξυπηρετητές (servers), βάσεις δεδομένων, υπηρεσίες και άλλα. Ο συνδυασμός αυτών των υποδομών σχηματίζει το σύνολο των μέσων που απαιτούνται για την ανάπτυξη και την εκτέλεση μιας διαδικτυακής εφαρμογής.

Οι υποδομές που χρησιμοποιήθηκαν για την παρούσα διαδικτυακή εφαρμογή, χωρίζονται σε δυο κατηγορίες. Η πρώτη κατηγορία αφορά τη δοκιμαστική λειτουργία της εφαρμογής (test) κατά το στάδιο της ανάπτυξης, ενώ η δεύτερη κατηγορία αφορά την παραγωγική λειτουργία της εφαρμογής, κατά την οποία η εφαρμογή είναι διαθέσιμη προς χρήση. Παρακάτω αναλύονται οι υποδομές που χρησιμοποιήθηκαν και για τις δυο κατηγορίες.

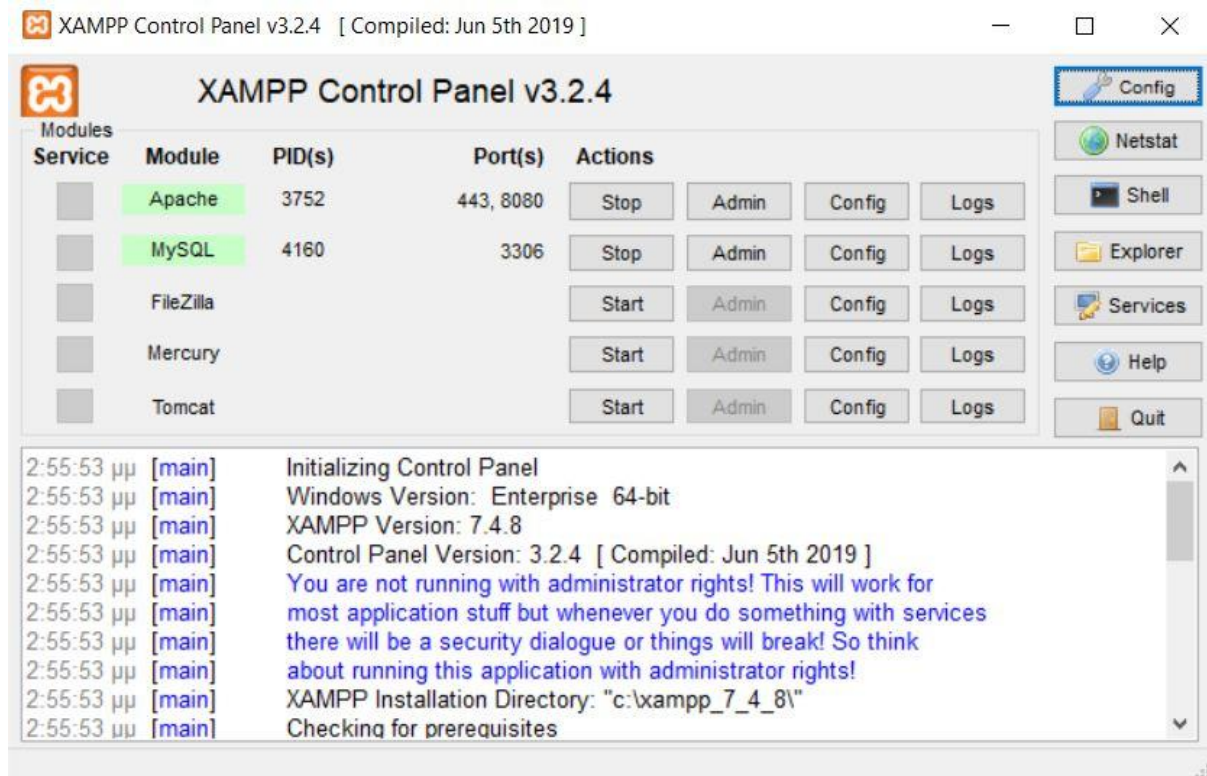
Υποδομές δοκιμαστικής λειτουργίας της εφαρμογής (test)

Οι υποδομές δοκιμαστικής λειτουργίας της εφαρμογής (test), χρησιμοποιήθηκαν στο αρχικό στάδιο της ανάπτυξης.

Ως βασική υποδομή χρησιμοποιήθηκε ένας προσωπικός υπολογιστής (pc) με λειτουργικό σύστημα windows 11 και το λογισμικό XAMPP για windows.

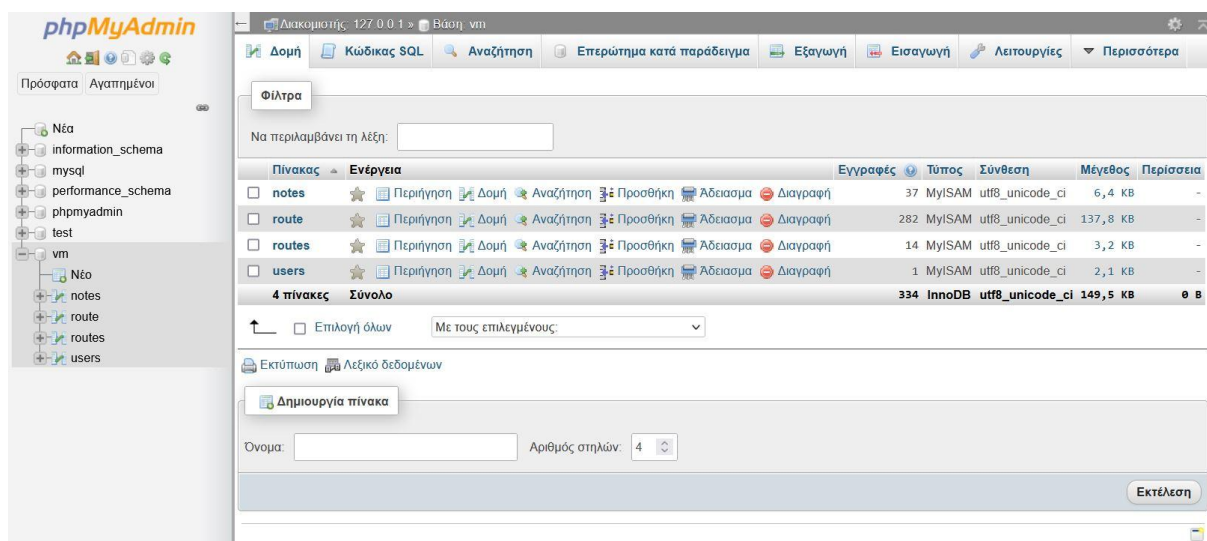
Το XAMPP είναι ένα ελεύθερο και ανοικτού κώδικα πακέτο λογισμικού που παρέχει έναν εύκολο τρόπο να εγκατασταθούν και να διαχειριστούν διάφορα λογισμικά σε έναν υπολογιστή για ανάπτυξη ιστοσελίδων και εφαρμογών. Τα βασικά συστατικά του XAMPP είναι Apache web server, MySQL και PHP.

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη



Εικόνα 2.3: το XAMPP σε λειτουργία

Επίσης το XAMPP ενσωματώνει το εργαλείο phpMyAdmin που παρέχεται ως μια διαδικτυακή εφαρμογή με γραφικό περιβάλλον διαχείρισης για τη MySQL και χρησιμοποιείται για πιο φιλικό προς τον χρήστη τρόπο διαχείρισης των βάσεων δεδομένων, όπως δημιουργία βάσεων δεδομένων, διαχείριση πινάκων και δεδομένων, εκτέλεση SQL ερωτημάτων, εισαγωγή και εξαγωγή Δεδομένων και άλλα.



Εικόνα 2.4: phpMyAdmin

Υποδομές παραγωγικής λειτουργίας (production)

Οι υποδομές παραγωγικής λειτουργίας (production), χρησιμοποιήθηκαν προκειμένου η εφαρμογή να δημοσιευτεί και να είναι διαθέσιμη προς χρήση στον παγκόσμιο ιστό, καθώς και να δοκιμαστεί η λειτουργία της σε πραγματικές συνθήκες.

Δημοσίευση της εφαρμογής στην υποδομή νέφους (Cloud), Microsoft Azure

Για τη δημοσίευση της εφαρμογής χρησιμοποιήθηκε η υποδομή νέφους (Cloud), Microsoft Azure, μέσω του φοιτητικού λογαριασμού που έχει παραχωρηθεί από το Πανεπιστήμιο.

Το Microsoft Azure είναι μια υπηρεσία cloud computing που προσφέρεται από τη Microsoft. Έχει σχεδιαστεί για να παρέχει μια ευέλικτη και κλιμακούμενη πλατφόρμα για την ανάπτυξη, τη διαχείριση και την εκτέλεση εφαρμογών μέσω του Internet.

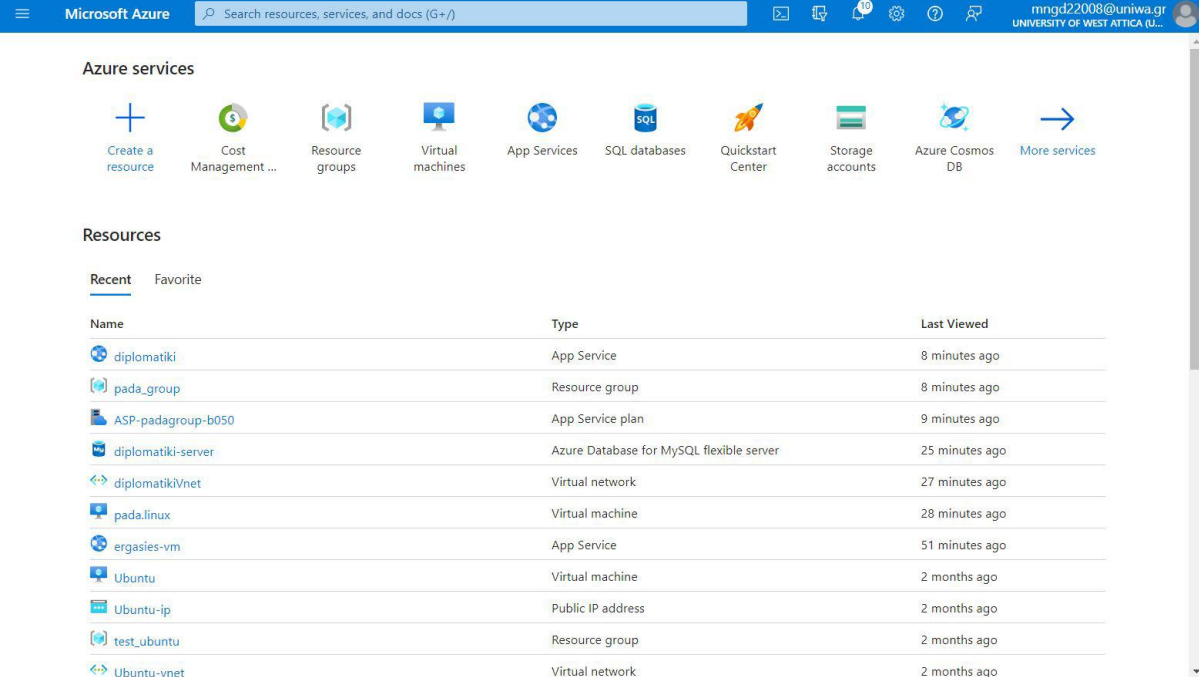
Οι υπηρεσίες Azure προσφέρουν λύσεις για ευρεία ποικιλία εφαρμογών, όπως web hosting, αποθήκευση δεδομένων, ανάλυση δεδομένων, τεχνητή νοημοσύνη, μηχανική μάθηση, ανάπτυξη εφαρμογών, και πολλές άλλες.

Ορισμένα βασικά χαρακτηριστικά του Microsoft Azure περιλαμβάνουν:

Υπηρεσίες Υποδομής ως Υπηρεσία (IaaS) που παρέχει εικονικές μηχανές, αποθήκευση και άλλες υποδομές, επιτρέποντας στις επιχειρήσεις να εκτελούν εφαρμογές χωρίς την ανάγκη να διατηρούν τις φυσικές υποδομές.

Πλατφόρμα ως Υπηρεσία (PaaS) που παρέχει περιβάλλοντα ανάπτυξης και εκτέλεσης για εφαρμογές χωρίς την ανάγκη για τις λεπτομέρειες της υποδομής.

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη



The screenshot shows the Microsoft Azure portal interface. At the top, there is a navigation bar with the Microsoft Azure logo, a search bar, and user information (mngd22008@uniwa.gr). Below the navigation bar, there are two main sections: "Azure services" and "Resources".

Azure services

- Create a resource
- Cost Management ...
- Resource groups
- Virtual machines
- App Services
- SQL databases
- Quickstart Center
- Storage accounts
- Azure Cosmos DB
- More services

Resources

Recent Favorite

Name	Type	Last Viewed
diplomatiki	App Service	8 minutes ago
pada_group	Resource group	8 minutes ago
ASP-padagroup-b050	App Service plan	9 minutes ago
diplomatiki-server	Azure Database for MySQL flexible server	25 minutes ago
diplomatiki/vnet	Virtual network	27 minutes ago
pada.linux	Virtual machine	28 minutes ago
ergasies-vm	App Service	51 minutes ago
Ubuntu	Virtual machine	2 months ago
Ubuntu-ip	Public IP address	2 months ago
test_ubuntu	Resource group	2 months ago
Ubuntu-vnet	Virtual network	2 months ago

Εικόνα 2.5: Microsoft Azure

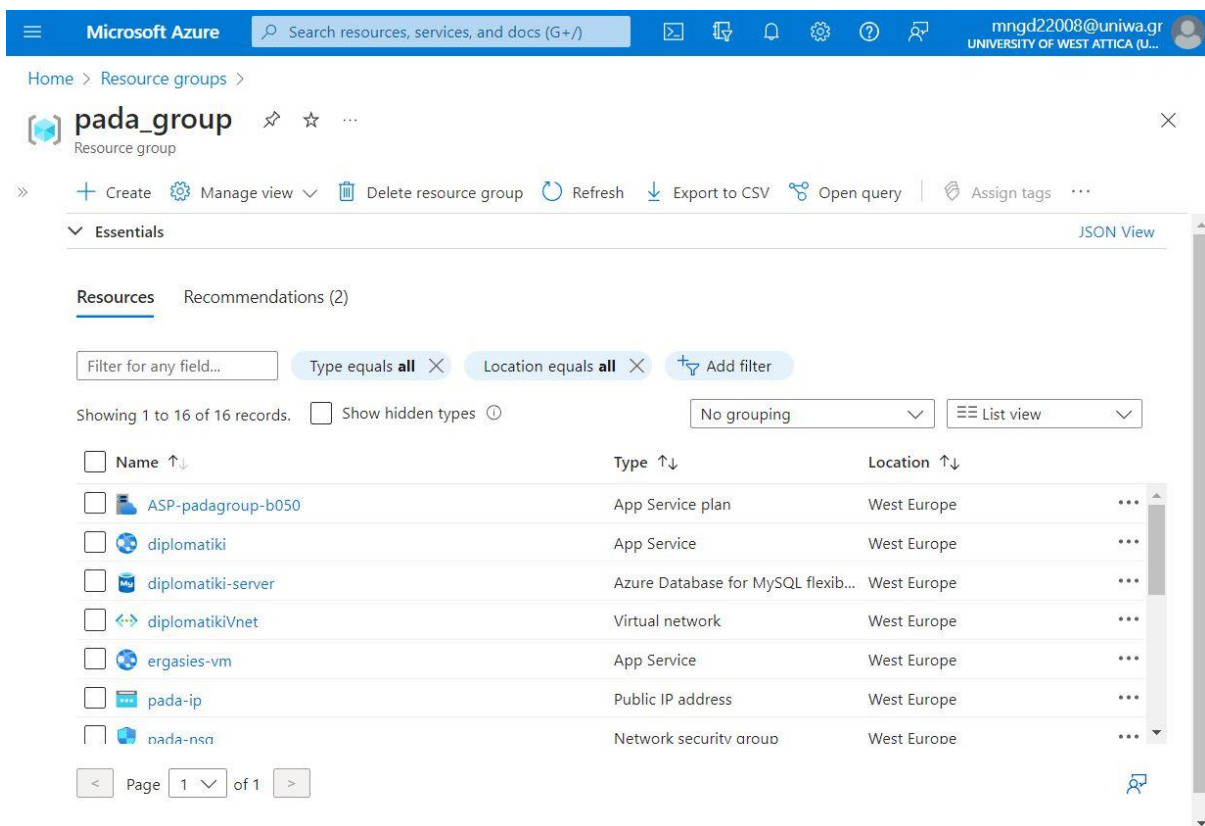
Η δημοσίευση της εφαρμογής στο Microsoft Azure αναλύεται παρακάτω:

Δημιουργία Resource group

Ένα Resource Group (Ομάδα Πόρων) είναι ένα λογικό κοντέινερ που χρησιμοποιείται για την οργάνωση, τη διαχείριση και την παρακολούθηση των πόρων που ανήκουν σε μια εφαρμογή.

Ένα Resource Group μπορεί να περιλαμβάνει όλους τους απαραίτητους πόρους για μια εφαρμογή.

Το Resource Group που δημιουργήθηκε ονομάστηκε pada_group κάτω από το οποίο πλαισιώνονται οι υπόλοιποι πόροι της εφαρμογής.



The screenshot shows the Microsoft Azure portal interface for a resource group named 'pada_group'. The page displays a list of resources under the 'Essentials' section. The resources are listed in a table with columns for Name, Type, and Location. The resources listed are:

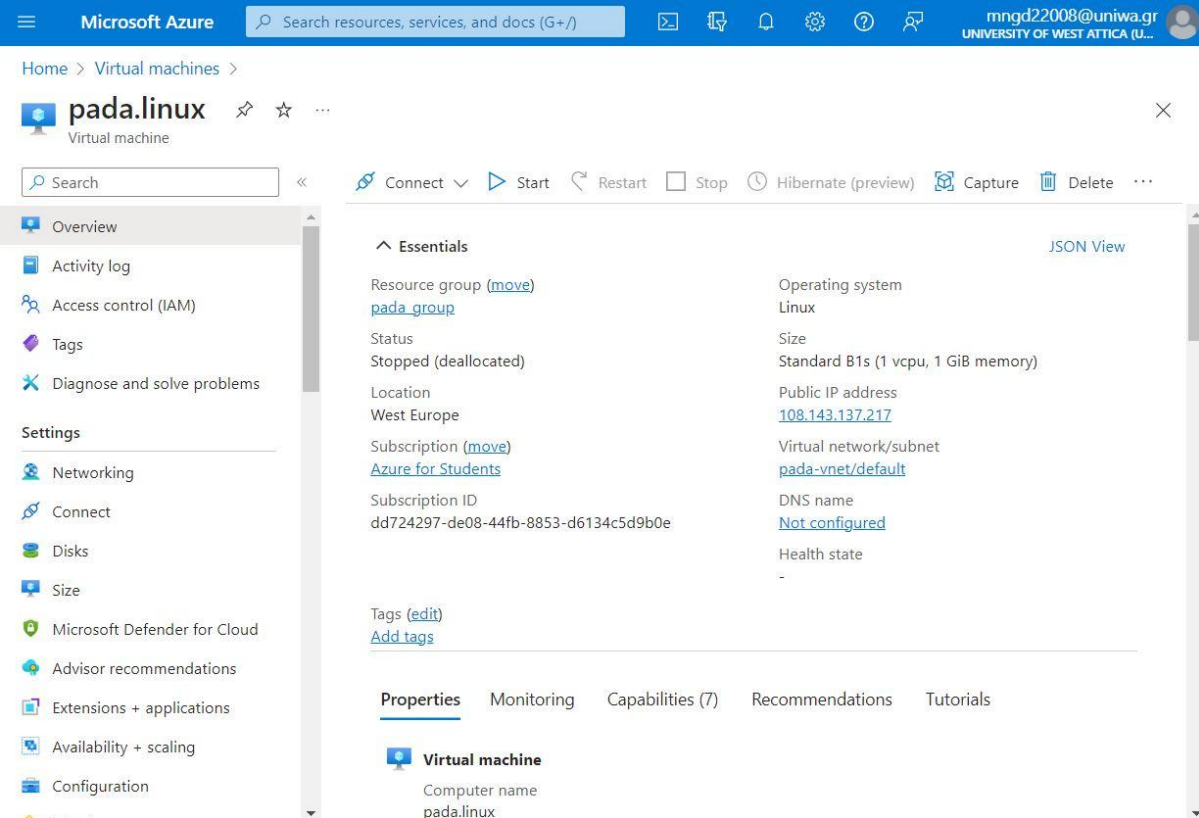
Name	Type	Location
ASP-padagroup-b050	App Service plan	West Europe
diplomatiki	App Service	West Europe
diplomatiki-server	Azure Database for MySQL flexib...	West Europe
diplomatikiVnet	Virtual network	West Europe
ergasies-vm	App Service	West Europe
pada-ip	Public IP address	West Europe
nada-nsd	Network security group	West Europe

Εικόνα 2.6: Resource group

Δημιουργία Virtual machine

Virtual Machine (Εικονική Μηχανή ή VM) είναι μια εικονική αναπαράσταση ενός φυσικού υπολογιστικού συστήματος, το οποίο εκτελεί λειτουργικό σύστημα. Οι VMs στο Azure παρέχουν ευελιξία για την εκτέλεση εφαρμογών και υπηρεσιών, χωρίς την ανάγκη αγοράς και συντήρησης φυσικού υλικού.

Το Virtual Machine που δημιουργήθηκε ονομάστηκε pada.linux, διαθέτει λειτουργικό σύστημα Linux 64bit, 1 vCPUs, 1 GiB RAM και Public IP address την 108.143.137.217.



The screenshot displays the Microsoft Azure portal interface. At the top, the 'Microsoft Azure' logo and a search bar are visible. The user's profile 'mngd22008@uniwa.gr' is shown in the top right corner. The main content area is titled 'pada.linux' and 'Virtual machine'. A navigation pane on the left lists various management options like 'Overview', 'Activity log', 'Access control (IAM)', and 'Settings'. The main panel shows the 'Essentials' section with a table of key properties:

Property	Value
Resource group	pada_group
Operating system	Linux
Status	Stopped (deallocated)
Size	Standard B1s (1 vcpu, 1 GiB memory)
Location	West Europe
Public IP address	108.143.137.217
Virtual network/subnet	pada-vnet/default
DNS name	Not configured
Health state	-

Below the table, there are sections for 'Tags' and 'Properties'. The 'Virtual machine' section shows the computer name as 'pada.linux'.

Εικόνα 2.7: Virtual machine

Δημιουργία Virtual network

Στο Microsoft Azure, ένα Virtual Network (Εικονικό Δίκτυο) είναι μια υπηρεσία που επιτρέπει τη δημιουργία ιδιωτικών δικτύων στον χώρο του cloud.

Τα Virtual Networks στο Azure είναι σχεδιασμένα για τη δημιουργία ενός ασφαλούς και ιδιωτικού περιβάλλοντος δικτύου.

Το Virtual Network που δημιουργήθηκε ονομάστηκε diplomatikiVnet.

The screenshot displays the Azure portal interface for a Virtual Network. The top navigation bar includes the Microsoft Azure logo, a search bar, and user information for 'mngd22008@uniwa.gr'. The breadcrumb trail shows 'Home > Resource groups > pada_group > diplomatikiVnet'. The main content area is divided into a left-hand navigation pane and a main configuration pane. The navigation pane lists various settings such as 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Address space', 'Connected devices', 'Subnets', 'Bastion', 'DDoS protection', 'Firewall', 'Microsoft Defender for Cloud', 'Network manager', and 'DNS servers'. The main pane shows the 'Essentials' section with a table of properties:

Property	Value
Resource group	pada_group
Location	West Europe
Subscription	Azure for Students
Subscription ID	dd724297-de08-44fb-8853-d6134c5d9b0e
Address space	10.0.0.0/16
DNS servers	Azure provided DNS service
Flow timeout	Configure
BGP community string	Configure
Virtual network ID	cd08f04e-f3ca-4cd2-b255-b635a18e1a39

Below the table, there are tabs for 'Topology', 'Properties', 'Capabilities (5)', 'Recommendations', and 'Tutorials'. The 'Capabilities' tab is active, showing two recommendations: 'DDoS protection' and 'Azure Firewall'.

Εικόνα 2.8: Virtual network

Δημιουργία Web App

Το Web App είναι μια υπηρεσία που παρέχει μια πλήρη πλατφόρμα για τη δημιουργία, την ανάπτυξη και τη λειτουργία web εφαρμογών και υπηρεσιών. Παρέχει μια απλή διαδικασία ανάπτυξης, διαχείρισης και κλιμάκωσης των εφαρμογών, που εστιάζει στην κατασκευή της λειτουργικότητας των εφαρμογών αντί της διαχείρισης της υποδομής.

Το Web App που δημιουργήθηκε ονομάστηκε diplomatiki, διαθέτει nginx web server, τη γλώσσα php έκδοση 8.0 και domain diplomatiki.azurewebsites.net.

The screenshot displays the Microsoft Azure portal interface for a Web App named 'diplomatiki'. The top navigation bar includes the Microsoft Azure logo, a search bar, and the user profile 'mngd22008@uniwa.gr UNIVERSITY OF WEST ATTICA (U...)'.

The breadcrumb trail shows: Home > Resource groups > pada_group > diplomatiki (Web App).

The left-hand navigation pane lists various management options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Microsoft Defender for Cloud, Events (preview), Deployment (Deployment slots, Deployment Center), and Settings (Configuration, Authentication, Application Insights, Identity, Backups).

The main content area shows the 'Essentials' section with the following details:

- Resource group: [pada_group](#)
- Status: Admin disabled
- Location: [West Europe](#)
- Subscription: [Azure for Students](#)
- Subscription ID: dd724297-de08-44fb-8853-d6134c5d9b0e
- Tags: [Add tags](#)

The 'Properties' section is expanded, showing:

- Default domain: [diplomatiki.azurewebsites.net](#)
- App Service Plan: ASP-padagroup-b050
- Operating System: Linux
- Health Check: Error fetching health check data. Please try again

Below the Properties section, there are tabs for Monitoring, Logs, Capabilities, Notifications, and Recommendations. The 'Web app' section is also visible, showing:

- Name: diplomatiki
- Publishing model: Code
- Runtime Stack: Php - 8.0

Εικόνα 2.9: Web App

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Αρχεία κώδικα

Τα αρχεία που περιέχουν τον κώδικα της εφαρμογής βρίσκονται στον κόμβο `waws-prod-am2-659.ftp.azurewebsites.windows.net`.

Για την αποστολή των αρχείων χρησιμοποιήθηκε το λογισμικό FileZilla.

Το FileZilla είναι ένα δωρεάν, ανοικτού κώδικα λογισμικό πελάτη FTP (File Transfer Protocol) που χρησιμοποιείται για τη μεταφορά αρχείων μεταξύ ενός υπολογιστή και ενός διακομιστή στο Διαδίκτυο. Επιτρέπει στους χρήστες να ανεβάζουν ή να κατεβάζουν αρχεία από ή προς έναν διακομιστή με ασφαλή τρόπο.

Το FileZilla παρέχει μια γραφική διεπαφή που καθιστά εύκολη τη χρήση του.

Οι βασικές λειτουργίες περιλαμβάνουν τη μεταφορά αρχείων, τη διαχείριση φακέλων, και την υποστήριξη για ασφαλείς συνδέσεις μέσω SSL/TLS. Το FileZilla είναι διαθέσιμο για πολλά λειτουργικά συστήματα, συμπεριλαμβανομένων των Windows, macOS και Linux.

Τέλος το domain name που τρέχει η εφαρμογή είναι το παρακάτω:

<https://diplomatiki.azurewebsites.net/>

Βάση δεδομένων

Η βάση δεδομένων που χρησιμοποιεί η εφαρμογή, εγκαταστάθηκε σε μια υποδομή Shared Hosting, ώστε να είναι ανεξάρτητη από την υποδομή νέφους (Cloud), Microsoft Azure. Ως βάση δεδομένων χρησιμοποιήθηκε η MariaDB.

Η MariaDB είναι ένα Σύστημα Διαχείρισης Σχεσιακών Βάσεων Δεδομένων (ΣΔΣΒΔ) που βασίζεται στην MySQL. Όπως και η MySQL, η MariaDB είναι ανοιχτό και ελεύθερο λογισμικό, προσφέροντας πρόσβαση στον πηγαίο κώδικα για τους χρήστες. Η κοινότητα της MariaDB είναι ενεργή και συνεισφέρει στη συνεχή ανάπτυξη και υποστήριξη του συστήματος. Η MariaDB διατηρεί υψηλή συμβατότητα με τη MySQL, προσφέροντας στους χρήστες τη δυνατότητα να μεταβούν από τη MySQL στη MariaDB χωρίς ιδιαίτερες αλλαγές στον κώδικα.

Τα στοιχεία που αφορούν τη βάση δεδομένων παρουσιάζονται παρακάτω:

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Διακομιστής βάσης δεδομένων

Διακομιστής: server.linux87.paraki.gr:3306

Τύπος διακομιστή: MariaDB

Έκδοση διακομιστή: 10.4.30-MariaDB-cll-lve - MariaDB Server

Έκδοση πρωτοκόλλου: 10

Διακομιστής ιστού

nginx/1.24.0

Έκδοση πελάτη βάσης δεδομένων: libmysql - mysqlnd 8.2.11

Έκδοση PHP: 8.2.11

Για τη διαχείριση της βάσης δεδομένων χρησιμοποιήθηκε το εργαλείο phpMyAdmin, το οποίο παρουσιάστηκε παραπάνω.

Δημοσίευση σε υποδομή hosting

Πέραν της υποδομής νέφους (Cloud), Microsoft Azure, η εφαρμογή δημοσιεύτηκε επίσης σε μια υποδομή Shared Hosting, σε έναν κοινό διακομιστή.

Βασικός λόγος της δημοσίευσης της εφαρμογής σε δευτερεύουσα υποδομή ήταν η διαθεσιμότητα. Η υποδομή νέφους (Cloud), Microsoft Azure παρέχεται δωρεάν μέσω του φοιτητικού λογαριασμού που έχει παραχωρηθεί από το Πανεπιστήμιο. Παρ' όλα αυτά η διάρκεια και η διαθεσιμότητα της υπηρεσίας είναι συγκεκριμένη και περιορισμένη. Σε περίπτωση λήξης της υπηρεσίας νέφους η εφαρμογή θα συνεχίσει να λειτουργεί σε ένα διαφορετικό domain name με τα ίδια δεδομένα.

Ορισμένα από τα βασικά χαρακτηριστικά της υποδομής hosting είναι τα εξής:

Λειτουργικό σύστημα: Linux redhat

Web Server: Apache

Έκδοση PHP: 8.0.30

Server API: CGI/FastCGI

Πιστοποιητικό TLS: Lets Encrypt

Τέλος το domain name που τρέχει η εφαρμογή είναι το παρακάτω:

<https://vm.arq.gr>

ΚΕΦΑΛΑΙΟ 3

ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ ΕΦΑΡΜΟΓΗΣ

Το εγχειρίδιο χρήσης της εφαρμογής περιέχει οδηγίες και πληροφορίες που βοηθούν τους χρήστες να κατανοήσουν και να χρησιμοποιήσουν την συγκεκριμένη εφαρμογή. Σκοπός του εγχειριδίου είναι να παρέχει σαφείς κατευθύνσεις σχετικά με το πώς να χρησιμοποιηθούν οι λειτουργίες της εφαρμογής.

3.1 Είσοδος στην εφαρμογή

Για είσοδο στην εφαρμογή, απαιτείται η χρήση ενός προγράμματος περιήγησης ιστού (browser), όπως Google Chrome, Mozilla Firefox, Microsoft Edge, Safari και άλλα.

Ο χρήστης εισάγει στη γραμμή διευθύνσεων του προγράμματος περιήγησης την παρακάτω διεύθυνση:

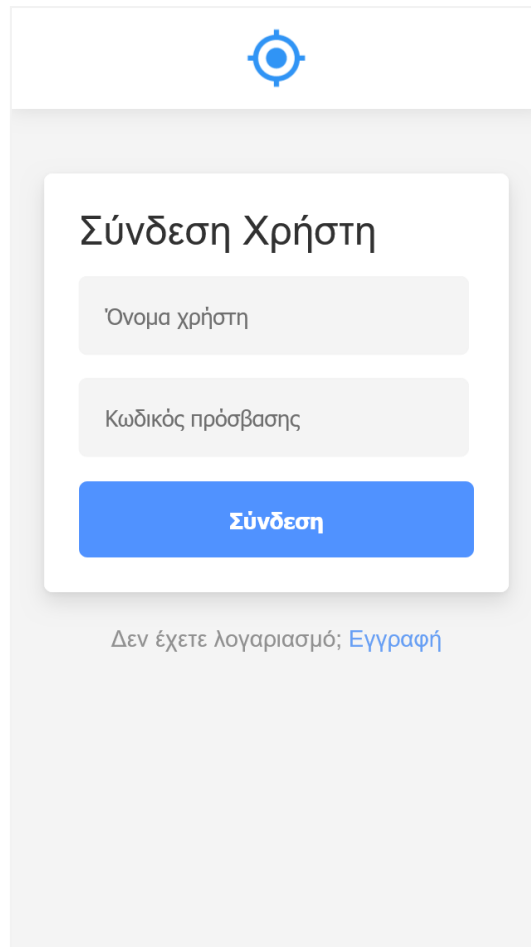
<https://diplomatiki.azurewebsites.net/> ή <https://vm.arq.gr/>

Η πρώτη οθόνη που προβάλλεται στον χρήστη, είναι η σελίδα σύνδεσης του χρήστη όπως εμφανίζεται στην Εικόνα 3.1.

Όταν ο χρήστης χρησιμοποιεί την εφαρμογή για πρώτη φορά, χρειάζεται να πραγματοποιήσει μια εγγραφή, ώστε να δημιουργηθεί ένας λογαριασμός που θα καταγράφονται τα δεδομένα της δραστηριότητάς του.

Για να γίνει αυτό, ο χρήστης θα πρέπει να επιλέξει τον σύνδεσμο Εγγραφή που βρίσκεται κάτω από τη φόρμα σύνδεσης και στη συνέχεια να συμπληρώσει τα στοιχεία του. Η διαδικασία αυτή αναλύεται στο κεφάλαιο 3.2 Εγγραφή Χρήστη.

Σε περίπτωση που ο χρήστης έχει πραγματοποιήσει εγγραφή, τότε θα πρέπει να συμπληρώσει τα στοιχεία σύνδεσης, ώστε να μεταβεί στο κύριο μέρος της εφαρμογής. Η διαδικασία αυτή αναλύεται στο κεφάλαιο 3.3 Σύνδεση Χρήστη.



Σύνδεση Χρήστη

Όνομα χρήστη

Κωδικός πρόσβασης

Σύνδεση

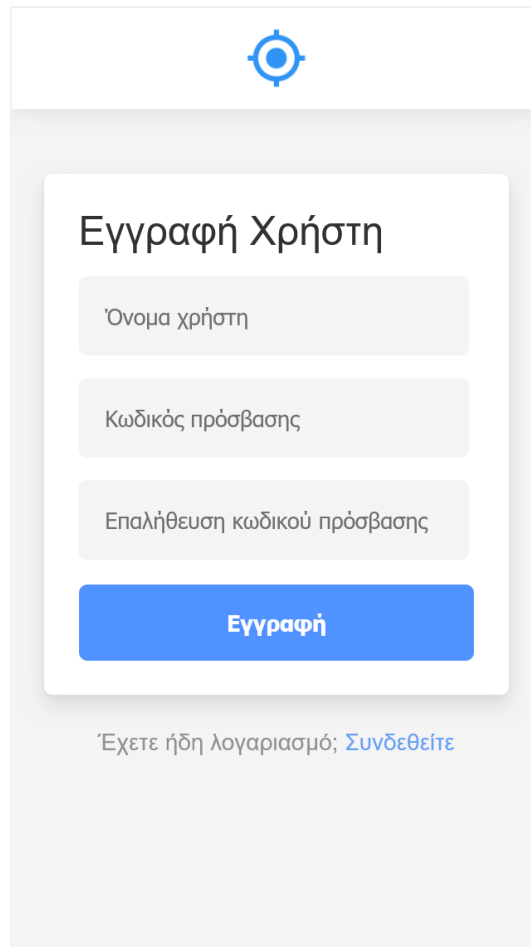
Δεν έχετε λογαριασμό; [Εγγραφή](#)

Εικόνα 3.1: Πρώτη οθόνη της εφαρμογής

3.2 Εγγραφή Χρήστη

Η εγγραφή χρήστη, χρησιμοποιείται προκειμένου ο χρήστης να διαθέτει προσωπικό λογαριασμό, ώστε να καταγράφονται τα δεδομένα της δραστηριότητάς του.

Στη σελίδα εγγραφής χρήστη, εμφανίζεται μια φόρμα, όπου ο χρήστης εισάγει ένα όνομα και έναν κωδικό πρόσβασης.



Εγγραφή Χρήστη

Όνομα χρήστη

Κωδικός πρόσβασης

Επαλήθευση κωδικού πρόσβασης

Εγγραφή

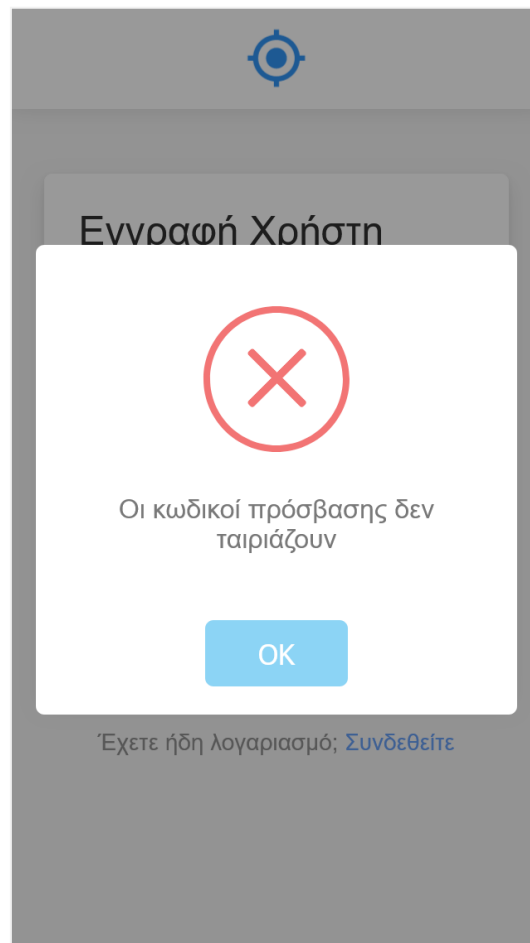
Έχετε ήδη λογαριασμό; [Συνδεθείτε](#)

Εικόνα 3.2: Οθόνη εγγραφής χρήστη

Τα στοιχεία αυτά έχουν κάποιους περιορισμούς κατά την καταχώρηση, όπως ότι το όνομα χρήστη πρέπει να περιέχει τουλάχιστον 3 χαρακτήρες, ενώ ο κωδικός πρόσβασης πρέπει να περιέχει τουλάχιστον 6 χαρακτήρες. Επίσης ο κωδικός πρόσβασης πρέπει να εισαχθεί από τον χρήστη εκ νέου στο πεδίο Επαλήθευση κωδικού πρόσβασης, προκειμένου να διασφαλιστεί ότι ο χρήστης δεν έχει λάθος κατά την καταχώρηση του κωδικού πρόσβασης.

Σε περίπτωση που ο χρήστης δεν έχει ακολουθήσει τους παραπάνω περιορισμούς, κατά την υποβολή της φόρμας από το κουμπί Εγγραφή, το σύστημα θα εμφανίσει μηνύματα πρόληψης σφαλμάτων και δεν θα καταχωρίσει τα στοιχεία έως ότου γίνει η διόρθωσή τους από τον χρήστη.

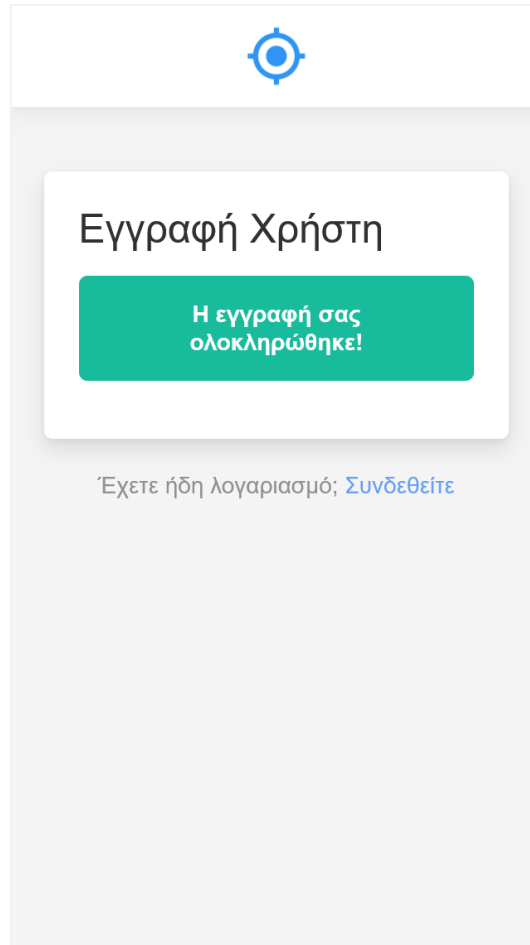
Στην Εικόνα 3.3 εμφανίζεται ένα ενδεικτικό μήνυμα πρόληψης σφάλματος, στην περίπτωση που ο χρήστης δεν έχει καταχωρήσει τον ίδιο κωδικό πρόσβασης και στα δυο πεδία.



Εικόνα 3.2.1: Μήνυμα πρόληψης σφάλματος

Κατά την ορθή καταχώρηση των στοιχείων, αυτά αποθηκεύονται, εμφανίζεται αντίστοιχο ενημερωτικό μήνυμα και στη συνέχεια γίνεται αυτόματη μετάβαση στη σελίδα σύνδεσης χρήστη, όπου ο χρήστης πλέον μπορεί να συνδεθεί στην εφαρμογή, συμπληρώνοντας τα στοιχεία που καταχώρησε κατά την εγγραφή του.

Στην Εικόνα 3.4 εμφανίζεται η οθόνη που περιέχει ενημερωτικό μήνυμα επιτυχούς εγγραφής του χρήστη.

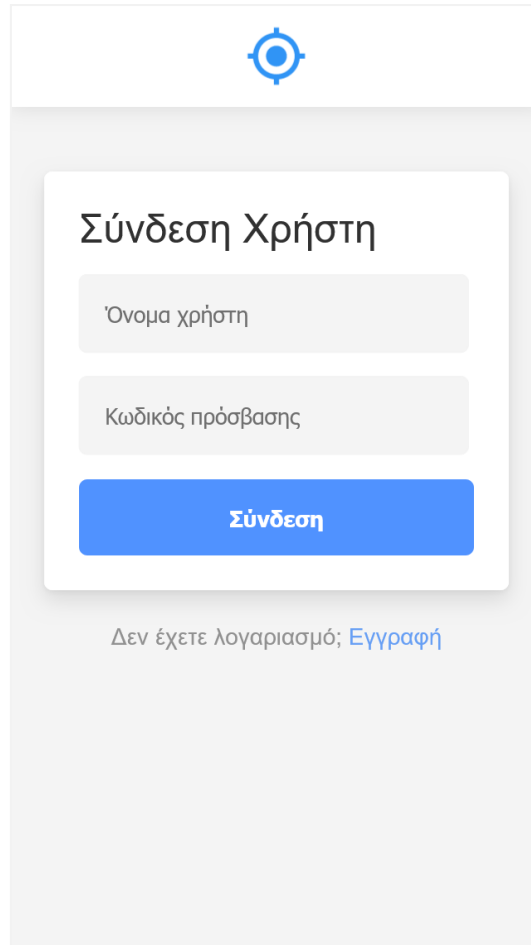


Εικόνα 3.2.2: Μήνυμα ολοκλήρωσης εγγραφής χρήστη

3.3 Σύνδεση Χρήστη

Στη σελίδα σύνδεσης χρήστη, εμφανίζεται μια φόρμα, όπου ο χρήστης πρέπει να εισάγει το όνομα και τον κωδικό πρόσβασης, προκειμένου να μεταβεί στο κύριο μέρος της εφαρμογής.

Στην Εικόνα 3.5 εμφανίζεται η οθόνη που περιέχει τη φόρμα σύνδεσης χρήστη.



Σύνδεση Χρήστη

Όνομα χρήστη

Κωδικός πρόσβασης

Σύνδεση

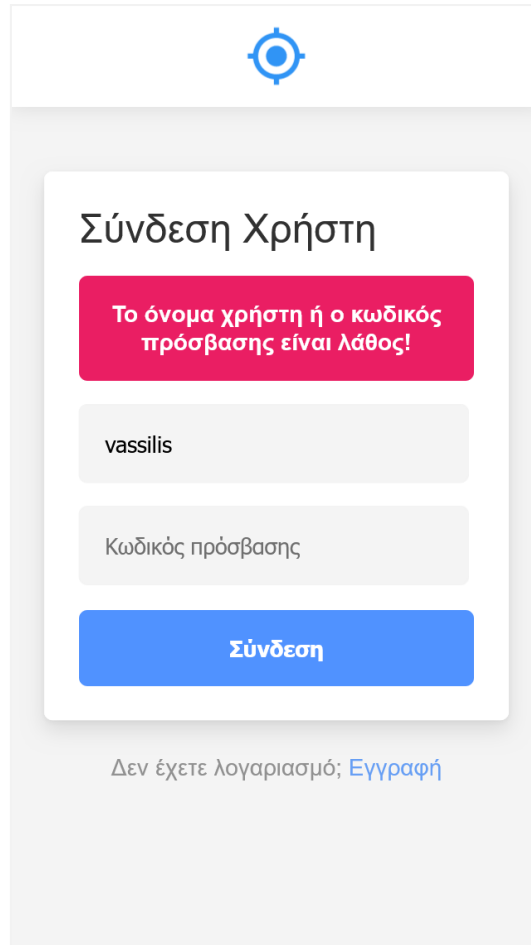
Δεν έχετε λογαριασμό; [Εγγραφή](#)

Εικόνα 3.3: Οθόνη σύνδεσης χρήστη

Κατά την εισαγωγή των στοιχείων και την υποβολή της φόρμας από το κουμπί Σύνδεση, το σύστημα αρχικά ελέγχει αν έχουν εισαχθεί δεδομένα από τον χρήστη. Σε περίπτωση που δεν έχουν εισαχθεί δεδομένα, το σύστημα θα εμφανίσει ενημερωτικά μηνύματα στον χρήστη.

Σε περίπτωση που ο χρήστης έχει εισάγει λάθος στοιχεία σύνδεσης, το σύστημα θα εμφανίσει μήνυμα σφάλματος, ενημερώνοντας τον χρήστη ότι έχει καταχωρήσει λάθος στοιχεία.

Στην Εικόνα 3.6 εμφανίζεται το μήνυμα λάθους έπειτα από καταχώρηση μη έγκυρων στοιχείων του χρήστη.



Εικόνα 3.3.1: Μήνυμα λάθους σε καταχώρηση στοιχείων

Κατά την ορθή εισαγωγή των στοιχείων, ο χρήστης θα μεταβεί αυτόματα στο κεντρικό μέρος της εφαρμογής όπου εμφανίζεται το κεντρικό μενού επιλογών, το οποίο αναλύεται στο κεφάλαιο 3.5 Κεντρικό μενού επιλογών.

3.4 Μενού επιλογών κορυφής

Το Μενού επιλογών κορυφής βρίσκεται σταθερά και μόνιμα στο πάνω μέρος της οθόνης και εμφανίζεται σε όλες τις σελίδες της εφαρμογής.



Εικόνα 3.4: Μενού επιλογών κορυφής

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

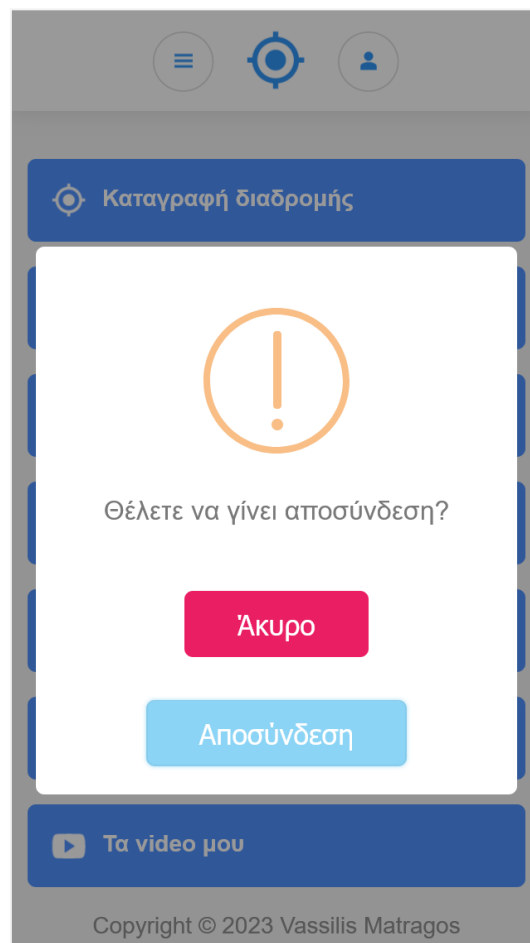
Το Μενού επιλογών κορυφής αποτελείται από τρία στοιχεία.

Το πρώτο στοιχείο εμφανίζεται αριστερά με ένα εικονίδιο που περιλαμβάνει τρεις οριζόντιες γραμμές. Το στοιχείο αυτό υποδηλώνει το κεντρικό μενού επιλογών της εφαρμογής και ο χρήστης μπορεί να μεταβεί σε αυτό από οποιαδήποτε σελίδα και αν βρίσκεται.

Το δεύτερο στοιχείο εμφανίζεται στο κέντρο με ένα εικονίδιο που ορίζει την τοποθεσία. Το στοιχείο αυτό δεν διαθέτει καμιά λειτουργικότητα και έχει χρησιμοποιηθεί ως ένα είδους λογότυπο που συμβολίζει το είδος της εφαρμογής.

Το τρίτο στοιχείο εμφανίζεται δεξιά με ένα εικονίδιο που περιλαμβάνει ένα άτομο. Το στοιχείο αυτό υποδηλώνει τον χρήστη που έχει συνδεθεί στην εφαρμογή.

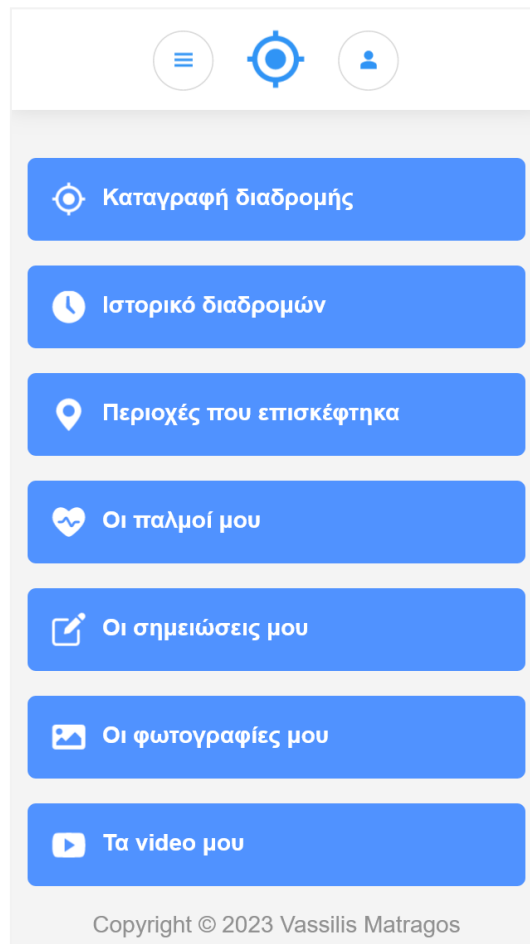
Το στοιχείο αυτό χρησιμοποιείται για την αποσύνδεση του χρήστη από την εφαρμογή (logout). Επιλέγοντας το στοιχείο αυτό, εμφανίζεται ένα παράθυρο διαλόγου, όπου ο χρήστης έχει τη δυνατότητα επιβεβαίωσης.



Εικόνα 3.4.1: Παράθυρο διαλόγου αποσύνδεσης χρήστη

3.5 Κεντρικό μενού επιλογών

Το κεντρικό μενού επιλογών είναι η κεντρική σελίδα της εφαρμογής και περιέχει όλα τα βασικά στοιχεία που συνθέτουν την εφαρμογή. Αντιπροσωπεύει τη διάταξη των διαφορετικών επιλογών και λειτουργιών που είναι διαθέσιμες για τον χρήστη. Στην Εικόνα 3.9 εμφανίζεται το κεντρικό μενού επιλογών.



Εικόνα 3.5: Κεντρικό μενού επιλογών

Στα κεφάλαια που ακολουθούν περιγράφονται αναλυτικά όλες οι σελίδες που συνθέτουν το κεντρικό μενού επιλογών.

3.6 Καταγραφή διαδρομής

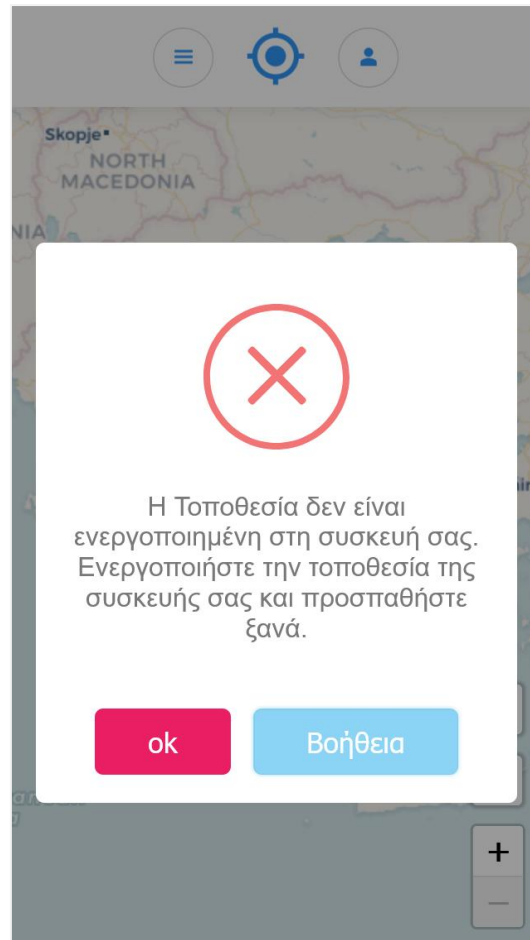
Η σελίδα καταγραφής διαδρομής, αποτελεί το βασικότερο μέρος της εφαρμογής, καθώς από εκεί προκύπτουν τα βασικά δεδομένα που συνθέτουν τα υπόλοιπα μέρη της εφαρμογής. Επίσης η σελίδα καταγραφής διαδρομής αποτελεί τον βασικό στόχο της παρούσας διπλωματικής εργασίας, που παρουσιάζει τη λειτουργικότητα και τις δυνατότητες που προσφέρει το Geolocation API της HTML, μέσω ενός φυλλομετρητή (browser), που σε συνδυασμό με ασύγχρονα αιτήματα Http (AJAX) της γλώσσας προγραμματισμού JavaScript, μπορεί να αποτελέσει τη βάση για μια πληθώρα εφαρμογών, όπως υπηρεσιών χαρτών και πλοήγησης, εφαρμογών υγείας, υπηρεσιών διαφήμισης, παρακολούθησης οχημάτων και άλλων εφαρμογών που εκμεταλλεύονται γεωγραφικά δεδομένα για πολλούς διαφορετικούς σκοπούς.

Κατά την είσοδο στην σελίδα της καταγραφής διαδρομής, προβάλλεται ένας διαδραστικός χάρτης και το σύστημα ζητά από τον χρήστη να παραχωρήσει πρόσβαση στην τοποθεσία του.

Η παραχώρηση πρόσβασης στην τοποθεσία είναι αναγκαία για τη λειτουργία της εφαρμογής και ο χρήστης πρέπει πάντα να παραχωρεί πρόσβαση στην τοποθεσία της συσκευής του.

Παρ' όλα αυτά αν ο χρήστης δεν παραχωρήσει πρόσβαση στην τοποθεσία του, το σύστημα θα προβάλει ένα αναδυόμενο παράθυρο με την κατάσταση της τοποθεσίας καθώς και αναλυτικές πληροφορίες για την χρήση της.

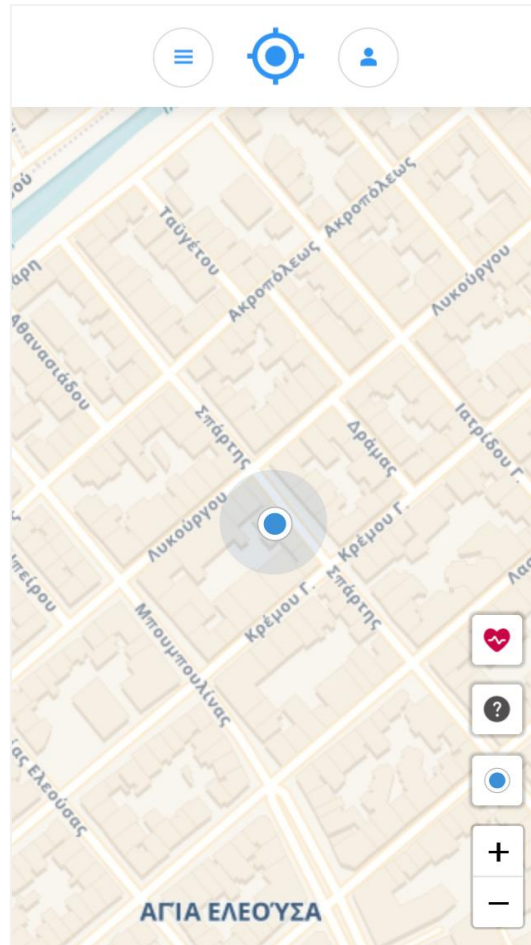
Η Εικόνα 3.10 παρουσιάζει το μήνυμα συστήματος στην περίπτωση που ο χρήστης δεν έχει παραχωρήσει πρόσβαση στην τοποθεσία του.



Εικόνα 3.6: Μήνυμα συστήματος για την κατάσταση της τοποθεσίας

Στην περίπτωση που ο χρήστης αποδεχθεί την πρόσβαση στην τοποθεσία, εμφανίζεται επάνω στον χάρτη μια μπλε κουκίδα, η οποία δηλώνει το σημείο που βρίσκεται ο χρήστης.

Καθώς ο χρήστης μετακινείται, αλλάζει και η θέση της μπλε κουκίδας επάνω στο χάρτη, υποδηλώνοντας το σημείο που βρίσκεται ο χρήστης την τρέχουσα χρονική στιγμή και παράλληλα αποθηκεύονται στο σύστημα τα δεδομένα του χρήστη (συντεταγμένες, ημερομηνία-ώρα, όνομα περιοχής κ.α.).



Εικόνα 3.6.1: Οθόνη καταγραφής διαδρομής

Η σελίδα καταγραφής διαδρομής αποτελείται από τα εξής στοιχεία:

1. Διαδραστικός χάρτης

Ο διαδραστικός χάρτης εμφανίζεται σε όλο το πλάτος και μήκος της οθόνης. Παρέχει πολλά επίπεδα πληροφορίας, όπως οδούς, ποτάμια, δάση, κτίρια και άλλα, καθώς και τη δυνατότητα στον χρήστη να αλληλεπιδρά με τον χάρτη και να προσαρμόζει τον τρόπο προβολής του.

2. Στίγμα τοποθεσίας

Το στίγμα τοποθεσίας, είναι ένα γραφικό στοιχείο που εμφανίζεται με τη μορφή μπλε κουκίδας και δείχνει τη θέση του χρήστη στον χάρτη. Καθώς ο χρήστης μετακινείται, αλλάζει και η θέση της μπλε κουκίδας επάνω στο χάρτη, υποδηλώνοντας το σημείο που βρίσκεται ο χρήστης την τρέχουσα χρονική στιγμή.

3. Κουμπιά εστίασης χάρτη (zoom in, zoom out)

Τα κουμπιά εστίασης χάρτη εμφανίζονται πάνω στον χάρτη με τα σύμβολα + και - και επιτρέπουν στον χρήστη να προσαρμόζει το επίπεδο μεγέθυνσης του χάρτη.

4. Κουμπί ένδειξης κατάστασης τοποθεσίας

Το κουμπί ένδειξης κατάστασης τοποθεσίας πραγματοποιεί διάφορες ενέργειες ανάλογα με μια σειρά από καταστάσεις.

Στην αρχική του κατάσταση, το κουμπί ένδειξης κατάστασης τοποθεσίας εμφανίζεται με μαύρο χρώμα και υποδηλώνει ότι η τοποθεσία δεν έχει ενεργοποιηθεί. Επιλέγοντάς το γίνεται προσπάθεια ενεργοποίησης της τοποθεσίας.

Στην περίπτωση που το κουμπί ένδειξης κατάστασης τοποθεσίας εμφανίζεται με μπλε χρώμα υποδηλώνει ότι η τοποθεσία έχει ενεργοποιηθεί και πραγματοποιείται καταγραφή διαδρομής. Επιλέγοντάς ο χάρτης εμφανίζει το στίγμα τοποθεσίας και η κατάσταση αλλάζει σε λειτουργία πεζοπορίας.

Στην περίπτωση που το κουμπί ένδειξης κατάστασης τοποθεσίας εμφανίζεται με μπλε χρώμα σε κύκλο υποδηλώνει ότι έχει ενεργοποιηθεί η λειτουργία πεζοπορίας. Αυτό σημαίνει πως η τοποθεσία είναι ενεργή και ο χάρτης κεντράρεται αυτόματα στο σημείο που βρίσκεται ο χρήστης.

Σε περίπτωση που γίνει zoom στον χάρτη, παύει η λειτουργία πεζοπορίας αλλά η τοποθεσία παραμένει ενεργοποιημένη.

5. Κουμπί βοήθειας σχετικά με την τοποθεσία

Το κουμπί βοήθειας σχετικά με την τοποθεσία εμφανίζεται με τη μορφή εικονιδίου που παρουσιάζει ένα ερωτηματικό και περιέχει μια σελίδα με αναλυτικές οδηγίες σχετικά με την χρήση της τοποθεσίας, καθώς και για προβλήματα που μπορεί να προκύψουν όπως σφάλματα στην εύρεση της τοποθεσίας.

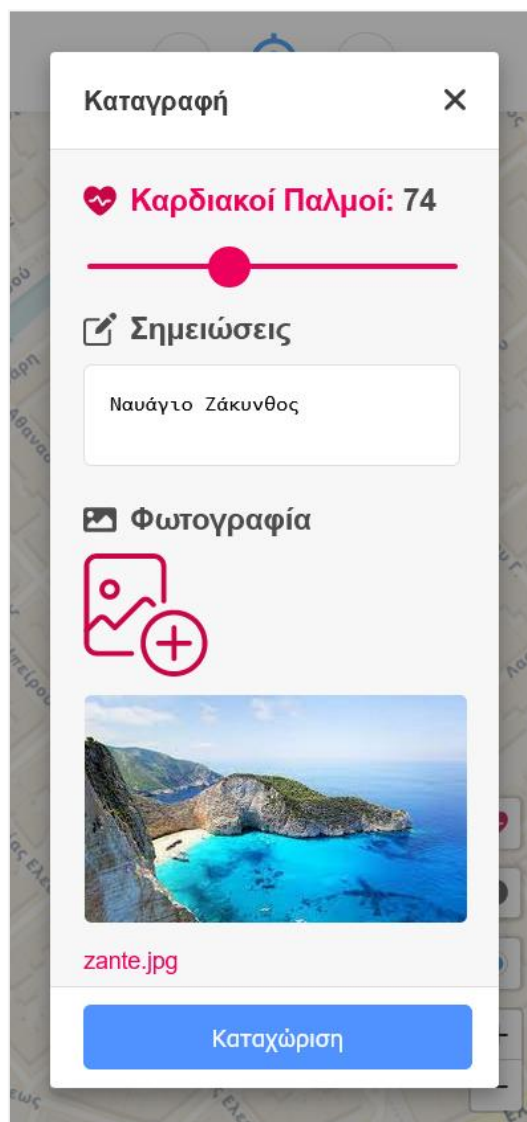
6. Κουμπί καταγραφής επιπλέον δεδομένων

Το κουμπί καταγραφής επιπλέον δεδομένων εμφανίζεται με τη μορφή εικονιδίου που παρουσιάζει μια κόκκινη καρδιά.

Επιλέγοντας το κουμπί καταγραφής επιπλέον δεδομένων, εμφανίζεται μια φόρμα στην οποία ο χρήστης έχει τη δυνατότητα να καταχωρήσει επιπλέον δεδομένα.

7. Φόρμα καταχώρησης επιπλέον δεδομένων

Η φόρμα καταχώρησης επιπλέον δεδομένων χρησιμοποιείται για να καταγραφούν επιπλέον δεδομένα του χρήστη όπως, καρδιακού παλμοί, σημειώσεις, φωτογραφίες και βίντεο κατά τη διάρκεια μια διαδρομής.



Εικόνα 3.6.2: Οθόνη καταγραφής επιπλέον δεδομένων

Για την καταχώρηση επιπλέον δεδομένων είναι υποχρεωτικό να έχει συμπληρωθεί τουλάχιστον ένα από τα στοιχεία της φόρμας.

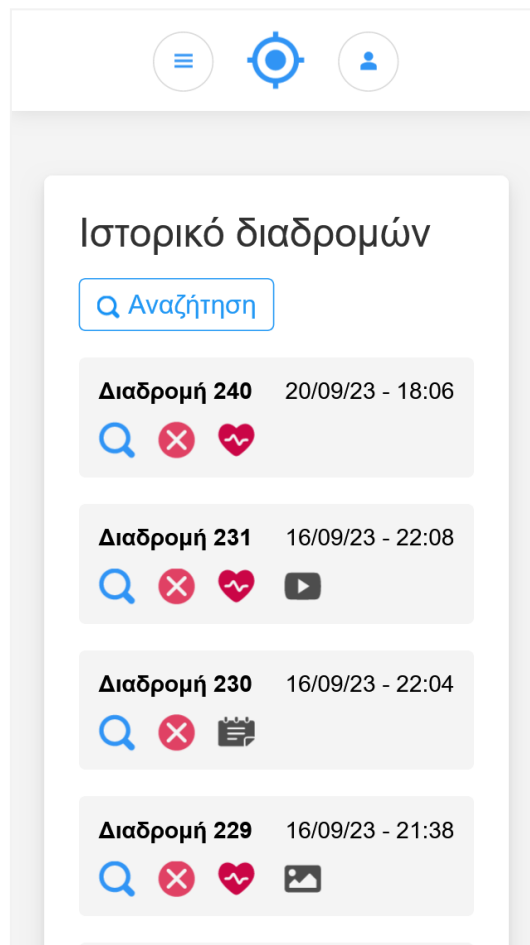
Στην περίπτωση που γίνει καταχώρηση, το σύστημα εμφανίζει ενημερωτικό μήνυμα που γνωστοποιεί στον χρήστη ότι τα δεδομένα αποθηκεύτηκαν, η φόρμα κλείνει και επανέρχεται στην αρχική της κατάσταση, ενώ ο χάρτης συνεχίζει να προβάλλει το στίγμα της τοποθεσίας του χρήστη και το σύστημα να πραγματοποιεί καταγραφή της διαδρομής.

Η καταχώρηση επιπλέον δεδομένων μπορεί να χρησιμοποιηθεί πολλές φορές κατά τη διάρκεια μιας διαδρομής.

3.7 Ιστορικό διαδρομών

Το ιστορικό διαδρομών παρουσιάζει μια λίστα με τις διαδρομές του χρήστη, καθώς και τα επιπλέον δεδομένα που έχει καταχωρίσει σε μια διαδρομή.

Επίσης υπάρχει δυνατότητα αναζήτησης σε δεδομένα διαδρομών όπως καρδιακοί παλμοί και σημειώσεις.



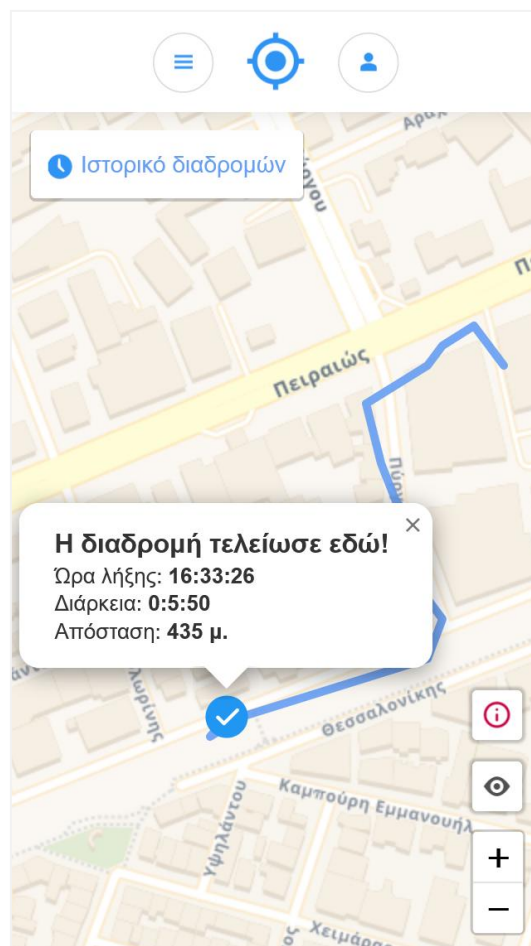
Εικόνα 3.7: Οθόνη ιστορικού διαδρομών

Κάθε διαδρομή οργανώνεται σε ένα γκρι πλαίσιο εμφανίζοντας το όνομα της διαδρομής, την ημερομηνία και την ώρα, καθώς επίσης και κουμπιά για προβολή της διαδρομής, διαγραφή, προβολή καρδιακών παλμών, σημειώσεων, φωτογραφιών και βίντεο.

Παρακάτω αναλύονται τα κουμπιά που υπάρχουν σε μια διαδρομή.

1. Προβολή διαδρομής

Το κουμπί προβολή διαδρομής εμφανίζεται ως ένα μπλε εικονίδιο με μεγεθυντικό φακό. Επιλέγοντας το κουμπί αυτό, γίνεται μετάβαση στη σελίδα προβολής της διαδρομής.

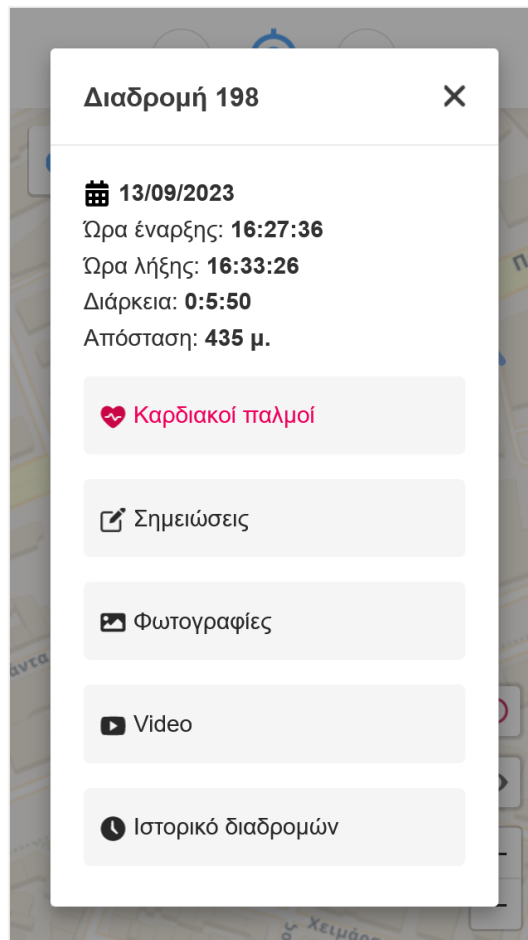


Εικόνα 3.7.1: Οθόνη της διαδρομής του χρήστη

Στη σελίδα προβολής διαδρομής εμφανίζεται ο διαδραστικός χάρτης και πάνω σε αυτόν είναι σχηματισμένη μια μπλε γραμμή η οποία απεικονίζει τη διαδρομή του χρήστη. Το σημείο που ξεκινά η μπλε γραμμή, είναι το σημείο όπου ξεκίνησε ο χρήστης, ενώ το μπλε κυκλικό εικονίδιο είναι το σημείο όπου τελείωσε η διαδρομή του χρήστη. Επιλέγοντας το μπλε κυκλικό εικονίδιο, εμφανίζονται επιπλέον πληροφορίες όπως η διάρκεια της διαδρομής και η απόσταση.

Επίσης η σελίδα προβολής διαδρομής περιλαμβάνει τα ακόλουθα κουμπιά.

- Κουμπί επιστροφής στο ιστορικό διαδρομών, που εμφανίζεται στο δεξιό πάνω μέρος του χάρτη.
- Κουμπί επανεστίασης. Απεικονίζεται με το εικονίδιο μάτι και χρησιμοποιείται ώστε να γίνεται επανεστίαση στη διαδρομή.
- Κουμπί για αναλυτικές πληροφορίες διαδρομής. Απεικονίζεται με το εικονίδιο i και χρησιμοποιείται για να προβάλει μια σελίδα με τις αναλυτικές πληροφορίες της διαδρομής.



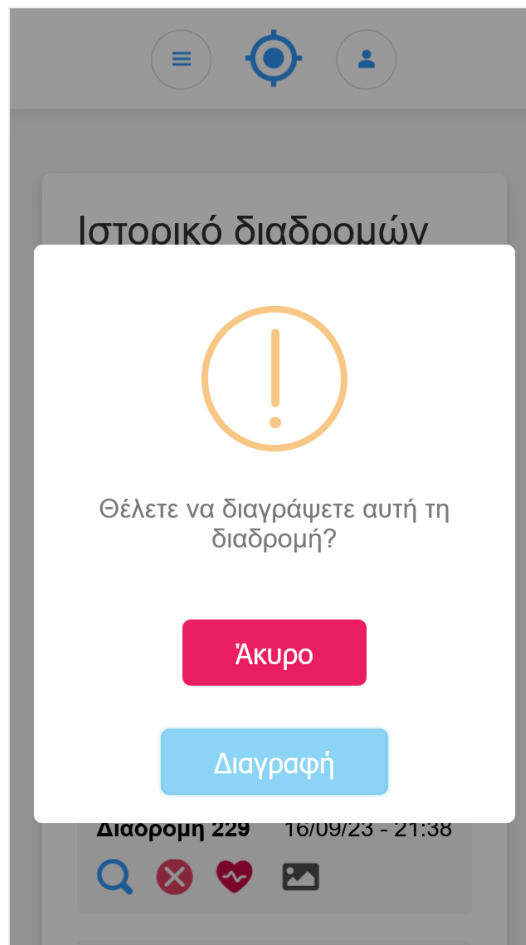
Εικόνα 3.7.2: Οθόνη με αναλυτικές πληροφορίες της διαδρομής του χρήστη

2. Διαγραφή διαδρομής

Το κουμπί διαγραφής διαδρομής εμφανίζεται ως ένα κόκκινο εικονίδιο με σύμβολο X.

Επιλέγοντας το κουμπί Διαγραφή διαδρομής, εμφανίζεται αναδυόμενο παράθυρο, όπου γίνεται ερώτηση επιβεβαίωσης διαγραφής.

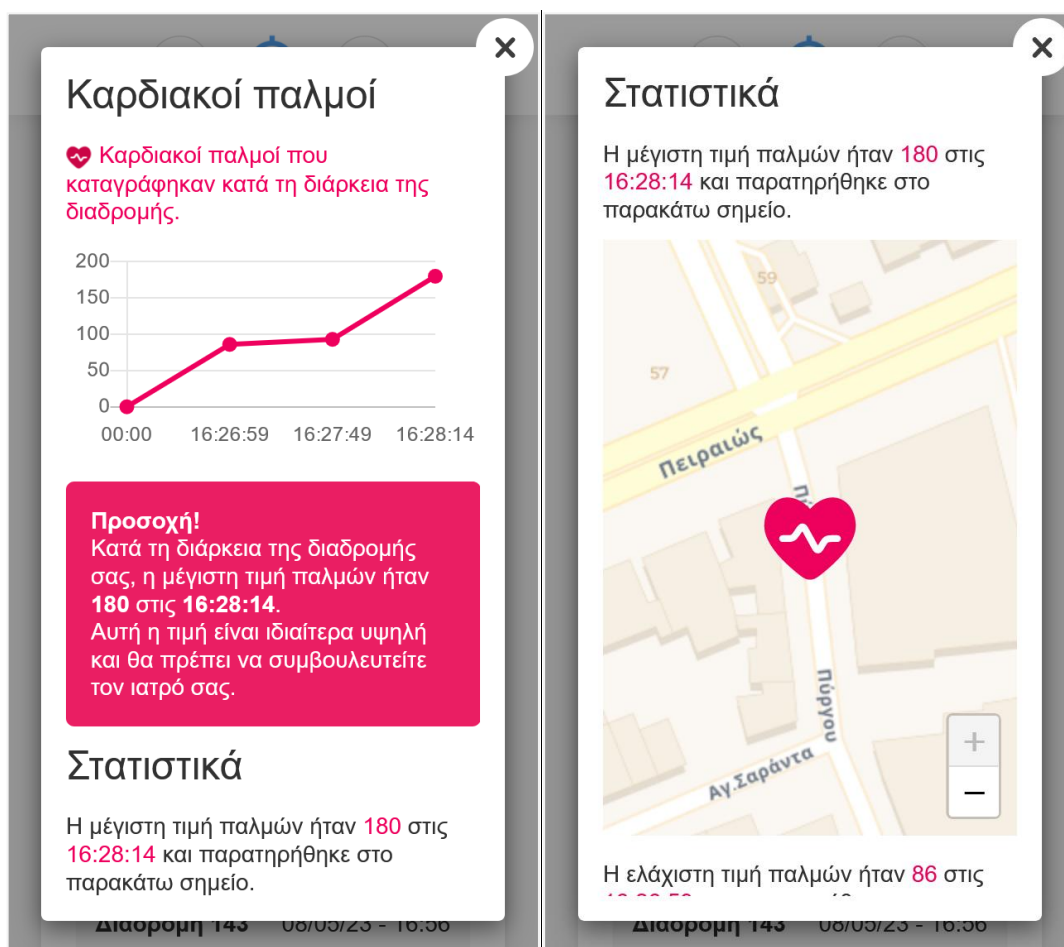
Επιλέγοντας Διαγραφή, διαγράφονται όλες τα στοιχεία που σχετίζονται με τη συγκεκριμένη διαδρομή του χρήστη.



Εικόνα 3.7.3: Παράθυρο επιβεβαίωσης διαγραφής της διαδρομής

3. Καρδιακοί παλμοί στη διαδρομή

Το κουμπί καρδιακοί παλμοί εμφανίζεται ως ένα κόκκινο εικονίδιο με σύμβολο μια καρδιά. Επιλέγοντας το κουμπί καρδιακοί παλμοί, εμφανίζεται μια σελίδα που περιλαμβάνει τους καρδιακούς παλμούς που καταχώρισε ο χρήστης κατά τη διάρκεια της διαδρομής.



Εικόνα 3.7.4: Οθόνη προβολής καρδιακών παλμών του χρήστη

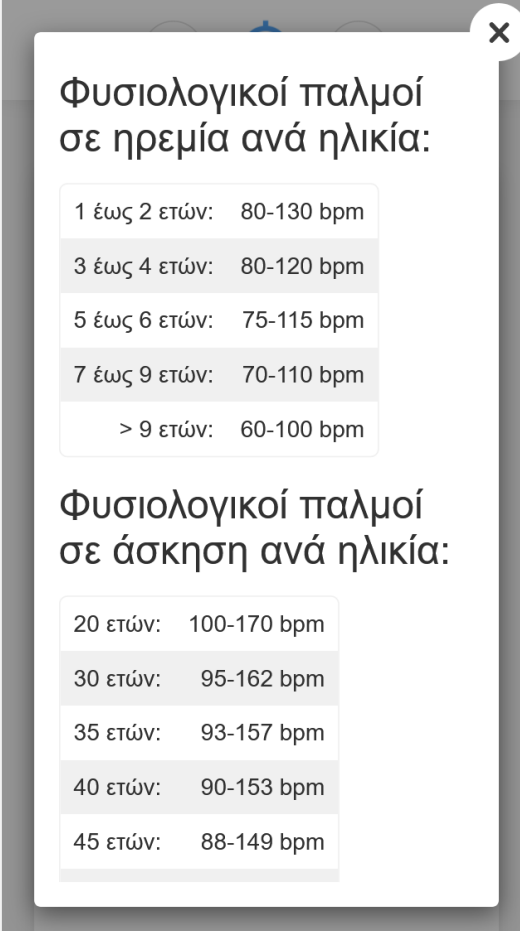
Η παραπάνω σελίδα απεικονίζει τους καρδιακούς παλμούς που καταχώρισε ο χρήστης κατά τη διάρκεια της διαδρομής του, με τη μορφή ενός γραφήματος. Στον άξονα Y απεικονίζονται οι τιμές των καρδιακών παλμών (0 έως 200), ενώ στον άξονα X απεικονίζονται οι ώρες.

Η συγκεκριμένη σελίδα εμφανίζει επίσης στατιστικά για τη μέγιστη και την ελάχιστη τιμή των καρδιακών παλμών του χρήστη, καθώς και σε ποιο σημείο παρουσιάστηκαν αυτές.

Το σημείο που παρουσιάστηκε η μέγιστη και η ελάχιστη τιμή των καρδιακών παλμών του χρήστη απεικονίζεται σε χάρτη.

Επίσης σε περίπτωση που παρατηρούνται ακραίες τιμές καρδιακών παλμών, το σύστημα εμφανίζει προειδοποιητικό μήνυμα, ενημερώνοντας τον χρήστη.

Τέλος η σελίδα περιλαμβάνει αναλυτικές πληροφορίες και χρήσιμες συμβουλές σχετικά με τους καρδιακούς παλμούς.



The screenshot shows a mobile application interface with a white background and a grey border. At the top right, there is a close button (X). The main content is divided into two sections, each with a title and a table of data.

Φυσιολογικοί παλμοί σε ηρεμία ανά ηλικία:

1 έως 2 ετών:	80-130 bpm
3 έως 4 ετών:	80-120 bpm
5 έως 6 ετών:	75-115 bpm
7 έως 9 ετών:	70-110 bpm
> 9 ετών:	60-100 bpm

Φυσιολογικοί παλμοί σε άσκηση ανά ηλικία:

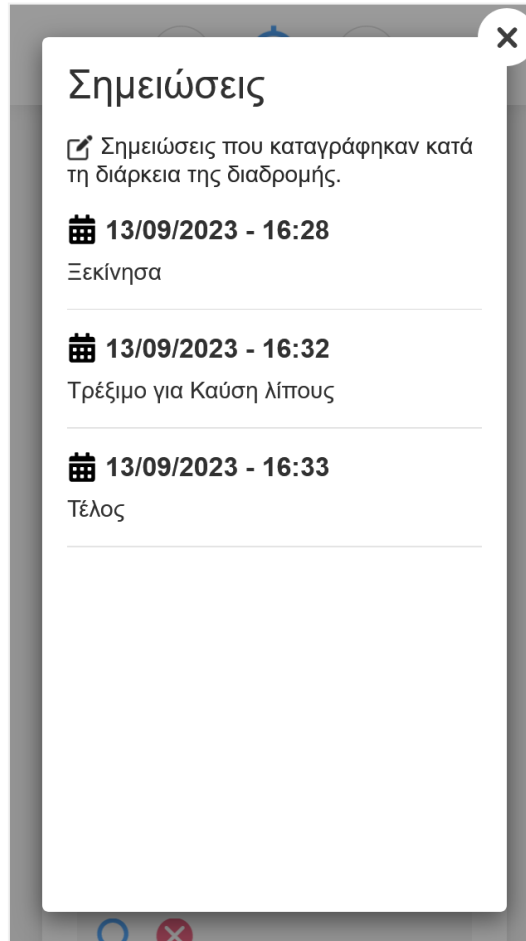
20 ετών:	100-170 bpm
30 ετών:	95-162 bpm
35 ετών:	93-157 bpm
40 ετών:	90-153 bpm
45 ετών:	88-149 bpm

Εικόνα 3.7.5: Οθόνη προβολής πληροφοριών καρδιακών παλμών

4. Σημειώσεις στη διαδρομή

Το κουμπί σημειώσεις εμφανίζεται ως ένα μαύρο εικονίδιο με σύμβολο ένα σημειωματάριο.

Επιλέγοντας το κουμπί σημειώσεις, εμφανίζεται μια σελίδα που περιλαμβάνει τις σημειώσεις που καταχώρισε ο χρήστης κατά τη διάρκεια της διαδρομής.

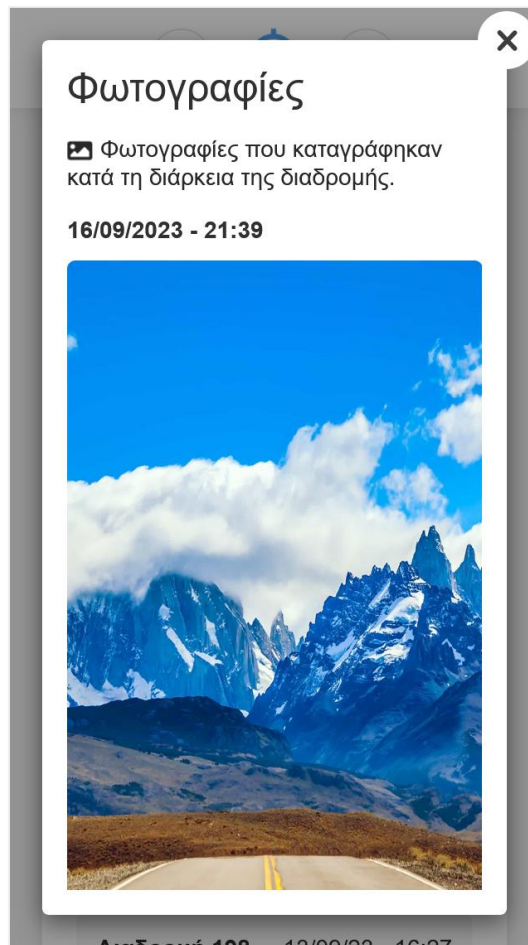


Εικόνα 3.7.6: Οθόνη προβολής σημειώσεων του χρήστη

5. Φωτογραφίες στη διαδρομή

Το κουμπί φωτογραφίες εμφανίζεται ως ένα μαύρο εικονίδιο με σύμβολο μια φωτογραφία.

Επιλέγοντας το κουμπί φωτογραφίες, εμφανίζεται μια σελίδα που περιλαμβάνει τις φωτογραφίες που καταχώρισε ο χρήστης κατά τη διάρκεια της διαδρομής.

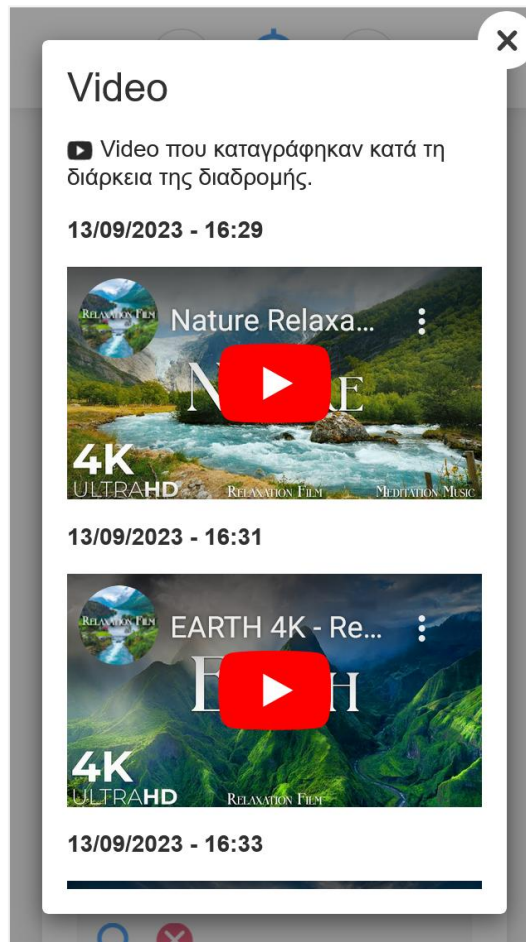


Εικόνα 3.7.7: Οθόνη προβολής φωτογραφιών του χρήστη

6. Βίντεο στην διαδρομή

Το κουμπί βίντεο εμφανίζεται ως ένα μαύρο εικονίδιο με σύμβολο ένα βίντεο.

Επιλέγοντας το κουμπί βίντεο, εμφανίζεται μια σελίδα που περιλαμβάνει τα βίντεο που καταχώρισε ο χρήστης κατά τη διάρκεια της διαδρομής.



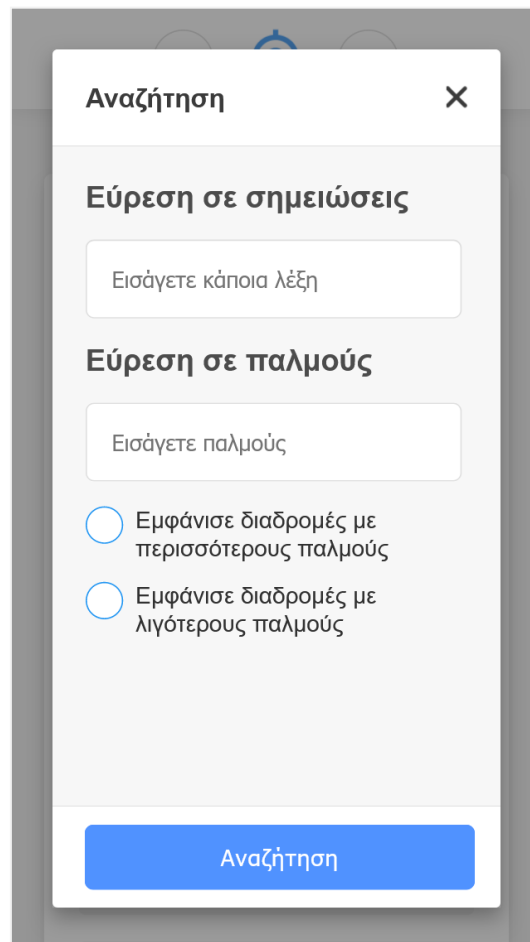
Εικόνα 3.7.8: Οθόνη προβολής βίντεο του χρήστη

Αναζήτηση

Το ιστορικό διαδρομών περιλαμβάνει επίσης τη δυνατότητα αναζήτησης σε δεδομένα διαδρομών όπως καρδιακοί παλμοί και σημειώσεις.

Το κουμπί αναζήτησης εμφανίζεται στο επάνω μέρος της σελίδας ιστορικό διαδρομών. Επιλέγοντας το κουμπί αναζήτησης, εμφανίζεται μια φόρμα εισαγωγής τιμών αναζήτησης.

Ο χρήστης έχει τη δυνατότητα αναζήτησης για συγκεκριμένες λέξεις σε σημειώσεις, καθώς και για τιμές καρδιακών παλμών που είναι απόλυτες, μικρότερες ή μεγαλύτερες της εισαχθείσας τιμής.



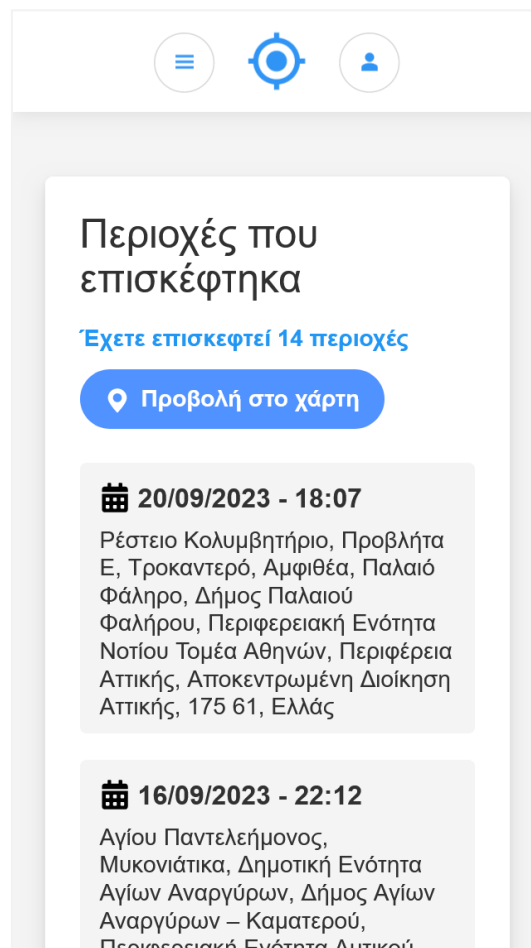
The image shows a mobile application search modal titled "Αναζήτηση" (Search). It features a close button (X) in the top right corner. The modal is divided into two main sections: "Εύρεση σε σημειώσεις" (Search in notes) and "Εύρεση σε παλμούς" (Search in heart rates). Each section has a text input field with a placeholder "Εισάγετε κάποια λέξη" (Enter a word) and "Εισάγετε παλμούς" (Enter heart rates) respectively. Below the heart rate section, there are two radio button options: "Εμφάνισε διαδρομές με περισσότερους παλμούς" (Show routes with more heart rates) and "Εμφάνισε διαδρομές με λιγότερους παλμούς" (Show routes with fewer heart rates). At the bottom of the modal is a blue button labeled "Αναζήτηση" (Search).

Εικόνα 3.7.9: Οθόνη αναζήτησης σε δεδομένα διαδρομών

Επιλέγοντας το κουμπί αναζήτηση, εμφανίζονται οι διαδρομές που πληρούν τα κριτήρια αναζήτησης του χρήστη.

3.8 Περιοχές που επισκέφτηκα

Η σελίδα περιοχές που επισκέφτηκα, παρουσιάζει μια λίστα με τις περιοχές που έχει επισκεφτεί ο χρήστης. Ως περιοχή επίσκεψης ορίζεται το τελευταίο σημείο σε μια διαδρομή του χρήστη. Επίσης εμφανίζεται το σύνολο των περιοχών και ένα κουμπί που δίνει τη δυνατότητα προβολής όλων των περιοχών σε χάρτη.

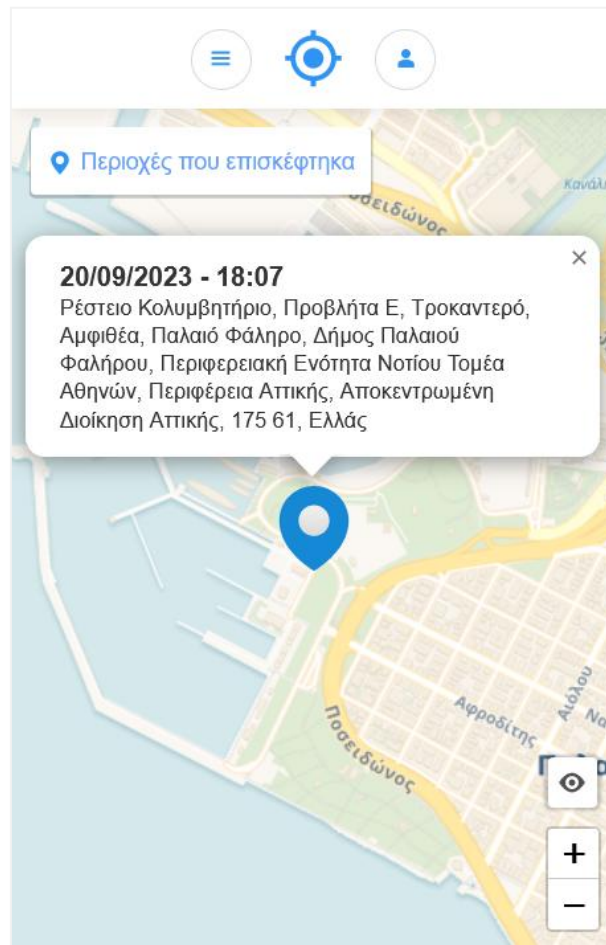


Εικόνα 3.8: Οθόνη περιοχών που έχει επισκεφτεί ο χρήστης

Κάθε περιοχή επίσκεψης οργανώνεται σε ένα γκρι πλαίσιο που περιλαμβάνει την ημερομηνία, την ώρα και το όνομα της περιοχής.

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Επιλέγοντας μια περιοχή, αυτή εμφανίζεται με τη μορφή μιας πινέζας σε χάρτη. Επιλέγοντας την πινέζα προβάλλονται οι παραπάνω πληροφορίες της περιοχής, όπως στην Εικόνα 3.8.1.



Εικόνα 3.8.1: Χάρτης περιοχής που έχει επισκεφτεί ο χρήστης

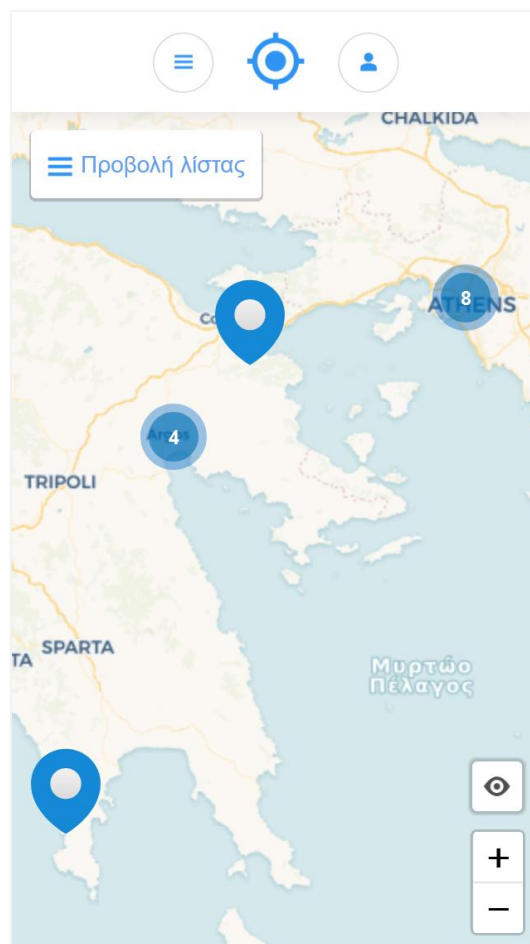
Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Επίσης επιλέγοντας το κουμπί Προβολή στο χάρτη, εμφανίζεται μια σελίδα στην οποία προβάλλεται ένας χάρτης που περιέχει σε μορφή πινέζας όλες τις περιοχές που έχει επισκεφτεί ο χρήστης.

Σε κάποιες περιπτώσεις οι περιοχές ενδέχεται να ομαδοποιούνται (clustering) με τη μορφή ενός μπλε κύκλου που περιλαμβάνει έναν αριθμό. Αυτό σημαίνει ότι για τη συγκεκριμένη περιοχή υπάρχουν τόσες περιοχές, όσες και ο αριθμός του κύκλου.

Αυτό συμβαίνει στην περίπτωση που υπάρχουν πολλές περιοχές, προκειμένου να είναι πιο ευδιάκριτες αλλά και για καλύτερη απόδοση της εφαρμογής.

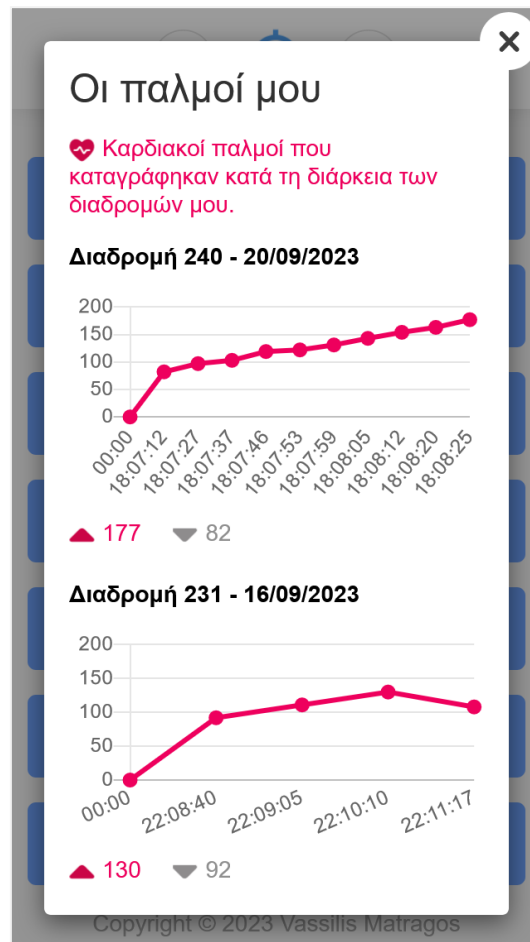
Επιλέγοντας τον κύκλο, ο χάρτης θα εστιαστεί στην συγκεκριμένη περιοχή και θα εμφανιστούν οι πινέζες που δηλώνουν την κάθε περιοχή.



Εικόνα 3.8.2: Προβολή σε χάρτη των περιοχών που έχει επισκεφτεί ο χρήστης

3.9 Οι παλμοί μου

Η σελίδα οι παλμοί μου, περιλαμβάνει τους καρδιακούς παλμούς που καταχώρισε ο χρήστης κατά τη διάρκεια μιας διαδρομής και προβάλλεται με μορφή γραφήματος ανά διαδρομή, με τίτλο το όνομα της διαδρομής και την ημερομηνία.



Εικόνα 3.9: Προβολή των καρδιακών παλμών του χρήστη ανά διαδρομή

Στον άξονα Υ απεικονίζονται οι τιμές των καρδιακών παλμών (0 έως 200), ενώ στον άξονα Χ απεικονίζονται οι ώρες.

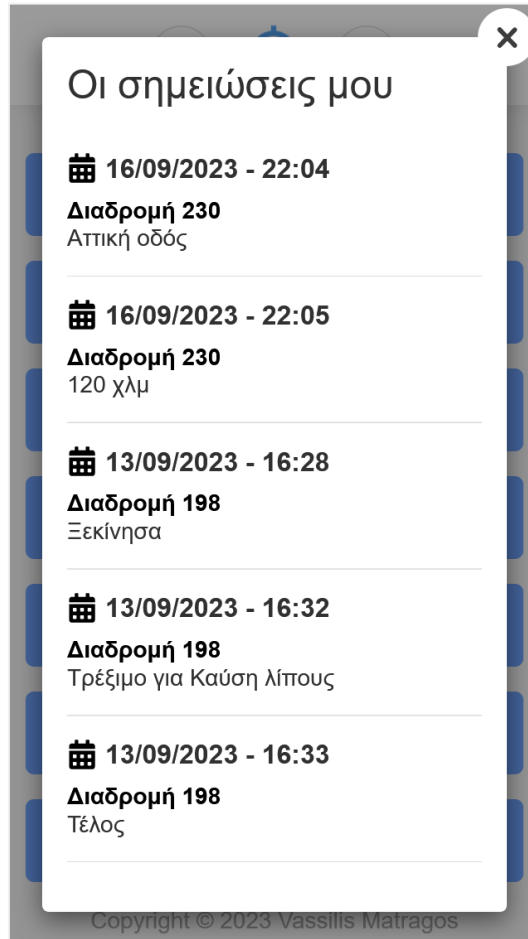
Επίσης σε κάθε διαδρομή απεικονίζεται η μέγιστη και η ελάχιστη τιμή των καρδιακών παλμών του χρήστη.

Τέλος η σελίδα περιλαμβάνει αναλυτικές πληροφορίες και χρήσιμες συμβουλές σχετικά με τους καρδιακούς παλμούς.

3.10 Οι σημειώσεις μου

Η σελίδα οι σημειώσεις μου, περιλαμβάνει όλες τις σημειώσεις που έχει καταχωρίσει ο χρήστης.

Οι σημειώσεις του χρήστη εμφανίζονται η μια κάτω από την άλλη με τίτλο την ημερομηνία και την ώρα, καθώς και το όνομα της διαδρομής.



Εικόνα 3.10: Προβολή όλων των σημειώσεων του χρήστη

3.11 Οι φωτογραφίες μου

Η σελίδα οι φωτογραφίες μου, περιλαμβάνει όλες τις φωτογραφίες που έχει καταχωρίσει ο χρήστης.

Οι φωτογραφίες του χρήστη εμφανίζονται η μια κάτω από την άλλη με τίτλο το όνομα της διαδρομής, καθώς και την ημερομηνία με την ώρα.

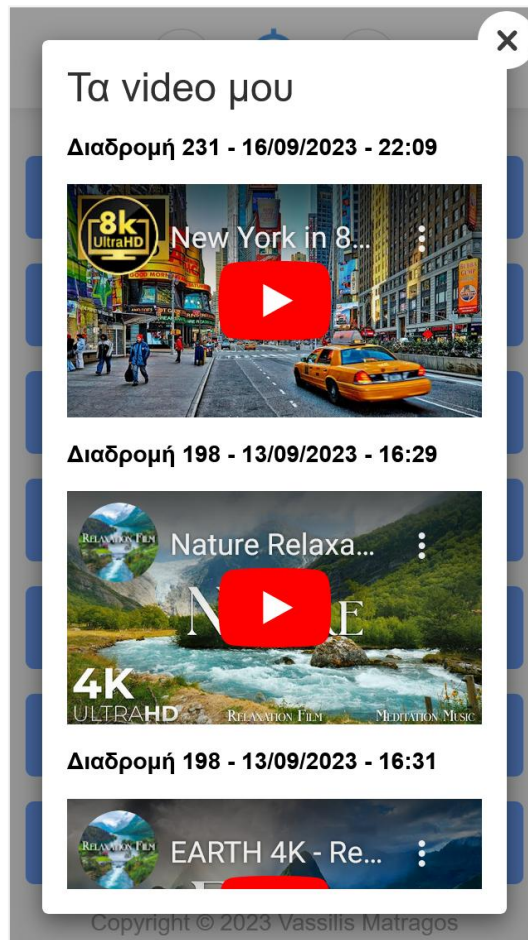


Εικόνα 3.11: Προβολή όλων των φωτογραφιών του χρήστη

3.12 Τα video μου

Η σελίδα τα video μου, περιλαμβάνει όλα τα βίντεο που έχει καταχωρίσει ο χρήστης.

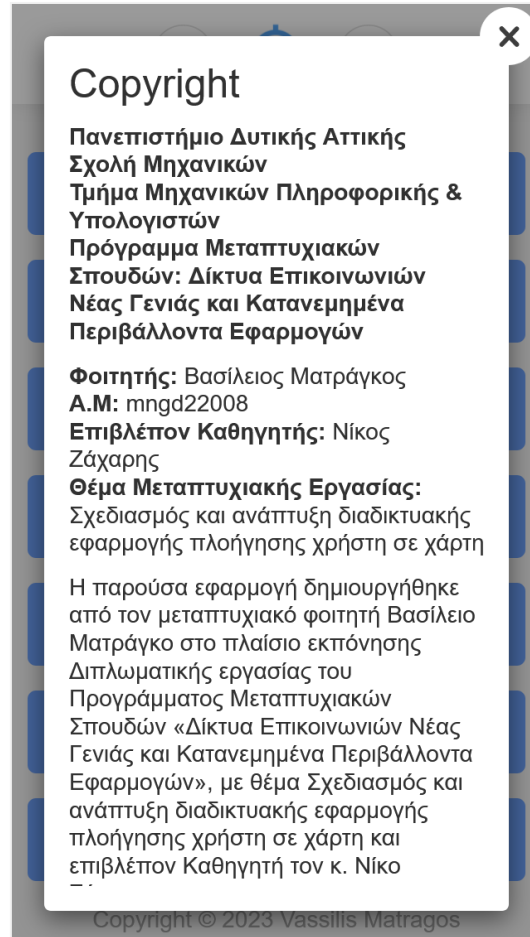
Τα βίντεο του χρήστη εμφανίζονται το ένα κάτω από το άλλο με τίτλο το όνομα της διαδρομής, καθώς και την ημερομηνία με την ώρα.



Εικόνα 3.12: Προβολή των όλων των βίντεο του χρήστη

3.13 Copyright

Η σελίδα Copyright, περιλαμβάνει πληροφορίες για την παρούσα διπλωματική εργασία και πληροφορίες για τις λειτουργίες της εφαρμογής.



Εικόνα 3.13: Copyright

ΚΕΦΑΛΑΙΟ 4

ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΙΘΑΝΕΣ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η παρούσα διπλωματική εργασία αφορά τον σχεδιασμό και την ανάπτυξη μιας διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη. Ο βασικός στόχος ήταν να παρουσιαστεί η λειτουργικότητα και οι δυνατότητες που προσφέρει το Geolocation API της HTML, μέσω ενός φυλλομετρητή (browser), που σε συνδυασμό την τεχνική των ασύγχρονων αιτημάτων Http (AJAX) της γλώσσας προγραμματισμού JavaScript, μπορεί να αποτελέσει τη βάση για μια πληθώρα εφαρμογών, όπως υπηρεσιών χαρτών και πλοήγησης, εφαρμογών υγείας, υπηρεσιών διαφήμισης, παρακολούθησης οχημάτων και πολλών άλλων εφαρμογών που εκμεταλλεύονται γεωγραφικά δεδομένα.

Στο κεφάλαιο 4.1 ακολουθούν τα συμπεράσματα για τον τρόπο λειτουργίας και χρήσης της εφαρμογής.

4.1 Συμπεράσματα

Το παρόν κεφάλαιο εξετάζει τα συμπεράσματα που προκύπτουν από τον τρόπο λειτουργίας και τη χρήση της παρούσας διαδικτυακής εφαρμογής.

Στο πλαίσιο των συμπερασμάτων, εξετάζονται η χρησιμότητα, η ευχρηστία, η απόδοση και η ασφάλεια της διαδικτυακής εφαρμογής.

Χρησιμότητα

Η χρησιμότητα της παρούσας διαδικτυακής εφαρμογής έγκειται στο γεγονός ότι επιτρέπει την πλοήγηση ενός χρήστη σε χάρτη, καταγράφοντας αυτόματα γεωγραφικά δεδομένα. Τα δεδομένα αυτά προβάλλονται με τη μορφή διαδρομών και περιοχών που επισκέφτηκε ο χρήστης, διατηρώντας ένα αρχείο ιστορικότητας. Επίσης η εφαρμογή παρέχει τη δυνατότητα καταχώρησης από τον χρήστη των καρδιακών παλμών του, που στη συνέχεια παρουσιάζονται με μορφή γραφημάτων, παρέχοντας στατιστικά από τα οποία μπορούν να εξαχθούν χρήσιμα συμπεράσματα για την υγεία του χρήστη.

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Ο χρήστης μπορεί επίσης να καταχωρεί και άλλα δεδομένα όπως σημειώσεις, φωτογραφίες και βίντεο. Με αυτόν τον τρόπο διατηρεί ένα αρχείο από προσωπικές στιγμές, που για κάποιους χρήστες οι πληροφορίες αυτές μπορεί να έχουν ανεκτίμητη συναισθηματική αξία.

Οι παραπάνω δυνατότητες καθιστούν την εφαρμογή χρήσιμη προς τον χρήστη.

Παρ' όλα αυτά, πέρα από τη συγκεκριμένη λειτουργικότητα που παρέχει η παρούσα εφαρμογή, θα μπορούσε να αποτελέσει τη βάση για μια πληθώρα εφαρμογών, όπως υπηρεσιών χαρτών και πλοήγησης, εφαρμογών υγείας, υπηρεσιών διαφήμισης, παρακολούθησης οχημάτων και πολλών άλλων εφαρμογών που εκμεταλλεύονται γεωγραφικά δεδομένα.

Ευχρηστία

Η εφαρμογή σχεδιάστηκε και υλοποιήθηκε με στόχο να προσφέρει μια θετική εμπειρία στον χρήστη. Δηλαδή ο χρήστης να μπορεί να χρησιμοποιήσει και να αλληλεπιδράσει με τις διάφορες λειτουργίες της εφαρμογής, εύκολα και γρήγορα.

Η διεπαφή χρήστη σχεδιάστηκε έτσι ώστε να είναι εύκολο να κατανοηθεί και να χρησιμοποιηθεί, ενώ διαθέτει συνέπεια στη διαμόρφωση των στοιχείων και των λειτουργιών που βοηθά στη μείωση της σύγχυσης και στη δημιουργία μιας πιο προβλέψιμης εμπειρίας χρήστη.

Επίσης παρέχονται οδηγίες και συμβουλές που μπορούν να βοηθήσουν τους χρήστες να κατανοήσουν τη λειτουργία της εφαρμογής και να εκμεταλλευτούν πλήρως τις δυνατότητές της.

Απόδοση

Η εφαρμογή σχεδιάστηκε και υλοποιήθηκε με στόχο να παρέχει γρήγορη απόκριση και απρόσκοπτη λειτουργία στους χρήστες.

Δόθηκε ιδιαίτερη έμφαση στην ταχύτητα με την οποία φορτώνουν οι σελίδες της εφαρμογής, καθώς αυτό επηρεάζει αισθητά την εμπειρία του χρήστη. Οι γρήγορες φορτώσεις συμβάλλουν στη θετική αίσθηση των χρηστών.

Επίσης η εφαρμογή ανταποκρίνεται γρήγορα στις ενέργειες του χρήστη, όπως εισαγωγή δεδομένων, άνοιγμα σελίδων και ενημερωτικά μηνύματα προς τον χρήστη.

Για την επίτευξη του παραπάνω στόχου δόθηκε ιδιαίτερη σημασία στη βελτιστοποίηση των εικόνων και των πολυμέσων, καθώς και στη δομή του κώδικα, ώστε να είναι γραμμένος με τρόπο που να εκτελείται γρήγορα και να ανταποκρίνεται αποτελεσματικά στα αιτήματα των χρηστών.

Ασφάλεια

Η εφαρμογή σχεδιάστηκε ώστε να παρέχει ασφάλεια ως προς τα δεδομένα του χρήστη καθώς για προστασία από άλλες απειλές ασφαλείας.

Για τον λόγο αυτό υιοθετήθηκαν μέτρα ασφαλείας, όπως η κρυπτογράφηση του κωδικού πρόσβασης χρήστη με hash function, προστασία από επιθέσεις SQL Injection και XSS ειδικούς χαρακτήρες σε HTML entities, επικύρωση δεδομένων σε επίπεδο πελάτη και διακομιστή, χρήση πιστοποιητικού TLS για επικοινωνία με HTTPS και ενημερωμένη έκδοση της γλώσσας προγραμματισμού, με την οποία διορθώνονται προβλήματα ασφαλείας σε σχέση με παλαιότερες εκδόσεις.

4.2 Πιθανές μελλοντικές επεκτάσεις

Ανεξάρτητα από τον τομέα ή τον σκοπό για τον οποίο δημιουργήθηκε μια εφαρμογή, πάντα υπάρχουν περιθώρια για βελτίωση, εξέλιξη και προσαρμογή στις ανάγκες των χρηστών.

Οι βελτιώσεις μπορεί να προκύψουν από πολλούς παράγοντες. Τεχνολογικές εξελίξεις, ανάδυση νέων αναγκών, αλλαγές στο περιβάλλον λειτουργίας και ανατροφοδότηση από τους χρήστες μπορούν να ενισχύσουν την ανάγκη για συνεχή βελτίωση. Ακόμα και αν μια εφαρμογή είναι επιτυχημένη, πρέπει να υπάρχει διαρκής αναζήτηση για τρόπους για που την κάνει πιο αποτελεσματική, χρηστική και ασφαλή.

Συνολικά, η ιδέα ότι όλες οι εφαρμογές έχουν πάντα περιθώρια βελτίωσης λειτουργεί ως κινητήριο για την καινοτομία και την εξέλιξη, προάγοντας έτσι τη διαρκή πρόοδο σε όλους τους τομείς της τεχνολογίας.

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Στην παρούσα διαδικτυακή εφαρμογή, προβλέπονται δυο μελλοντικές επεκτάσεις. Η πρώτη επέκταση αφορά την επίδοση (performance) της εφαρμογής, ενώ η δεύτερη αφορά τη λειτουργικότητα και την περαιτέρω αυτοματοποίηση.

Επίδοση (performance)

Η παρούσα διαδικτυακή εφαρμογή βασίζεται σε μια σχεσιακή βάση δεδομένων για την αποθήκευση και την ανάκτηση των δεδομένων των χρηστών.

Κατά το πέρασμα του χρόνου και εφόσον προστίθενται όλο και περισσότερες εγγραφές στη βάση δεδομένων, η ανάκτηση αυτών θα επιφέρει καθυστερήσεις που θα επηρεάσουν την επίδοση της εφαρμογής.

Σε αυτό το πρόβλημα, έρχεται να δώσει λύση μια μη σχεσιακή βάση δεδομένων.

Σε ορισμένες περιπτώσεις, όπως στην παρούσα διαδικτυακή εφαρμογή, οι μη σχεσιακές βάσεις δεδομένων μπορούν να προσφέρουν γρηγορότερη ανάκτηση δεδομένων σε σχέση με τις σχεσιακές βάσεις δεδομένων, ειδικά όταν πρόκειται για μεγάλα σύνολα δεδομένων.

Οι μη σχεσιακές βάσεις δεδομένων επιτρέπουν την πιο ευέλικτη και βελτιστοποιημένη διαχείριση των δεδομένων σε περιβάλλοντα όπου η απόδοση και η ταχύτητα είναι κρίσιμοι παράγοντες.

Η ανάγκη για γρήγορη ανάκτηση δεδομένων είναι κρίσιμη σε πολλές εφαρμογές, και αυτός είναι ένας από τους κύριους λόγους που κάποιες φορές προτιμάται η χρήση μη σχεσιακών βάσεων δεδομένων.

Στις σχεσιακές βάσεις δεδομένων, η δομή των πινάκων και οι συσχετίσεις μεταξύ τους μπορεί να απαιτήσει περισσότερο χρόνο για την ανάκτηση δεδομένων, ειδικά όταν οι συνθήκες των ερωτήσεων είναι πολύπλοκες. Στις μη σχεσιακές βάσεις, όπως οι βάσεις δεδομένων γράφων ή κειμένου, οι δομές δεδομένων μπορούν να παρουσιάζονται με τρόπο που επιτρέπει γρήγορη πρόσβαση σε συγκεκριμένα σημεία των δεδομένων.

Σε μη σχεσιακές βάσεις δεδομένων που υποστηρίζουν sharding, τα δεδομένα μπορούν να κατανέμονται σε πολλαπλούς κόμβους. Αυτό επιτρέπει την παράλληλη ανάκτηση δεδομένων από διάφορες πηγές, βελτιώνοντας τον χρόνο απόκρισης των ερωτημάτων.

Μια καλή επιλογή υλοποίησης μιας μη σχεσιακής βάσης δεδομένων είναι η MongoDB. Η MongoDB είναι μια δημοφιλής μη σχεσιακή βάση δεδομένων που ανήκει στην κατηγορία των βάσεων δεδομένων NoSQL. Έχει κυκλοφορήσει υπό τη γενική άδεια χρήσης Apache, η οποία είναι μια άδεια ανοικτού κώδικα. Αυτό σημαίνει ότι το λογισμικό είναι δωρεάν για χρήση, με διάφορους όρους και περιορισμούς που ορίζονται στην άδεια Apache.

Η κύρια χαρακτηριστική της ιδιότητα είναι ότι αποθηκεύει τα δεδομένα σε μορφή BSON (Binary JSON), που είναι ένα δυαδικό JSON (JavaScript Object Notation) και υποστηρίζει δομές δεδομένων όπως γραφήματα και έγγραφα.

Η MongoDB είναι ιδανική για εφαρμογές που απαιτούν ευελιξία στο σχήμα των δεδομένων, η ανάπτυξη γρήγορων εφαρμογών, η ανάλυση μεγάλων όγκων δεδομένων, και η διαχείριση πολύπλοκων δεδομένων.

Κάποια βασικά χαρακτηριστικά της MongoDB περιλαμβάνουν:

1. Ευελιξία του Σχήματος

Τα έγγραφα (documents) αποθηκεύονται σε μορφή BSON και δεν χρειάζεται να ακολουθούν ένα συγκεκριμένο σχήμα. Αυτό σημαίνει ότι μπορεί να αποθηκεύει διάφορους τύπους δεδομένων μέσα σε ένα έγγραφο χωρίς να χρειάζεται προσαρμογή στο σχήμα της βάσης δεδομένων.

2. Σειρά Δεδομένων (Indexing)

Η MongoDB υποστηρίζει τη δημιουργία δεικτών (indexes) για γρήγορη ανάκτηση δεδομένων. Αυτό είναι σημαντικό για επιτάχυνση των ερωτημάτων και τη βελτιστοποίηση της απόδοσης της βάσης δεδομένων.

3. Διάταξη (Sharding)

Η MongoDB υποστηρίζει το sharding, που είναι η δυνατότητα κατανεμημένης αποθήκευσης δεδομένων σε πολλαπλούς κόμβους. Αυτό επιτρέπει την αποτελεσματική διαχείριση τεράστιων όγκων δεδομένων και την επέκταση του συστήματος όσο αναπτύσσεται η εφαρμογή.

4. Γρήγορα ερωτήματα

Η MongoDB είναι σχεδιασμένη για να παρέχει γρήγορη απόδοση σε ερωτήματα και ενέργειες εγγραφής. Η δυνατότητα διανομής φορτίου μεταξύ πολλαπλών κόμβων σε συνδυασμό με τη διαχείριση μνήμης και διαχείριση δεδομένων σε δίσκο επιτρέπει αποδοτική απόδοση.

5. Ευκολία Χρήσης

Η MongoDB παρέχει ευέλικτες επιλογές για την αλληλεπίδραση με τη βάση δεδομένων, συμπεριλαμβανομένου του γραφικού περιβάλλοντος εκτέλεσης εντολών (Mongo Shell) και διάφορων οδηγών (drivers) για πολλές γλώσσες προγραμματισμού.

Αυτοματοποίηση

Η δεύτερη μελλοντική επέκταση που προβλέπεται στην παρούσα διαδικτυακή εφαρμογή, αφορά τη λειτουργικότητα και την περαιτέρω αυτοματοποίηση.

Όπως αναλύθηκε στα παραπάνω κεφάλαια, η εφαρμογή χρησιμοποιεί το GPS της κινητής συσκευής για την λήψη και αποθήκευση των γεωγραφικών δεδομένων.

Η παραπάνω διαδικασία απαιτεί τη χρήση υλικού της συσκευής, που στην προκειμένη περίπτωση είναι το GPS. Το συγκεκριμένο υλικό διατίθεται σε όλες τις κινητές συσκευές που υπάρχουν αυτή τη στιγμή στην αγορά.

Παρ' όλα αυτά, πολλές σύγχρονες κινητές συσκευές διαθέτουν ενσωματωμένους αισθητήρες καρδιακών παλμών. Οι περισσότεροι από αυτούς τους αισθητήρες λειτουργούν με τη χρήση φωτοπηγών, όπως το LED, που προσαρτώνται στην κινητή συσκευή, και ανιχνεύουν τις μικροκαρδιακές αλλαγές.

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Οι εφαρμογές υγείας που είναι διαθέσιμες για πολλές κινητές συσκευές εκμεταλλεύονται αυτούς τους αισθητήρες για να παρέχουν πληροφορίες σχετικά με τον καρδιακό παλμό του χρήστη κατά τη διάρκεια δραστηριοτήτων, προπονήσεων ή ακόμα και σε ανάπαυση.

Με βάση τα παραπάνω, η δεύτερη μελλοντική επέκταση που προβλέπεται στην παρούσα διαδικτυακή εφαρμογή, αφορά τη λειτουργικότητα και την περεταίρω αυτοματοποίηση, μέσω του αισθητήρα της κινητής συσκευής για αυτόματη αποστολή των καρδιακών παλμών.

Όσον αφορά το τεχνικό κομμάτι που περιλαμβάνει τον τρόπο με τον οποίο θα γίνεται η λήψη των δεδομένων καρδιακών παλμών από το υλικό της κινητής συσκευής, θα μπορούσε να υλοποιηθεί μέσω των API (Διεπαφή Προγραμματισμού Εφαρμογής) που παρέχουν οι κατασκευαστές και επιτρέπουν στους προγραμματιστές να αποκτήσουν πρόσβαση σε αυτά τα δεδομένα.

Ορισμένοι κατασκευαστές παρέχουν SDKs (Σύνολα Ανάπτυξης Λογισμικού) που καθιστούν ευκολότερη την ενσωμάτωση αυτής της λειτουργικότητας στις εφαρμογές.

Με αυτόν τον τρόπο δεν θα χρειάζεται πλέον να γίνεται καταχώρηση καρδιακών παλμών από τον χρήστη και έτσι η παρούσα διαδικτυακή εφαρμογή θα αναβαθμιζόταν, προσφέροντας παράλληλα υπηρεσίες παρακολούθησης υγείας.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1]** Αβούρης, Ν., Κατσάνος, Χ., Τσέλιος, Ν., & Μουστάκας, Κ. "Εισαγωγή στην Αλληλεπίδραση Ανθρώπου-Υπολογιστή", 2015

- [2]** Preece, J., Rogers, Y., & Sharp, H., "Interaction Design: Beyond Human-Computer Interaction.", Wiley, 2015

- [3]** L. Welling and L. Thomson, "PHP and MySQL Web Development," Addison-Wesley, 2016

- [4]** Lie, H., "CSS and Documents." ACM Transactions on Internet Technology, 2003

- [5]** Viega, J., & McGraw, G., "Building secure software: How to avoid security problems the right way", Addison-Wesley, 2002

- [6]** Σταυρακούδης, Α, "Βάσεις δεδομένων και SQL: Μια πρακτική προσέγγιση - 2η έκδοση", Εκδόσεις Κλειδάριθμος, 2015

- [7]** Λιακέας, Γ., "Η ΓΛΩΣΣΑ JAVASCRIPT (3η ΕΚΔΟΣΗ)", Εκδόσεις Κλειδάριθμος, 2021

- [8]** Arwa, A., Ghaida, A., Sarah, A., " Walking Sensing Using Smart Phones Built-in Sensors", LAP LAMBERT Academic Publishing, 2015

ΠΑΡΑΡΤΗΜΑ Α΄

Στο παράρτημα αυτό παρατίθενται αποσπάσματα από τον κώδικα της παρούσας εφαρμογής, που παρουσιάζουν ιδιαίτερο ενδιαφέρον.

Το αρχείο geolocation.js

Το αρχείο geolocation.js βρίσκεται στον φάκελο js και χρησιμοποιείται σε μια από τις βασικότερες λειτουργίες της εφαρμογής, η οποία είναι η παρακολούθηση της θέσης του χρήστη και η αποθήκευση των γεωγραφικών δεδομένων.

Αναλυτικότερα, πριν ξεκινήσει η χρήση των υπηρεσιών τοποθεσίας, ελέγχεται αν ο browser υποστηρίζει αυτές τις υπηρεσίες. Σε περίπτωση που δεν υποστηρίζονται, εμφανίζεται ένα μήνυμα σφάλματος.

Η συνάρτηση watchCurentPosition ξεκινά την παρακολούθηση της τρέχουσας θέσης του χρήστη, χρησιμοποιώντας τη μέθοδο watchPosition του αντικειμένου navigator.geolocation. Η θέση του χρήστη ενημερώνεται συνεχώς, και κάποιες λειτουργίες εκτελούνται όποτε αλλάζει η θέση.

Όταν η θέση του χρήστη ενημερώνεται, ο χάρτης ανανεώνεται και η νέα θέση του χρήστη εμφανίζεται στον χάρτη. Επίσης, ελέγχονται και ανανεώνονται ορισμένα στοιχεία στη διεπαφή του χρήστη, όπως το εικονίδιο της θέσης και το επίπεδο ζουμ του χάρτη.

Η συνάρτηση CenterPosition χρησιμοποιείται για να επικεντρώσει τον χάρτη στην τρέχουσα θέση του χρήστη.

Τα δεδομένα της τρέχουσας θέσης του χρήστη αποθηκεύονται στη βάση δεδομένων με τη χρήση αιτημάτων HTTP στις συναρτήσεις saveRouteData και getLat, getLon. Τα δεδομένα αποθηκεύονται κάθε 3 δευτερόλεπτα.

Επίσης χρησιμοποιείται η συνάρτηση locError για να εμφανίσει μηνύματα σφαλμάτων στον χρήστη σχετικά τις υπηρεσίες τοποθεσίας.

```
//Geolocation
var marker=null;
var lat=null;
var long=null;
var myLatLng=null;
var Spead=null;
var default_zoom=null;
var driving_marker=null;
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
function watchCurentPosition(){
if (navigator.geolocation) {
var watchID = navigator.geolocation.watchPosition(win, locError,{enableHighAccuracy: true,
maximumAge: 1000});
}
else {
swal({ title: "", text: "Ο φυλλομετρητής σας δεν υποστηρίζει υπηρεσίες τοποθεσίας.", type:
"error", });
}
}

function check_coords(){
var geo_lat = window.lat;
var speed = window.Speed;

if(geo_lat == null){
$('#position_opened').css({'display': 'none'});
$('#driving_mode').css({'display': 'none'});
$('#bmp_opened').css({'display': 'none'});
$('#find_my_position').css({'display': 'block'});
}
else{
map_zoom = map.getZoom();
if(window.driving_marker == 'on'){
$('#find_my_position').css({'display': 'none'});
$('#position_opened').css({'display': 'none'});
$('#bmp_opened').css({'display': 'block'});
$('#driving_mode').css({'display': 'block'});
}
else{
$('#find_my_position').css({'display': 'none'});
$('#driving_mode').css({'display': 'none'});
$('#bmp_opened').css({'display': 'block'});
$('#position_opened').css({'display': 'block'});
}
}
}

function CenterPosition(){
map.setView(marker.getLatLng(),17);
map.invalidateSize();
driving_marker="on";
}

function clearCurentPosition(){
map.setZoom(17);
navigator.geolocation.clearWatch(watchID);
//marker.setVisible(false);
}

function check_position_zoom(){
if(window.lat != null){
map_zoom = map.getZoom();
if(map_zoom == '17'){
map.setZoom(16);
}
}
}
}
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
var win = function(position) {
  lat = position.coords.latitude;
  long = position.coords.longitude;
  myLatLng = new L.LatLng(lat, long);
  myLatLngDef = lat+', '+long;
  Speed = Math.round(position.coords.speed*3.6);
  default_zoom = map.getZoom();
  //var Altitude = Math.round(position.coords.altitude);
  //var Accuracy = position.coords.accuracy;
  //var Altitude_Accuracy = position.coords.altitudeAccuracy;
  //var Heading = position.coords.heading;
  //var Timestamp = position.timestamp;

  if(marker==null){
    var geoMarker_bg = L.Icon.extend({ options: { iconSize: [66, 66], iconAnchor: [35, 35],
popupAnchor: [0, -42] } });
    var geoMarker = L.Icon.extend({ options: { iconSize: [22, 22], iconAnchor: [12, 12],
popupAnchor: [0, -42] } });

    marker_bg = L.marker([lat, long], {icon: new geoMarker_bg({iconUrl:
'img/icons/my_pos_bg.svg'}) }).addTo(map);
    marker = L.marker([lat, long], {icon: new geoMarker({iconUrl: 'img/icons/my_pos.svg'})
}).addTo(map);
    centerPopup(myLatLng);

    driving_marker="on";
    map.setZoom(17);
    //map.panTo(myLatLng);
    map.setView(marker.getLatLng());
    map.setView(marker_bg.getLatLng());

    map.invalidateSize();

    // Write to DataBase the main record of the route
    var xmlhttp = new XMLHttpRequest();
    xmlhttp.open("GET", "lib/add_main_route.php", true);
    xmlhttp.send();
  }
  else{

    marker.setLatLng(myLatLng);
    marker_bg.setLatLng(myLatLng);

    marker_bg.setZIndexOffset(-100);
    marker.setZIndexOffset(1000);

    map.on('zoomstart', function() {
    driving_marker="off";
    });

    map.on('dragstart', function() {
    driving_marker="off";
    })

    if(window.driving_marker == 'on'){
    map.setView(marker.getLatLng());
    }

    check_coords();
    map.invalidateSize();
  }
}
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
}  
};  
  
function saveRouteData(){  
if(lat != null){  
    // Write to DataBase the route data  
    var xmlhttp = new XMLHttpRequest();  
    xmlhttp.open("GET", "lib/add_route.php?lat="+lat+"&lon="+long, true);  
    xmlhttp.send();  
}  
}  
  
function getLat(){  
if(lat != null){  
    let in_lat = document.getElementById("in_lat");  
    //in_lat.innerText = lat;  
    in_lat.value = lat;  
}  
else{  
    in_lat.value = '';  
}  
}  
  
function getLon(){  
if(long != null){  
    let in_lon = document.getElementById("in_lon");  
    //in_lon.innerText = long;  
    in_lon.value = long;  
}  
else{  
    in_lon.value = '';  
}  
}  
  
function locError(error) {  
switch(error.code){  
  
case error.PERMISSION_DENIED:  
swal({  
    title: "",  
    text: "Η Τοποθεσία δεν είναι ενεργοποιημένη στη συσκευή σας. Ενεργοποιήστε την τοποθεσία της  
συσκευής σας και προσπαθήστε ξανά.",  
    type: "error",  
    showCancelButton: true,  
    confirmButtonColor: "#8cd4f5",  
    confirmButtonText: "Βοήθεια",  
    cancelButtonText: "ok",  
    closeOnConfirm: false,  
    closeOnCancel: true  
}),  
function(isConfirm){  
    if (isConfirm) {  
        //document.location.href = "geolocation_help.php";  
        $(".geolocation_help_button").click();  
    }  
    else{
```

```
    //location.reload(false);
  }
});
break;

case error.POSITION_UNAVAILABLE:
swal({ title: "", text: "Δεν υπάρχουν διαθέσιμες πληροφορίες για την τοποθεσία σας.", type:
"error", });
break;

case error.TIMEOUT:
  swal({ title: "", text: "Το αίτημα προβολής της τοποθεσίας σας έληξε.", type: "error", });
break;

case error.UNKNOWN_ERROR:
swal({ title: "", text: "Προέκυψε ένα άγνωστο σφάλμα.", type: "error", });
break;
}
}

function centerPopup(id){

  map.on('popupopen', function(e) {
    var px = map.project(e.popup._latlng);
    px.y -= e.popup._container.clientHeight/1.65;
    //console.log(px);
    map.panTo(map.unproject(px),{animate: true});
  });
}

const interval = setInterval(function() {
  saveRouteData();
}, 3000);
```

Το αρχείο L.Polyline.measuredDistance.js

Το αρχείο L.Polyline.measuredDistance.js βρίσκεται στον φάκελο js/leaflet και χρησιμοποιείται για να υπολογίζει τη συνολική απόσταση μεταξύ των σημείων που ορίζουν μια γραμμή (polyline) σε έναν χάρτη.

```
L.Polyline.prototype.measuredDistance = function (options) {
  // Default options
  this.defaultOptions = L.extend(this.defaultOptions || {}, {
    formatted: true,
    metric: true,
  });
  // Extend options
  this.options = L.extend(this.defaultOptions, this.options, options);

  var distance = 0, coords = null, coordsArray = this._latlngs;

  for (i = 0; i < coordsArray.length - 1; i++) {
    coords = coordsArray[i];
    distance += coords.distanceTo(coordsArray[i + 1]);
  }

  // Return formatted distance
```


Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
    return this.options.formatted === true ? L.PolylineUtil.readableDistance(distance,
this.options.metric) : distance;
};

L.PolylineUtil = {
  readableDistance: function (distance, isMetric) {
    var distanceStr;

    // show metres when distance is < 1km, then show km
    if (distance > 1000) {
      distanceStr = (distance / 1000).toFixed(2) + ' χλμ.';
    } else {
      distanceStr = Math.ceil(distance) + ' μ.';
    }

    return distanceStr;
  }
};
```

Αποσπάσματα κώδικά από το αρχείο scripts.js

Το αρχείο scripts.js βρίσκεται στον φάκελο js και χρησιμοποιείται για να ενσωματώσει βασικό κώδικα javascript που αφορούν την εφαρμογή γενικότερα.

Έλεγχος για κενά πεδία φόρμας:

```
function checkFields(form) {
var checks_radios = form.find(':checkbox, :radio'),
    inputs =
form.find(':input').not(checks_radios).not('[type="submit"],[type="button"],[type="reset"],[type
="range"],[id="in_lat"],[id="in_lon"]');

var range = form.find(':input[type="range"]');
if(range.val() == 0){
  range = 0;
}
else{
  range = 1;
}

var checked = checks_radios.filter(':checked');
var filled = inputs.filter(function(){
  return $.trim($(this).val()).length > 0;
});

if(range + checked.length + filled.length === 0) {
  return false;
}

return true;
}
```

Ασύγχρονη αποστολή στοιχείων φόρμας προς αποθήκευση των δεδομένων:

```
$(function(){
    $('#addNotesForm').on('submit',function(e){
        validateYouTubeUrl(event);
        var oneFilled = checkFields($(this));
        if(oneFilled === false) {
            e.preventDefault();
            //alert('Δεν έχετε εισάγει δεδομένα');
            swal({ title: "", text: "Δεν έχετε εισάγει δεδομένα", type: "error", },
            function(isConfirm){
                if (isConfirm){
                    window.setTimeout(function () {
                        $('html,body').animate({ scrollTop: $(field).offset().top-115 }, 1000);
                        $(field).css({"border-color": "#ea1e63", "border-width": "1px", "border-
style":"solid"});
                        $(field).focus();
                    }, 0);
                }
            });
            return false;
        }
        else{
            e.preventDefault();

            document.body.style.overflow='auto';
            $(".hidden_page").attr("style", "top: -9999px; left: -9999px;");
            $(".loading").css({"display": "block"});
            $(".loading").html("Παρακαλώ περιμένετε...");
            $(".bmp_opened_status").css({"display": "none"});

            var formData = new FormData($("#addNotesForm")[0]);
            $.ajax({
                url: 'lib/add_note.php',
                type: 'post',
                //data: $('#addNotesForm').serialize(),
                data: formData,
                processData: false,
                contentType: false,
                success: function () {
                    //alert('form was submitted');
                    $(".loading").css({"display": "none"});
                    $(".loading").html("");
                    $(".bmp_opened_status").css({"display": "block"});

                    document.body.style.overflow='auto';
                    $(".hidden_page").attr("style", "top: -9999px; left: -9999px;");
                    clear_form_data();
                    $(".success_ok").css({"display": "block"});
                    $(".success_ok").html("Τα δεδομένα αποθηκεύτηκαν!");
                    setTimeout(function(){
                        $(".success_ok").css({"display": "none"});
                        $(".success_ok").html("");
                    }, 3000);
                },
                error: function (request, status, error) {
                    //alert(request.responseText);
                    $(".loading").css({"display": "none"});
                    $(".loading").html("");
                    $(".bmp_opened_status").css({"display": "block"});
                }
            });
        }
    });
});
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
document.body.style.overflow='auto';
$(".hidden_page").attr("style", "top: -9999px; left: -9999px;");
clear_form_data();
$(".success_error").css({"display": "block"});
$(".success_error").html("Τα δεδομένα δεν αποθηκεύτηκαν!");
setTimeout(function(){
    $(".success_error").css({"display": "none"});
    $(".success_error").html("");
}, 3000);
}
});
}
});
});
```

Καθαρισμός των πεδίων και των γραφικών της φόρμας:

```
function clear_form_data(){
    $("#addNotesForm").trigger('reset');
    $("#bpm_val").html("");
    $(".video_object").attr("style", "display: none;");
    $("#add_image_preview_container").attr("style", "display: none;");
    $("#add_image_name").html("");
}
```

Αποσπάσματα κώδικά από το αρχείο styles.css

Το αρχείο styles.css βρίσκεται στον φάκελο css και χρησιμοποιείται για να ενσωματώσει βασικό κώδικα css που αφορούν την εφαρμογή γενικότερα.

Προσαρμοσμένα γραφικά για το πεδίο input type="range" της φόρμας καταχώρησης δεδομένων:

```
/* Custom Range Slider */
.slider-range {
    -webkit-appearance: none;
    width: calc(100% - 7px);
    height: 3px;
    padding: 0px;
    border-radius: 5px;
    background: #ee005d;
    outline: none;
    border: none;
    pointer-events: none; /* Disable pointer events on the track */
}

.slider-range::-webkit-slider-thumb {
    -webkit-appearance: none;
    appearance: none;
    width: 25px;
    height: 25px;
    border-radius: 50%;
    background: #ee005d;
    cursor: pointer;
    pointer-events: auto; /* Enable pointer events on the thumb */
}
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
}  
.slider-range::-moz-range-thumb {  
  width: 25px;  
  height: 25px;  
  border: 0;  
  border-radius: 50%;  
  background: #ee005d;  
  cursor: pointer;  
  pointer-events: auto; /* Enable pointer events on the thumb */  
}  
  
.range_value{  
  color:#4c4c4c;  
  font-weight: bold;  
}
```

Προσαρμοσμένα γραφικά για το πεδίο input type="file" της φόρμας καταχώρησης δεδομένων:

```
/* Upload Image */  
input[type="file"]{  
  display: none;  
}  
.add_image_label{  
  cursor:pointer;  
  color: #ee005d;  
}  
#add_image_name{  
  margin-top: 10px;  
  color:#ee005d;  
}  
.add_image{  
  width: 75px;  
  margin-right: 8px;  
  vertical-align: middle;  
  color: #ee005d;  
  filter: invert(10%) sepia(88%) saturate(5624%) hue-rotate(331deg) brightness(102%)  
  contrast(104%);  
}  
#add_image_preview_container{  
  display: none;  
  margin-top: 10px;  
}  
#add_image_preview{  
  width: 100%;  
  border:1px solid #f7f7f7;  
  border-radius: 5px;  
  -moz-border-radius: 5px;  
  -webkit-border-radius: 5px;  
}
```

Το αρχείο cfg.php

Το αρχείο cfg.php βρίσκεται στον φάκελο lib και χρησιμοποιείται για τον ορισμό των βασικών ρυθμίσεων της εφαρμογής.

```
<?php
header("Cache-Control: no-store, no-cache, must-revalidate, max-age=0");
header("Cache-Control: post-check=0, pre-check=0", false);
header("Pragma: no-cache");

// Data
$app_title = "Διπλωματική ΠΜΣ-ngnda";
$def_map_point = "38.003650,23.677697";
$server_root = $_SERVER["DOCUMENT_ROOT"];
$host = $_SERVER['HTTP_HOST'];
$uploads_folder = "uploads/";
$uploads_dir = $server_root.'/'."diplomatiki/".$uploads_folder;

// Tiles
$map_tile = "https://{s}.basemaps.cartocdn.com/rastertiles/voyager/{z}/{x}/{y}.png";

// Data Base
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "vm";
?>
```

Αποσπάσματα κώδικά από το αρχείο functions.php

Το αρχείο functions.php βρίσκεται στον φάκελο lib και χρησιμοποιείται για να ενσωματώσει βασικό κώδικα php που αφορούν την εφαρμογή γενικότερα.

Σύνδεση με τη βάση δεδομένων:

```
// Connection with Data Base
try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname;charset=utf8", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    //echo "Connected successfully";
} catch(PDOException $e) {
    echo "Connection failed: " . $e->getMessage();
}
```

Διαγραφή των επιπλέον δεδομένων μιας διαδρομής:

```
function notesDelete($conn, $user_id, $route_id, $uploads_dir){

    $records = $conn->query("SELECT image FROM notes WHERE user_id='".$user_id.'" AND
route_id='".$route_id.'" AND image!='' ");
    $records_count = $records->rowCount();
    $note_image = "";

    if($records_count > 0){
        foreach($records as $row) {
            $note_image = $row['image'];
            unlink($uploads_dir.$note_image);
        }
    }
}
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
    }  
  }  
  
  $query="DELETE FROM notes WHERE user_id='".$user_id.'" AND route_id='".$route_id.'";  
  $conn->exec($query);  
  $conn = null;  
}
```

Μετατροπή συντεταγμένων σε όνομα περιοχής (reverse geocoding):

```
function getAddress($lat, $lon){  
  $json =  
  "https://nominatim.openstreetmap.org/reverse?format=json&lat=".$lat."&lon=".$lon."&zoom=27&addressdetails=1";  
  
  $ch = curl_init($json);  
  curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);  
  curl_setopt($ch, CURLOPT_USERAGENT, "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:59.0)  
  Gecko/20100101 Firefox/59.0");  
  $jsonfile = curl_exec($ch);  
  curl_close($ch);  
  
  $RG_array = json_decode($jsonfile,true);  
  
  $display_name = $RG_array['display_name'];  
  //$road = $RG_array['address']['road'];  
  //$town = $RG_array['address']['town'];  
  //$municipality = $RG_array['address']['municipality'];  
  //$county = $RG_array['address']['county'];  
  //$state_district = $RG_array['address']['state_district'];  
  //$state = $RG_array['address']['state'];  
  //$postcode = $RG_array['address']['postcode'];  
  //$country = $RG_array['address']['country'];  
  //$country_code = $RG_array['address']['country_code'];  
  
  return $display_name;  
}
```

Έλεγχος και αποκατάσταση μη έγκυρης διαδρομής:

```
function checkRouteStatus($conn, $user_id, $uploads_dir){  
  // Delete routes with cords_num < 2  
  $records = $conn->query("SELECT * FROM routes WHERE user_id='".$user_id.'" AND cords_num<2 ");  
  $records_count = $records->rowCount();  
  if($records_count > 0){  
    foreach($records as $row) {  
      $route_id = $row['id'];  
      routeDelete($conn, $user_id, $route_id);  
      notesDelete($conn, $user_id, $route_id, $uploads_dir);  
    }  
  }  
}
```

Υπολογισμός της διάρκειας μιας διαδρομής:

```
function getRouteDuration($route_date_start, $route_date_end){
    $start_date = new DateTime($route_date_start);
    $since_start = $start_date->diff(new DateTime($route_date_end));

    $hours = $since_start->h.':';
    $minutes = $since_start->i.':';
    $seconds = $since_start->s;

    $duration = $hours.$minutes.$seconds;

    return $duration;
}
```

Καρδιακοί παλμοί χρήστη:

```
function getNotesBpm($conn, $user_id, $route_id){
    $records = $conn->query("SELECT bpm, date FROM notes WHERE user_id='".$user_id.'" AND
route_id='".$route_id.'" AND (bpm!='' OR bpm!=0) ORDER BY date ASC ");
    $records_count = $records->rowCount();
    $note_bpm = "";

    if($records_count > 0){
        foreach($records as $row) {
            $note_bpm = $row['bpm'];
            echo $note_bpm.', ';
        }
    }
}
```

```
function getNotesBpmTime($conn, $user_id, $route_id){
    $records = $conn->query("SELECT date FROM notes WHERE user_id='".$user_id.'" AND
route_id='".$route_id.'" AND (bpm!='' OR bpm!=0) ORDER BY date ASC ");
    $records_count = $records->rowCount();
    $note_bpm_time = "";

    if($records_count > 0){
        foreach($records as $row) {
            $note_bpm_time = date("H:i:s", strtotime($row['date']));
            echo "".$note_bpm_time.", ";
        }
    }
}
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Οι συντεταγμένες που παρατηρήθηκαν οι μέγιστοι καρδιακοί παλμοί χρήστη:

```
function getNotesBpmMaxPoint($conn, $user_id, $route_id, $max_bpm){
    $records = $conn->query("SELECT lat, lon FROM notes WHERE user_id='".$user_id."' AND
route_id='".$route_id."' AND bpm='".$max_bpm.'" ");
    $records_count = $records->rowCount();
    $note_bpm_lat = "";
    $note_bpm_lon = "";

    if($records_count > 0){
        foreach($records as $row) {
            $note_bpm_lat = $row['lat'];
            $note_bpm_lon = $row['lon'];
            return $note_bpm_lat.', '.$note_bpm_lon;
        }
    }
}
```

Αποσπάσματα κώδικά από το αρχείο places.php

Το αρχείο places.php βρίσκεται στον κεντρικό φάκελο και χρησιμοποιείται για να παρουσιάσει τις περιοχές που επισκέφτηκε ο χρήστης.

Ερώτημα SQL για την εμφάνιση των περιοχών που επισκέφτηκε ο χρήστης:

```
WITH t AS (
SELECT r.*, ROW_NUMBER()
OVER (PARTITION BY route_id ORDER BY date DESC) AS rn
FROM route AS r
)
SELECT * FROM t WHERE rn = 1 AND user_id='".$user_id.'"
ORDER BY date DESC
```

Αποσπάσματα κώδικά από το αρχείο history.php

Το αρχείο history.php βρίσκεται στον κεντρικό φάκελο και χρησιμοποιείται για να παρουσιάσει το ιστορικό διαδρομών.

Σύνθετος κώδικας σε php που συνδυάζει ερωτήματα SQL υπό συνθήκες. Η αρχική κατάσταση είναι η προβολή όλων των διαδρομών. Στην περίπτωση που έχουν τεθεί κριτήρια αναζήτησης, παρουσιάζονται μόνο οι διαδρομές που πληρούν τα κριτήρια αναζήτησης:

```
$select = "SELECT * FROM routes";
$where = "user_id='".$user_id.'"";
$notes = "";
$bpm = "";
$bpm_m = "";

if(isset($_POST["notes"])){
    $notes = $_POST["notes"];
```


Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
}

if(isset($_POST["bpm"])){
    $bpm = $_POST["bpm"];
}

if(isset($_POST["bpm_m"])){
    $bpm_m = $_POST["bpm_m"];
}

if($notes != "" || $bpm != ""){
    $select = "SELECT DISTINCT r.id, r.user_id, r.date FROM routes as r INNER JOIN notes as n ON
r.id=n.route_id";
    $where = "r.user_id='". $user_id. "' ";
}

if($notes != ""){
    $where .= " AND notes!='' AND notes LIKE '%" . $notes. "%' ";
}

if($bpm != ""){
    $where .= " AND (n.bpm!='' OR n.bpm!=0) AND bpm"."$bpm_m"."=" . $bpm. " ";
}

$query = $select. " WHERE ".$where. " ORDER BY date DESC";
```

Το αρχείο add_route.php

Το αρχείο add_route.php βρίσκεται στον φάκελο lib και χρησιμοποιείται για την αποθήκευση των δεδομένων της διαδρομής.

```
<?php
session_start();
if (isset($_SESSION['authenticated']) && ($_SESSION['authenticated'] == '1')){
include("functions.php");

$user_id = $_SESSION['user_id'];
$route_id = getLastRouteId($conn, $user_id);
if($route_id == ""){
    return false;
}

if(isset($_GET["lat"])){
    $lat = mb_substr($_GET["lat"], 0, 7);
}
else{
    return false;
}

if(isset($_GET["lon"])){
    $lon = mb_substr($_GET["lon"], 0, 7);
}
else{
    return false;
}
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
$last_route_lat = getLastRouteLat($conn, $user_id, $route_id);
$last_route_lon = getLastRouteLon($conn, $user_id, $route_id);
$loc_name = '';

if(($lat!=" " && $lat!=$last_route_lat) && ($lon!=" " && $lon!=$last_route_lon)){
$loc_name = getAddress($lat, $lon);

$query="INSERT INTO `route` ( `id` , `user_id` , `route_id` , `lat` , `lon` , `date` , `loc_name` )
VALUES ( ' , '$user_id', '$route_id', '$lat', '$lon', NOW(), '$loc_name' )";
$conn->exec($query);

$last_cords_num = getLastRoutes_cords_num($conn, $user_id, $route_id);
$new_cords_num = $last_cords_num+1;

$query="UPDATE routes SET cords_num = '$new_cords_num' WHERE id='$route_id' ";
$conn->exec($query);

$conn = null;
}

}
?>
```

Το αρχείο add_note.php

Το αρχείο add_note.php βρίσκεται στον φάκελο lib και χρησιμοποιείται για την αποθήκευση των επιπλέον δεδομένων του χρήστη.

```
<?php
session_start();
if (isset($_SESSION['authenticated']) && ($_SESSION['authenticated'] == '1')){

include("functions.php");
$user_id = $_SESSION['user_id'];
$route_id = getLastRouteId($conn, $user_id);
if($route_id == ""){
    return false;
}

$lat = $_POST['lat'];
$lon = $_POST['lon'];
$bpm = htmlspecialchars($_POST['bpm']);
$notes = htmlspecialchars($_POST['note']);
$video = htmlspecialchars($_POST['video']);
$image = htmlspecialchars($_FILES["image"]["size"]);

$error = '';
$newfilename = '';

// Upload Image
if($image > 0){
    $image = $_FILES['image']['name'];
    $target_file = $uploads_dir.basename($image);
    $imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));
    $newfilename = $route_id.'_' .date('Ymd_His').'.'.$imageFileType;

    if (move_uploaded_file($_FILES["image"]["tmp_name"], $uploads_dir.$newfilename)) {
        $image_upload_ok = 1;
    } else {
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
    $error = "Η φωτογραφία δεν καταχωρήθηκε!";
}

}

if($bpm == 0 && $notes == '' && $image == '' && $video == '') {
$error = "Δεν έχετε εισάγει δεδομένα";
}

else{
$query="INSERT INTO `notes` ( `id` , `user_id` , `route_id`, `lat`, `lon` , `date`, `bpm`,
`notes`, `image`, `video` ) VALUES (', '$user_id', '$route_id', '$lat', '$lon', NOW(), '$bpm',
'$notes', '$newfilename', '$video')";
$conn->exec($query);
$conn = null;
$success = "Τα δεδομένα αποθηκεύτηκαν!";
}

}
?>
```

Αποσπάσματα κώδικά από το αρχείο route.php

Το αρχείο route.php βρίσκεται στον κεντρικό φάκελο και χρησιμοποιείται για να παρουσιάσει μια διαδρομή στον χάρτη.

Κώδικας σε java script που συνδυάζει δεδομένα από μεταβλητές php:

```
<script>
var map = L.map('map', {minZoom: 6, zoomControl: false}).setView([<?php echo $def_map_point;
?>], 15);
L.tileLayer('<?php echo $map_tile; ?>').addTo(map);
L.control.zoom({ position: 'bottomright'}).addTo(map);

var polylinePoints = [<?php echo $polyline_points; ?>];

var polyline = L.polyline(polylinePoints,{
    color: '#679ff4',
    weight: 5,
    opacity: .9,
    lineJoin: 'round'
}).addTo(map);

map.fitBounds(polyline.getBounds());

// Returns the distance in metric units
var distance = polyline.measuredDistance();
document.getElementById("distance").innerHTML = distance;

// Show a marker for route end
var geoMarker = L.Icon.extend({ options: { iconSize: [26,26], iconAnchor: [13, 13],
popupAnchor: [0, -16] } });
marker = L.marker([<?php echo $last_route_lat.', '$last_route_lon; ?>], {icon: new
geoMarker({iconUrl: 'img/icons/check.svg'}) }).addTo(map);
marker.bindPopup('<div class="route_title">Η διαδρομή τελείωσε εδώ!</div>Όρα λήξης:
<strong><?php echo $route_time_end; ?></strong><br>Διάρκεια: <strong><?php echo $duration;
?></strong><br>Απόσταση: <strong>'+distance+'</strong>');
</script>
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

```
function focusToRoute(){
map.fitBounds(polyline.getBounds());
}
</script>
```

Αποσπάσματα κώδικά από το αρχείο route_bpm.php

Το αρχείο route_bpm.php βρίσκεται στον κεντρικό φάκελο και χρησιμοποιείται για να παρουσιάσει το γράφημα με τους καρδιακούς παλμούς του χρήστη σε μια διαδρομή, την ελάχιστη και μέγιστη τιμή των καρδιακών παλμών, καθώς και τα σημεία στον χάρτη που παρουσιάστηκαν αυτές.

Κώδικας σε java script που συνδυάζει δεδομένα από συναρτήσεις και μεταβλητές php για την προβολή του γραφήματος καρδιακών παλμών:

```
<script src="js/chart.js"></script>
<canvas id="bpmChart" style="width:100%; max-width:600px; margin-top:15px;"></canvas>

<script>
const xValues = ['00:00', <?php getNotesBpmTime($conn, $user_id, $route_id); ?>];
const yValues = [0, <?php getNotesBpm($conn, $user_id, $route_id); ?>];

new Chart("bpmChart", {
  type: "line",
  data: {
    labels: xValues,
    datasets: [{
      fill: false,
      lineTension: 0,
      backgroundColor: "rgba(238, 0, 93, 1.0)",
      borderWidth: 3,
      borderColor: "rgba(238, 0, 93, 1.0)",
      data: yValues
    }]
  },
  options: {
    legend: {display: false},
    scales: {
      xAxes: [{ gridLines: { display: false } }],
      yAxes: [{
        ticks: {min: 0, max:200, stepSize: 50},
        gridLines: { display: true }
      }]
    }
  }
});
</script>
```

Σχεδιασμός και ανάπτυξη διαδικτυακής εφαρμογής πλοήγησης χρήστη σε χάρτη

Κώδικας σε java script που συνδυάζει δεδομένα από μεταβλητές php για την προβολή του χάρτη στο σημείο που παρατηρήθηκε η μέγιστη τιμή καρδιακών παλμών:

```
<script>
var map = L.map('map_max_bpm', {minZoom: 6, zoomControl: false, dragging:false, scrollWheelZoom:
false }).setView([<?php echo $max_bpm_point; ?>], 18);
L.tileLayer('<?php echo $map_tile; ?>').addTo(map);
L.control.zoom({ position: 'bottomright'}).addTo(map);
// Show marker
var markerIcon = L.Icon.extend({ options: { iconSize: [56,56], iconAnchor: [28, 28],
popupAnchor: [0, -30] } });
var markerIconUrl = 'img/icons/heart_c.svg';
marker = L.marker([<?php echo $max_bpm_point; ?>], {icon: new markerIcon({iconUrl:
markerIconUrl}) }).addTo(map).bindPopup('<div class="route_title"><?php echo $max_bpm; ?>
παλμοί</div>Ωρα: <?php echo $max_bpm_time; ?>');
</script>
```