



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Διαδικτυακή εφαρμογή καταγραφής και παρουσίασης συμβάντων με χρήση μη σχεσιακής βάσης δεδομένων
Όνοματεπώνυμο Φοιτητή	Γιώργος Κουτούζος
Πατρώνυμο	Νικόλαος
Αριθμός Μητρώου	ΜΠΠΛ/11047
Επιβλέπων	Χρήστος Δουληγέρης, Καθηγητής

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Περίληψη

Το Παγκόσμια Διαδίκτυο, ή Internet, αποτελεί το σημαντικότερο και επιδραστικότερο δημιούργημα του ανθρώπου τα τελευταία πενήντα χρόνια. Η εφεύρεση του Παγκόσμιου Ιστού (World Wide Web) και η εύκολη πρόσβαση στο Διαδίκτυο συνέβαλαν στη συντέλεση τεράστιων δομικών αλλαγών σε πάρα πολλούς τομείς της κοινωνίας. Μια δυνατότητα που δεν έχει αναπτυχθεί ιδιαίτερα και θα μπορούσε να προσφερθεί εκτενέστερα μέσω του Internet στους χρήστες του είναι η λεπτομερής καταγραφή και παρουσίαση των εμπειριών τους.

Στην παρούσα μεταπτυχιακή διατριβή παρουσιάζεται μια εφαρμογή η οποία δίνει τη δυνατότητα στους χρήστες της να διατηρούν ένα προσωπικό προφίλ, να καταχωρούν τις διάφορες εμπειρίες τους και τα σημαντικά γι' αυτούς γεγονότα, να επεξεργάζονται αυτές τις καταχωρήσεις και να βλέπουν λίστες αποτελούμενες από αυτές.

Η υλοποίηση της εφαρμογής έγινε με βάση τις αρχές της σχεδιαστικής αρχιτεκτονικής MVC ενώ παράλληλα χρησιμοποιήθηκαν δημοφιλή πλαίσια εργασίας (Codeigniter, Bootstrap), βάσεις δεδομένων (MongoDB) και άλλες σύγχρονες τεχνολογίες (AJAX, JQuery, Google Maps).

Abstract

The Internet is the most important and the most influential human invention since the 1960's. The World Wide Web invention and the easy Internet access contributed to the completion of huge structural changes in many sectors of our society. A feature that is not well developed and could be offered more extensively through the Internet to its users is the detailed recording and presentation of their experiences.

This thesis is about a web application that enables its users to maintain a personal profile, register their experiences and important events, edit these entries and watch lists consisting of them.

The application was implemented using the principles of the MVC architecture and popular frameworks (Codeigniter, Bootstrap), databases (MongoDB) and other modern technologies (AJAX, JQuery, Google Maps).

Περιεχόμενα

1. Εισαγωγή	7
2. Σχετική Βιβλιογραφία	9
Σύνοψη	11
3. Τεχνολογίες	12
3.1 Codeigniter	12
3.2 Η βιβλιοθήκη JQuery	13
3.3 AJAX.....	13
3.4 Bootstrap.....	14
3.5 MongoDB	14
3.6 Η υπηρεσία χαρτών Google Maps.....	16
Σύνοψη	17
4 Αρχιτεκτονική του προτεινόμενου συστήματος.....	18
4.1 Αρχιτεκτονική συστήματος	18
4.1.1 Επίπεδο εξυπηρέτησης HTTP αιτημάτων	18
4.1.2 Επίπεδο πελατών	18
4.1.3 Επίπεδο αποθήκευσης δεδομένων	19
4.2 Ανάλυση και σχεδιασμός εφαρμογής.....	19
4.2.1 Σύλληψη απαιτήσεων	20
4.2.2 Περιπτώσεις χρήσης	22
4.2.2.1 Περίπτωση χρήσης: Εγγραφή στο σύστημα.....	22
4.2.2.2 Περίπτωση χρήσης: Είσοδος στην εφαρμογή	23
4.2.2.3 Περίπτωση χρήσης: Εισαγωγή νέου lifebit	23
4.2.2.4 Περίπτωση χρήσης: Επεξεργασία lifebit.....	24
4.2.2.5 Περίπτωση χρήσης: Διαγραφή lifebit.....	25
4.2.2.6 Περίπτωση χρήσης: Προβολή των lifebits ταξινομημένων με βάση την ημερομηνία συμβάντος.....	25
4.2.2.7 Περίπτωση χρήσης: Προβολή των lifebits ταξινομημένων με βάση την ημερομηνία εισαγωγής	25
4.2.2.8 Περίπτωση χρήσης: Προβολή των lifebits ενός συγκεκριμένου επεισοδίου	26
4.2.2.9 Περίπτωση χρήσης: Χρονική αναζήτηση.....	26

4.2.2.10 Περίπτωση χρήσης: Επεξεργασία στοιχείων προφίλ	26
4.2.2.11 Περίπτωση χρήσης: Αλλαγή κωδικού	27
4.2.3 Διαγράμματα Δραστηριοτήτων	27
4.2.3.1 Διάγραμμα δραστηριοτήτων: Εγγραφή στο σύστημα	28
4.2.3.2 Διάγραμμα δραστηριοτήτων: Είσοδος στην εφαρμογή.....	29
4.2.3.3 Διάγραμμα δραστηριοτήτων: Εισαγωγή lifebit.....	31
4.2.3.4 Διάγραμμα δραστηριοτήτων: Επεξεργασία lifebit	33
4.2.3.5 Διάγραμμα δραστηριοτήτων: Διαγραφή lifebit.....	35
4.2.4 Διαγράμματα Ακολουθίας	36
4.2.4.1 Διάγραμμα ακολουθίας: Εγγραφή χρήστη	37
4.2.4.2 Διάγραμμα ακολουθίας: Είσοδος χρήστη	38
4.2.4.3 Διάγραμμα ακολουθίας: Επεξεργασία lifebit	39
4.2.4.4 Διάγραμμα ακολουθίας: Διαγραφή lifebit.....	40
4.3 Διάταξη δεδομένων	41
Σύνοψη	42
5. Παράδειγμα χρήσης της εφαρμογής.....	43
Σύνοψη	49
6. Επίλογος	50
Σύνοψη	52
7. Βιβλιογραφία	53

1. Εισαγωγή

Το Παγκόσμιο Διαδίκτυο, ή Internet (Leiner, B. et al., 2009), αναφέρεται σε ένα σύνολο υπολογιστών και δικτύων που συνδέονται μεταξύ τους σε ένα παγκόσμιο δίκτυο έτσι ώστε να μπορούν να επικοινωνούν και να μοιράζονται πληροφορίες. Είναι ουσιαστικά ένα δίκτυο δικτύων υπολογιστών και αποτελεί ίσως το μεγαλύτερο εργαλείο που αναπτύχθηκε τις τελευταίες δεκαετίες. Το 1993 το εργαστήριο CERN στην Ελβετία παρουσίασε τον Παγκόσμιο Ιστό (World Wide Web - www) (Leiner, B. et al., 2009) ο οποίος αναπτύχθηκε από τον Tim Berners-Lee. Πρόκειται για ένα σύστημα διασύνδεσης πληροφοριών σε μορφή πολυμέσων (multimedia) που βρίσκονται αποθηκευμένες σε χιλιάδες υπολογιστές του Internet σε ολόκληρο τον κόσμο και παρουσιάσής τους σε ηλεκτρονικές σελίδες. Από τότε, η εξέλιξη του παγκόσμιου ιστού ήταν ραγδαία. Η εφεύρεσή του σε συνδυασμό με την ευκολία απόκτησης πρόσβασης στο Internet προσέλκυσε έναν μεγάλο αριθμό καινούργιων χρηστών και επέφερε μια επανάσταση σε πάρα πολλούς τομείς (επικοινωνία, ψυχαγωγία, εμπόριο, διαφήμιση, έρευνα, εκπαίδευση, μουσική, ιατρική κ.α.)

Ένας τομέας στον οποίο θα μπορούσε να βοηθήσει το Internet είναι η καταγραφή και η παρουσίαση των διαφόρων εμπειριών των χρηστών του. Η ζωή του ανθρώπου αποτελείται από πάρα πολλές, μικρές και μεγάλες εμπειρίες. Ο τόσο μεγάλος αριθμός τους δημιουργεί πολλές φορές την ανάγκη καταγραφής τους αφού η ανθρώπινη μνήμη είναι περιορισμένη και δεν είναι δυνατόν να αποθηκεύει το σύνολό τους. Ένας ακόμα λόγος για τον οποίο θα ήταν χρήσιμη η καταγραφή των διαφόρων γεγονότων είναι οι πολλές δυσκολίες που αντιμετωπίζουν σχεδόν καθημερινά οι άνθρωποι. Όταν έρχονται αντιμέτωποι με δύσκολες καταστάσεις, πολλές φορές καταφεύγουν σε παλαιότερες εμπειρίες τους, είτε ευχάριστες τις οποίες απόλαυσαν είτε δυσάρεστες τις οποίες κατάφεραν να ξεπεράσουν, έτσι ώστε να πάρουν κουράγιο και να ανταπεξέλθουν σε αυτές. Τέλος, η καταγραφή των διαφόρων περιστατικών και καταστάσεων που βιώνει ένας άνθρωπος θα βοηθούσε στην ανάκτηση σχετικών λεπτομερειών που θα μπορούσαν να του φανούν πολύ χρήσιμες.

Εκτός ελαχίστων εξαιρέσεων, όπως η εφαρμογή www.mylifelisted.com, υπάρχει έλλειψη διαδικτυακών εφαρμογών που προσφέρουν στους χρήστες τους τη δυνατότητα καταγραφής και παρουσίασης εμπειριών. Το συγκεκριμένο γεγονός υπήρξε κίνητρο και ουσιαστικά έδωσε την ιδέα για το θέμα της συγκεκριμένης μεταπτυχιακής διατριβής.

Η παρούσα μεταπτυχιακή διατριβή αφορά το σχεδιασμό και την υλοποίηση μιας εφαρμογής η οποία δίνει τη δυνατότητα στους χρήστες της να καταχωρούν γεγονότα προσθέτοντας μια τοποθεσία για κάθε ένα από αυτά έτσι ώστε να ξέρουν πού ακριβώς συνέβη. Μπορούν επίσης να τα επεξεργάζονται, να τα διαγράφουν, να βλέπουν λίστες με αυτά αλλά και να βλέπουν, μαζεμένες σε ένα χάρτη, τις τοποθεσίες των γεγονότων που βρίσκονται σε ένα συγκεκριμένο εύρος ημερομηνιών. Τέλος, οι

χρήστες έχουν τη δυνατότητα να ομαδοποιούν σε επεισόδια τα γεγονότα που καταχωρούν αλλά και να διατηρούν ένα προφίλ στο οποίο μπορούν να επεξεργάζονται τις προσωπικές τους πληροφορίες.

Στο πρώτο κεφάλαιο παρουσιάστηκε η ιδέα και τα βασικά χαρακτηριστικά της εφαρμογής που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας.

Στο δεύτερο κεφάλαιο γίνεται μία συνοπτική παρουσίαση της βιβλιογραφίας η οποία συνέβαλε στην υλοποίηση της εφαρμογής.

Στο τρίτο κεφάλαιο παρουσιάζονται οι τεχνολογίες οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής της παρούσας διπλωματικής εργασίας. Αναφορικά, αυτές είναι το PHP πλαίσιο εργασίας Codeigniter, η Javascript βιβλιοθήκη JQuery, η μέθοδος AJAX, το πλαίσιο εργασίας διεπαφών Bootstrap, το Google Maps και η μη σχεσιακή βάση δεδομένων MongoDB.

Στο τέταρτο κεφάλαιο παρουσιάζεται η αρχιτεκτονική του συστήματος και η διαδικασία σχεδίασης και ανάπτυξης της εφαρμογής με χρήση της γλώσσας μοντελοποίησης UML και της μεθόδου RUP.

Στο πέμπτο κεφάλαιο αναφέρονται μελλοντικές επεκτάσεις και βελτιώσεις που επιδέχεται η εφαρμογή καθώς και τα τελικά συμπεράσματα που βγήκαν από την εκπόνηση της παρούσας μεταπτυχιακής διατριβής.

2. Σχετική Βιβλιογραφία

Σε αυτό το κεφάλαιο παρουσιάζονται συνοπτικά τα συγγράμματα, το αντικείμενο των οποίων είναι σχετικό με θέματα και τεχνολογίες που χρησιμοποιήθηκαν κατά την εκπόνηση της παρούσας διπλωματικής εργασίας και τα οποία μελετήθηκαν και βοήθησαν στην υλοποίησή της.

Getting started with NoSQL Vaish G., 2011

Το σύγγραμμα αυτό ουσιαστικά εισάγει τον αναγνώστη στον κόσμο των NoSQL βάσεων δεδομένων. Αρχικά, γίνεται μια περιγραφή των κυριότερων χαρακτηριστικών τους και των πλεονεκτημάτων που προσφέρουν στους χρήστες τους. Στη συνέχεια, όσον αφορά ένα συγκεκριμένο παράδειγμα που περιγράφεται, παρουσιάζεται η παραδοσιακή προσέγγιση (δηλαδή η προσέγγιση μιας σχεσιακής βάσης) για την οργάνωση του επιπέδου δεδομένων (data layer), τα προβλήματα που μπορεί να εμφανιστούν αν ακολουθηθεί και ο τρόπος που μπορούν να αντιμετωπιστούν αν χρησιμοποιηθεί μια NoSQL βάση. Επίσης, περιγράφονται οι τύποι αποθήκευσης (storage types) των διαφόρων NoSQL βάσεων και τα πλεονεκτήματα του καθενός ενώ γίνεται και μια σύγκριση μεταξύ τους. Εν συνεχεία, παρουσιάζονται κάποια συγκεκριμένα είδη εφαρμογών, αναλύονται οι απαιτήσεις που υπάρχουν στο σχεδιασμό του επιπέδου δεδομένων τους και αναφέρονται τα πλεονεκτήματα και τα μειονεκτήματα που μπορεί να έχει η χρησιμοποίηση μιας βάσης NoSQL στην κάθε περίπτωση. Επιπλέον, γίνεται σύγκριση των διαθέσιμων NoSQL βάσεων και, τέλος, παρουσιάζεται η υλοποίηση μιας εφαρμογής με τη χρήση της μη σχεσιακής βάσης δεδομένων MongoDB.

Το συγκεκριμένο σύγγραμμα βοήθησε τα μέγιστα, κυρίως μέσω της παρουσίασης διαφόρων παραδειγμάτων, στην κατανόηση των χαρακτηριστικών και των βασικών αρχών των NoSQL βάσεων αλλά και των διαφορών τους από τις σχεσιακές. Επίσης, συνέβαλε στην απόφαση να χρησιμοποιηθεί η MongoDB, και όχι κάποια από τις υπόλοιπες NoSQL βάσεις, στην υλοποίηση της παρούσας εφαρμογής.

The Little MongoDB Book Seguin K., 2011

Το σύγγραμμα αυτό αποτελεί ένα πλήρες εγχειρίδιο της μη σχεσιακής βάσης δεδομένων MongoDB. Σκοπός του είναι να έρθει ο αναγνώστης σε επαφή και να πάρει μια πρώτη εικόνα της συγκεκριμένης βάσης. Αρχικά, δίνονται κάποιες οδηγίες εγκατάστασης και εξηγούνται οι βασικοί μηχανισμοί της MongoDB. Στη συνέχεια γίνεται μια παρουσίαση των πιο βασικών query selectors και δίνονται σχετικά

παραδείγματα εφαρμογής τους. Έπειτα, γίνεται μια εισαγωγή και αναλύεται ο τρόπος λειτουργίας των διαθέσιμων CRUD operators (create, read, update, delete) της MongoDB και παράλληλα παρουσιάζονται συγκεκριμένα παραδείγματα χρήσης τους για την καλύτερη κατανόησή τους. Εν συνεχεία, παρουσιάζονται και εξηγούνται κάποια χαρακτηριστικά μοντελοποίησης των δεδομένων στη MongoDB και γίνεται σύγκριση με τα αντίστοιχα των σχεσιακών βάσεων ενώ δίνονται σχετικές κατευθυντήριες γραμμές. Τέλος, γίνεται παρουσίαση κάποιων μηχανισμών και εργαλείων που προσφέρει η MongoDB (indexes, replication, sharding, profiler κ.α.) καθώς και μια γενική σύγκριση με τις σχεσιακές βάσεις δεδομένων ενώ παράλληλα δίνονται συμβουλές σχετικά με το ποιο είδος βάσης πρέπει να επιλέγεται σε κάθε περίπτωση.

Το σύγγραμμα αυτό αποτέλεσε οδηγό εγκατάστασης και λειτουργίας της MongoDB και συνέβαλε στην κατανόηση των βασικών αρχών της και στη σωστή χρήση της στα πλαίσια της παρούσας εφαρμογής.

MongoDB and PHP Francia S., 2012

Το συγκεκριμένο σύγγραμμα αποτελεί έναν οδηγό για τη χρησιμοποίηση της βάσης δεδομένων MongoDB σε συνδυασμό με την PHP για την ανάπτυξη διαδικτυακών εφαρμογών. Αρχικά, παρουσιάζονται τα κυριότερα χαρακτηριστικά της MongoDB και τα πλεονεκτήματά της ενώ επίσης αναλύονται κάποια προβλήματα των σχεσιακών βάσεων τα οποία μπορεί να επιλύσει. Στη συνέχεια, καταγράφονται οι τρόποι με τους οποίους ο χρήστης μπορεί, μέσω της PHP, να καταχωρήσει, ανακτήσει, επεξεργαστεί και διαγράψει δεδομένα μιας τέτοιας βάσης ενώ παρουσιάζονται και σχετικά παραδείγματα για την καλύτερη κατανόηση των παραπάνω λειτουργιών. Τέλος, περιγράφονται κάποιες λειτουργίες της MongoDB όπως ευρετήρια (indexes), regular expressions, aggregation εντολές κ.α. και οι τρόποι με τους οποίους ο χρήστης μπορεί να τις χειριστεί μέσω της PHP.

Το συγκεκριμένο σύγγραμμα, μέσω των πολλών παραδειγμάτων και των αναλυτικών οδηγιών που περιέχει, βοήθησε στη σωστή χρήση της MongoDB μέσω της PHP κατά την υλοποίηση της παρούσας εφαρμογής.

Codeigniter for Rapid PHP Application Development Upton D.

Στο σύγγραμμα αυτό γίνεται μια λεπτομερής παρουσίαση του PHP πλαισίου εργασίας (framework) Codeigniter. Σκοπός του είναι να δώσει τη δυνατότητα σε αρχάριους προγραμματιστές να το μάθουν και να το χρησιμοποιήσουν για την ανάπτυξη διαδικτυακών εφαρμογών. Αρχικά, αναφέρει κάποια γενικά στοιχεία για το Codeigniter και δίνει οδηγίες για την εγκατάστασή του. Κάνει μια παρουσίαση της MVC (Model-View-Controller) αρχιτεκτονικής, εξηγεί πώς το Codeigniter την

εφαρμόζει αλλά και με ποιο τρόπο δέχεται τις αιτήσεις και τις δρομολογεί. Στη συνέχεια, περιγράφει τον τρόπο με τον οποίο ο προγραμματιστής μπορεί να επικοινωνήσει και να διαχειριστεί μια βάση δεδομένων με τη βοήθεια του Codeigniter. Έπειτα, παρουσιάζει αναλυτικά διάφορα χαρακτηριστικά και δυνατότητες που προσφέρει το Codeigniter στους χρήστες του, όπως εύκολη δημιουργία παρουσιάσεων (views), δημιουργία HTML φορμών και validation τους, εύκολη διαχείριση των session μεταβλητών και των PHP αντικειμένων (objects), αποστολή ηλεκτρονικού ταχυδρομείου κ.α.. Τέλος, παρουσιάζει τα testing εργαλεία (tools) που παρέχει το Codeigniter στους χρήστες του και διάφορους helpers για εύκολο χειρισμό ημερομηνιών, αλφαριθμητικών, πινάκων, εικόνων, αρχείων κ.α.

Το συγκεκριμένο σύγγραμμα αποτέλεσε σημείο αναφοράς καθόλη τη διάρκεια υλοποίησης της παρούσας εφαρμογής. Η άριστη τεκμηρίωσή του, οι αναλυτικές οδηγίες και το πλήθος παραδειγμάτων που περιέχει συνέβαλαν στην αξιοποίηση μεγάλου μέρους των δυνατοτήτων που παρέχει το Codeigniter.

Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκαν τα συγγράμματα που μελετήθηκαν για τις ανάγκες της παρούσας διπλωματικής εργασίας καθώς το αντικείμενό τους αφορούσε θέματα με τα οποία αυτή καταπιάνεται. Στο επόμενο κεφάλαιο παρουσιάζονται οι τεχνολογίες που χρησιμοποιήθηκαν, εμβαθύνοντας σε πιο τεχνικές λεπτομέρειες για κάθε μία από αυτές.

3. Τεχνολογίες

Σε αυτό το κεφάλαιο παρουσιάζονται συνοπτικά οι τεχνολογίες οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής. Επιγραμματικά, αυτές είναι το πλαίσιο εργασίας PHP Codeigniter, η εγγραφοκεντρική μη σχεσιακή βάση δεδομένων MongoDB, η Javascript βιβλιοθήκη JQuery, η υπηρεσία Google Maps, η μέθοδος AJAX και το Bootstrap, ένα πλαίσιο εργασίας για την ανάπτυξη της διεπαφής του χρήστη με την εφαρμογή.

3.1 Codeigniter

Το Codeigniter (Urton D.) είναι ένα ανοιχτού κώδικα (open-source) πλαίσιο εργασίας PHP το οποίο χρησιμοποιείται για τη δημιουργία δυναμικών ιστοσελίδων. Επιλέχτηκε για την υλοποίηση της συγκεκριμένης εφαρμογής αφού είναι ελαφρύ, εύχρηστο και εύκολα επεκτάσιμο. Παρέχει ένα μεγάλο αριθμό βιβλιοθηκών, έχει πολύ καλή τεκμηρίωση και μεγάλη κοινότητα υποστήριξης.

Ένας από τους λόγους για τους οποίους η παρούσα εφαρμογή επιλέχτηκε να αναπτυχθεί πάνω στο πλαίσιο εργασίας Codeigniter ήταν για να εκμεταλλευτεί τα οφέλη της αρχιτεκτονικής Μοντέλο-Παρουσίαση-Ελεγκτής (Model-View-Controller, MVC) (Urton D.) πάνω στην οποία στηρίζεται το Codeigniter. Σύμφωνα με αυτήν, η εφαρμογή χωρίζεται σε τρία επίπεδα έτσι ώστε να υπάρχει διαχωρισμός των τριών βασικών λειτουργιών της αρχιτεκτονικής της. Οι λειτουργίες αυτές είναι η απεικόνιση, η επεξεργασία και η πρόσβαση και διαχείριση των δεδομένων.

Το Μοντέλο (Model) είναι το επίπεδο που είναι υπεύθυνο για την επικοινωνία με τη βάση δεδομένων. Περιέχει τις απαραίτητες λειτουργίες για ανάκτηση, εισαγωγή και ενημέρωση των πληροφοριών σε αυτήν. Επικοινωνεί με τον Ελεγκτή (Controller), στέλνει δεδομένα σε αυτόν και δέχεται οδηγίες από αυτόν ώστε να ενημερώνει την κατάστασή του.

Η Παρουσίαση (View) είναι το επίπεδο που σχετίζεται με την απεικόνιση των δεδομένων στο χρήστη. Μια Παρουσίαση είναι συνήθως μια σελίδα HTML ενώ στον Codeigniter μπορεί να είναι ένα μέρος μιας τέτοιας σελίδας. Συνήθως μία Παρουσίαση επικοινωνεί με έναν ελεγκτή, ο οποίος της στέλνει δεδομένα για να τα εμφανίσει.

Ο Ελεγκτής (Controller) ουσιαστικά συνδέει το Μοντέλο με την Παρουσίαση και είναι υπεύθυνος για τη ροή των ενεργειών που πρέπει να γίνουν ώστε να εξυπηρετηθεί μια αίτηση. Δέχεται την αίτηση, παίρνει δεδομένα από το Μοντέλο, τα επεξεργάζεται και τα στέλνει στην Παρουσίαση ώστε να εμφανιστούν στο χρήστη. Επίσης, δίνει οδηγίες στο Μοντέλο ώστε να ενημερώνει την κατάστασή του και είναι υπεύθυνος για τη μεταβίβαση του ελέγχου σε κάποιον άλλο Ελεγκτή.

Η χρησιμοποίηση ενός πλαισίου εργασίας το οποίο στηρίζεται στην αρχιτεκτονική MVC προσφέρει πολλά πλεονεκτήματα. Το βασικότερο είναι ο διαχωρισμός προβλημάτων (separation of concerns): το κάθε ένα από τα τρία επίπεδα στα οποία διαχωρίζεται η εφαρμογή επιτελεί ξεχωριστό έργο, είναι ξεκάθαρα καθορισμένο και δε συμπλέκεται με τα υπόλοιπα. Η επεκτασιμότητα είναι ακόμα ένα πλεονέκτημα αφού είναι εύκολο να προστεθούν καινούργιες λειτουργίες ή να αλλάξουν κάποιες από τις υπάρχουσες. Επιπλέον, εξασφαλίζεται η μέγιστη δυνατή επαναχρησιμοποίηση του κώδικα αλλά και η εύκολη συντήρησή του. Τέλος, η χρήση της MVC αρχιτεκτονικής διευκολύνει σε μεγάλο βαθμό τις διαδικασίες ελέγχου της τελικής εφαρμογής λόγω της μεγάλης ανεξαρτησίας των συστατικών μερών της.

3.2 Η βιβλιοθήκη JQuery

Η JQuery (Chaffer J., Swedberg K., 2013) είναι μια ανοιχτού κώδικα βιβλιοθήκη Javascript η οποία διατίθεται δωρεάν και χρησιμοποιείται για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών που απαιτούν μεγάλη διαδραστικότητα με τους χρήστες τους. Βοηθάει την αλληλεπίδραση μεταξύ Javascript και HTML και στόχος της είναι η απλούστευση του προγραμματισμού σε Javascript μέσω των διαφόρων μηχανισμών και εντολών της. Διαθέτει ένα μεγάλο αριθμό από πρόσθετα (plugins) ενώ, λόγω της μεγάλης επεκτασιμότητάς της, δίνει τη δυνατότητα στους προγραμματιστές να δημιουργήσουν νέα. Επιπλέον, παρέχει πλήρη και πολύ αναλυτική τεκμηρίωση και, όντας μια πολύ διαδεδομένη βιβλιοθήκη Javascript, υποστηρίζεται από μια πολύ μεγάλη και ενεργή κοινότητα χρηστών. Είναι μια μικρού μεγέθους βιβλιοθήκη και ένα από τα πλεονεκτημά της είναι ότι δίνει τη δυνατότητα στο χρήστη να χρησιμοποιήσει σχεδόν το σύνολο των δυνατοτήτων που προσφέρει η Javascript. Κατά την υλοποίηση της παρούσας εφαρμογής έγινε εκτενής χρήση της JQuery με σκοπό τη διαχείριση των χαρτών και των ημερολογίων σε διάφορες σελίδες, την εμφάνιση των διαφόρων μηνυμάτων στους χρήστες, την πραγματοποίηση των AJAX κλήσεων, τη σελιδοποίηση των λιστών κ.α.

3.3 AJAX

Το AJAX (Asynchronous Javascript and XML) (Keith J.) είναι μία τεχνολογία που χρησιμοποιείται ευρέως σε διαδικτυακές εφαρμογές που αλληλεπιδρούν με το χρήστη. Προσφέρει τη δυνατότητα επικοινωνίας και ανταλλαγής δεδομένων μεταξύ του πελάτη (client) και του εξυπηρετητή χωρίς να είναι απαραίτητη η ανανέωση της σελίδας. Ουσιαστικά δίνει τη δυνατότητα μερικής ανανέωσης του περιεχομένου που προβάλλεται σε μια σελίδα χωρίς αυτή να χρειάζεται να ξαναφορτώσει. Ως αποτέλεσμα, οι εφαρμογές που χρησιμοποιούν AJAX είναι πολύ γρήγορες, δυναμικές

και έχουν πολλές δυνατότητες αλληλεπίδρασης με το χρήστη. Τέλος, το AJAX συμβάλλει καθοριστικά στη μείωση της χρήσης εύρους ζώνης (reduced bandwidth usage) λόγω της μειωμένης κίνησης από και προς τον εξυπηρετητή. Στα πλαίσια της παρούσας εφαρμογής η συγκεκριμένη τεχνολογία χρησιμοποιήθηκε στην υλοποίηση της λειτουργίας της χρονικής αναζήτησης, στην υποβολή όλων των φορμών της και στην υλοποίηση της λειτουργίας διαγραφής ενός lifebit.

3.4 Bootstrap

Το Bootstrap είναι ένα ισχυρό πλαίσιο εργασίας διεπαφών (front-end framework) το οποίο χρησιμοποιείται για την ταχύτερη και ευκολότερη ανάπτυξη ιστοσελίδων. Βασίζεται στις τεχνολογίες HTML, CSS, Javascript και υποστηρίζει όλους τους φυλλομετρητές και σχεδόν όλες τις εκδόσεις τους. Επιλέχτηκε για την υλοποίηση της εφαρμογής γιατί είναι κατάλληλο για νέους προγραμματιστές αφού είναι πολύ εύκολο στην εγκατάσταση και στη χρήση. Προσφέρει ένα σταθερό και ευέλικτο σύστημα διάταξης (grid system) το οποίο είναι αποκρίσιμο σε όλες τις διαστάσεις του φυλλομετρητή. Ως αποτέλεσμα, το Bootstrap είναι κατάλληλο για ανάπτυξη εφαρμογών φιλικών προς τα κινητά και τις ταμπλέτες (tablets). Περιλαμβάνει πλήθος από συστατικά ιστού (web components), εικονίδια σε μορφή συμβολοσειρών (glyph icons) και είναι επεκτάσιμο με τη βοήθεια των Javascript plugins που διαθέτει. Παρέχει πολλές κλάσεις CSS οι οποίες μπορούν να χρησιμοποιηθούν σε πίνακες, φόρμες, κουμπιά, εικόνες και άλλα στοιχεία ώστε να διευκολύνουν σε μεγάλο βαθμό την παραγωγή των ζητούμενων αποτελεσμάτων. Τέλος, έχει πολύ καλό εγχειρίδιο (documentation) και μια μεγάλη και δυναμική κοινότητα χρηστών.

3.5 MongoDB

Η MongoDB (Seguin K., 2011) είναι ένα από τα πιο γνωστά open source document oriented database systems και αποτελεί κομμάτι των NoSQL βάσεων δεδομένων. Οι NoSQL βάσεις δεδομένων (Vaish G., 2011) ονομάζονται έτσι από το γεγονός ότι διαφέρουν αρκετά από τις παραδοσιακές σχεσιακές βάσεις. Τα πλεονεκτήματά τους έναντι των σχεσιακών βάσεων, τα οποία περιγράφονται παρακάτω, συνέβαλλαν στην επιλογή μιας βάσης αυτού του είδους για την υλοποίηση της παρούσας εφαρμογής.

Στις σχεσιακές βάσεις, τα δεδομένα διαχωρίζονται σε αλληλένδετους πίνακες οι οποίοι αποτελούνται από γραμμές και στήλες και αναφέρονται μεταξύ τους μέσω των ξένων κλειδιών (foreign keys). Έτσι, η ανάγνωση και η εγγραφή δεδομένων συχνά απαιτεί συνεργασία και εκτέλεση από πολλούς διαφορετικούς πίνακες, διαδικασία η οποία πολλές φορές είναι αρκετά χρονοβόρα. Αντίθετα, οι NoSQL βάσεις λειτουργούν

τελείως διαφορετικά. Για παράδειγμα, μια εγγραφοστρεφής (document oriented) βάση δεδομένων, όπως η MongoDB, αθροίζει σε έγγραφα τα δεδομένα που πρέπει να αποθηκευτούν χρησιμοποιώντας JSON. Το κάθε έγγραφο JSON μπορεί να θεωρηθεί ως ένα αντικείμενο που χρησιμοποιείται από την εφαρμογή. Παρόλο που με αυτόν τον τρόπο μπορεί να δημιουργηθούν διπλές εγγραφές, η λογική αυτή προτιμάται για εφαρμογές βασισμένες στο διαδίκτυο οι οποίες πρέπει να διαχειριστούν μεγάλο πλήθος δεδομένων λόγω του φθηνού αποθηκευτικού χώρου, της βελτίωσης της απόδοσης των λειτουργιών ανάγνωσης και εγγραφής και της εύκολης κατανομής των προκυπτόντων εγγράφων.

Μία ακόμη σημαντική διαφορά είναι πως οι σχεσιακές βάσεις δεδομένων έχουν άκαμπτα σχήματα ενώ το μοντέλο των NoSQL έχει δυναμικό σχήμα (dynamic schema). Η σχεσιακή τεχνολογία απαιτεί αυστηρό καθορισμό ενός σχήματος πριν από την αποθήκευση δεδομένων σε μία βάση. Η αλλαγή του σχήματος, αφού τα δεδομένα έχουν εισαχθεί, είναι προβληματική και συνήθως αποφεύγεται, κάτι το οποίο αποτελεί μεγάλο πρόβλημα στην εποχή των μεγάλων δεδομένων (big data), κατά την οποία οι προγραμματιστές εφαρμογών χρειάζεται να εισάγουν συνεχώς και ταχέως νέους τύπους δεδομένων για τον εμπλουτισμό των εφαρμογών τους. Αντιθέτως, οι βάσεις δεδομένων εγγράφων έχουν δυναμικό σχήμα, γεγονός που τις καθιστά ιδανικές για την ελεύθερη προσθήκη πεδίων σε JSON έγγραφα χωρίς να απαιτείται πρώτα καθορισμός των αλλαγών. Έτσι, διευκολύνεται κατά πολύ η δημιουργία οποιασδήποτε βάσης και η μετέπειτα διαχείριση και συντήρησή της.

Ένα ακόμα βασικό χαρακτηριστικό των NoSQL βάσεων δεδομένων είναι η δυνατότητα του αυτόματου κατακερματισμού (auto-sharding), δηλαδή η δυνατότητα που έχουν να διανέμουν αυτόματα τα δεδομένα και τα ερωτήματα σε διάφορους εξυπηρετητές χωρίς να υπάρχει η απαίτηση η εφαρμογή να έχει ορίσει από πριν τη διάταξή τους. Σε περίπτωση κατάρρευσης ενός εξυπηρετητή, υπάρχει η δυνατότητα γρήγορης αντικατάστασής του, χωρίς να επηρεαστεί η λειτουργία της εφαρμογής.

Τέλος, ένα από τα πλεονεκτήματα των βάσεων αυτού του είδους είναι οι ενσωματωμένες δυνατότητες προσωρινής αποθήκευσης (caching) που διαθέτουν, δηλαδή η δυνατότητα διατήρησης στη μνήμη του συστήματος των δεδομένων που χρησιμοποιούνται πιο συχνά από το χρήστη. Με αυτό τον τρόπο εξαλείφεται η ανάγκη διατήρησης ενός ξεχωριστού επιπρόσθετου επιπέδου προσωρινής αποθήκευσης.

Στα πλαίσια της παρούσας εφαρμογής επιλέχθηκε να χρησιμοποιηθεί η MongoDB η οποία αποτελεί μια από τις πιο γνωστές και ευρέως διαδεδομένες NoSQL βάσεις δεδομένων. Ως τέτοια δεν ακολουθεί το κλασικό σχεσιακό μοντέλο και αποθηκεύει τα δεδομένα σε μορφή αρχείων BSON (Binary JSON). Η συγκεκριμένη μορφή είναι μια δυαδική αναπαράσταση των JSON εγγράφων αλλά περιέχει πολύ περισσότερους τύπους δεδομένων από το JSON.

Κάθε σύστημα MongoDB μπορεί να φιλοξενεί πολλές βάσεις. Κάθε μια από αυτές αποτελείται από συλλογές αρχείων (collections) οι οποίες είναι οι αντίστοιχοι

πίνακες των σχεσιακών βάσεων. Κάθε συλλογή περιέχει ένα σύνολο από BSON έγγραφα (documents), κάθε ένα από τα οποία θα μπορούσε να χαρακτηριστεί ως το αντίστοιχο μιας εγγραφής (record) ενός σχεσιακού πίνακα. Κάθε έγγραφο αποτελείται από μια σειρά από ζεύγη κλειδιού-τιμής (key-value pairs). Ουσιαστικά, το κλειδί αποτελεί το όνομα του πεδίου και η τιμή το περιεχόμενό του. Η τιμή ενός πεδίου μπορεί να είναι οποιουδήποτε τύπου δεδομένων BSON, όπως αλφαριθμητικά, αριθμοί, άλλα έγγραφα, αναφορές σε άλλα έγγραφα, πίνακες (arrays), πίνακες από έγγραφα κ.α.. Η συγκεκριμένη δομή επιτρέπει την αποτύπωση, μέσα στο ίδιο έγγραφο μίας οντότητας, μια σειρά από δεδομένα και σχέσεις που την αφορούν. Με αυτόν τον τρόπο η ανάκτηση των συγκεκριμένων δεδομένων που αφορούν την οντότητα αυτή γίνεται εύκολα και αποφεύγονται πολλαπλά joins μεταξύ πινάκων και πολλαπλές αναγνώσεις από το δίσκο προκειμένου να συγκεντρωθούν τα αποτελέσματα. Επιπλέον, η δυνατότητα χρησιμοποίησης ενός εγγράφου ως τιμή πεδίου ενός άλλου εγγράφου επιτρέπει και διευκολύνει τη δημιουργία συσχετίσεων μεταξύ τους.

Ένα πολύ βασικό χαρακτηριστικό των βάσεων MongoDB, όπως και όλων των NoSQL βάσεων όπως αναφέρθηκε παραπάνω, είναι το δυναμικό σχήμα των εγγράφων τους. Αυτό σημαίνει ότι τα έγγραφα που περιέχονται μέσα σε μια συγκεκριμένη συλλογή δεν είναι απαραίτητο να έχουν το ίδιο σύνολο πεδίων και την ίδια δομή. Ακόμα και κοινά πεδία μιας συλλογής εγγράφων μπορούν να περιέχουν διαφορετικούς τύπους δεδομένων. Έτσι, είναι πολύ εύκολη η προσθήκη, η μετονομασία ή η αλλαγή κάποιου πεδίου.

Τέλος, ένα σημαντικό πλεονέκτημα της MongoDB είναι η δυνατότητα χρήσης ευρετηρίων (indexes), δηλαδή μιας δομής δεδομένων η οποία διευκολύνει το γρήγορο εντοπισμό εγγράφων βάσει των αποθηκευμένων τιμών συγκεκριμένων πεδίων. Έτσι, αυξάνεται η απόδοση των διαδικασιών ανάγνωσης όταν πρόκειται για συχνά επαναλαμβανόμενα ερωτήματα προς τη βάση.

3.6 Η υπηρεσία χαρτών Google Maps

Το google maps είναι μια υπηρεσία χαρτών που παρέχεται από την Google και δίνει τη δυνατότητα στους χρήστες της να προβάλλουν στον browser τους χάρτες από όλο τον κόσμο και να περιηγηθούν σε αυτούς, να δουν πληροφορίες τοπικών επιχειρήσεων, να βρουν τη συντομότερη διαδρομή μεταξύ δύο σημείων κ.α.. Η Google δημιούργησε επίσης το Google Maps API, ένα Javascript API αποτελούμενο από Ιδιότητες (Properties), Κλάσεις (Classes), Συμβάντα (Events) και Μεθόδους (Methods), ώστε να επιτρέψει στους προγραμματιστές, μέσω της σωστής χρήσης των παραπάνω, να ενσωματώσουν και να διαχειριστούν το google maps στις διαδικτυακές εφαρμογές τους. Στα πλαίσια της παρούσας εφαρμογής η συγκεκριμένη τεχνολογία χρησιμοποιήθηκε στην επιλογή και στην παρουσίαση της τοποθεσίας του κάθε Lifebit,

στην επιλογή και στην παρουσίαση του τόπου γέννησης του κάθε χρήστη και στην παρουσίαση των lifebits στα πλαίσια της λειτουργίας της χρονικής αναζήτησης.

Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκαν οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής της παρούσας διπλωματικής εργασίας. Στο επόμενο κεφάλαιο θα περιγραφεί λεπτομερώς η αρχιτεκτονική του συστήματος της εφαρμογής που υλοποιήθηκε βάσει των τεχνολογιών που επιλέχτηκαν, η διαδικασία ανάλυσης και σχεδίασης με τη μέθοδο RUP και διαγραμμάτων UML και η διάταξη των δεδομένων της εφαρμογής.

4 Αρχιτεκτονική του προτεινόμενου συστήματος

Στο κεφάλαιο αυτό παρουσιάζεται και αναλύεται η αρχιτεκτονική του συστήματος. Αρχικά, αναφέρονται τα επίπεδα που την αποτελούν και, στη συνέχεια, παρουσιάζεται η διαδικασία ανάλυσης και σχεδιασμού της εφαρμογής με τη χρήση διαγραμμάτων ακολουθίας, δραστηριοτήτων και περιπτώσεων χρήσης της γλώσσας μοντελοποίησης UML. Τέλος, παρουσιάζεται ο τρόπος διάταξης των δεδομένων (data layout) της εφαρμογής που ακολουθήθηκε.

4.1 Αρχιτεκτονική συστήματος

Τα επίπεδα που αποτελούν την εφαρμογή της παρούσας διπλωματικής εργασίας είναι τα εξής: επίπεδο εξυπηρέτησης HTTP αιτημάτων, επίπεδο πελατών και επίπεδο αποθήκευσης δεδομένων.

4.1.1 Επίπεδο εξυπηρέτησης HTTP αιτημάτων

Το επίπεδο αυτό είναι υπεύθυνο για την εξυπηρέτηση των αιτημάτων που στέλνονται από το επίπεδο πελατών, δηλαδή από τους χρήστες, μέσω του πρωτοκόλλου HTTP με τη βοήθεια ενός φυλλομετρητή (web browser). Αποτελείται από τον Apache, ένα ανοιχτού κώδικα λογισμικό εξυπηρετητή (server), ο οποίος είναι υπεύθυνος για τη λήψη των αιτήσεων, την επεξεργασία τους και τη δημιουργία και αποστολή των κατάλληλων απαντήσεων στο φυλλομετρητή. Το επίπεδο αυτό ουσιαστικά περιέχει όλα τα αρχεία της εφαρμογής, δηλαδή όλα τα αρχεία του πλαισίου εργασίας Codeigniter, τα PHP αρχεία, τις παρουσιάσεις (views), τα CSS και Javascript αρχεία. Μια απάντηση σε ένα αίτημα που αποστέλλεται σε ένα φυλλομετρητή μπορεί να αποτελείται από δεδομένα, κώδικα HTML, CSS και Javascript.

4.1.2 Επίπεδο πελατών

Το επίπεδο αυτό αποτελείται από ένα φυλλομετρητή και είναι υπεύθυνο για την επικοινωνία και την αλληλεπίδραση του χρήστη με την εφαρμογή. Στόχος του φυλλομετρητή είναι η αποστολή των HTTP αιτημάτων στον εξυπηρετητή, η λήψη των αντίστοιχων απαντήσεων και η προβολή των σελίδων που προκύπτουν από την επεξεργασία των δεδομένων και των αρχείων HTML, CSS και Javascript που περιέχονται στις απαντήσεις και φτάνουν σε αυτόν.

4.1.3 Επίπεδο αποθήκευσης δεδομένων

Το επίπεδο αυτό είναι υπεύθυνο για την αποθήκευση των δεδομένων που εισάγει ο χρήστης στην εφαρμογή. Η MongoDB αποθηκεύει τα δεδομένα σε μορφή BSON, μια δυαδική μορφή των εγγράφων JSON, ενώ, όπως όλες οι NoSQL βάσεις, έχει την ενσωματωμένη δυνατότητα προσωρινής αποθήκευσης (caching), δηλαδή τη δυνατότητα διατήρησης στη μνήμη των δεδομένων που χρησιμοποιούνται πιο συχνά από το χρήστη.

4.2 Ανάλυση και σχεδιασμός εφαρμογής

Η ανάλυση και ο σχεδιασμός της εφαρμογής έγινε με τη χρήση της UML (Unified Modeling Language) (Schmuller J.), μιας γραφιστικής γλώσσας μοντελοποίησης που χρησιμοποιείται για τη σχηματική αναπαράσταση των προδιαγραφών και των απαιτήσεων μιας εφαρμογής καθώς και για τη δημιουργία και τεκμηρίωση των τμημάτων ενός συστήματος λογισμικού. Η UML ορίζει ένα σύνολο διαγραμμάτων, σκοπός των οποίων είναι η αναπαράσταση των διαφορετικών απόψεων του συστήματος κατά τη διάρκεια της ανάπτυξης και της υλοποίησης ενός λογισμικού. Τα παρεχόμενα διαγράμματα είναι τα εξής:

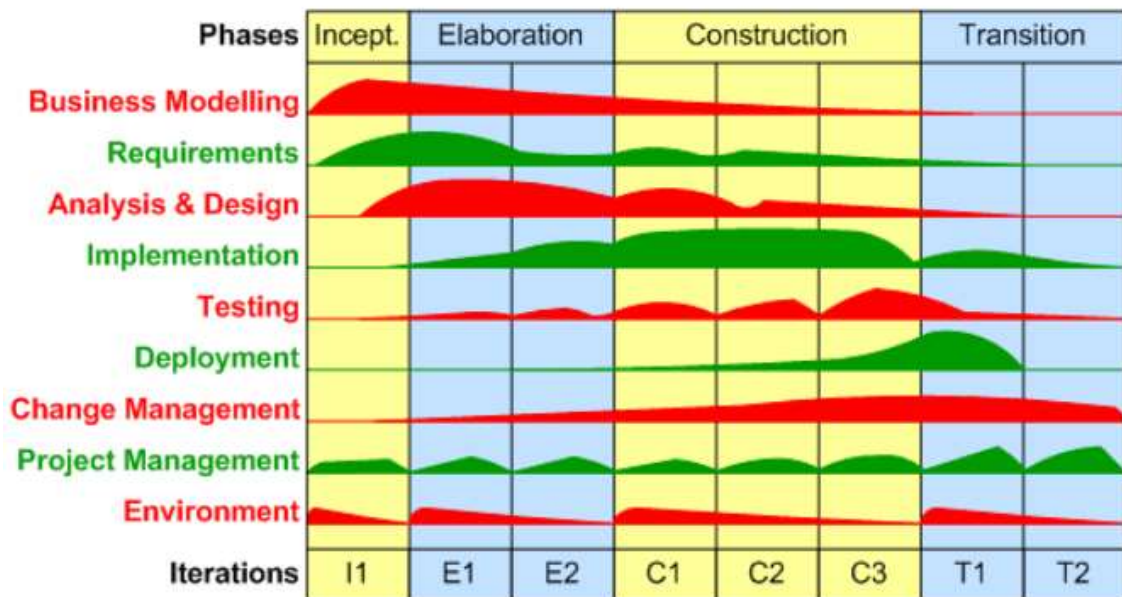
- Διαγράμματα περιπτώσεων χρήσης (use case)
- Διαγράμματα δομής
 - o Διαγράμματα κλάσεων (class)
 - o Διαγράμματα αντικειμένων (object)
- Διαγράμματα συμπεριφοράς
 - o Διαγράμματα καταστάσεων (state chart)
 - o Διαγράμματα δραστηριοτήτων (activity)
 - o Διαγράμματα αλληλεπίδρασης
 - Διαγράμματα ακολουθίας (sequence)
 - Διαγράμματα συνεργασίας (collaboration)
- Διαγράμματα δομής υλοποίησης
 - o Διαγράμματα εξαρτημάτων (component)
 - o Διαγράμματα ανάπτυξης (deployment)

Το μοντέλο RUP (Rational Unified Process) (Shuja A.) έχει αναπτυχθεί από τους δημιουργούς της UML και συνίσταται ως διαδικασία ανάπτυξης λογισμικού. Η διαδικασία RUP χρησιμοποιεί τη UML ως γλώσσα μοντελοποίησης, αφορά την ανάλυση απαιτήσεων και το σχεδιασμό ενός λογισμικού και χωρίζεται σε τέσσερις φάσεις:

Διαδικτυακή εφαρμογή καταγραφής και παρουσίασης συμβάντων με χρήση μη σχεσιακής βάσης δεδομένων

- Έναρξη (Inception): Καθορισμός του έργου
- Επεξεργασία (Elaboration): Σχεδιασμός μεθόδου υλοποίησης του έργου, καθορισμός χαρακτηριστικών του έργου, σχεδιασμός της αρχιτεκτονικής του συστήματος
- Κατασκευή (Construction): Υλοποίηση του έργου
- Μετάβαση (Transition): Ανάπτυξη του προϊόντος στην κοινότητα χρηστών

Η RUP είναι δομημένη σε δύο διαστάσεις, το χρόνο και τα τμήματα της διαδικασίας. Το παρακάτω σχήμα παρουσιάζει μια γραφική αναπαράσταση των διαστάσεων αυτών. Ο οριζόντιος άξονας αναπαριστά το χρόνο και παρουσιάζει τις πτυχές του κύκλου ζωής της διαδικασίας ανάπτυξης λογισμικού. Η διάσταση αυτή περιγράφεται σε σχέση με τις φάσεις (phases) και τις επαναλήψεις (iterations). Ο κάθετος άξονας παρουσιάζει το περιεχόμενο και δείχνει τους κλάδους-τομείς που ομαδοποιούν λογικά τη διαδικασία του περιεχομένου.



Κύκλος ζωής ανάπτυξης λογισμικού

4.2.1 Σύλληψη απαιτήσεων

Στην ενότητα αυτή παρουσιάζονται οι λειτουργικές απαιτήσεις της εφαρμογής.

- Ένας νέος χρήστης θα πρέπει να πραγματοποιήσει εγγραφή και να δημιουργήσει προσωπικό λογαριασμό εισόδου. Ο λογαριασμός αυτός θα αποτελείται από μια διεύθυνση ηλεκτρονικού ταχυδρομείου και έναν κωδικό.
- Ένας ήδη εγγεγραμμένος χρήστης θα πρέπει να υποβάλει την καταχωρημένη διεύθυνση ηλεκτρονικού ταχυδρομείου και το σωστό κωδικό ώστε να εισέλθει στην εφαρμογή.
- Στην αρχική σελίδα της εφαρμογής ο χρήστης θα πρέπει να βλέπει το συνολικό αριθμό των καταχωρήσεών του (lifebits) και την ημερομηνία εισαγωγής του τελευταίου lifebit.
- Στην κορυφή κάθε σελίδας θα πρέπει να υπάρχει μια αναπτυσσόμενη λίστα στο δεξί μέρος από όπου ο χρήστης θα μπορεί να επιλέξει είτε να προβάλει τα στοιχεία του προφίλ του είτε να αποσυνδεθεί από την εφαρμογή. Επίσης, θα πρέπει να υπάρχει ένα κεντρικό μενού από το οποίο ο χρήστης θα μπορεί να πλοηγηθεί στις εξής σελίδες:
 - Αρχική
 - Η ιστορία μου (λίστα lifebits ταξινομημένων με βάση την ημερομηνία συμβάντος)
 - Πρόσφατα lifebits (λίστα lifebits ταξινομημένων με βάση την ημερομηνία καταχώρησης)
 - Νέο lifebit
 - Χρονική αναζήτηση
- Ο χρήστης θα μπορεί να εισάγει ένα νέο lifebit.
- Ο χρήστης θα μπορεί να επεξεργάζεται ένα ήδη καταχωρημένο lifebit.
- Ο χρήστης θα μπορεί να διαγράφει ένα ήδη καταχωρημένο lifebit.
- Το κάθε lifebit θα έχει τα εξής χαρακτηριστικά:
 - Περιγραφή συμβάντος
 - Τοποθεσία συμβάντος (προαιρετικά)
 - Θέση του συμβάντος στο χάρτη (προαιρετικά)
 - Ημερομηνία συμβάντος
 - Το χρήστη που το καταχώρησε
- Ο χρήστης θα μπορεί να βλέπει τα στοιχεία προφίλ του (όνομα, επώνυμο κ.τ.λ.) και να τα επεξεργάζεται ενώ, από την ίδια σελίδα, θα μπορεί να αλλάζει τον κωδικό του.
- Στο δεξί μέρος της αρχικής σελίδας και των σελίδων που περιέχουν λίστες από lifebits θα πρέπει να υπάρχει μια φόρμα γρήγορης εισαγωγής lifebit.

- Η υποβολή στο σύστημα των διαφόρων φορμών της εφαρμογής θα πρέπει να γίνεται με τη μέθοδο AJAX. Οι περιπτώσεις αυτές είναι οι εξής:
 - o Εισαγωγή νέου lifebit
 - o Εισαγωγή νέου lifebit από τη φόρμα γρήγορης εισαγωγής
 - o Επεξεργασία ήδη καταχωρημένου lifebit
 - o Διαγραφή ήδη καταχωρημένου lifebit
 - o Χρονική αναζήτηση
- Ο χρήστης θα μπορεί να ομαδοποιεί όσα lifebits επιθυμεί σε επεισόδια αναφέροντας το όνομα του επεισοδίου στην περιγραφή του lifebit προς ομαδοποίηση κατά την εισαγωγή του. Επίσης, θα έχει τη δυνατότητα να βλέπει σε μορφή λίστας όλα τα lifebits ενός συγκεκριμένου επεισοδίου.

4.2.2 Περιπτώσεις χρήσης

Οι παρακάτω περιπτώσεις χρήσης, οι οποίες περιγράφουν λειτουργικότητες του συστήματος από την οπτική γωνία του χρήστη, προέκυψαν από την ανάλυση των απαιτήσεων.

4.2.2.1 Περίπτωση χρήσης: Εγγραφή στο σύστημα

Βασική ροή

- Το σύστημα εμφανίζει την οθόνη εισόδου-εγγραφής.
- Ο χρήστης πληκτρολογεί τη διεύθυνση ηλεκτρονικού ταχυδρομείου και τον επιθυμητό κωδικό στα αντίστοιχα πεδία και πατάει το κουμπί “Εγγραφή”
- Το σύστημα δημιουργεί ένα νέο χρήστη.
- Ο χρήστης συνδέεται αυτόματα στο σύστημα και μεταφέρεται στην αρχική σελίδα.

Εναλλακτική ροή 1

- Η διεύθυνση ηλεκτρονικού ταχυδρομείου που πληκτρολόγησε ο χρήστης χρησιμοποιείται ήδη. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η εισαγωγή διαφορετικής διεύθυνσης.

Εναλλακτική ροή 2

- Ο χρήστης δε συμπλήρωσε τουλάχιστον ένα από τα απαιτούμενα πεδία ή η τιμή του πεδίου “Επιβεβαίωση κωδικού” δεν είναι ίδια με την τιμή του πεδίου “Κωδικός” ή η συμπλήρωση των πεδίων δεν είναι σύμφωνη με τις απαιτήσεις

του συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

4.2.2.2 Περίπτωση χρήσης: Είσοδος στην εφαρμογή

Βασική ροή

- Το σύστημα εμφανίζει την οθόνη εισόδου-εγγραφής.
- Ο χρήστης πληκτρολογεί τη διεύθυνση ηλεκτρονικού ταχυδρομείου και τον κωδικό στα αντίστοιχα πεδία και πατάει το κουμπί “Σύνδεση”.
- Ο χρήστης συνδέεται στο σύστημα και μεταφέρεται στην αρχική σελίδα.

Εναλλακτική ροή

- Ο χρήστης δε συμπλήρωσε τουλάχιστον ένα από τα δύο απαιτούμενα πεδία ή η συμπλήρωση των πεδίων δεν είναι σύμφωνη με τις απαιτήσεις του συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

4.2.2.3 Περίπτωση χρήσης: Εισαγωγή νέου lifebit

α. Εισαγωγή νέου lifebit από τη σελίδα νέου lifebit της εφαρμογής

Βασική ροή

- Ο χρήστης πατάει από το κεντρικό μενού την επιλογή “Νέο lifebit”.
- Το σύστημα εμφανίζει τη σελίδα εισαγωγής νέου lifebit.
- Ο χρήστης συμπληρώνει τα πεδία της φόρμας (περιγραφή, ημερομηνία) και, προαιρετικά, επιλέγει μια τοποθεσία στο χάρτη.
- Αποστέλλει τα στοιχεία πατώντας το κουμπί “Δημιούργησε νέο lifebit”.
- Το σύστημα δέχεται την αίτηση και αποθηκεύει το καινούργιο lifebit.
- Εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και μεταφέρει το χρήστη στη σελίδα “Η ιστορία μου”.

Εναλλακτική ροή

- Ο χρήστης δε συμπλήρωσε τουλάχιστον ένα από τα απαιτούμενα πεδία ή η συμπλήρωση των πεδίων δεν ήταν σύμφωνη με τις απαιτήσεις του συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

β. Εισαγωγή νέου lifebit από τη φόρμα γρήγορης εισαγωγής lifebit

Βασική ροή

- Ο χρήστης βρίσκεται σε μια από τις σελίδες της εφαρμογής στις οποίες εμφανίζεται η φόρμα γρήγορης εισαγωγής lifebit.
- Συμπληρώνει τα πεδία της φόρμας (περιγραφή, ημερομηνία).
- Αποστέλλει τα στοιχεία πατώντας το κουμπί “Δημιούργησε νέο lifebit”.
- Το σύστημα δέχεται την αίτηση και αποθηκεύει το καινούργιο lifebit.
- Το σύστημα, ανάλογα με τη σελίδα της εφαρμογής στην οποία βρισκόταν ο χρήστης κατά την υποβολή:
 - o Εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και ανανεώνει το συνολικό αριθμό lifebits και την ημερομηνία εισαγωγής του τελευταίου lifebit (αρχική σελίδα).
 - o Εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και ανανεώνει τη σελίδα (σελίδες που περιέχουν λίστες από lifebits).

Εναλλακτική ροή

- Ο χρήστης δε συμπλήρωσε τουλάχιστον ένα από τα απαιτούμενα πεδία ή η συμπλήρωση των πεδίων δεν ήταν σύμφωνη με τις απαιτήσεις του συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

4.2.2.4 Περίπτωση χρήσης: Επεξεργασία lifebit**Βασική ροή**

- Ο χρήστης βρίσκεται σε μια από τις σελίδες που περιέχουν λίστες από lifebits.
- Πατάει το link “Επεξεργασία” του lifebit που θέλει να επεξεργαστεί.
- Το σύστημα εμφανίζει τη σελίδα επεξεργασίας lifebit. Τα πεδία της φόρμας είναι προσυμπληρωμένα με τα στοιχεία του lifebit προς επεξεργασία.
- Ο χρήστης αλλάζει τα πεδία που θέλει και υποβάλλει τη φόρμα πατώντας το κουμπί “Αποθήκευση αλλαγών”.
- Το σύστημα δέχεται την αίτηση και αποθηκεύει τις αλλαγές.
- Εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και μεταφέρει το χρήστη στη σελίδα “Η ιστορία μου”.

Εναλλακτική ροή

- Ο χρήστης δε συμπλήρωσε τουλάχιστον ένα από τα απαιτούμενα πεδία ή η συμπλήρωση των πεδίων δεν ήταν σύμφωνη με τις απαιτήσεις του

συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

4.2.2.5 Περίπτωση χρήσης: Διαγραφή lifebit

Βασική ροή

- Ο χρήστης βρίσκεται σε μια από τις σελίδες που περιέχουν λίστες από lifebits.
- Πατάει το link “Επεξεργασία” του lifebit που θέλει να επεξεργαστεί.
- Το σύστημα εμφανίζει τη σελίδα επεξεργασίας lifebit.
- Ο χρήστης πατάει το κουμπί “Διαγραφή”.
- Εμφανίζεται ένα αναδυόμενο μήνυμα επιβεβαίωσης. Ο χρήστης πατάει “Ναι”.
- Το σύστημα διαγράφει το lifebit και μεταφέρει το χρήστη στη σελίδα “Πρόσφατα lifebits”.

4.2.2.6 Περίπτωση χρήσης: Προβολή των lifebits ταξινομημένων με βάση την ημερομηνία συμβάντος

Βασική ροή

- Ο χρήστης πατάει από το κεντρικό μενού την επιλογή “Η ιστορία μου”.
- Το σύστημα ανακτά τα κατάλληλα δεδομένα από τη βάση δεδομένων.
- Εμφανίζει στο χρήστη τη σελίδα με τη λίστα των lifebits.

Εναλλακτική ροή

- Δεν υπάρχουν κατάλληλα δεδομένα στη βάση δεδομένων.
- Το σύστημα εμφανίζει στο χρήστη τη σελίδα με ένα σχετικό μήνυμα ενημέρωσης για το χρήστη.

4.2.2.7 Περίπτωση χρήσης: Προβολή των lifebits ταξινομημένων με βάση την ημερομηνία εισαγωγής

Βασική ροή

- Ο χρήστης πατάει από το κεντρικό μενού την επιλογή “Πρόσφατα lifebits”.
- Το σύστημα ανακτά τα κατάλληλα δεδομένα από τη βάση δεδομένων.
- Εμφανίζει στο χρήστη τη σελίδα με τη λίστα των lifebits.

Εναλλακτική ροή

Διαδικτυακή εφαρμογή καταγραφής και παρουσίασης συμβάντων με χρήση μη σχεσιακής βάσης δεδομένων

- Δεν υπάρχουν κατάλληλα δεδομένα στη βάση δεδομένων.
- Το σύστημα εμφανίζει στο χρήστη τη σελίδα με ένα σχετικό μήνυμα ενημέρωσης για το χρήστη.

4.2.2.8 Περίπτωση χρήσης: Προβολή των lifebits ενός συγκεκριμένου επεισοδίου

Βασική ροή

- Ο χρήστης βρίσκεται σε μια από τις σελίδες που περιέχουν λίστες από lifebits.
- Πατάει το link με το όνομα του επεισοδίου που βρίσκεται στην περιγραφή ενός από τα lifebits του επεισοδίου.
- Το σύστημα ανακτά όλα τα lifebits του συγκεκριμένου επεισοδίου από τη βάση δεδομένων.
- Εμφανίζει στο χρήστη τη σελίδα με τη λίστα των lifebits.

4.2.2.9 Περίπτωση χρήσης: Χρονική αναζήτηση

Βασική ροή

- Ο χρήστης πατάει από το κεντρικό μενού την επιλογή “Χρονική αναζήτηση”.
- Επιλέγει ένα συγκεκριμένο εύρος ημερομηνιών και πατάει το κουμπί “Αναζήτηση”.
- Εμφανίζεται ένας χάρτης που περιέχει τους markers όλων των lifebits η ημερομηνία συμβάντος των οποίων είναι μέσα στο επιθυμητό εύρος ημερομηνιών.

4.2.2.10 Περίπτωση χρήσης: Επεξεργασία στοιχείων προφίλ

Βασική ροή

- Ο χρήστης πατάει, από την αναπτυσσόμενη λίστα στο δεξί μέρος της κορυφής της σελίδας, την επιλογή “Προφίλ”.
- Το σύστημα εμφανίζει τη σελίδα επεξεργασίας προσωπικών στοιχείων και αλλαγής κωδικού. Τα πεδία της φόρμας επεξεργασίας προσωπικών στοιχείων είναι προσυμπληρωμένα με τα ήδη καταχωρημένα στοιχεία.
- Ο χρήστης αλλάζει τα πεδία που θέλει και υποβάλλει τη φόρμα πατώντας το κουμπί “Αποθήκευση αλλαγών”.

- Το σύστημα δέχεται την αίτηση και αποθηκεύει τις αλλαγές.
- Εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και μεταφέρει το χρήστη στη σελίδα “Προσωπικά στοιχεία”.

Εναλλακτική ροή

- Η συμπλήρωση των πεδίων από το χρήστη δεν ήταν σύμφωνη με τις απαιτήσεις του συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

4.2.2.11 Περίπτωση χρήσης: Αλλαγή κωδικού

Βασική ροή

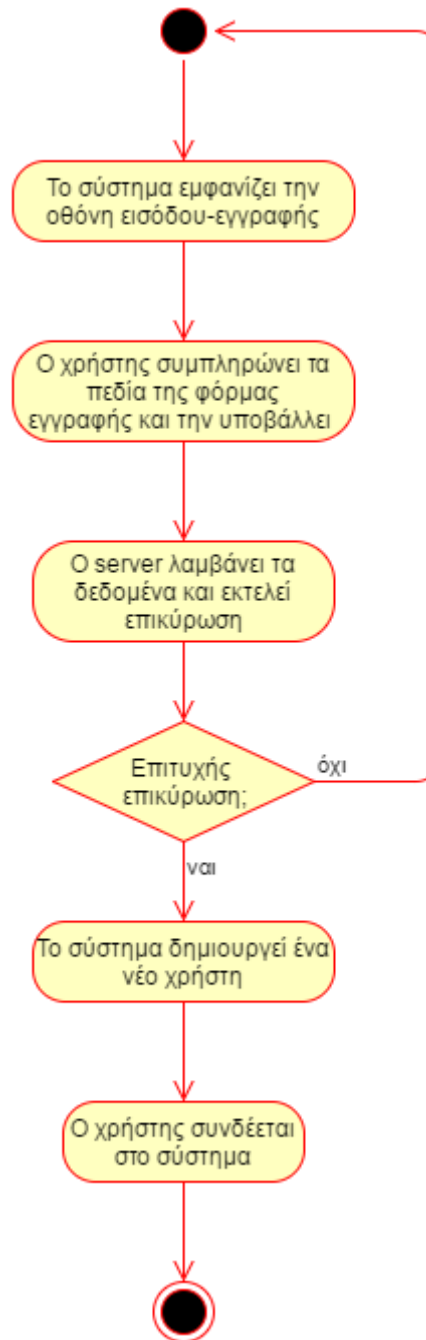
- Ο χρήστης πατάει, από την αναπτυσσόμενη λίστα στο δεξί μέρος της κορυφής της σελίδας, την επιλογή “Προφίλ”.
- Το σύστημα εμφανίζει τη σελίδα επεξεργασίας προσωπικών στοιχείων και αλλαγής κωδικού.
- Ο χρήστης πληκτρολογεί τα απαιτούμενα πεδία και πατάει το κουμπί “Αποθήκευση νέου κωδικού”.
- Το σύστημα δέχεται την αίτηση και αποθηκεύει τις αλλαγές.
- Εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και μεταφέρει το χρήστη στην αρχική σελίδα.

Εναλλακτική ροή

- Ο χρήστης δε συμπλήρωσε τουλάχιστον ένα από τα απαιτούμενα πεδία ή η τιμή του πεδίου “Επιβεβαίωση νέου κωδικού” δεν ήταν ίδια με την τιμή του πεδίου “Νέος κωδικός” ή η συμπλήρωση των πεδίων δεν ήταν σύμφωνη με τις απαιτήσεις του συστήματος. Ο χρήστης ειδοποιείται με σχετικό μήνυμα και απαιτείται η σωστή συμπλήρωση των πεδίων.

4.2.3 Διαγράμματα Δραστηριοτήτων

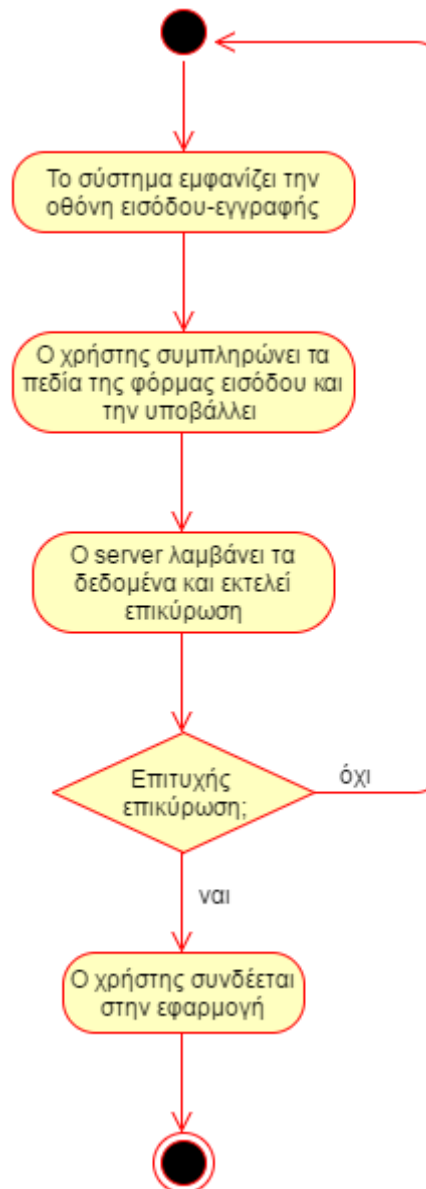
Σε αυτό το υποκεφάλαιο παρουσιάζονται τα διαγράμματα δραστηριοτήτων για τις βασικότερες περιπτώσεις χρήσης. Σκοπός τους είναι η παρουσίαση της γενικής λειτουργίας μιας οντότητας με βάση απλούστερες λειτουργίες που ονομάζονται δραστηριότητες.

4.2.3.1 Διάγραμμα δραστηριοτήτων: Εγγραφή στο σύστημα

Στο παραπάνω διάγραμμα παρουσιάζεται η ροή των εργασιών για την εγγραφή ενός χρήστη στο σύστημα. Αρχικά, εμφανίζεται η σελίδα εισόδου-εγγραφής, ο χρήστης πληκτρολογεί τα απαραίτητα πεδία της φόρμας και την υποβάλλει. Στη συνέχεια, το σύστημα ελέγχει αν ο χρήστης έχει συμπληρώσει όλα τα απαραίτητα

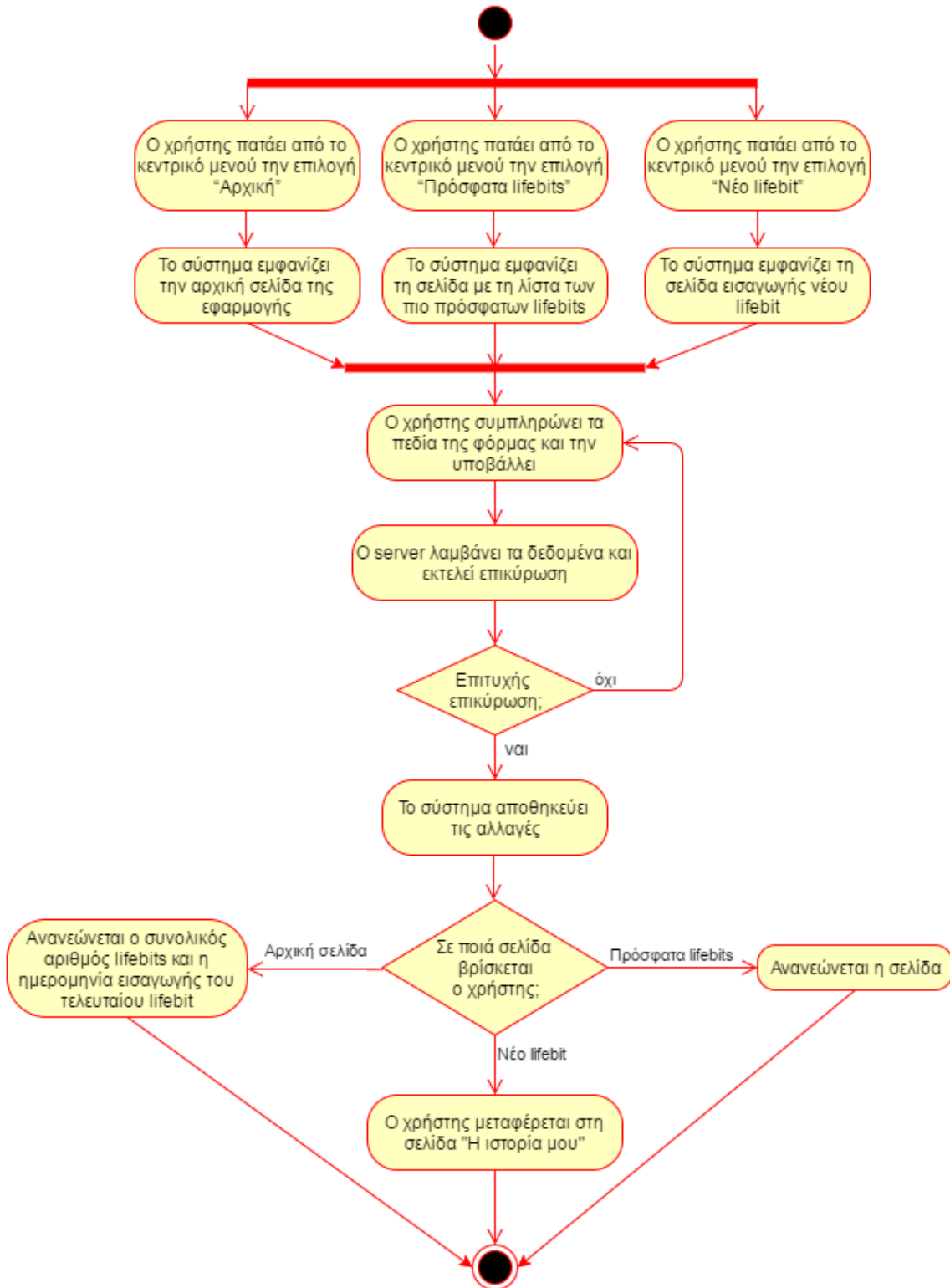
πεδία, αν η διεύθυνση ηλεκτρονικού ταχυδρομείου που πληκτρολόγησε χρησιμοποιείται ήδη, αν η τιμή του πεδίου “Επιβεβαίωση κωδικού” είναι ίδια με την τιμή του πεδίου “Κωδικός” και αν ο κωδικός αποτελείται από τουλάχιστον τέσσερις χαρακτήρες. Αν η επικύρωση είναι επιτυχής, πραγματοποιείται η εγγραφή και ο χρήστης συνδέεται αυτόματα στο σύστημα. Σε αντίθετη περίπτωση, εμφανίζεται η φόρμα εγγραφής και ένα κατάλληλο μήνυμα αποτυχίας.

4.2.3.2 Διάγραμμα δραστηριοτήτων: Είσοδος στην εφαρμογή



Στο παραπάνω διάγραμμα παρουσιάζεται η ροή των εργασιών για την είσοδο ενός χρήστη στην εφαρμογή. Αρχικά, εμφανίζεται η σελίδα εισόδου-εγγραφής, ο χρήστης πληκτρολογεί τα απαραίτητα πεδία της φόρμας και την υποβάλλει. Στη συνέχεια, το σύστημα ελέγχει αν ο χρήστης έχει συμπληρώσει όλα τα απαραίτητα πεδία και αν οι τιμές των πεδίων που πληκτρολόγησε είναι σωστές. Αν η επικύρωση είναι επιτυχής, ο χρήστης συνδέεται στην εφαρμογή. Σε αντίθετη περίπτωση, εμφανίζεται η φόρμα εισόδου και ένα κατάλληλο μήνυμα αποτυχίας.

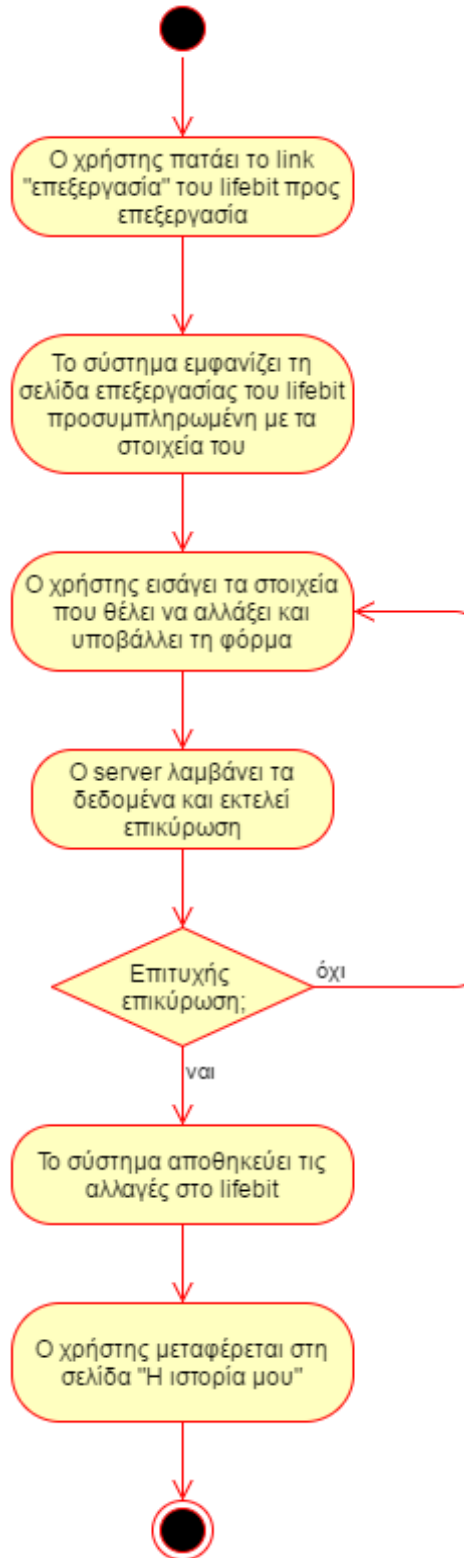
4.2.3.3 Διάγραμμα δραστηριοτήτων: Εισαγωγή lifebit



Διαδικτυακή εφαρμογή καταγραφής και παρουσίασης συμβάντων με χρήση μη σχεσιακής βάσης δεδομένων

Ο χρήστης μπορεί να εισαγάγει ένα νέο lifebit με δύο τρόπους: από τη σελίδα “Νέο lifebit” και από τη φόρμα γρήγορης εισαγωγής που εμφανίζεται στην αρχική σελίδα και στις σελίδες που περιέχουν λίστες από lifebits. Και στις δύο περιπτώσεις ο χρήστης συμπληρώνει τα πεδία της αντίστοιχης φόρμας και την υποβάλλει. Το σύστημα λαμβάνει τα δεδομένα και ελέγχει αν ο χρήστης έχει συμπληρώσει όλα τα απαραίτητα πεδία και αν οι τιμές των πεδίων που πληκτρολόγησε είναι σωστές. Αν η επικύρωση είναι επιτυχημένη τότε, ανάλογα με τη σελίδα που βρίσκεται ο χρήστης, το σύστημα είτε ανανεώνει το συνολικό αριθμό των lifebits του χρήστη και την ημερομηνία εισαγωγής του τελευταίου lifebit (αρχική σελίδα), είτε ανανεώνει τη σελίδα (“Πρόσφατα lifebits”, “ Η ιστορία μου”, σελίδα επεισοδίου), είτε μεταφέρει το χρήστη στη σελίδα “Η ιστορία μου” (“Νέο lifebit”). Αν η επικύρωση είναι ανεπιτυχής, εμφανίζεται ένα σχετικό μήνυμα αποτυχίας.

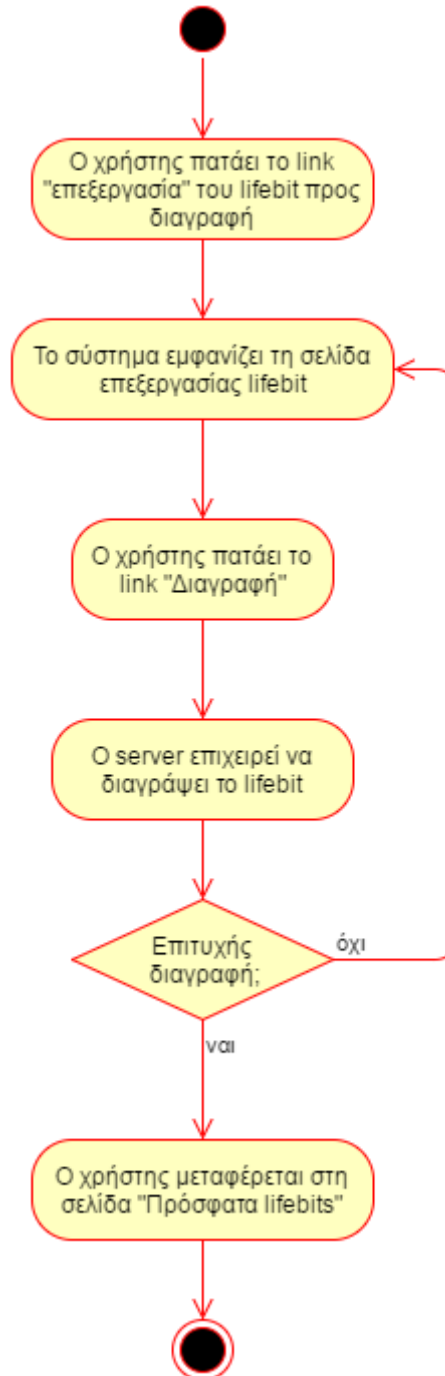
4.2.3.4 Διάγραμμα δραστηριοτήτων: Επεξεργασία lifebit



Διαδικτυακή εφαρμογή καταγραφής και παρουσίασης συμβάντων με χρήση μη σχεσιακής βάσης δεδομένων

Στο παραπάνω διάγραμμα παρουσιάζεται η ροή των εργασιών για την επεξεργασία ενός lifebit. Ο χρήστης, βρισκόμενος σε μια από τις σελίδες της εφαρμογής που περιέχουν λίστες από lifebits, πατάει το link “επεξεργασία” του επιθυμητού lifebit. Ως αποτέλεσμα, εμφανίζεται η σελίδα επεξεργασίας lifebit η οποία περιέχει την αντίστοιχη φόρμα προσυμπληρωμένη με τα στοιχεία του συγκεκριμένου lifebit. Στη συνέχεια ο χρήστης εισάγει τα στοιχεία που θέλει, υποβάλλει τη φόρμα και το σύστημα ελέγχει αν ο χρήστης έχει συμπληρώσει όλα τα απαραίτητα πεδία και αν η συμπλήρωση των πεδίων είναι σύμφωνη με τις απαιτήσεις του συστήματος. Αν η επικύρωση είναι επιτυχής, το σύστημα αποθηκεύει τις αλλαγές, εμφανίζει ένα αναδυόμενο μήνυμα επιτυχίας και μεταφέρει το χρήστη στη σελίδα “Η ιστορία μου”. Σε αντίθετη περίπτωση, εμφανίζεται ένα σχετικό μήνυμα αποτυχίας.

4.2.3.5 Διάγραμμα δραστηριοτήτων: Διαγραφή lifebit



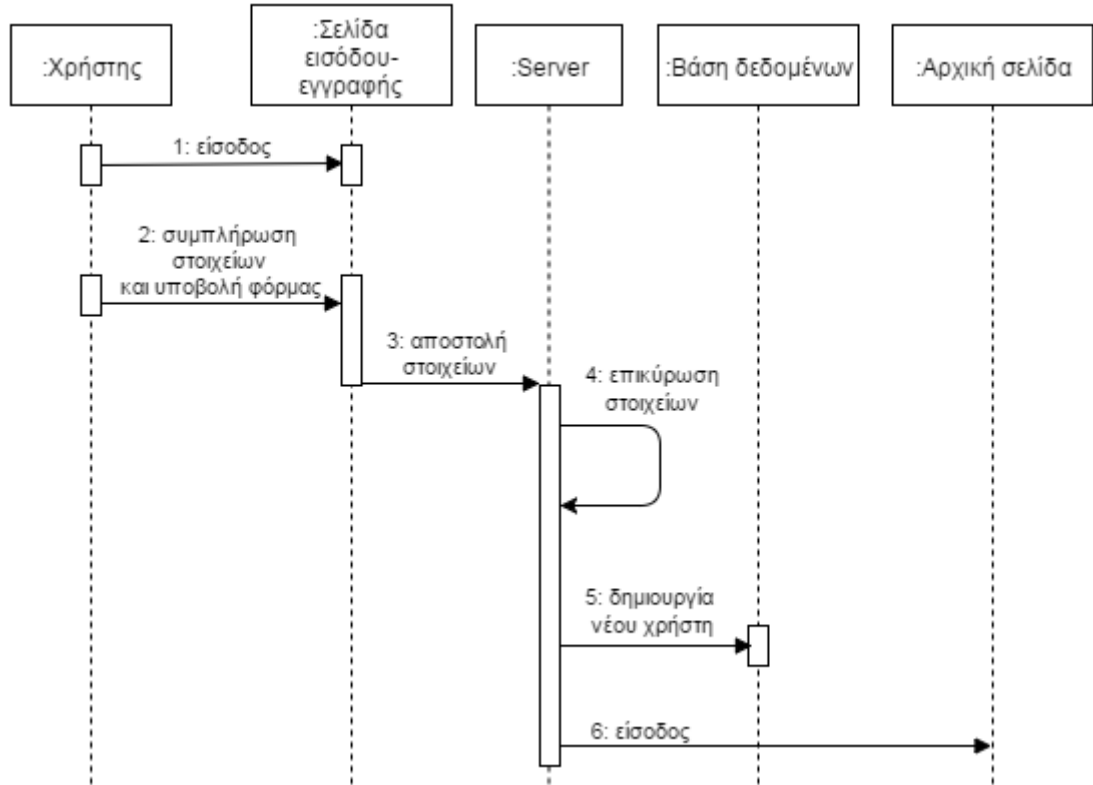
Στο παραπάνω διάγραμμα παρουσιάζεται η ροή των εργασιών για τη διαγραφή ενός lifebit. Ο χρήστης, βρισκόμενος σε μια από τις σελίδες της εφαρμογής που περιέχουν λίστες από lifebits, πατάει το link "επεξεργασία" του lifebit προς

διαγραφή και εμφανίζεται η αντίστοιχη σελίδα. Στη συνέχεια, πατάει το κουμπί “Διαγραφή” και το σύστημα επιχειρεί να διαγράψει το lifebit. Αν η διαγραφή είναι επιτυχής εμφανίζεται ένα αναδυόμενο μήνυμα επιτυχίας και ο χρήστης μεταφέρεται στη σελίδα “Πρόσφατα lifebits”. Σε αντίθετη περίπτωση, εμφανίζεται μήνυμα σφάλματος.

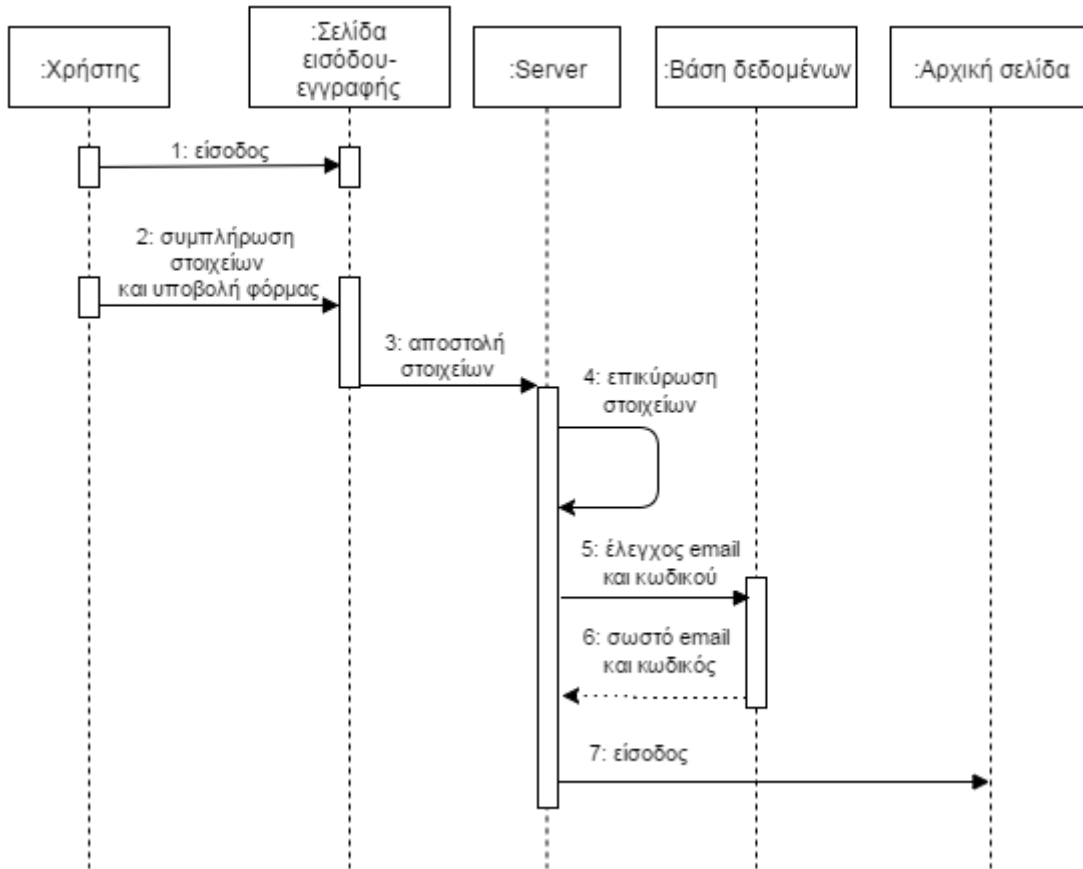
4.2.4 Διαγράμματα Ακολουθίας

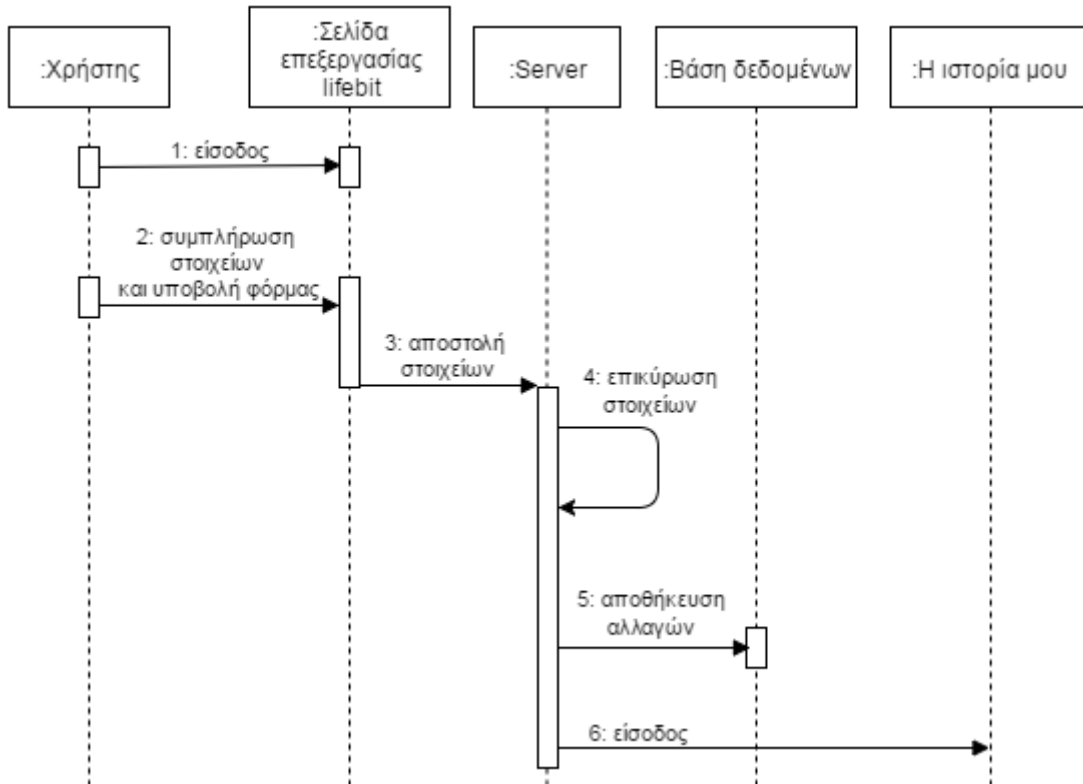
Τα διαγράμματα ακολουθίας είναι διαγράμματα τα οποία περιγράφουν την αλληλεπίδραση μεταξύ αντικειμένων, δηλαδή τον τρόπο που διαφορετικά αντικείμενα συνεργάζονται μεταξύ τους σε μια χρονική ακολουθία με σκοπό την πραγματοποίηση μιας συγκεκριμένης ενέργειας. Η αλληλεπίδραση μεταξύ των αντικειμένων παρουσιάζεται σε δύο διαστάσεις: η κάθετη διάσταση αναπαριστά την κλίμακα του χρόνου ενώ η οριζόντια αναπαριστά τα διάφορα αντικείμενα.

4.2.4.1 Διάγραμμα ακολουθίας: Εγγραφή χρήστη

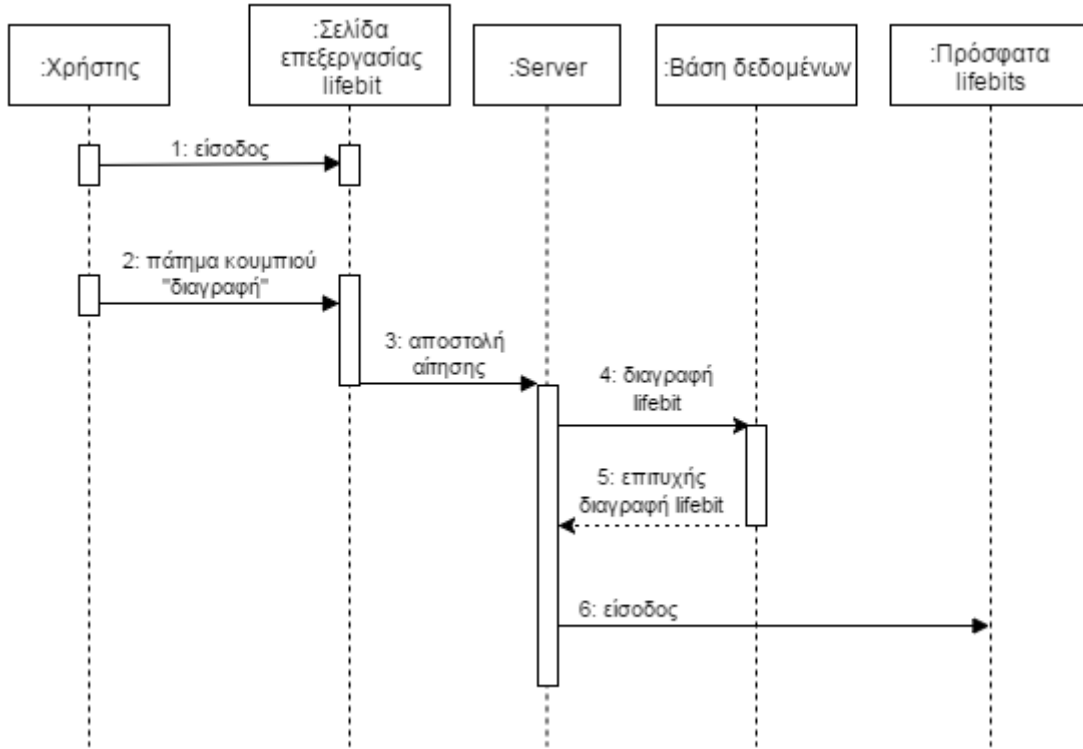


4.2.4.2 Διάγραμμα ακολουθίας: Είσοδος χρήστη



4.2.4.3 Διάγραμμα ακολουθίας: Επεξεργασία lifebit

4.2.4.4 Διάγραμμα ακολουθίας: Διαγραφή lifebit



4.3 Διάταξη δεδομένων

Όπως έχει ήδη αναφερθεί, η MongoDB είναι η βάση δεδομένων που χρησιμοποιήθηκε στα πλαίσια της εφαρμογής της παρούσας εργασίας για την αποθήκευση δεδομένων. Το lifebit αποτελεί τη βασική οντότητα της εφαρμογής και ενσωματώνει όλες τις πληροφορίες που εισάγει ο χρήστης σχετικά με αυτό. Η δεύτερη οντότητα περιέχει όλες τις πληροφορίες σχετικά με τον κάθε χρήστη της εφαρμογής.

Lifebit		
Collection με πληροφορίες για το lifebit		
Field	Field description	Value
_id	Μοναδικό id	Mongo id
description	Περιγραφή συμβάντος	string
location	Τοποθεσία συμβάντος	string
date_inserted	Ημερομηνία εισαγωγής	string
date_fact	Ημερομηνία συμβάντος	string
userId	Id χρήστη	Mongo id
latitude	Γεωγραφικό πλάτος τοποθεσίας	string
longitude	Γεωγραφικό μήκος τοποθεσίας	string

Χρήστης		
Collection με πληροφορίες για το χρήστη		
Field	Field description	Value
_id	Μοναδικό id	Mongo id
email	Ηλεκτρονικό ταχυδρομείο	string
password	Κωδικός	string
firstName	Όνομα	string
lastName	Επίθετο	string
dateBirth	Ημερομηνία γέννησης	string
placebirth	Τοποθεσία γέννησης	string
sex	Φύλο	string
town	Πόλη	string
latitude	Γεωγραφικό πλάτος τοποθεσίας γέννησης	string
longitude	Γεωγραφικό μήκος τοποθεσίας γέννησης	string
dateRegistration	Ημερομηνία εγγραφής	string

Σύνοψη

Στο κεφάλαιο αυτό αρχικά παρουσιάστηκαν τα τρία επίπεδα που αποτελούν την αρχιτεκτονική του συστήματος ενώ, στη συνέχεια, έγινε μια συνοπτική παρουσίαση της γλώσσας μοντελοποίησης UML και του μοντέλου RUP. Επίσης, έγινε παρουσίαση και περιγραφή των λειτουργικών απαιτήσεων της εφαρμογής με τη βοήθεια των περιπτώσεων χρήσης και των διαγραμμάτων ακολουθίας και δραστηριοτήτων. Τέλος, παρουσιάστηκε η διάταξη δεδομένων που σχεδιάστηκε και υλοποιήθηκε με σκοπό την αποθήκευση πληροφοριών.

Στο επόμενο κεφάλαιο παρουσιάζεται ένα μέρος των δυνατοτήτων που προσφέρει η εφαρμογή μέσω της καταγραφής μιας σειράς ενεργειών ενός χρήστη της.

5. Παράδειγμα χρήσης της εφαρμογής

Στο κεφάλαιο αυτό παρουσιάζεται ένα παράδειγμα χρήσης της εφαρμογής, δηλαδή μια ακολουθία ενεργειών που μπορεί να κάνει ένας χρήστης κατά την περιήγησή του σε αυτή.

Αρχικά, ο χρήστης, βρισκόμενος στη σελίδα εγγραφής-σύνδεσης της εφαρμογής, πληκτρολογεί το email του και έναν κωδικό στα κατάλληλα πεδία ώστε να δημιουργήσει ένα λογαριασμό.

Lifebits

Δημιούργησε τα δικά σου lifebits
Μοιράσου με τους φίλους σου τις πιο σημαντικές στιγμές σου

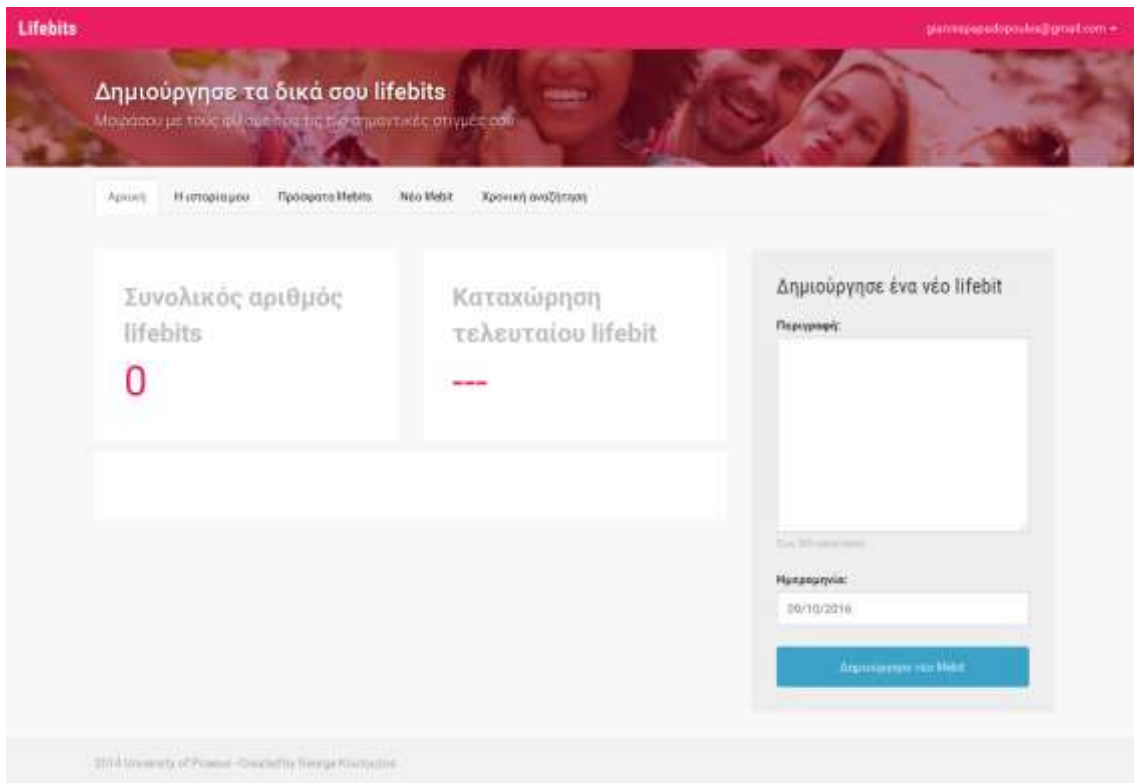
Δημιούργησε το λογαριασμό σου και άρχισε αμέσως τη δημιουργία των δικών σου **lifebits**

Εγγραφή

Έχεις λογαριασμό;

© 2014 University of Pirene - Created by George Koustouzos

Τα στοιχεία που πληκτρολόγησε είναι έγκυρα οπότε δημιουργείται ο λογαριασμός του και ο ίδιος μεταφέρεται στην αρχική σελίδα της εφαρμογής.



Στη συνέχεια, αποφασίζει να καταγράψει, με τη βοήθεια της εφαρμογής, ένα ταξίδι που πραγματοποίησε στο Λονδίνο. Έτσι, πατάει την επιλογή “Νέο lifebit” του κεντρικού μενού και μεταφέρεται στην αντίστοιχη σελίδα όπου προσθέτει τα αντίστοιχα lifebits. Σε κάθε lifebit που καταχωρεί προσθέτει την ετικέτα “#London2015” στην περιγραφή του ομαδοποιώντας με αυτόν τον τρόπο τα συγκεκριμένα lifebits στο επεισόδιο “London2015”.

Αρχική
Η ιστορία μου
Πρόσφατα lifebits
Νέο lifebit
Χρονική αναζήτηση

Δημιούργησε ένα νέο lifebit

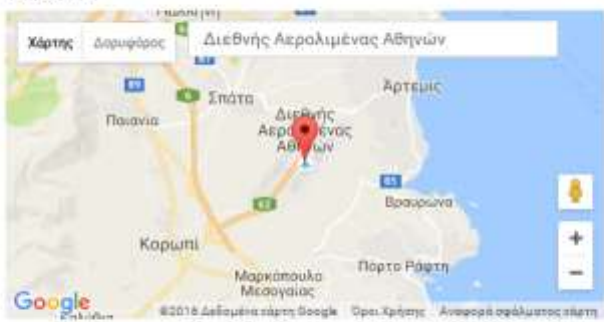
Περιγραφή*

Επιτέλους, μετά από τόσες προετοιμασίες, το ταξίδι ξεκινάει!
#London2015

Έως 300 χαρακτήρες

Απόδειξη κάρτη

Τοποθεσία:

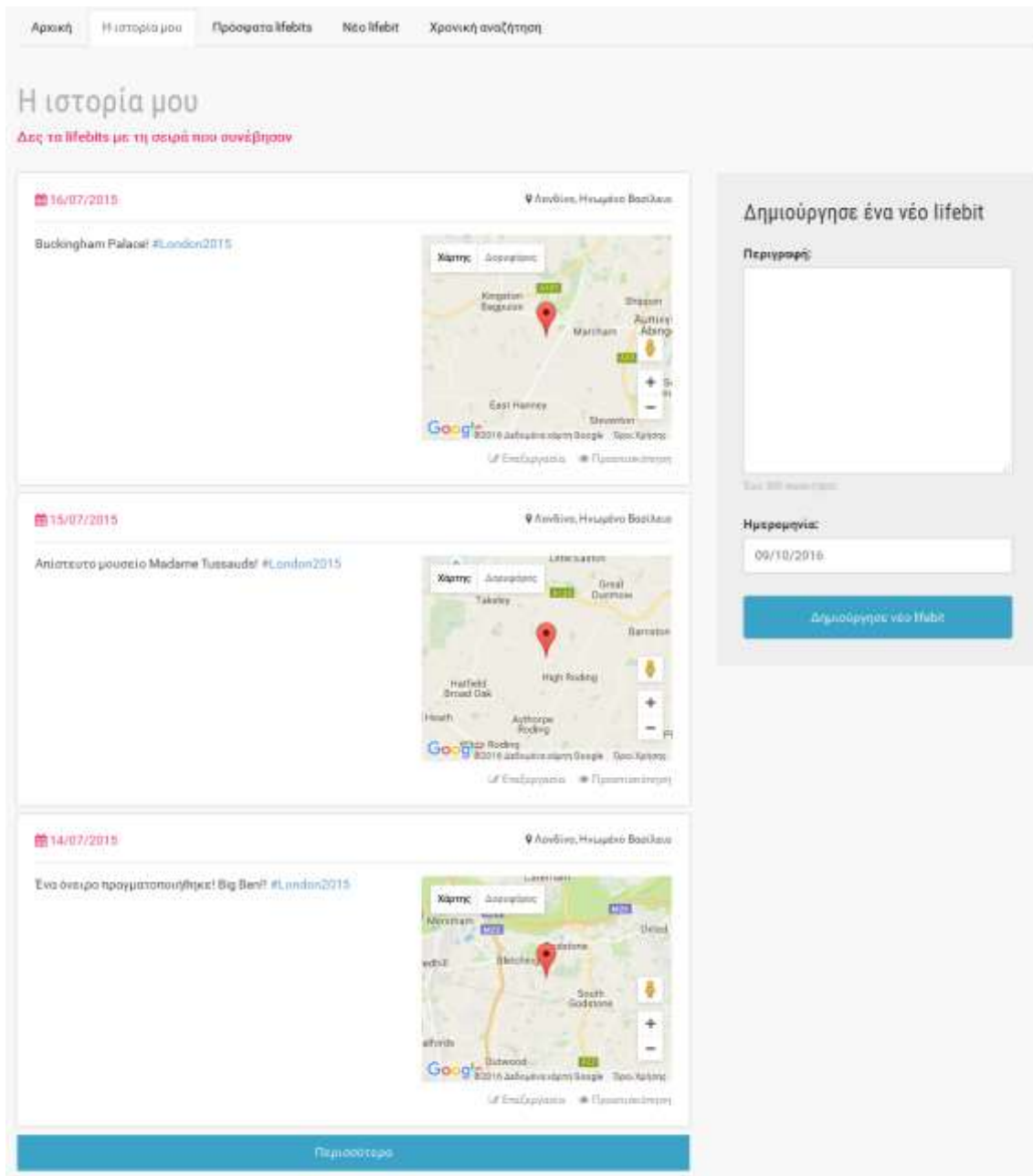


Ημερομηνία*

13/07/2015

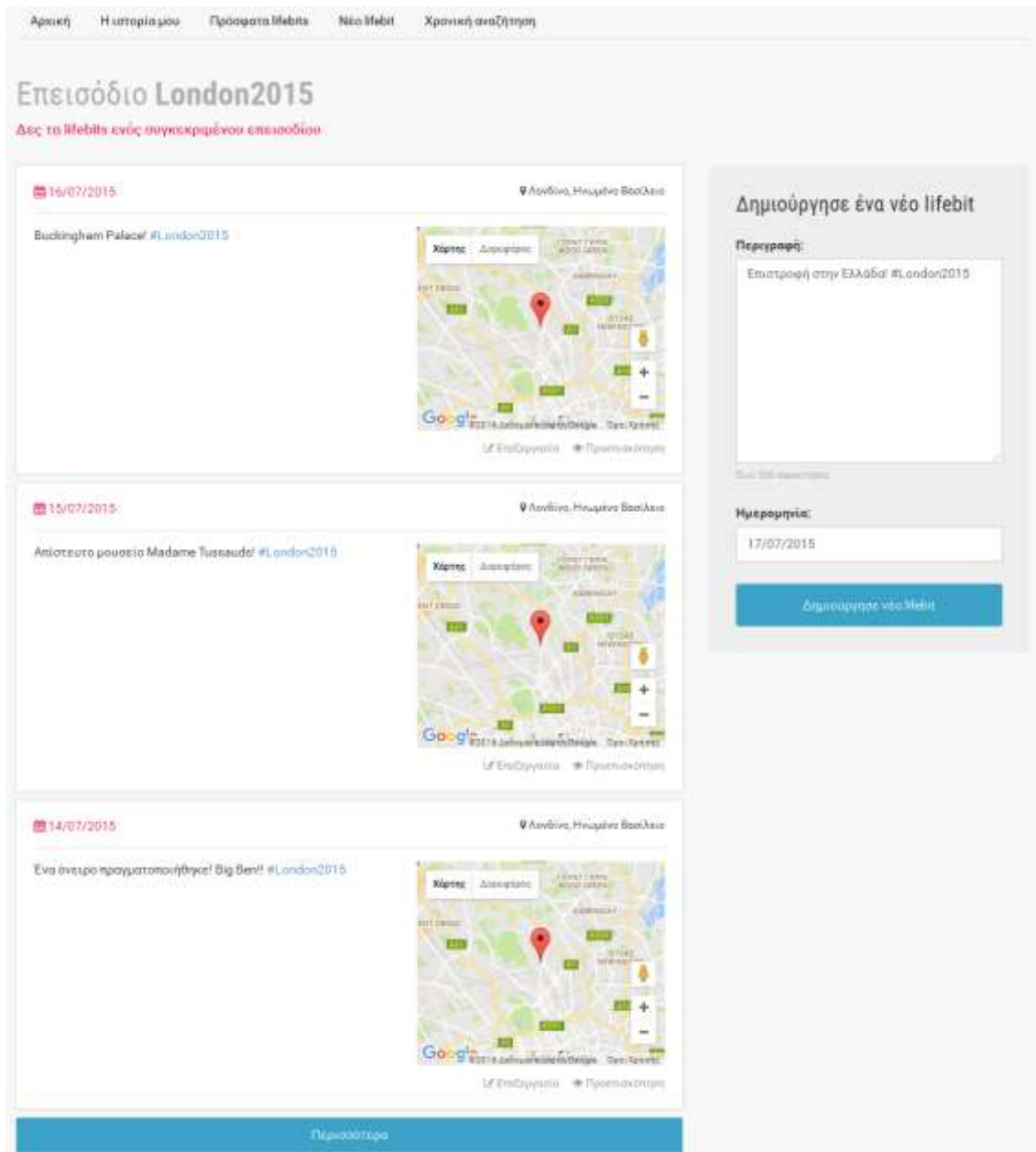
Δημιούργησε νέο lifebit

Πατώντας την επιλογή “Η ιστορία μου” του κεντρικού μενού μεταφέρεται σε μια σελίδα όπου μπορεί να δει όλα τα lifebits που έχει καταχωρήσει έως τώρα.



Διαδικτυακή εφαρμογή καταγραφής και παρουσίασης συμβάντων με χρήση μη σχεσιακής βάσης δεδομένων

Στη συνέχεια πατάει το σύνδεσμο “#London2015” που βρίσκεται στις περιγραφές των σχετικών με το ταξίδι lifebits και μεταφέρεται σε μια νέα σελίδα όπου μπορεί να δει όλα τα lifebits του συγκεκριμένου επεισοδίου. Χρησιμοποιώντας τη φόρμα γρήγορης αναζήτησης που βρίσκεται στο δεξί μέρος της σελίδας καταχωρεί άλλο ένα lifebit.



Πατώντας το link “Επεξεργασία” ενός από τα lifebits μεταφέρεται σε μια σελίδα όπου μπορεί να αλλάξει τα δεδομένα του συγκεκριμένου lifebit.

Αρχική Η ιστορία μου Πρόσφατα lifebits Νέο lifebit Χρονική αναζήτηση

Επεξεργασία


Περιγραφή:

Απίστευτο μουσείο Madame Tussauds! #London2015

Έως 500 χαρακτήρες

[Απόκρυψη χάρτη](#)

Τοποθεσία:

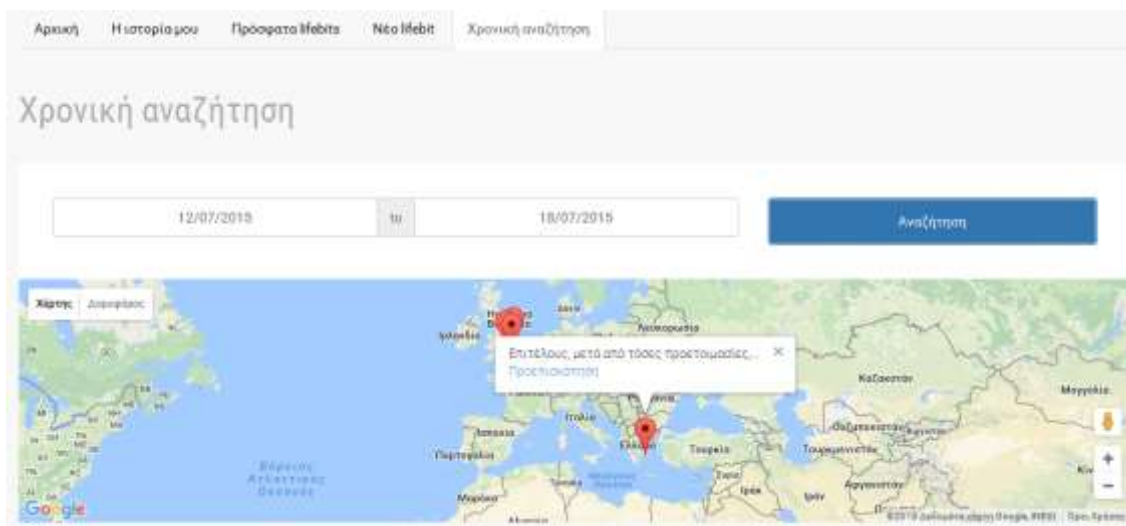


Ημερομηνία:

15/07/2015

Αποθήκευση αλλαγών Διαγραφή

Στη συνέχεια, πατώντας την επιλογή “Χρονική αναζήτηση” του κεντρικού μενού, μεταφέρεται σε μια σελίδα όπου, επιλέγοντας τις ημερομηνίες του ταξιδιού, βλέπει σε ένα χάρτη τις τοποθεσίες των σχετικών με το ταξίδι lifebits. Πατώντας πάνω σε έναν από τους markers που εμφανίζονται βλέπει ένα κομμάτι της περιγραφής του αντίστοιχου lifebit και ένα link που, πατώντας το, θα μεταφερθεί στη σελίδα προεπισκόπησης του.



Σύνοψη

Στο κεφάλαιο αυτό παρουσιάζονται, μέσα από την καταγραφή μιας σειράς ενεργειών ενός χρήστη, κάποιες από τις δυνατότητες που προσφέρει η εφαρμογή σε αυτόν.

Στο επόμενο κεφάλαιο παρουσιάζονται τα συμπεράσματα που προέκυψαν από την εκπόνηση της παρούσας διπλωματικής εργασίας, αναφέρονται κάποιες λειτουργίες που δεν υλοποιήθηκαν και γίνονται ορισμένες προτάσεις για βελτίωση και επέκταση της εφαρμογής.

6. Επίλογος

Στο κεφάλαιο αυτό παρουσιάζονται τα συμπεράσματα που προέκυψαν από την υλοποίηση της παρούσας διπλωματικής εργασίας, γίνεται αναφορά των λειτουργιών της εφαρμογής που δεν υλοποιήθηκαν και περιγράφονται μελλοντικές επεκτάσεις και βελτιώσεις που θα μπορούσαν να υλοποιηθούν σε μια επόμενη έκδοση της εφαρμογής.

Συμπεράσματα

Ο σκοπός της παρούσας εργασίας ήταν ο σχεδιασμός και η υλοποίηση μιας διαδικτυακής εφαρμογής η οποία θα προσέφερε στο χρήστη τη δυνατότητα να καταχωρεί τις εμπειρίες του και σημαντικά για αυτόν γεγονότα αλλά και να ανακτά και να βλέπει τις συγκεκριμένες καταχωρήσεις. Η ενασχόληση με την υλοποίηση μιας εφαρμογής “από το μηδέν” ήταν μια από τις πιο σημαντικές εμπειρίες που αποκομίσθηκαν. Οι ανάγκες της εργασίας οδήγησαν, υπό την κατάλληλη καθοδήγηση, στην έρευνα και στη γνωριμία με σύγχρονες τεχνολογίες του διαδικτύου αλλά και στο συνδυασμό τους για την επίτευξη του καλύτερου δυνατού αποτελέσματος. Η ενασχόληση με τα διάφορα μέρη που αποτελούν τη συγκεκριμένη εφαρμογή είχε ως αποτέλεσμα την απόκτηση πολλών νέων γνώσεων αλλά και τη δημιουργία της επιθυμίας για απόκτηση ακόμα περισσότερων.

Η σχεδιαστική αρχή του διαχωρισμού προβλημάτων (Separation of Concerns), σύμφωνα με την οποία μια πληροφοριακή εφαρμογή πρέπει να χωρίζεται σε διακριτά επιμέρους τμήματα και κάθε ένα από αυτά να αντιμετωπίζει ένα ξεχωριστό πρόβλημα, οδήγησε στην επιλογή ενός πλαισίου εργασίας που να βασίζεται και να ακολουθεί την αρχιτεκτονική Μοντέλο-Παρουσίαση-Ελεγκτής (Model-View-Controller, MVC), η οποία αποτελεί τη δημοφιλέστερη υλοποίηση αυτής της αρχής. Έτσι, επιλέχθηκε το PHP πλαίσιο εργασίας Codeigniter ως ένα από τα δημοφιλέστερα και αποτελεσματικότερα PHP πλαίσια εργασίας το οποίο προσφέρει πολλές ευκολίες και δυνατότητες στους χρήστες του.

Η μη σχεσιακή βάση δεδομένων MongoDB αποτελεί ιδανική επιλογή για εφαρμογές, όπως η εφαρμογή που αναπτύχθηκε για τις ανάγκες της συγκεκριμένης διπλωματικής εργασίας, οι οποίες διαχειρίζονται μεγάλο και συνεχώς αυξανόμενο όγκο δεδομένων. Σε αυτές τις εφαρμογές είναι απαραίτητη η γρήγορη είσοδος και ανάκτηση των δεδομένων για τη βελτιστοποίηση της απόδοσή τους. Έτσι, μετά από έρευνα και σύγκριση μεταξύ των διάφορων διαθέσιμων NoSQL βάσεων, αποφασίστηκε να χρησιμοποιηθεί η MongoDB στην υλοποίηση της παρούσας εφαρμογής.

Το πλαίσιο εργασίας Bootstrap αποτελεί μια από τις πιο δημοφιλείς προτάσεις για εύκολη, γρήγορη και φιλική προς το χρήστη ανάπτυξη ιστοσελίδων,

προσφέροντας ένα σταθερό, ευέλικτο και αποκρίσιμο σύστημα διάταξης (grid system) ώστε να είναι πολύ εύκολη η υποστήριξη συσκευών όπως κινητά ή ταμπλέτες.

Η JQuery είναι μια από τις πιο διαδεδομένες Javascript βιβλιοθήκες αφού διευκολύνει σε μεγάλο βαθμό την επικοινωνία μεταξύ Javascript και HTML και αποτελεί την πλέον κατάλληλη επιλογή για ανάπτυξη διαδικτυακών εφαρμογών οι οποίες απαιτούν μεγάλη διαδραστικότητα με τους χρήστες τους. Η JQuery, όπως και οι τεχνολογίες HTML, CSS και Bootstrap, έπαιξε καθοριστικό ρόλο στον τρόπο παρουσίασης και διαχείρισης των δεδομένων της εφαρμογής αλλά και στην εμφάνιση των διαφόρων σελίδων της.

Το AJAX είναι μια τεχνολογία ιδιαίτερα δημοφιλής στις διαδικτυακές εφαρμογές που έχουν έντονη αλληλεπίδραση με τους χρήστες τους αφού, λόγω της δυνατότητας που δίνει για ανταλλαγή δεδομένων χωρίς να είναι απαραίτητη η φόρτωση της σελίδας, κάνει τις εφαρμογές που τη χρησιμοποιούν ιδιαίτερα γρήγορες και δυναμικές.

Το Google Maps είναι μια υπηρεσία χαρτών που προσφέρεται από την Google ενώ η ενσωμάτωσή της στις διαδικτυακές εφαρμογές και η διαχείρισή της από τους προγραμματιστές γίνεται μέσω του Javascript API Google Maps API. Η συγκεκριμένη υπηρεσία βοήθησε στην επιλογή και στην παρουσίαση των διαφόρων τοποθεσιών πάνω σε χάρτη σε πολλά σημεία της εφαρμογής.

Περιορισμοί - Μελλοντικές βελτιώσεις και επεκτάσεις

Υπάρχουν κάποιες λειτουργίες οι οποίες, παρόλο που αρχικά υπήρχε η σκέψη να συμπεριληφθούν στο σύνολο των λειτουργιών της εφαρμογής που παρουσιάζεται σε αυτή την εργασία, δεν υλοποιήθηκαν και, ως αποτέλεσμα, αποτελούν πιθανές μελλοντικές βελτιώσεις και επεκτάσεις της. Ο κυριότερος λόγος της μη υλοποίησής τους είναι ο μεγάλος όγκος χρόνου και πολυπλοκότητας που θα απαιτούσαν και θα προσέθεταν στην εργασία σε επίπεδο έρευνας, σχεδιασμού και υλοποίησης.

Κατά το σχεδιασμό της εφαρμογής υπήρχε η ιδέα ο κάθε χρήστης να έχει διαθέσιμο ένα χρονοδιάγραμμα (timeline) όπου θα μπορούσε να βλέπει με χρονολογική σειρά τις καταχωρήσεις του και τις λεπτομέρειές τους σε ένα ωραίο γραφικό περιβάλλον. Μια ακόμα βελτίωση θα ήταν η αλλαγή και η βελτιστοποίηση του τρόπου σελιδοποίησης (pagination) στις σελίδες που παρουσιάζονται οι καταχωρήσεις του χρήστη. Επίσης, μια βελτίωση που θα μπορούσε να έχει μια επόμενη έκδοση της εφαρμογής θα ήταν η βελτιστοποίηση της λειτουργίας κατηγοριοποίησης-ομαδοποίησης των καταχωρήσεων μέσω της εμφάνισης, ως προτεινόμενων, των ήδη υπαρχόντων επεισοδίων στους χρήστες κατά τη δημιουργία ενός lifebit. Τέλος, η δημιουργία ενός δομημένου επιπέδου αναζήτησης (structured search layer) της εφαρμογής θα έδινε στο χρήστη τη δυνατότητα αναζήτησης καταχωρήσεων σύμφωνα με την περιγραφή τους αλλά και την ημερομηνία.

Μια μελλοντική επέκταση που θα μπορούσε να εφαρμοστεί, μέσω της υλοποίησης των ανάλογων λειτουργιών, θα ήταν ουσιαστικά η εξέλιξη της εφαρμογής σε κοινωνικό δίκτυο. Οι χρήστες θα μπορούσαν να συνδέονται μέσω της σχέσης follower-following (ακόλουθος-αυτός που ακολουθείται), να γίνονται διαδικτυακοί φίλοι, να αλληλεπιδρούν μεταξύ τους δηλώνοντας την αρέσκειά τους και σχολιάζοντας τις καταχωρήσεις άλλων χρηστών ενώ θα μπορούσαν να αναφέρουν άλλους χρήστες στις καταχωρήσεις τους. Θα μπορούσε επίσης να υλοποιηθεί ένα σύστημα ανταλλαγής μηνυμάτων μεταξύ των χρηστών το οποίο θα διευκόλυε τη μεταξύ τους επικοινωνία και θα παρείχε στον κάθε χρήστη ένα πλήρες ιστορικό των συνομιλιών του. Τέλος, ο χρήστης θα μπορούσε να εισέρχεται και να εγγράφεται στην εφαρμογή με τα στοιχεία λογαριασμού άλλων πλατφορμών (facebook, google+, twitter κ.α.).

Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκαν τα συμπεράσματα που αποκομίσθηκαν από την εκπόνηση της παρούσας διπλωματικής εργασίας ενώ, επίσης, αναφέρθηκαν διάφορες λειτουργίες που δεν υλοποιήθηκαν αλλά και πιθανές μελλοντικές προσθήκες και βελτιώσεις της εφαρμογής. Στη συνέχεια, παρουσιάζονται οι βιβλιογραφικές πηγές που μελετήθηκαν.

7. Βιβλιογραφία

Leiner B. et al., 2009. A Brief History of the Internet. Available at:
<http://www.cs.ucsb.edu/~almeroth/classes/F10.176A/papers/internet-history-09.pdf>

EllisLab. A Brief History Of CodeIgniter. Retrieved from:
<https://ellislab.com/codeigniter>

Vaish G., 2011. Getting started with NoSQL. Available at:
<http://pdf.th7.cn/download/files/1411/Getting%20Started%20with%20NoSQL.pdf>

Seguin K., 2011. The Little MongoDB Book. Available at:
<http://openmymind.net/mongodb.pdf>

Upton D. Codeigniter for Rapid PHP Application Development. Available at:
<http://www.azpromo.cz/seminare/1249657909.pdf>

Francia S., 2012. MongoDB and PHP. Available at:
<http://img105.job1001.com/upload/adminnew/2015-04-07/1428398776-AAI3MVO.pdf>

Chaffer J., Swedberg K., 2013. Learning JQuery. Available at:
<http://www.saigontech.edu.vn/faculty/FITM/XML/Ebook/Learning%20jQuery,%204th%20Edition.pdf>

Keith J.. Bulletproof Ajax. Available at:
<http://ommolketab.ir/aaf-lib/6lf5q1zifzpo8zcgvt5ad09gcu6i3u.pdf>

Spurlock J., 2013. Bootstrap. Available at:
<http://wiki.ifs.hsr.ch/APF/files/bootstrap.pdf>

Google Maps API
<https://developers.google.com/maps/documentation/javascript>

Schmuller J., 2004. Sams Teach Yourself UML in 24 Hours. Available at:
http://www4.cookman.edu/faculty/banisakher/Electronic_Book.pdf

Ahmad Shuja, 2007. RUP Reference and Certification Guide. Available at:
http://ptgmedia.pearsoncmg.com/images/9780131562929/samplechapter/0131562924_01.pdf