



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Ανάπτυξη Mobile εφαρμογής για εφαρμογές M-Health Development of Mobile App for M-Health Applications
Όνοματεπώνυμο Φοιτητή	Ευάγγελος Κιτσούλης
Πατρώνυμο	Βασίλειος
Αριθμός Μητρώου	ΜΠΠΛ15028
Επιβλέπων	Αλέπης Ευθύμιος Επίκουρος Καθηγητής

Ημερομηνία Παράδοσης **Οκτώβριος 2018**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Ευθύμιος Αλέπης
Επίκουρος Καθηγητής

Μαρία Βίρβου
Καθηγήτρια

Κωνσταντίνος Πατσάκης
Επίκουρος Καθηγητής

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία έχει ως κύριο στόχο την σχεδίαση και την ανάπτυξη μιας Web εφαρμογής ιατρικού περιεχομένου, η οποία θα είναι συμβατή για χρήση από οποιονδήποτε browser όπως και κινητές συσκευές.

Σκοπός της εφαρμογής είναι να μπορεί ο χρήστης να καταχωρεί το ιστορικό από τις παθήσεις τις οποίες έχει νοσήσει, καθώς και να βλέπει τις πιθανότητες που έχει για να αποκτήσει μια πάθηση μέσω της κληρονομικότητας. Επίσης μέσα από την εφαρμογή θα δίνεται η δυνατότητα στον χρήστη να δηλώνει μέσω του ΑΜΚΑ τα συγγενικά του πρόσωπα έτσι ώστε να προκύψει η κληρονομικότητα. Η εφαρμογή θα διαθέτει ιατρικά άρθρα όπου ο χρήστης θα μπορεί να διαβάσει, όπως επίσης και σχετικά άρθρα που θα εμφανίζονται στο προφίλ του κάθε χρήστη με βάση τις παθήσεις που έχει δηλώσει ο ίδιος.

Βασικό πλεονέκτημα για το χρήστη, θα είναι όχι μόνο να γνωρίζει από ποιες παθήσεις έχει νοσήσει ο ίδιος αλλά και από ποιες παθήσεις θα μπορούσε να νοσήσει με βάση την κληρονομικότητα που προκύπτει.

Η παρούσα μεταπτυχιακή εργασία έχει υλοποιηθεί σε δύο κομμάτια, την Web έκδοση και την Android. Η Web έκδοση υλοποιήθηκε από τον Ποσνακίδη Ηρακλή (ΜΠΠΛ 15061) και παρουσιάζεται στην μεταπτυχιακή διατριβή του.

ABSTRACT

The current thesis has as main target the design and development of a medical web application, which will be compatible for use by any browser and mobile devices.

The purpose of the application is to let the user record the history of the illnesses he has suffered from, and also to be able to see the chances of getting a condition due to heredity. Moreover, the user will have the choice to declare his relatives' personal registration numbers in order for the heredity data to be resulted. The application will also have medical articles available to the user, as well as other related articles that will pop up in the profile of the user based on the illnesses that he has declared.

A major advantage for the user, will not only be the fact that he will be able each time to know from what illnesses he has already suffered, but also to be able to know the potential diseases that may affect him based on his heredity.

In this dissertation we have created two applications, Web versions and Android version. The Web version is developed by Posnakidis Iraklis (MPPL 15061) and has been presented in his own dissertation.

Περιεχόμενα

Ακρωνύμια	7
Λέξεις κλειδιά	8
ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ.....	9
1.1 Περιγραφή της εργασίας.....	9
1.2 Κίνητρο για την ανάπτυξη της εργασίας.....	9
1.3 Στόχος εργασίας.....	9
1.4 Δομή εργασίας.....	10
ΚΕΦΑΛΑΙΟ 2. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ	11
2.1 Web εφαρμογή	11
2.2 Android εφαρμογή	11
ΚΕΦΑΛΑΙΟ 3. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ	12
3.1 Medical Records.....	12
3.2 My Medical History	16
ΚΕΦΑΛΑΙΟ 4. ΠΑΡΟΥΣΙΑΣΗ ΚΑΙ ΧΡΗΣΗ ΕΦΑΡΜΟΓΗΣ	18
4.1 Εικονίδιο εφαρμογής και Splash Screen	18
4.2 Αρχική οθόνη – Login screen	19
4.3 Register.....	21
4.4 Δήλωση πάθησης και ιστορικό όλων των παθήσεων.....	22
4.5 Δήλωση Συγγένειας.....	24
4.6 Το προφίλ του χρήστη	25
4.7 Αλλαγή κωδικού	27
4.8 Contact form.....	29
4.9 Top news.....	29
4.10 Προτεινόμενα Άρθρα	31
4.11 Σφάλματα από την μεριά του χρήστη.....	32
4.12 Σφάλματα από την μεριά των servers.....	35
ΚΕΦΑΛΑΙΟ 5. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ.....	36
5.1 Οι τεχνολογίες που χρησιμοποιήθηκαν για τοWeb κομμάτι της εφαρμογής	36
5.1.1 Vue.js.....	33

5.1.2 JavaScript Framework.....	34
5.1.3 Η γλώσσα μορφοποίησης HTML.....	35
5.1.4 JavaScript.....	35
5.1.5 CSS.....	36
5.1.6 Vuetify Framework.....	36
5.1.7 Webstorm.....	37
5.2 Τεχνολογίες που χρησιμοποιήθηκαν για την εφαρμογή Android.....	41
5.2.1 Το Android Studio	41
5.2.2 Java	42
5.2.3 XML	42
5.3 Τεχνολογίες που χρησιμοποιήθηκαν στο Backend	42
5.3.1 Ruby	43
5.3.2 Rails.....	43
5.3.3 JSON	44
5.3.4 MySQL.....	44
ΚΕΦΑΛΑΙΟ 6. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....	48
ΑΝΑΦΟΡΕΣ	49
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	50
ΠΑΡΑΡΤΗΜΑΤΑ.....	47

Ακρωνύμια

ΑΜΚΑ - Αριθμός Μητρώου Κοινωνικής Ασφάλισης

HTML - HyperText Markup Language

CSS - Cascading Style Sheets

API - Application Programming Interface

SGML - Standard Generalized Markup Language

XML - eXtensible Markup Language

JVM - Java Virtual Machine

JSON - JavaScript Object Notation

DBMS - DataBase Management Systems

Λέξεις κλειδιά

frontend, backend, AMKA, web application, browser, splash screen, email, password, gender, Full name, password confirmation, framework, component, Vue.js, Angular, React

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

1.1 Περιγραφή της εργασίας

Το αντικείμενο της εργασίας είναι η σχεδίαση και η ανάπτυξη μιας ιατρικής εφαρμογής όπου η κύρια λειτουργία της θα είναι το ιστορικό των παθήσεων του κάθε χρήστη όπως επίσης και η εμφάνιση των παθήσεων που μπορεί να νοσήσει με βάση την κληρονομικότητα.

Πέρα από αυτές τις δύο βασικές λειτουργίες, ο χρήστης θα έχει την δυνατότητα να βλέπει κάποια γενικά ιατρικά άρθρα τα οποία θα βρίσκονται στην αρχική οθόνη, όπως και άρθρα τα οποία θα σχετίζονται με τις παθήσεις που έχει δηλώσει ο ίδιος. Είναι σημαντικό να αναφερθεί ότι ο χρήστης για να μπορεί να χρησιμοποιήσει την εφαρμογή και για να αξιοποιήσει τις δυνατότητές της, θα πρέπει να πραγματοποιήσει εγγραφή και κατά της είσοδο του να δώσει τα κατάλληλα στοιχεία όπως το ΑΜΚΑ των συγγενικών του προσώπων.

Επίσης εδώ θα πρέπει να αναφερθεί ότι η συγκεκριμένη εφαρμογή είναι Web Application[1] και έχει την δυνατότητα να τρέχει σε όλες τις κινητές συσκευές όπως επίσης και σε οποιοδήποτε browser. Με λίγα λόγια ο χρήστης θα έχει την δυνατότητα να έχει πρόσβαση στην εφαρμογή από οποιοδήποτε λειτουργικό σύστημα εκείνος επιθυμεί.

1.2 Κίνητρο για την ανάπτυξη της εργασίας

Κίνητρο για την ανάπτυξη της εφαρμογής αποτέλεσε το μάθημα της Ιατρικής Πληροφορικής[2] όπου σε συνεργασία με τον επιβλέποντα καθηγητή, δόθηκε το θέμα της εργασίας. Το σημαντικό είναι ότι πέρα της αρχικής ιδέας, έγιναν κάποιες χρήσιμες προσθήκες όπως είναι η κληρονομικότητα με τις παθήσεις και τα άρθρα που θα μπορεί να διαβάζει ο χρήστης. Φυσικά πέρα από αυτά που έχουν αναφερθεί μέχρι τώρα η εφαρμογή διαθέτει και άλλες δυνατότητες οι οποίες όμως θα αναφερθούν πιο κάτω.

1.3 Στόχος εργασίας

Στόχος της εργασίας όπως έχει αναφερθεί και πιο πάνω είναι η ανάπτυξη μιας ιατρικής εφαρμογής η οποία θα κρατάει το ιστορικό των παθήσεων που έχει νοσήσει ο χρήστης καθώς και τις παθήσεις τις οποίες έχει πιθανότητα να τις αποκτήσει με βάση την κληρονομικότητα.

Πιο αναλυτικά:

- Web Application : Δημιουργία ενός Web Application ώστε ο χρήστης να μπορεί να έχει πρόσβαση από την κινητή του συσκευή αλλά και από οποιοδήποτε υπολογιστή.
- Ιστορικό και κληρονομικότητα : Καταγραφή του ιστορικού των παθήσεων των χρηστών . Επίσης ο χρήστης θα μπορεί να δηλώνει ο ίδιος τα συγγενικά του πρόσωπα ώστε να προκύπτει η κληρονομικότητα.
- Γενικά και σχετικά άρθρα: Θα υπάρχουν γενικά ιατρικά άρθρα όπου ο χρήστης θα μπορεί να τα διαβάζει στην κύρια οθόνη της εφαρμογής. Επίσης όταν ο χρήστης θα δηλώνει μια πάθηση θα υπάρχουν σχετικά ιατρικά άρθρα τα οποία θα έχουν σχέση με τις παθήσεις που έχει δηλώσει ο χρήστης.
- Γενικές λειτουργίες της εφαρμογής : Θα υπάρχουν κάποιες γενικές λειτουργίες όπως ότι ο χρήστης θα μπορεί να ανεβάσει φωτογραφία προφίλ , επίσης θα μπορεί να αλλάξει τον κωδικό πρόσβασης όπως ακόμα θα υπάρχει και μια φόρμα επικοινωνίας που θα μπορεί να επικοινωνήσει με τον διαχειριστή.

1.4 Δομή εργασίας

Όπως είδαμε και παραπάνω στο πρώτο κεφάλαιο αναφέρονται η περιγραφή της εργασίας, το κίνητρο καθώς και ο στόχος αυτής για το καλύτερο δυνατό αποτέλεσμα.

Στο δεύτερο κεφάλαιο καταγράφεται η ανάλυση των απαιτήσεων της εργασίας καθώς και κάποιες βασικές έννοιες που θα τις αναλύσουμε με μεγαλύτερη προσοχή στα υπόλοιπα κεφάλαια.

Στο τρίτο κεφάλαιο έχουμε την ανασκόπηση πεδίου. Εδώ θα παρουσιάσουμε κάποιες παρόμοιες εφαρμογές σε Android[3] και θα αναλύσουμε τις λειτουργίες τους.

Στο τέταρτο κεφάλαιο θα παρουσιάσουμε και θα αναλύσουμε την εφαρμογή που φτιάξαμε καθώς και θα δούμε αναλυτικά όλες τις λειτουργίες της.

Στο πέμπτο κεφάλαιο γίνεται αναφορά στην αρχιτεκτονική του συστήματος. Εδώ θα παρουσιάσουμε τα εργαλεία που χρησιμοποιήθηκαν, σε ποιες γλώσσες έχει γραφτεί η εφαρμογή καθώς και πως πραγματοποιήθηκε η υλοποίησή της.

ΚΕΦΑΛΑΙΟ 2. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

2.1 Web εφαρμογή

Η εφαρμογή χωρίζεται σε δύο κομμάτια όσο αναφορά το frontend. Αυτό έχει να κάνει με την κύρια εφαρμογή η οποία είναι γραμμένη σε Vue.js[4] και στο δεύτερο κομμάτι το οποίο είναι γραμμένο σε Android και θα δούμε πιο κάτω.

Όσο αναφορά το web κομμάτι της εφαρμογής που όπως αναφέραμε και πιο πάνω είναι γραμμένο σε Vue.js θα παρέχει τις παρακάτω δυνατότητες στον χρήστη:

- Θα μπορεί να πραγματοποιεί εγγραφή στο σύστημα δίνοντας τα σωστά του στοιχεία και να πραγματοποιεί είσοδο στην εφαρμογή
- Θα πραγματοποιεί είσοδο στην εφαρμογή με τα ανάλογα σωστά του στοιχεία
- Στην κεντρική σελίδα της εφαρμογής θα υπάρχει ένας χαιρετισμός με το ανάλογο email του χρήστη και στην συνέχεια θα ακολουθούν ένα σύντομο κείμενο το οποίο θα αναφέρεται στην εφαρμογή. Ακόμη θα υπάρχουν τα top news όπου ο χρήστης θα μπορεί να διαβάσει κάποια γενικά ιατρικά άρθρα. Τέλος θα υπάρχουν κάποια κουμπιά με τα οποία ο χρήστης θα μπορεί να κινείται και στα υπόλοιπα παράθυρα
- Με το κουμπί το οποίο θα βρίσκεται μέσα στην αρχική σελίδα της εφαρμογής , ο χρήστης θα μπορεί να προσθέτει τις παθήσεις από τις οποίες έχει νοσήσει καθώς και να βλέπει το ιστορικό του. Επίσης εκεί θα υπάρχει και ένα άλλο κουμπί όπου θα ανοίγει μια φόρμα επικοινωνίας σε περίπτωση που ο χρήστης θα θέλει να επικοινωνήσει με τον διαχειριστή. Τέλος θα υπάρχει και ένα κουμπί με το οποίο θα μπορεί να πραγματοποιεί έξοδο από την εφαρμογή.
- Στα αριστερά είτε με την χρήση κουμπού είτε τραβώντας την οθόνη από αριστερά προς τα δεξιά θα ανοίγει ένα πλευρικό παράθυρο στο οποίο θα βρίσκονται το προφίλ του χρήστη , οι ρυθμίσεις του χρήστη , τα προτεινόμενα άρθρα και ένα κουμπί με τα σχετικά της εφαρμογής.
- Στο προφίλ του χρήστη θα εμφανίζονται όλα τα στοιχεία του καθώς και το όνομα των συγγενικών του προσώπων εφόσον αυτός τα έχει δηλώσει. Θα υπάρχει μια φωτογραφία προφίλ την οποία θα μπορεί να την αλλάζει όποτε θέλει ο χρήστης. Στην συνέχεια θα μπορεί να βλέπει τις πιθανότητες από τις παθήσεις τις οποίες μπορεί να νοσήσει με χρήση διαγράμματος αλλά και σε κείμενο απλής μορφής.
- Στις ρυθμίσεις ο χρήστης μπορεί να αλλάξει τον κωδικό πρόσβασης και να δηλώσει το ΑΜΚΑ των συγγενικών του προσώπων.
- Στα articles θα μπορεί να βλέπει τα προτεινόμενα άρθρα με βάσει τις παθήσεως που έχει δηλώσει .
- Στο τέλος θα υπάρχει ένα κουμπί με τα σχετικά της εφαρμογής όπου θα αναγράφονται και οι δημιουργοί της.

2.2 Android εφαρμογή

Η Android εφαρμογή θα δίνει την δυνατότητα στον χρήστη να έχει όλες τις παραπάνω λειτουργίες στην κινητή του συσκευή μόνο που σε αυτή την περίπτωση ο χρήστης δεν θα χρειάζεται να ανοίξει κάποιον browser για να περιηγηθεί στην εφαρμογή. Εδώ θα πρέπει να αναφέρουμε ότι οι παραμετροποιήσεις που έχουν γίνει ώστε να γίνει σε τελικό στάδιο εφαρμογή για κινητό τηλέφωνο , είναι λίγες καθώς το μόνο που έχει προστεθεί παραπάνω στον κώδικα Android είναι κυρίως έλεγχοι που έχουν να κάνουν με την συνδεσιμότητα της κινητής συσκευής με τον server καθώς και κάποιες επιπλέον ρυθμίσεις για να λειτουργεί η εφαρμογή σε όλες τις οθόνες κινητών τηλεφώνων.

Τέλος θα υπάρχει το εικονίδιο της εφαρμογής για να μπορεί ο χρήστης να βρίσκει και να ξεχωρίζει την εφαρμογή οπτικά , με πιο εύκολο τρόπο. Βέβαια θα υπάρχει και το απαραίτητο

εικονίδιο (splash screen[5]) που θα εμφανίζεται κατά την έναρξη της εφαρμογής και μόνο , για αισθητικούς λόγους.

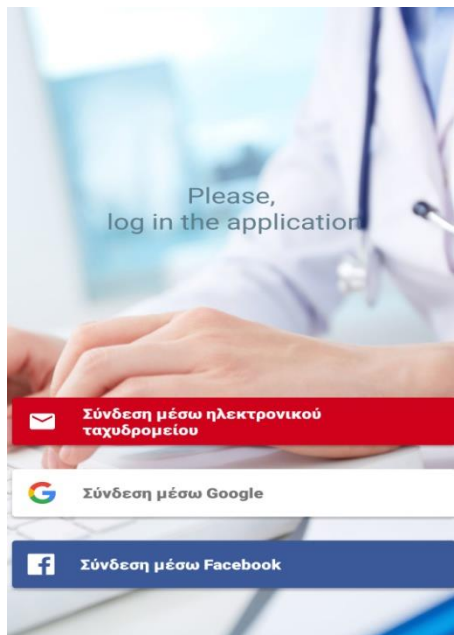
ΚΕΦΑΛΑΙΟ 3. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ

Υπάρχει αρκετό ενδιαφέρον στον κόσμο της ιατρικής πληροφορικής τα τελευταία χρόνια και αυτό φαίνεται από τις εφαρμογές που δημιουργούνται είτε για χρήση από τους ίδιους ασθενείς είτε για χρήση από επαγγελματίες στον χώρο της υγείας. Στην παρούσα φάση θα αναφέρουμε και θα αναλύσουμε κάποιες ιατρικές εφαρμογές οι οποίες θα είναι παρόμοιες με αυτή που αναπτύχθηκε σε αυτή την διπλωματική και όπως έχει αναφερθεί και πιο πάνω οι εφαρμογές αυτές θα έχουν να κάνουν με την χρήση της κινητής συσκευής .

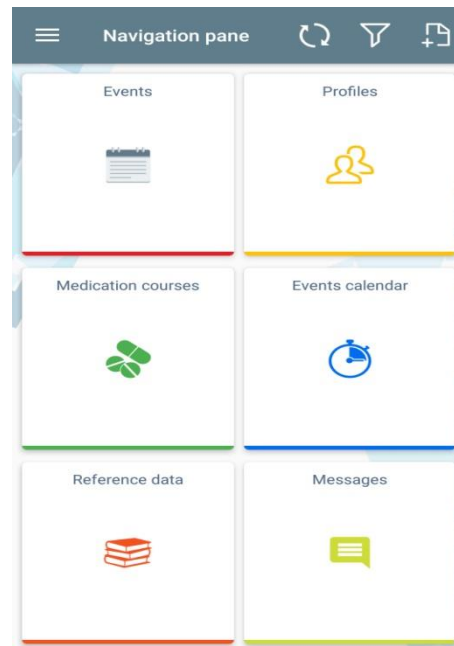
3.1 Medical Records

Η εφαρμογή Medical records[6] αποτελεί μια πολύ καλή επιλογή για όποιον θέλει να οργανώσει τα ραντεβού με τους γιατρούς , να κρατήσει επικοινωνία με τους γιατρούς μέσω της εφαρμογής αλλά και να μπορεί να βλέπει , και να διαχειρίζεται το ιστορικό των παθήσεων είτε αυτό πρόκειται για μια απλή πάθηση , είτε πρόκειται για εξετάσεις οι οποίες αποτελούνται από αρχεία , όπως είναι οι ακτινογραφίες.

Εικόνα 3.1



Εικόνα 3.2

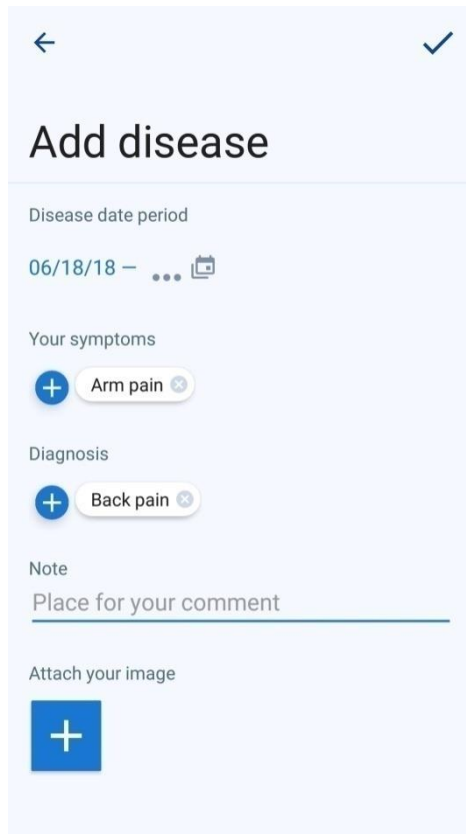


Όπως βλέπουμε και πιο πάνω η συγκεκριμένη εφαρμογή χρειάζεται εγγραφή χρήστη για να μπορεί να την χρησιμοποιήσει και αυτό είναι πολύ σημαντικό γιατί τα δεδομένα της εφαρμογής δεν αποθηκεύονται μέσα στην κινητή συσκευή αλλά σε κάποιον απομακρυσμένο server. Η σύνδεση στην εφαρμογή μπορεί να επιτευχθεί μέσω email του χρήστη ή μέσω λογαριασμού Google, ακόμη και μέσω του λογαριασμού Facebook που μπορεί να έχει ο χρήστης.

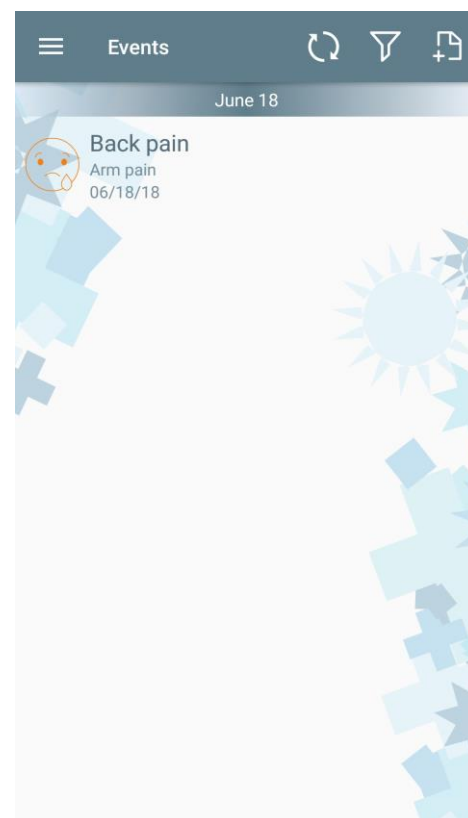
Στην διπλανή εικόνα (εικόνα 3.2) βλέπουμε την αρχική οθόνη της εφαρμογής με τις περισσότερες επιλογές που μπορεί να έχει ο χρήστης όπως είναι τα events, το προφίλ του χρήστη, την φαρμακευτική αγωγή, το ημερολόγιο με τα event, το reference data και τα μηνύματα όπου μπορεί να επικοινωνεί με τον γιατρό του.

Στην συνέχεια θα δούμε πιο αναλυτικά το καθένα από αυτά σε τι αναφέρεται:

Εικόνα 3.3



Εικόνα 3.4

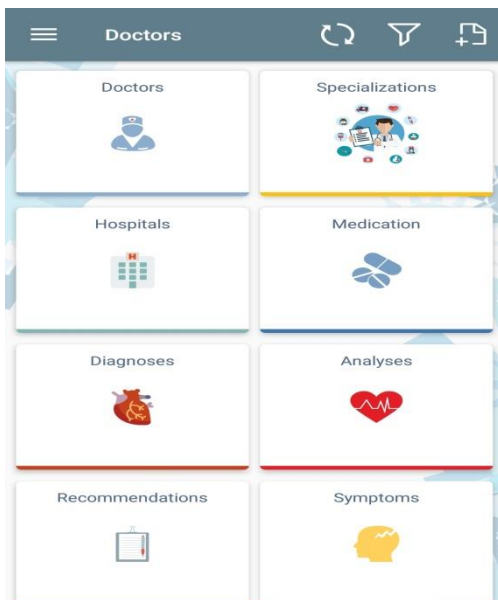


- **Event:** Με αυτή την επιλογή ο χρήστης έχει την επιλογή να προσθέσει κάποια πάθηση από την οποία έχει νοσήσει. Πιο αναλυτικά μπορεί να βάλει την περίοδο στην οποία έχει νοσήσει, τα συμπτώματα της πάθησης, η διάγνωση που έχει κάνει ο προσωπικό του γιατρός και κάποιες σημειώσεις αν αυτό είναι απαραίτητο. Επίσης υπάρχει επιλογή για να προσθέσει αρχείο όπως είναι η ακτινογραφία ή κάποιο άλλο έγγραφο που θα αναφέρεται στην γνωμάτευση της πάθησης. Όπως βλέπουμε στην εικόνα 3.3 ο χρήστης μπορεί ακόμη να επεξεργαστεί τη πάθηση. Στην εικόνα 3.4 βλέπουμε το ιστορικό που έχει ο χρήστης με τις παθήσεις τις οποίες έχει την δυνατότητα αν θέλει και να τις διαγράψει.
- **Προφίλ χρήστη:** Σε αυτήν την περίπτωση ο χρήστης μπορεί να επεξεργαστεί το προφίλ του αλλάζοντας τον κωδικό πρόσβασης το email και βάζοντας μια φωτογραφία προφίλ. Αρκετές δυνατότητες βέβαια από αυτές που αναφέραμε αλλά και από κάποιες που θα αναφέρουμε πιο κάτω χρειάζεται η pro έκδοση ώστε να τις χρησιμοποιήσει ο χρήστης. Μια από αυτές είναι και η επεξεργασία του προφίλ του χρήστη.

- Event calendar: Εδώ υπάρχει ένα ημερολόγιο στο οποίο ο χρήστης μπορεί να βλέπει το ιστορικό με τις παθήσεις όπως και την φαρμακευτική του αγωγή με βάση την ημερομηνίες που είναι δηλωμένες από τον ίδιο.

Εικόνα 3.5

- Medication course: Εδώ ο χρήστης μπορεί να προσθέσει την φαρμακευτική αγωγή που μπορεί να παίρνει για κάποια πάθηση. Πιο αναλυτικά μπορεί να αποθηκεύσει τον τίτλο της εργασίας , την φαρμακευτική αγωγή τον τύπο του φαρμάκου που παίρνει , την δοσολογία , την ημερομηνία που ξεκίνησε την αγωγή καθώς και την διάρκεια της αγωγής όπως επίσης και το κάθε πότε πρέπει να παίρνει την συγκεκριμένη αγωγή. Στην συνέχεια ο χρήστης και μετά την αποθήκευση της θεραπεία μπορεί να την επιλέξει και να την επεξεργαστεί όπως και να την διαγράψει.



Εικόνα 3.6

- **Reference data:** Εδώ ο χρήστης μπορεί έχει αρκετές ρυθμίσεις, όπως βλέπουμε στην εικόνα 3.6 που μπορεί να κάνει στην εφαρμογή και κυρίως να προσθέσει καινούριες πληροφορίες όπως είναι το όνομα του θεράποντα γιατρού, η ειδικότητα των γιατρών, τα νοσοκομεία, την φαρμακευτική αγωγή, το όνομα της διάγνωσης. Επίσης μπορεί να μάθει για τις ιατρικές εξετάσεις, να δημιουργήσει σημειώσεις με τις συστάσεις που μπορεί να του έχει δώσει ο γιατρός καθώς να πάρει διάφορες πληροφορίες σχετικά με τα συμπτώματα μιας πάθησης.

Εικόνα 3.7

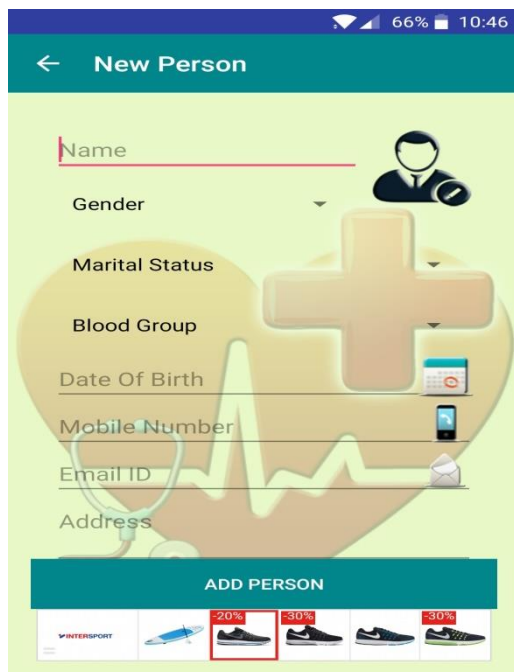
- **Messages:** Στην συγκεκριμένη επιλογή του χρήστη του δίνεται η δυνατότητα να μπορεί να επικοινωνήσει με τον θεράποντα γιατρό. Στην δική μας περίπτωση, όπως φαίνεται και την εικόνα 3.7 δεν υπάρχουν νέα μηνύματα καθώς δεν έχουμε δηλώσει κάποιον θεράποντα γιατρό.

- Επίσης η εφαρμογή διαθέτει και ένα πλαϊνό παράθυρο με το οποίο υπάρχουν όλες οι παραπάνω ρυθμίσεις και μπορεί έτσι ο χρήστης να τις χρησιμοποιήσει για μεγαλύτερη ευκολία στην περιήγησή του.

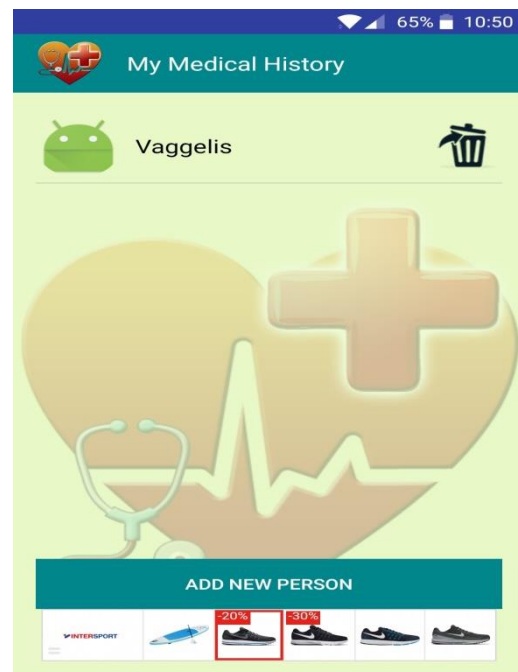
3.2 My Medical History

Η εφαρμογή My Medical History[7] είναι παρόμοια αυτής που παρουσιάσαμε πιο πάνω αν και στην συγκεκριμένη περίπτωση δεν υπάρχουν τόσες ρυθμίσεις που μπορεί να κάνει ο χρήστης σχετικά με τις παθήσεις και το ιστορικό του. Ωστόσο μπορεί εύκολα κάποιος να κρατήσει το ιστορικό από τις παθήσεις του , να τις επεξεργαστεί ή να τις διαγράψει , να προσθέσει και άλλους ασθενείς. Πιο αναλυτικά οι δυνατότητες θα παρουσιαστούν στη συνέχεια:

Εικόνα 3.8

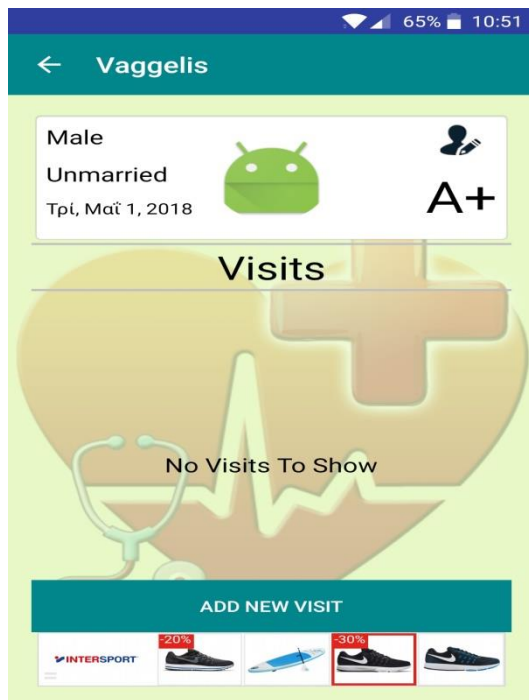


Εικόνα 3.9

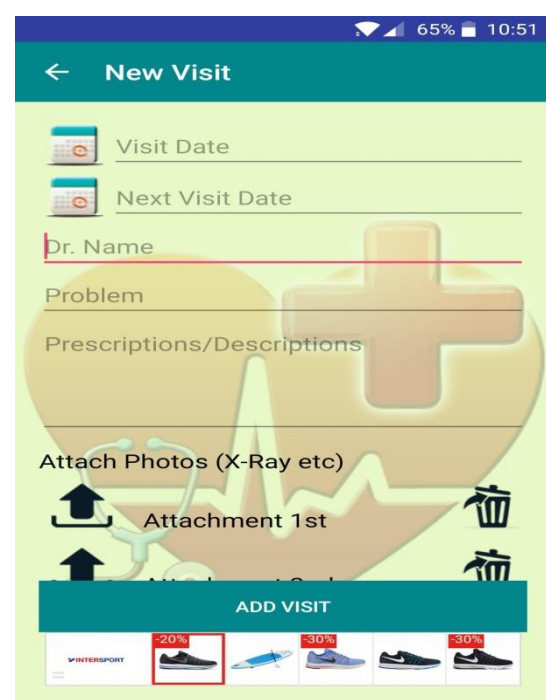


- Στην εικόνα 3.8 βλέπουμε πως μπορεί ο χρήστης να προσθέσει έναν ασθενή. Για την εγγραφή του χρήστη είναι απαραίτητες μερικές πληροφορίες όπως είναι το όνομα του χρήστη, το φύλο , η οικογενειακή του κατάσταση (αν είναι παντρεμένος ή όχι) , την ομάδα αίματος. Επίσης ο χρήστης θα πρέπει να συμπληρώσει την ημερομηνία της γέννησής του όπως ακόμη τον αριθμό του τηλεφώνου του , το email του και την διεύθυνση την οποία μένει. Στην εικόνα 3.9 βλέπουμε έναν από τους χρήστες που έχουμε δημιουργήσει.

Εικόνα 3.10



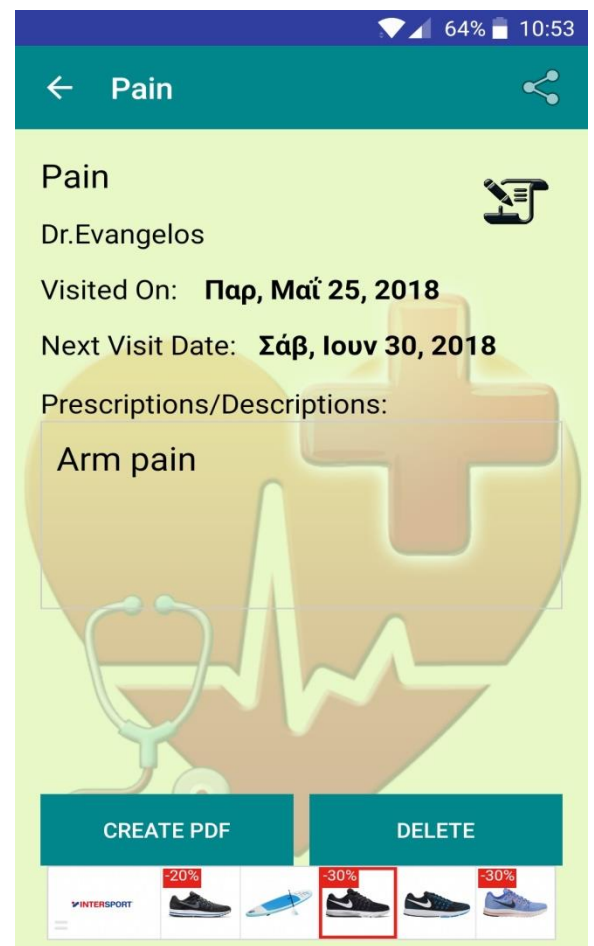
Εικόνα 3.11

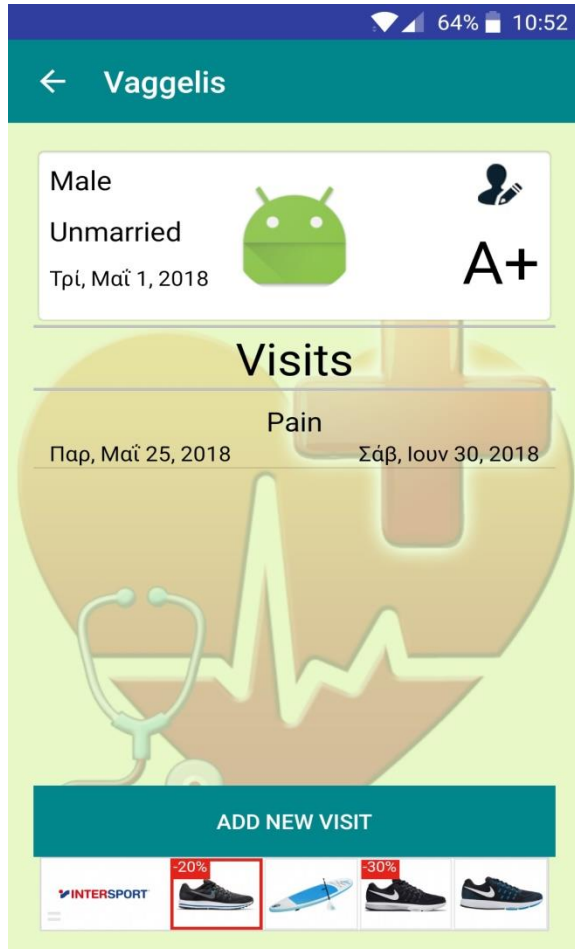


- Στην εικόνα 3.10 βλέπουμε την καρτέλα του ασθενή που έχει επιλέξει ο χρήστης να δει. Εδώ εμφανίζονται κάποιες βασικές πληροφορίες του ασθενή όπως είναι το φύλο του , η οικογενειακή του κατάσταση , η ημερομηνία γέννηση του η ομάδα αίματος και μια φωτογραφία προφίλ που θα έχει ανεβάσει ο ίδιος. Στην κάτω πλευρά της οθόνης υπάρχει ένα κουμπί με το οποίο ανοίγει μια καινούρια φόρμα όπου ο χρήστης μπορεί να δώσει τα στοιχεία για μια καινούρια επίσκεψη στον γιατρό , κρατώντας έτσι το ιστορικό με τις παθήσεις του. Σε αυτήν την περίπτωση μπορεί να βάλει την ημερομηνία που επισκέφτηκε τον θεράποντα γιατρό και την ημερομηνία που είναι προγραμματισμένη η επόμενη επίσκεψη. Το όνομα του γιατρού μπορεί να δηλωθεί καθώς δεν είναι από τις απαραίτητες πληροφορίες για να αποθηκευτεί στο ιστορικό αλλά είναι από τις σημαντικές. Τέλος και βασικό ο χρήστης μπορεί να δηλώσει την πάθηση που έχει καθώς και μια περιγραφή αυτής όπως επίσης και να ανεβάσει κάποιο αρχείο που σχετίζεται με τις ιατρικές εξετάσεις που έχει χρειαστεί να κάνει για την συγκεκριμένη πάθηση.

Εικόνα 3.12

Εικόνα 3.13





Στην εικόνα 3.12 βλέπουμε το ιστορικό των παθήσεων του χρήστη Vaggelis που στην συγκεκριμένη περίπτωση αποτελείται μόνο από μία πάθηση. Στην εικόνα 3.13 βλέπουμε ότι ο χρήστης μπορεί να επεξεργαστεί την συγκεκριμένη πάθηση αλλά και να δει όλες την πληροφορίες που στην αρχική οθόνη δεν φαίνονται. Εδώ επίσης υπάρχουν άλλες δύο επιλογές που επιτρέπει στον χρήστη να κατεβάσει την πάθηση που έχει δηλώσει σε αρχείο pdf ή να την διαγράψει.

ΚΕΦΑΛΑΙΟ 4. ΠΑΡΟΥΣΙΑΣΗ ΚΑΙ ΧΡΗΣΗ ΕΦΑΡΜΟΓΗΣ

Σε αυτό το κεφάλαιο θα αναλύσουμε και θα παρουσιάσουμε το web application που έχουμε φτιάξει. Για να γίνει η παρουσίαση της εφαρμογής έχουμε φτιάξει ένα σενάριο όπου θα υπάρχουν τρεις χρήστες, father, mother,

son1. Οι δύο από τους χρήστες οι οποίοι θα είναι συγγενείς με τον son1 θα έχουν δηλώσει από τρεις παθήσεις ο καθένας ώστε ο son1 να μπορεί να δει τις πιθανότητες που θα έχει να πάθει κάποιες από αυτές τις παθήσεις. Θα δούμε επίσης πως δηλώνεται αυτή η κληρονομικότητα, θα αλλάξουμε κωδικούς πρόσβασης, θα δηλώσουμε παθήσεις όπως και θα δούμε το ιστορικό και θα επικοινωνήσουμε με τον διαχειριστή μέσα από την φόρμα επικοινωνίας.

4.1 Εικονίδιο εφαρμογής και Splash Screen

Εικόνα 4.1.1

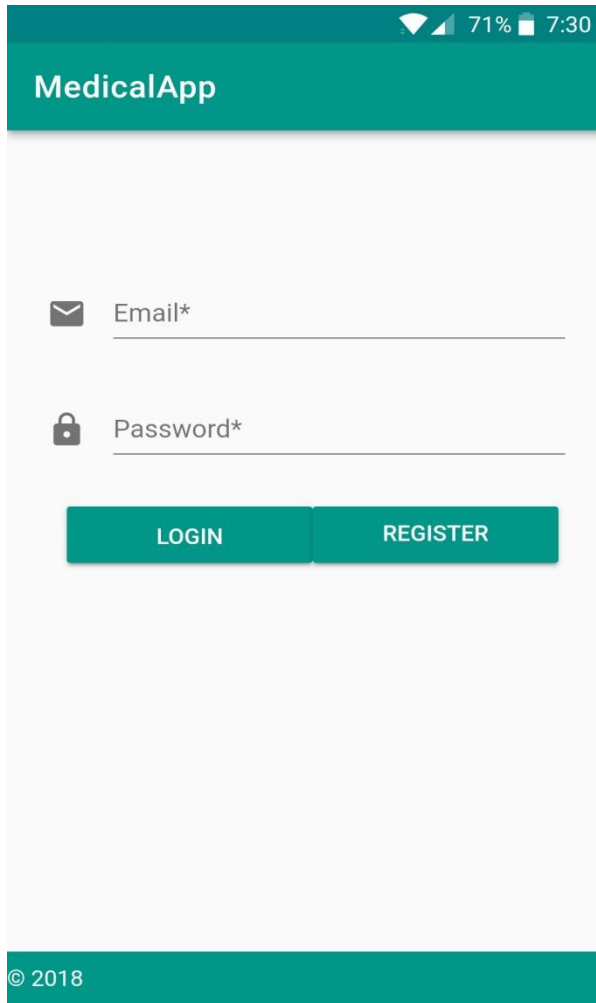


Στην πιο πάνω εικόνα βλέπουμε το εικονίδιο το οποίο θα εμφανίζεται όταν θα ανοίγει η εφαρμογή και μόνο. Πιο συγκεκριμένα το splash screen θα εμφανίζεται για λίγα δευτερόλεπτα και στην συνέχεια θα περνάμε στην αρχική οθόνη της εφαρμογής που θα δούμε και πιο κάτω. Στην δική μας περίπτωση το splash screen δεν μπήκε μόνο από άποψη ομορφιάς αλλά και για να αποφύγουμε την λευκή οθόνη που θα υπήρχε κατά την εκκίνηση της εφαρμογής. Πιο κάτω θα αναφέρουμε και τι γίνεται σε περίπτωση που υπάρξει κάποιο πρόβλημα στην σύνδεση με το server.

Το συγκεκριμένο εικονίδιο μιας και είναι αυτό που ο χρήστης το έχει συνδέσει άμεσα με την εφαρμογή, θα είναι και το εικονίδιο το οποίο θα εμφανίζεται στο μενού της κινητής συσκευής. Αυτό βοηθάει τον χρήστη να βρίσκει την εφαρμογή με μεγαλύτερη ευκολία.

4.2 Αρχική οθόνη – Login screen

Εικόνα 4.2.1



Στην εικόνα 4.2.1 βλέπουμε στην αρχική οθόνη της εφαρμογής. Αυτή η οθόνη θα εμφανίζεται μόλις κλείσει το splash screen. Όπως βλέπουμε αποτελείται από δύο πεδία τα οποία θα συμπληρώνει ο χρήστης και στην συνέχεια από δύο κουμπιά το οποίο το ένα θα πραγματοποιεί την σύνδεση στην εφαρμογή ενώ το άλλο θα μας πηγαίνει σε μια καινούργια οθόνη που ο χρήστης θα κάνει εγγραφή στο σύστημα και θα το δούμε στην επόμενη υποενοότητα. Στα δύο πεδία ο χρήστης καλείται να συμπληρώσει το email και το password του ώστε να πραγματοποιήσει σύνδεση στην εφαρμογή.

Η σύνδεση πραγματοποιείται πατώντας το κουμπί LOGIN και εφόσον τα πεδία που συμπλήρωσε ο χρήστης είναι σωστά με βάση τους περιορισμούς που θα δούμε πιο κάτω αλλά και με βάση τα στοιχεία που έχει δηλώσει ο ίδιος.

4.3 Register

Εικόνα 4.3.1

Εικόνα 4.3.2

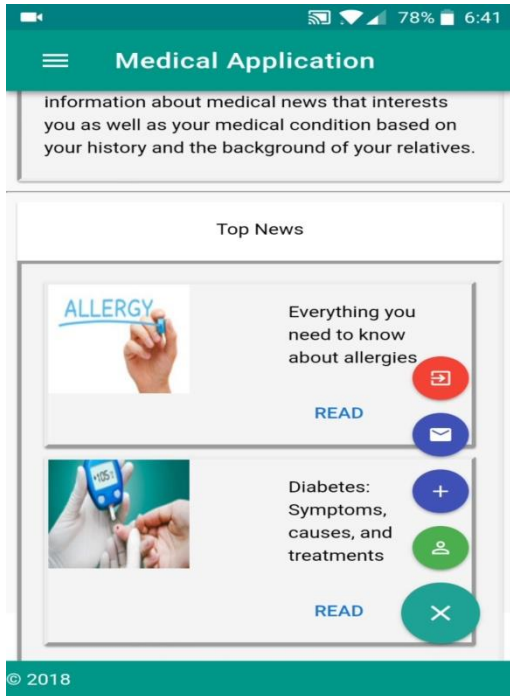
Εδώ βλέπουμε την επόμενη οθόνη που θα εμφανίζετε πατώντας το κουμπί REGISTER στην αρχική οθόνη. Σε αυτό το βήμα ο χρήστης θα πρέπει να δώσει όλα τα στοιχεία τα οποία υπάρχουν στην φόρμα καθώς όλα είναι υποχρεωτικά. Στην περίπτωση που ο χρήστης δεν θα συμπληρώσει έστω και ένα από τα παραπάνω ή θα συμπληρώσει κάτι λάθος, θα εμφανίζεται ένα παράθυρο το οποίο θα λέει στον χρήστη τι λάθος έχει κάνει. Για να πραγματοποιήσει εγγραφή στο σύστημα θα πρέπει να πατήσει το κουμπί REGISTER. Το κουμπί BACK μας γυρνάει πίσω στην αρχική οθόνη της εφαρμογής που οι χρήστης μπορεί να πραγματοποιήσει είσοδο εφόσον έχει λογαριασμό. Στο δικό μας παράδειγμα θα δημιουργήσουμε πρώτα τον λογαριασμό του πατέρα όπως φαίνεται στην εικόνα 4.3.2. Έχουμε δώσει το email, το password και password confirmation, το όνομα του πατέρα το οποίο στο παράδειγμά μας είναι το Father, το φύλλο και τέλος το ΑΜΚΑ το οποίο αποτελείται μόνο από έντεκα αριθμούς. Στην συνέχεια και αφού ο χρήστης κάνει εγγραφή, μπαίνουμε στην εφαρμογή όπως φαίνεται στην παρακάτω εικόνα. Εδώ υπάρχει ένας χαιρετισμός με το email του χρήστη, μια μικρή παράγραφος για την εφαρμογή, τα Top News και ένα κουμπί στο κάτω δεξιό μέρος της οθόνης που θα να αναλύσουμε πιο κάτω τι κάνει.

Εικόνα 4.3.3

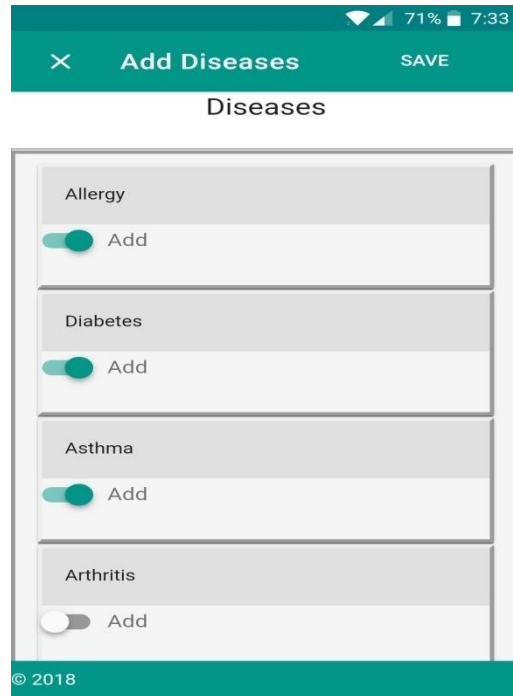
Εικόνα 4.3.4

4.4 Δήλωση πάθησης και ιστορικό όλων των παθήσεων

Εικόνα 4.4.1



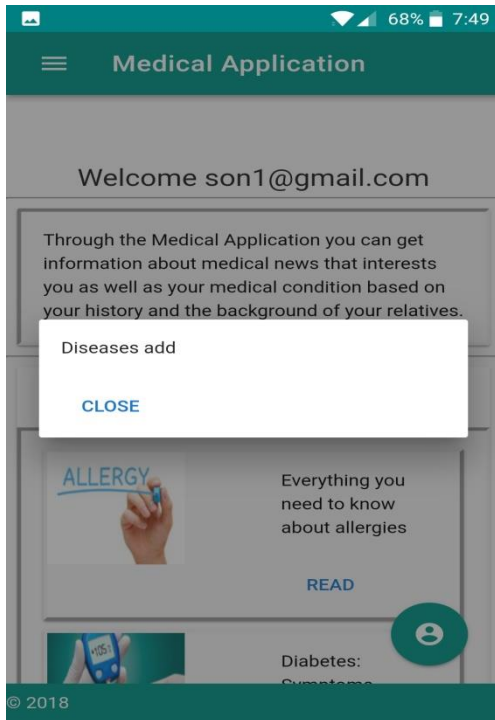
Εικόνα 4.4.2



Εικόνα 4.4.3

Εικόνα 4.4.4





Σε αυτή την υποενότητα θα δούμε πως ο χρήστης μπορεί να δηλώσει μια πάθηση και στην συνέχεια πως μπορεί να δει το ιστορικό με τις παθήσεις του. Στην εικόνα 4.4.1 βλέπουμε το κουμπί που βρίσκεται στην κάτω δεξιά πλευρά της οθόνης. Αυτό μας δίνει τέσσερις επιλογές από κάτω προς τα πάνω:

- Ιστορικό των παθήσεων
- Δήλωση κάποιας πάθησης από τον χρήστη
- Contact form
- Έξοδος από την εφαρμογή

Στην εικόνα 4.4.2 βλέπουμε τις παθήσεις που είναι αποθηκευμένες στην βάση δεδομένων. Εδώ ο χρήστης δεν μπορεί να προσθέσει κάποια πάθηση η οποία δεν υπάρχει στην βάση δεδομένων, οπότε αυτό που έχει να κάνει είναι να δηλώσει κάποιες από αυτές. Στην συγκεκριμένη περίπτωση ο χρήστης σε αυτό το βήμα δηλώνει τις τρεις πρώτες παθήσεις που υπάρχουν για επιλογή. Η δήλωση γίνεται με πολύ εύκολο τρόπο αφού αυτό που έχει να κάνει ο χρήστης είναι να πατήσει το διακόπτη, όπου και φάνετε ποιες παθήσεις έχει επιλέξει και στην συνέχεια να πατήσει το κουμπί SAVE που βρίσκετε στην πάνω δεξιά μεριά της οθόνης. Υπάρχει και η επιλογή όπου ο χρήστης μπορεί να βγει από αυτήν την οθόνη αν πατήσει το κουμπί στην πάνω αριστερή μεριά της οθόνης.

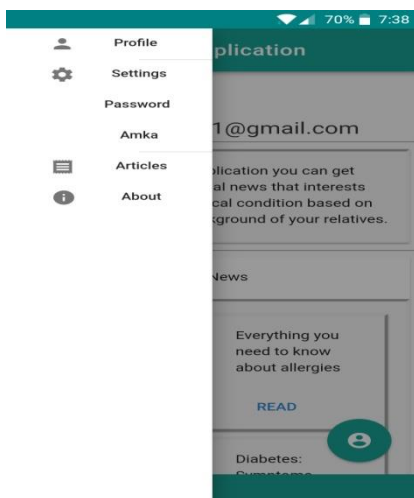
Στην συνέχεια και αφού ο χρήστης θα αποθηκεύσει τις παθήσεις που έχει, θα επιστρέψει στην αρχική οθόνη της εφαρμογής και θα εμφανιστεί ένα μήνυμα επιβεβαίωσης για να ξέρει ότι οι παθήσεις έχουν αποθηκευτεί στον λογαριασμό του. Φυσικά αυτή είναι μια διαδικασία που ο χρήστης μπορεί να την κάνει πολλές φορές ακόμη και για όλες τις παθήσεις. Αντίθετα δεν μπορεί ο χρήστης να διαγράψει μια πάθηση από την οποία έχει νοσήσει και αυτό γιατί είναι η λογική της εφαρμογής να κρατάει το ιστορικό του κάθε χρήστη από τις παθήσεις που έχει. Σε περίπτωση που έχει γίνει κάποιο λάθος υπάρχει και η φόρμα επικοινωνίας με την οποία μπορεί

να επικοινωνήσει με τον διαχειριστή, αυτό όμως θα το αναλύσουμε πιο κάτω βλέποντας και την αντίστοιχη φόρμα.

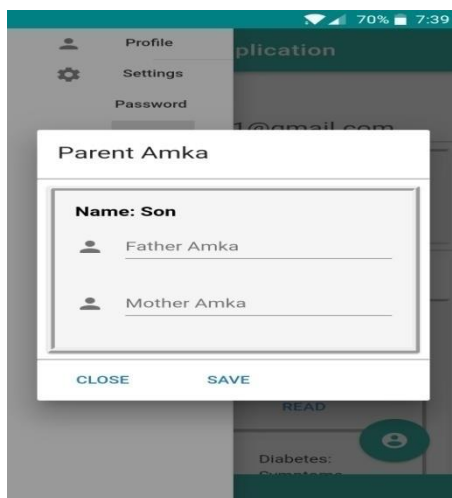
Στην εικόνα 4.4.4 βλέπουμε την οθόνη όπου ο χρήστης μπορεί να δει τις παθήσεις που έχει δηλώσει όπως το περιγράψαμε και στα προηγούμενα βήματα. Η διαδικασία του ιστορικού των παθήσεων αναφέρεται μόνο στον χρήστη και όχι στις παθήσεις που προκύπτουν από την συγγένεια, με κάποιο άλλο πρόσωπο. Επίσης αυτό το ιστορικό για λόγους προσωπικών δεδομένων μπορεί να το δει μόνο ο χρήστης. Με βάση τα βήματα που ακολουθήσαμε πιο πάνω βλέπουμε ότι στο ιστορικό του συγκεκριμένου χρήστη υπάρχουν οι τρεις παθήσεις που έχουμε δηλώσει. Σε αυτήν την οθόνη ο χρήστης δεν έχει κάποια επιλογή που μπορεί να διαλέξει και το μόνο που μπορεί να κάνει είναι να δει το ιστορικό που υπάρχει και ένα κουμπί εξόδου όπως και στα περισσότερα παράθυρα στην πάνω αριστερή μεριά της οθόνης.

4.5 Δήλωση Συγγένειας

Εικόνα 4.5.1



Εικόνα 4.5.2



Στο συγκεκριμένο βήμα για να δηλώσουμε την συγγένεια που μπορεί να έχουμε πρέπει να επιλέξουμε και να ανοίξουμε το μενού που φαίνεται στην εικόνα 4.5.1. Αυτό μπορεί να γίνει με δύο τρόπους:

- Πατώντας το κουμπί που βρίσκεται στην πάνω αριστερή πλευρά της οθόνης
- Τραβώντας την οθόνη από αριστερά προς τα δεξιά

Αφού προχωρήσουμε στο επόμενο βήμα και ανοίξουμε το παράθυρο όπως φαίνεται και πιο πάνω, υπάρχουν κάποιες επιλογές που μπορεί να κάνει ο χρήστης όπως:

- Το προφίλ του χρήστη
- Ρυθμίσεις
 - Αλλαγή κωδικού πρόσβασης
 - Δήλωση συγγένειας με τα ΑΜΚΑ
- Προτεινόμενα άρθρα με βάση τις παθήσεις που έχει δηλώσει ο χρήστης
- Επιλογή σχετικά για την εφαρμογή

Αφού ανοίξουμε το παράθυρο πατάμε στο κουμπί ρυθμίσεις και μας ανοίγουν οι επιλογές που βλέπουμε και που αναφέραμε πιο πάνω. Στην συνέχεια επιλέγουμε το κουμπί που λέει ΑΜΚΑ και αναφέρεται στην δήλωση των συγγενικών προσώπων.

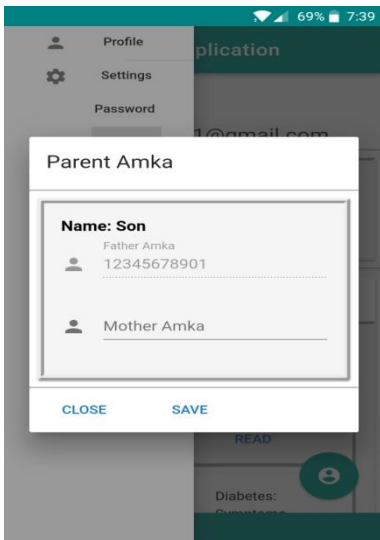
Στην εικόνα 4.5.2 βλέπουμε το παράθυρο που έχει ανοίξει ώστε ο χρήστης δίνοντας τα ΑΜΚΑ των συγγενικών του προσώπων να δηλώσει έτσι την συγγένεια. Εδώ ο χρήστης μπορεί να κάνει τα εξής βήματα:

- Να δώσει το ΑΜΚΑ μόνο του πατέρα του
- Να δώσει το ΑΜΚΑ μόνο της μητέρας του

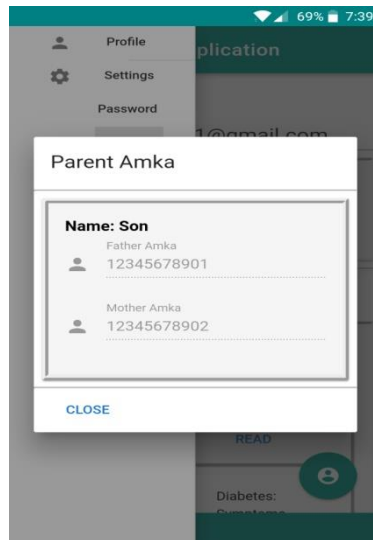
- Να δώσει το ΑΜΚΑ και των δύο ή και τίποτα

Εδώ θα πρέπει να υπενθυμίσουμε πως το ΑΜΚΑ αποτελείται ακριβώς από έντεκα αριθμούς και δεν μπορεί να είναι γράμματα ή χαρακτήρες. Φυσικά σε περίπτωση λάθους υπάρχουν και τα ανάλογα μηνύματα που θα δούμε στις παρακάτω ενότητες. Επίσης δίνοντας μια φορά το ΑΜΚΑ των συγγενικών του προσώπων δεν μπορεί να το αλλάξει μετά παρά μόνο το δει. Κάτι άλλο επίσης που δεν μπορεί να κάνει ο χρήστης σε αυτή την περίπτωση είναι να δώσει και στα δύο πεδία το ίδιο ΑΜΚΑ. Αυτό είναι ένα βήμα όπου ο χρήστης μπορεί να το κάνει όποτε θέλει είτε για τον έναν γονιό είτε και για τους δύο. Κάτι ακόμα που πρέπει να αναφέρουμε για όσο αναφορά την κληρονομικότητα μεταξύ των χρηστών είναι πως δεν φτάνει μόνο το ΑΜΚΑ αλλά θα πρέπει ο χρήστης (συγγενής) να έχει και αυτός λογαριασμό στην εφαρμογή και να έχει δηλώσει από τουλάχιστον μία πάθηση. Παρακάτω βλέπουμε δύο από αυτές τις περιπτώσεις που αναφέραμε πιο πάνω. Μια που ο χρήστης έχει δηλώσει ένα συγγενικό του πρόσωπο και άλλη μια και για τους δύο γονείς.

Εικόνα 4.5.3

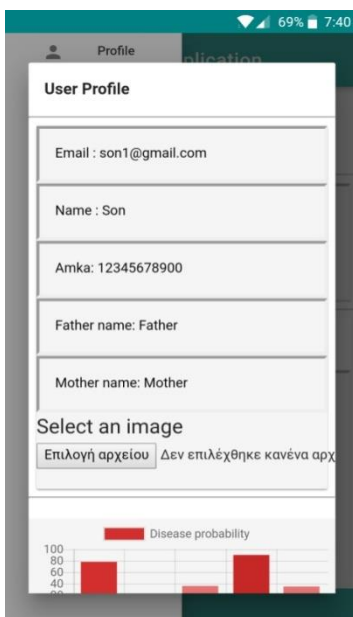


Εικόνα 4.5.4

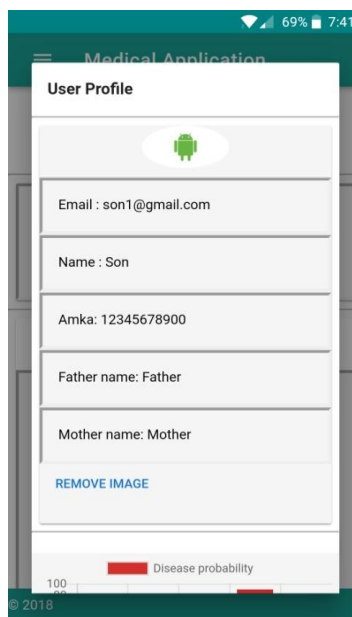


4.6 Το προφίλ του χρήστη

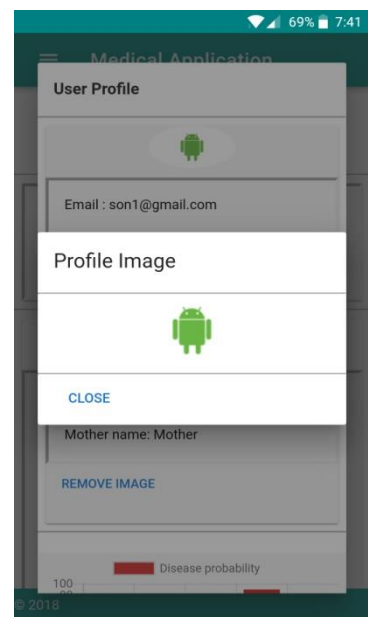
Εικόνα 4.6.1



Εικόνα 4.6.2



Εικόνα 4.6.3



Όπως φαίνεται στην εικόνα 4.6.1 αυτό είναι το προφίλ του χρήστη. Για να μπει κάποιος εκεί θα πρέπει να ανοίξει το πλαϊνό παράθυρο και να πατήσει πάνω στο κουμπί που λέει Profile. Το προφίλ χωρίζεται σε τέσσερα μέρη που είναι τα εξής:

- Πληροφορίες του χρήστη
- Επιλογή φωτογραφία προφίλ
- Διάγραμμα με τις πιθανότητες των παθήσεων
- Οι παθήσεις και οι πιθανότητες τους σε μορφή κειμένου

Στο πρώτο μέρος υπάρχουν οι πληροφορίες του χρήστη που είναι :

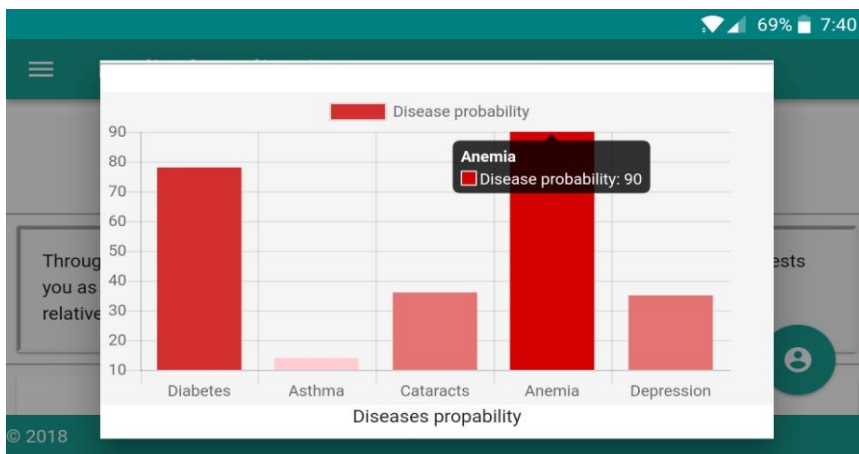
- Το email του
- Το όνομα και το επίθετό του σε ένα πεδίο όπως τα έδωσε όταν έκανε εγγραφή στο σύστημα
- Το ΑΜΚΑ του
- Το όνομα του πατέρα του
- Τα όνομα της μητέρας του

Σε περίπτωση φυσικά που δεν υπάρχουν δηλωμένα συγγενικά πρόσωπα τα πεδία με το όνομα του πατέρα και της μητέρας είναι κενά.

Στο δεύτερο μέρος ο χρήστης έχει την επιλογή να ανεβάσει μια φωτογραφία που θα φαίνεται στο προφίλ του. Αυτό γίνεται πατώντας το κουμπί επιλογής αρχείου και διαλέγοντας την φωτογραφία της επιλογής του. Στην εικόνα 4.6.2 φαίνεται μια φωτογραφία που έχει ανεβάσει ο χρήστης. Υπάρχει και η επιλογή να την αφαιρέσει και να ξανά ανεβάσει μια καινούρια και αυτό μπορεί να το επαναλάβει όσες φορές θέλει. Εδώ υπάρχει άλλη μία επιλογή η οποία όταν ο χρήστης θα πατάει πάνω στην φωτογραφία του προφίλ του , θα εμφανίζεται ένα παράθυρο όπου η φωτογραφία θα μεγαλώνει όπως φαίνεται και στην εικόνα 4.6.3.

Στην συνέχεια και στο τρίτο μέρος του προφίλ του χρήστη βλέπουμε το διάγραμμα το οποίο μας δείχνει τις παθήσεις και τις πιθανότητες που έχει ο χρήστης να νοσήσει από αυτές. Τα ονόματα των παθήσεων φαίνονται κάτω από κάθε μπάρα ενώ οι πιθανότητες φαίνονται είτε στο πλάι του διαγράμματος είτε πατώντας πάνω σε μια πάθηση όπως βλέπουμε και στην εικόνα 4.6.4.

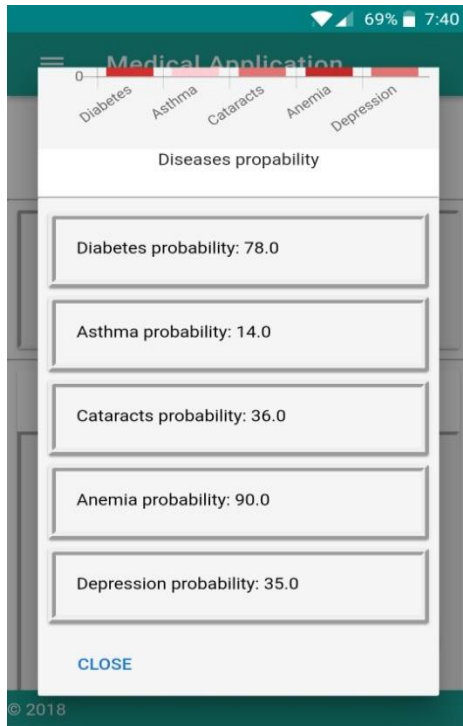
Εικόνα 4.6.4



Εδώ θα πρέπει να αναφέρουμε ότι όσο μεγαλύτερη είναι η πιθανότητα της πάθησης από την κληρονομικότητα τόσο πιο κόκκινη φαίνεται η μπάρα στο διάγραμμα. Σε περίπτωση βέβαια που ο χρήστης θα μπει και θα δηλώσει μια πάθηση την οποία την έχει και κάποιο συγγενικό του πρόσωπο , τότε μπαίνοντας στο προφίλ του για να δει και πάλι τις πιθανότητες , αυτή η πάθηση δεν θα υπάρχει πλέον εφόσον την έχει και ο ίδιος. Στην εικόνα 4.6.4 έχουμε γυρίσει την κινητή

συσκευή σε οριζόντιο προσανατολισμό για να φανούν όλα τα ονόματα των παθήσεων με μεγαλύτερη ευκολία.

Εικόνα 4.6.5



Στην εικόνα 4.6.5 βλέπουμε το τέταρτο μέρος στο προφίλ του χρήστη το οποίο μας δείχνει όλες τις παθήσεις που μας έδειχνε και το διάγραμμα αλλά σε μορφή κειμένου. Έτσι ο χρήστης θα έχει δύο επιλογές για να δει τις πιθανότητες, είτε μέσω διαγράμματος, είτε μέσω μορφή κειμένου όπως βλέπουμε και στην εικόνα. Εδώ το κουμπί της εξόδου βρίσκεται στο κάτω μέρος του παραθύρου το οποίο πατώντας το επιστρέφουμε στην αρχική οθόνη.

4.7 Αλλαγή κωδικού

Εικόνα 4.7.1

The screenshot shows a mobile application interface with a dark green header. Below the header, there is a horizontal bar with two categories: Profile and Settings. Below this bar, the text 'Change your password' is displayed. The main content area consists of a white box with a dark border. Inside the box, the email address 'Email: son1@gmail.com' is displayed. Below the email address, there are three input fields, each with a lock icon and a label: 'Password*', 'Confirm Password*', and 'Current Password*'. At the bottom of the box are two buttons: 'CLOSE' and 'SAVE'. The status bar at the top shows 69% battery and 7:42.

Εικόνα 4.7.2

The screenshot shows a mobile application interface with a dark green header. Below the header, there is a horizontal bar with two categories: Profile and Settings. Below this bar, the text 'Change your password' is displayed. The main content area consists of a white box with a dark border. Inside the box, the email address 'Email: son1@gmail.com' is displayed. Below the email address, there are three input fields, each with a lock icon and a label: 'Password*', 'Confirm Password*', and 'Current Password*'. The 'Password*' and 'Confirm Password*' fields are filled with dots. The 'Current Password*' field is filled with dots and has a blue underline. At the bottom of the box are two buttons: 'CLOSE' and 'SAVE'. The status bar at the top shows 69% battery and 7:42.

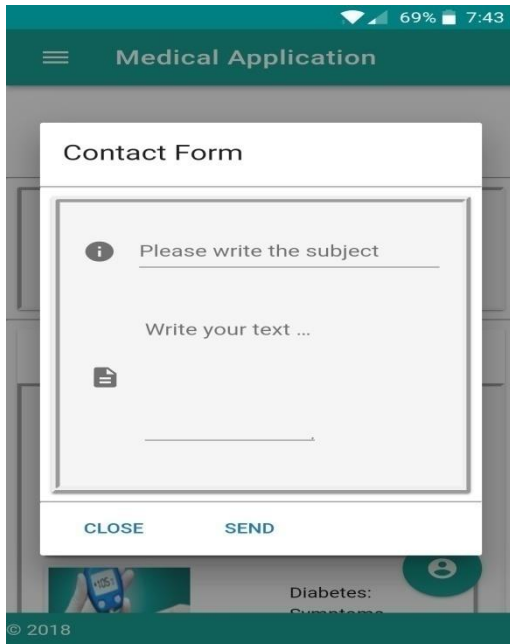
Στην εικόνα 4.7.1 βλέπουμε την φόρμα στην οποία ο χρήστης θα δώσει δύο φορές τον καινούριο κωδικό πρόσβασης και στην συνέχεια τον παλιό για λόγους ασφαλείας , ώστε να αλλάξει τον κωδικό πρόσβασης. Στην περίπτωση που ο χρήστης πατήσει SAVE και αποθηκεύσει στην βάση τον καινούργιο κωδικό πρόσβασης κλείνει το συγκεκριμένο παράθυρο και επιστρέφει στην αρχική οθόνη όπως έχει γίνει και με τα προηγούμενα βήματα. Βέβαια εδώ θα αναφέρουμε ότι όταν η αλλαγή του κωδικού είναι επιτυχής θα εμφανιστεί ένα μήνυμα επιβεβαίωσης στον χρήστη. Σε περίπτωση όμως που ο χρήστης έχει κάνει κάποιο λάθος θα εμφανιστεί το αντίστοιχο μήνυμα λάθους. Πιο συγκεκριμένα υπάρχουν συγκεκριμένα κριτήρια για τον κωδικό όπως:

- Όλοι οι κωδικοί θα αποτελούνται τουλάχιστον από 6 χαρακτήρες
- Ο κωδικός πρόσβασης μπορεί να αποτελείται από αριθμούς , γράμματα ή ακόμη και σύμβολα
- Όπως και στην διαδικασία της εγγραφής , το password και το confirm password θα πρέπει να είναι ακριβώς τα ίδια
- Το current password θα πρέπει να είναι ο παλιός κωδικός πρόσβασης του χρήστη ο οποίος δεν μπορεί να είναι ίδιος με τους δύο καινούργιους κωδικούς πρόσβασης

Επίσης ο χρήστης έχει την δυνατότητα ακύρωσης της διαδικασίας αν πατήσει στο κάτω αριστερό μέρος της φόρμας το κουμπί CLOSE. Στην περίπτωση αυτή και αφού ο χρήστης θα ξανά μπει στην φόρμα να αλλάξει τον κωδικό του , η φόρμα δεν θα έχει κρατήσει τα στοιχεία που υπήρχαν πριν λίγο. Το ίδιο θα συμβεί όταν ο χρήστης θα αλλάξει τον κωδικό πρόσβασης.

4.8 Contact form

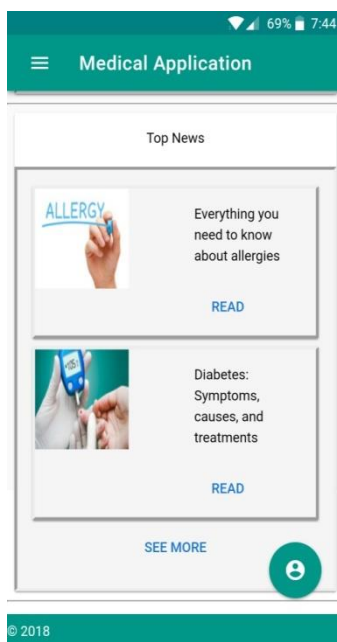
Εικόνα 4.8.1



Στην εικόνα 4.8.1 βλέπουμε την φόρμα επικοινωνίας όπου ο χρήστης το μόνο που έχει να κάνει για να επικοινωνήσει με τον διαχειριστή είναι να γράψει το θέμα του προβλήματος, το πρόβλημα που αντιμετωπίζει σχετικά με την εφαρμογή και στην συνέχεια να πατήσει το κουμπί SEND ώστε να αποσταλεί με email από την εφαρμογή. Στον διαχειριστή αυτό που θα φανεί είναι το email του χρήστη από τον οποίο ήρθε και στην συνέχεια όλα αυτά που θα του έχει γράψει ο χρήστης. Επίσης και εδώ που σε περίπτωση ο χρήστης ακυρώσει την διαδικασία ή την ολοκληρώσει αυτά τα οποία έχει γράψει ο χρήστης θα διαγραφούν.

4.9 Top news

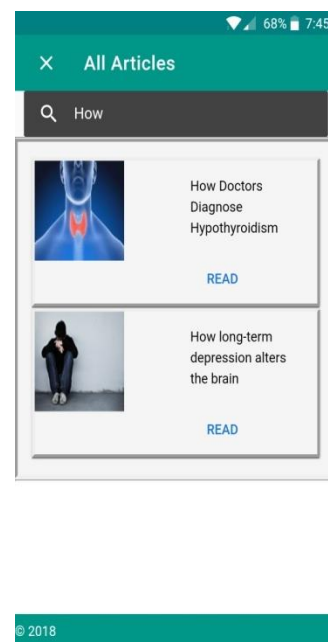
Εικόνα 4.9.1



Εικόνα 4.9.2



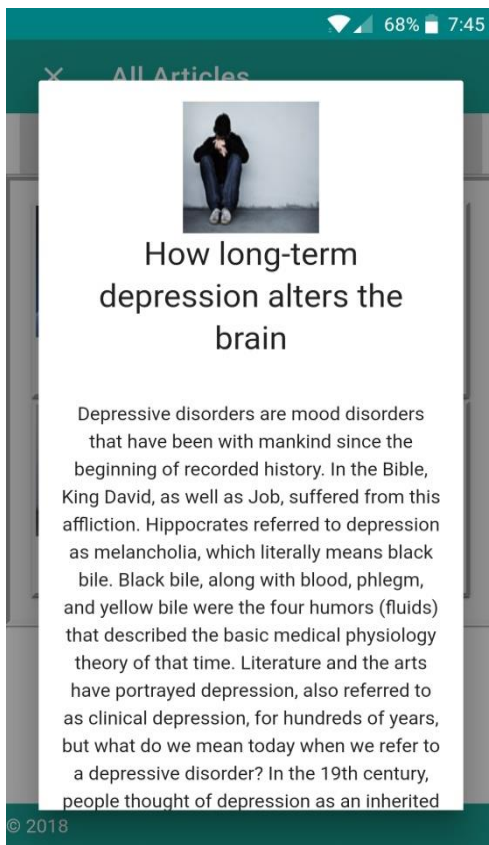
Εικόνα 4.9.3



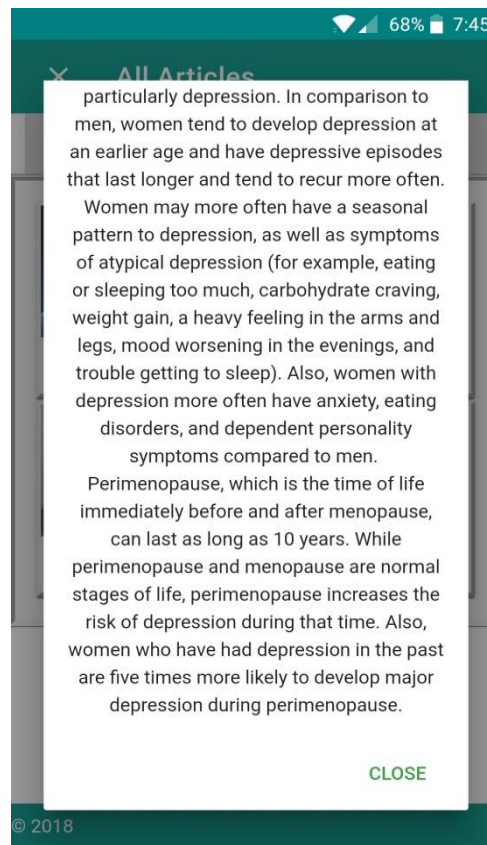
Στην εικόνα 4.9.1 βλέπου τα Top news τα οποία υπάρχουν στην αρχική οθόνη της εφαρμογής και είναι από τα πρώτα πράγματα που μπορεί να δει ο χρήστης. Τα Top news αποτελούνται από δύο άρθρα τα οποία τα ορίζει ο διαχειριστής της εφαρμογής όπου πατώντας το κουμπί READ μπορεί να διαβάσει το άρθρο που έχει επιλέξει ή πατώντας το κουμπί SEE MORE ανοίγει ένα άλλο παράθυρο με όλα τα άρθρα που είναι αποθηκευμένα στην βάση. Αυτό το βλέπουμε στην εικόνα 4.9.2 όπου το κάθε άρθρο αποτελείται από μια ξεχωριστή καρτέλα με το κουμπί που πατάει ο χρήστης για να το διαβάσει, μια μπάρα στην πάνω πλευρά της οθόνης και το κουμπί εξόδου.

Στην εικόνα 4.9.4 βλέπουμε την αναζήτηση που μπορεί να κάνει ο χρήστης η οποία γίνεται με δυναμικό τρόπο. Δηλαδή δεν χρειάζεται να πατήσουμε κάποιο κουμπί για να μας βγάλει τα αποτελέσματα.

Εικόνα 4.9.4



Εικόνα 4.9.5

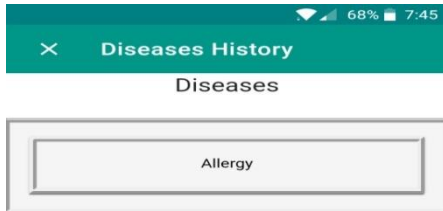


Στην εικόνα 4.9.4 βλέπουμε πως φαίνεται ένα άρθρο όταν ο χρήστης θα πατήσει το READ για να το διαβάσει. Εμφανίζεται η εικόνα του άρθρου στην πάνω μεριά μετά ο τίτλος του άρθρου και στην συνέχεια το κυρίως σώμα του άρθρου. Εδώ αυτό που μπορεί να κάνει ο χρήστης είναι να διαβάσει το άρθρο και στο τέλος υπάρχει ένα κουμπί CLOSE όπως φαίνετε στην εικόνα 4.9.5 το οποίο επιστρέφει στην οθόνη με τα άρθρα.

Την ίδια μορφή έχουν όλα τα άρθρα της εφαρμογής τα οποία μπορεί να διαβάσει ο χρήστης είτε είναι γενικής ιατρικής φύσεως όπως βλέπουμε παραπάνω, είτε είναι προτεινόμενα άρθρα που θα δούμε στην επόμενη υποενότητα.

4.10 Προτεινόμενα Άρθρα

Εικόνα 4.10.1



Εικόνα 4.10.2



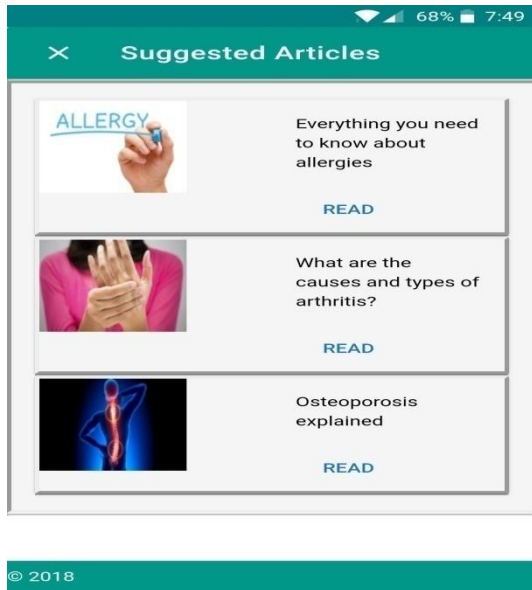
© 2018

© 2018

Σε αυτήν την υποενότητα θα δούμε τα προτεινόμενα άρθρα τα οποία γίνονται με αυτοματοποιημένο τρόπο στον κάθε χρήστη και έχουν να κάνουν με τις παθήσεις που έχει δηλώσει ο ίδιος και όχι με την κληρονομικότητα που έχει.

Για να υπάρχουν προτεινόμενα άρθρα θα πρέπει ο χρήστης να έχει δηλώσει τουλάχιστον από μία πάθηση, όπως βλέπουμε και στην εικόνα 4.10.1. Ο χρήστης έχει δηλώσει την πάθηση Allergy και κατά συνέπεια στην εικόνα 4.10.2 βλέπουμε ότι στα προτεινόμενα άρθρα υπάρχει ένα σχετικό άρθρο το οποίο αναφέρεται στην πάθηση που έχει ο χρήστης. Στην συγκεκριμένη περίπτωση το άρθρο το οποίο πρότεινε η εφαρμογή είναι μόνο ένα, όμως όπως θα αναλύσουμε και στο παρακάτω κεφάλαιο αυτό δεν είναι και ο κανόνας. Υπάρχουν άρθρα στην βάση τα οποία η εφαρμογή μπορεί να τα προτείνει σε κάποιον χρήστη και να αναφέρονται σε πάνω από μία πάθηση όπως βλέπουμε στην εικόνα 4.10.3. Εδώ ο χρήστης έχει δηλώσει δύο παθήσεις και τα προτεινόμενα άρθρα είναι τρία.

Εικόνα 4.10.3



4.11 Σφάλματα από την μεριά του χρήστη

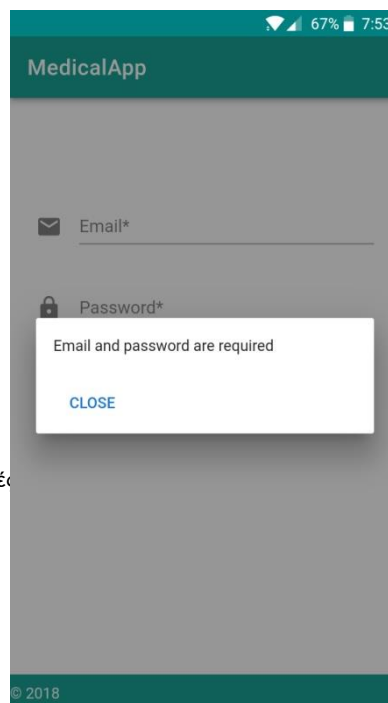
Σε αυτό το κεφάλαιο θα δούμε κάποια από τα λάθη που μπορεί να κάνει ο χρήστης και τα μηνύματα λάθους ή τα βοηθητικά μηνύματα που εμφανίζονται ανάλογα με την περίπτωση και το λάθος. Ας δούμε πρώτα κάποιες από τις περιπτώσεις αυτές:

- Λάθος κωδικός ή email στο login
- Κενά και τα δύο πεδία ή ένα από τα δύο πεδία στο login
- Κενά όλα ή ένα από τα πεδία κατά την εγγραφή του χρήστη
- Εγγραφή με email που είδη υπάρχει
- Λάθος password confirmation στην διαδικασία εγγραφής
- Σε περίπτωση που ο χρήστης θα βάλει κάποιον αριθμό στο πεδίο που του ζητάει το όνομά του
- Ο κωδικός θα είναι μικρότερος από έξι χαρακτήρες
- Το ΑΜΚΑ θα είναι μικρότερο ή μεγαλύτερο από έντεκα χαρακτήρες
- Κάτω από έξι χαρακτήρες κατά την αλλαγή κωδικού
- Λάθος τρέχων κωδικός κατά την αλλαγή κωδικού
- Οι δύο καινούργιοι κωδικοί οι οποίοι έχει δώσει ο χρήστης κατά την αλλαγή κωδικού δεν είναι ίδιοι
- Λάθος ΑΜΚΑ στην δήλωση συγγένειας. Εδώ το μήνυμα μπορεί να είναι και βοηθητικό δείχνοντας στον χρήστη το λάθος που έκανε

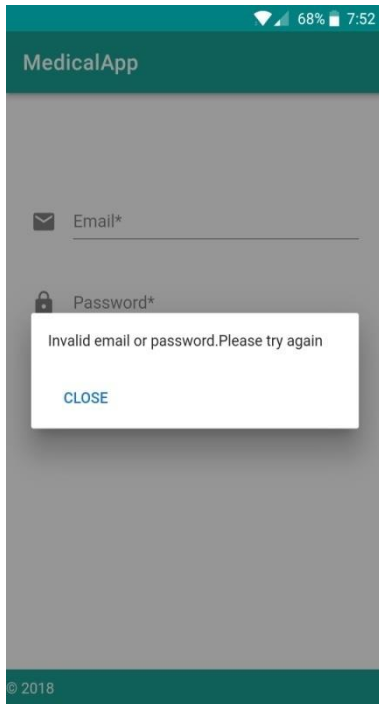
Εικόνα 4.11.1

Εικόνα 4.11.2

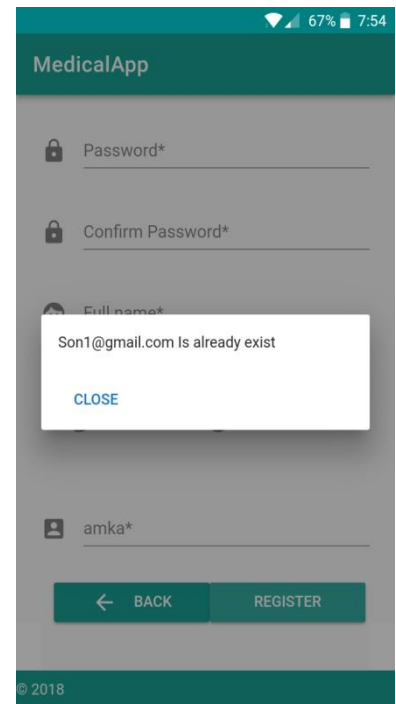
Εικόνα 4.11.3



Μεταπτυχιακή Διατριβή



Κιτσούλης Ευάγγελος

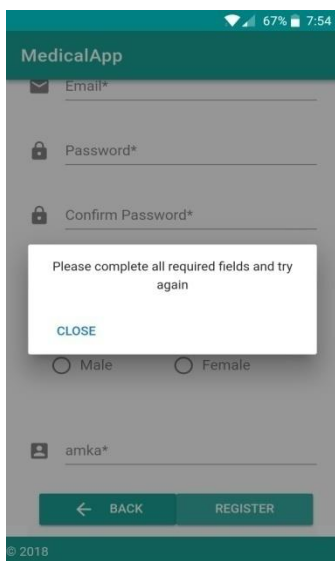


Εικόνα 4.11.1 : Ο χρήστης έχει δώσει λάθος κωδικό πρόσβασης ή email

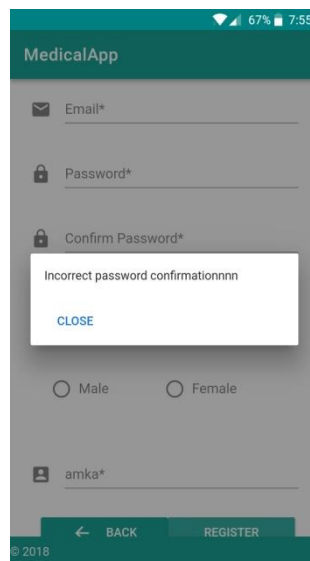
Εικόνα 4.11.2 : Ο χρήστης δε έχει δώσει κάποιο από τα δύο υποχρεωτικά πεδία για να κάνει σύνδεση στην εφαρμογή

Εικόνα 4.11.3 : Ο χρήστης προσπαθεί να κάνει εγγραφή στο σύστημα με email το οποίο υπάρχει ήδη στην βάση

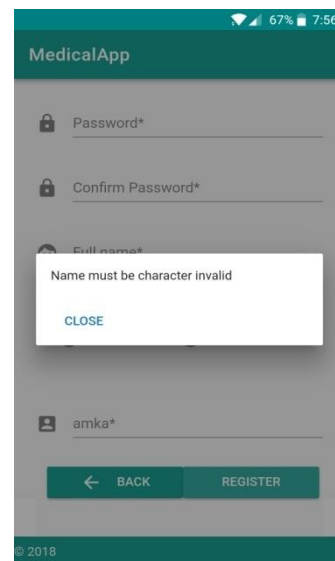
Εικόνα 4.11.4



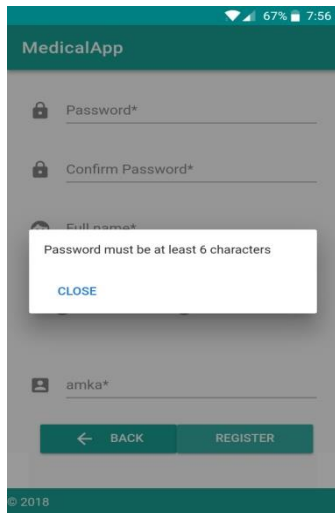
Εικόνα 4.11.5



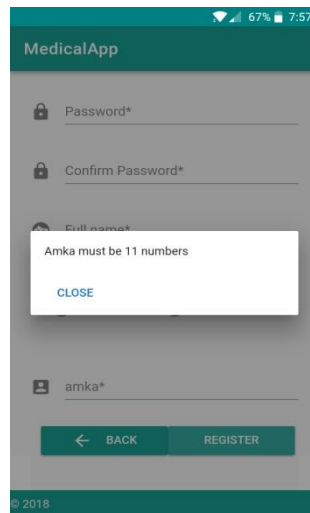
Εικόνα 4.11.6



Εικόνα 4.11.7



Εικόνα 4.11.8



Εικόνα 4.11.4 : Ο χρήστης δεν έχει συμπληρώσει κάποιο από τα πεδία της φόρμας

Εικόνα 4.11.5 : Οι δύο κωδικοί που έχει συμπληρώσει είναι διαφορετικοί

Εικόνα 4.11.6 : Στο πεδίο όνομα έχει συμπληρώσει και κάποιον αριθμό

Εικόνα 4.11.7 Ο κωδικός είναι μικρότερος από έξι χαρακτήρες

Εικόνα 4.11.8 : Το ΑΜΚΑ που έχει συμπληρώσει ο χρήστης δεν είναι ακριβώς έντεκα αριθμοί

Πιο πάνω αναφέραμε μερικές περιπτώσεις που μπορεί να κάνει λάθος ο χρήστης. Παρόμοια μηνύματα λάθους ή βοηθητικά μηνύματα λάθους υπάρχουν και όταν ο χρήστης είναι συνδεδεμένος στην εφαρμογή. Αυτά έχουν να κάνουν με την αλλαγή κωδικού πρόσβασης καθώς και με την δήλωση συγγένειας.

Στην επόμενη ενότητα θα δούμε κάποια σφάλματα επικοινωνίας που μπορούν να προκύψουν από την εφαρμογή και τους servers.

4.12 Σφάλματα από την μεριά των servers

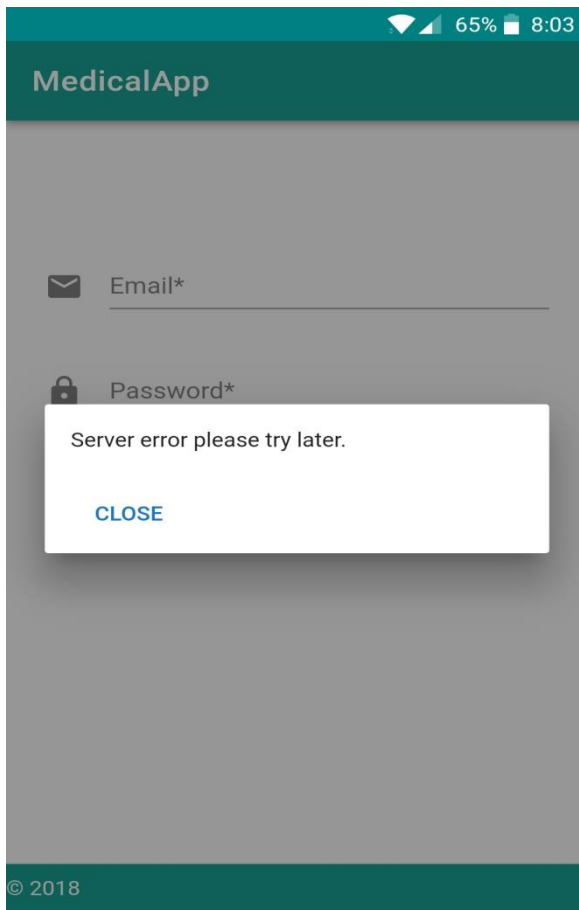
Εδώ μπορούμε να αναφέρουμε δύο περιπτώσεις σε γενικές γραμμές όπου η εφαρμογή να μην μπορεί να επικοινωνήσει με τον server όπως:

- Δεν επικοινωνεί με το Rails
- Δεν επικοινωνεί με το Node.js

Στην πρώτη περίπτωση όπου η εφαρμογή δεν θα μπορέσει να ανταλλάξει δεδομένα με το Rails και κατά συνέπεια δεν θα μπορεί να πάρει ούτε να δώσει πληροφορίες στην βάση δεδομένων, θα βγάλει ένα μήνυμα που θα ενημερώνει τον χρήστη ότι υπάρχει κάποιο πρόβλημα με τον server και ότι θα πρέπει να ξανά δοκιμάσει αργότερα.

Εδώ θα πρέπει να πούμε ότι η εφαρμογή του χρήστη θα φαίνεται κανονικά στο κινητό του και αυτό εφόσον ο Node server λειτουργεί κανονικά. Στην δεύτερη περίπτωση ο χρήστης δεν θα μπορεί να έχει πρόσβαση στην εφαρμογή εφόσον η επικοινωνία έχει χαθεί με το Node server. Αυτό γιατί η εφαρμογή σηκώνεται ουσιαστικά από τον συγκεκριμένο server. Έτσι σε αυτήν την περίπτωση θα εμφανίζεται ένα εικονίδιο το οποίο θα δείχνει στον χρήστη ότι ο server είναι υπό κατασκευή για κάποιον λόγο και κατά συνέπεια δεν λειτουργεί.

Εικόνα 4.12.1



Εικόνα 4.12.2



Στην εικόνα 4.12.1 βλέπουμε την πρώτη περίπτωση που αναφέραμε πιο πάνω ενώ στην εικόνα 4.12.2 βλέπουμε την δεύτερη. Φυσικά αυτό μπορεί να συμβεί και όταν ο χρήστης είναι είδη μέσα στην εφαρμογή όπου θα εμφανιστούν τα ίδια μηνύματα.

ΚΕΦΑΛΑΙΟ 5. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ

5.1 Οι τεχνολογίες που χρησιμοποιήθηκαν για το Web κομμάτι της εφαρμογής

Για την δημιουργία της εφαρμογής στο web κομμάτι και για την ανάπτυξη των απαιτήσεων που έχουμε αναφέρει στην προηγούμενη ενότητα θα χρησιμοποιήσουμε JavaScript framework[8] που χρησιμοποιείται για την ανάπτυξη σελίδων στο internet αλλά και για την ανάπτυξη εφαρμογών web όπως είναι και η δική μας. Στην συγκεκριμένη περίπτωση επιλέξαμε να χρησιμοποιήσουμε το Vue.js για την ανάπτυξη της εφαρμογής. Φυσικά υπάρχουν και άλλα JavaScript framework όπως είναι το React[9] ή το Angular[10]. Φυσικά σε όλο αυτό το πακέτο όπως θα δούμε και πιο κάτω έχουμε ασχοληθεί με JavaScript[11], HTML[12], CSS[13] και πιο συγκεκριμένα για την ανάπτυξη τους styling της εφαρμογής, χρησιμοποιήσαμε το Vuetify[14] framework. Εδώ θα πρέπει να αναφέρουμε ότι ο λόγος που χρησιμοποιήσαμε αυτά τα εργαλεία για την ανάπτυξη της εφαρμογής είναι για να γίνει responsive και φιλικό περιβάλλον για τον χρήστη. Για την υποστήριξη του Vue.js χρησιμοποιήσαμε το Node.js [15] webserver. Επίσης χρησιμοποιήθηκε και το Android ώστε η εφαρμογή να μπορεί να τρέξει σε κινητή συσκευή Android που και αυτός ήταν και ο σκοπός μας απ την αρχή.

Όλα αυτά που αναφέραμε πιο πάνω είναι για την ανάπτυξη της εφαρμογής στο frontend κομμάτι. Στο Backend κομμάτι χρησιμοποιήσαμε την γλώσσα Ruby[16] καθώς και το Ruby on Rails[17] webserver. Η βάση δεδομένων της εφαρμογής είναι φτιαγμένη σε MySQL[18] η οποία υποστηρίζεται από τον server όπου χρησιμοποιούμε. Πιο κάτω θα αναλύσουμε όσα έχουμε αναφέρει μέχρι τώρα και θα δούμε τους λόγους που επιλέξαμε τις συγκεκριμένες τεχνολογίες.

5.1.1 Vue.js

Εικόνα 5.1.1



Το Vue.js όπως αναφέραμε και πιο πάνω είναι ένα JavaScript framework το οποίο θα μας βοηθήσει να αναπτύξουμε την web εφαρμογή. Πιο συγκεκριμένα το συγκεκριμένο framework είναι open source και χρησιμοποιεί JavaScript, HTML και CSS για την ανάπτυξη πιο φιλικών και responsive user interfaces. Το Vue.js είναι συμβατό με όλες τις βιβλιοθήκες JavaScript που χρησιμοποιούνται αυτήν την στιγμή από άλλα αντίστοιχα framework που αναφέραμε πιο πάνω. Επίσης θεωρείται το ιδανικό framework για ανάπτυξη εφαρμογής μιας σελίδας (Single-Page - Application) με χρήση σύγχρονων εργαλείων και βιβλιοθηκών JavaScript που είναι συμβατές για τέτοια χρήση. Βέβαια στην δική μας περίπτωση η εφαρμογή αποτελείται από δύο σελίδες όπως είδαμε και πιο πάνω. Απαραίτητο για να το χρησιμοποιήσει κάποιος είναι η καλή γνώση της JavaScript, HTML και CSS όπως και των τεχνολογιών διαδικτύου. Η δομή με την οποία χτίζεται μια εφαρμογή στο Vue.js είναι πολύ απλή και ξεκάθαρη ακόμη και σε κάποιον ο οποίος δεν έχει

ασχοληθεί ποτέ με κάποιο αντίστοιχο framework. Επίσης αυτό που βοηθάει πολύ είναι και το πολύ καλό documentation που υπάρχει διαθέσιμο. Απαραίτητο για να τρέξουμε στον υπολογιστή μας μια εφαρμογή γραμμένη στο Vue.js είναι το node.js server.

5.1.2 JavaScript Framework

Εικόνα 5.1.2



Εικόνα 5.1.2



Σε αυτήν την υποενότητα θα δούμε τα κοινά και τις διαφορές στα χαρακτηριστικά που έχει το Vue.js σε σχέση με τα άλλα δυο πολύ γνωστά JavaScript framework που υπάρχουν αυτήν την στιγμή.

React vs Vue.js

Το React framework έχει αρκετά κοινά με το Vue.js όπως:

- Χρησιμοποιούν και τα δύο ένα εικονικό DOM[19]
- Παρέχουν και τα δύο component για την ανάπτυξη της εφαρμογής
- Είναι και τα δύο εξίσου γρήγορα
- Συμβατότητα των βιβλιοθηκών JavaScript

Υπάρχουν βέβαια και διαφορές όπως :

- Τα δύο προγράμματα χρησιμοποιούν διαφορετικά template για την ανάπτυξη της εφαρμογής και αυτό έχει να κάνει με την δομή του project. Ένα React project αποτελείται κυρίως από JavaScript ενώ ένα Vue project αποτελείται από JavaScript , HTML και CSS στο ίδιο component.
- Στο Vue.js μπορούμε να έχουμε μεγαλύτερη παραμετροποίηση πάνω στο project
- Το Vue διαθέτει δικό του CSS framework για την ανάπτυξη native mobile application

Angular vs Vue.js

Όπως με το React έτσι και με το angular framework υπάρχουν και εδώ κοινά. Στην συγκεκριμένη περίπτωση βέβαια είναι δικαιολογημένο διότι οι άνθρωποι που ανέπτυξαν το Vue είχαν πολύ μεγάλη εμπειρία πάνω στο Angular. Παρακάτω θα δούμε μερικά κοινά και διαφορές:

- Και τα δύο μοιάζουν πολύ στο συντακτικό κομμάτι
- Το Vue είναι πιο απλό όσο από πλευράς API[20] αλλά και σχεδιασμό
- Το Vue χρησιμοποιεί έναν τρόπο αποθήκευσης δεδομένων μεταξύ component
- Τα δύο framework έχουν διαφορετική λογική και λειτουργία στον τρόπο που επιδρούν πάνω στο DOM και κατά συνέπεια στα Directives και Components

Όπως προκύπτει από τα παραπάνω και τα υπόλοιπα framework είναι εξίσου αξιόλογα είτε πρόκειται από άποψη λογικής είτε από άποψη λειτουργίας. Το σημαντικό όμως είναι ότι το Vue.js ήρθε να διορθώσει όλα τα προβλήματα και τις ασυμβατότητες που υπήρχαν μέχρι τώρα στα δύο προηγούμενα framework. Επίσης ένα ακόμη πολύ σημαντικό που μας έκανε να το επιλέξουμε και να αναπτύξουμε την Web εφαρμογή σε αυτό, είναι η απλότητα που παρέχει, στην λογική του αλλά και στην χρήση του.

5.1.3 Η γλώσσα μορφοποίησης HTML

Η HTML (HyperText Markup Language) είναι μια γλώσσα μορφοποίησης η οποία χρησιμοποιείται για την ανάπτυξη ιστοσελίδων αλλά και σε ανάπτυξη web applications τα τελευταία χρόνια. Στην απλή της μορφή, αποτελείται από ένα αρχείο που είναι γραμμένη η HTML όπου μπορεί να το ανοίξεις με οποιοδήποτε πρόγραμμα περιήγησης. Η HTML βασίζεται πάνω στην γλώσσα SGML[21] (Standard Generalized Markup Language) όπου αποτελεί ένα μεγαλύτερο σύστημα επεξεργασίας εγγράφων από αυτό της HTML. Τέλος με την γλώσσα μορφοποίησης μπορείς να ορίζεις που θα στηθεί το κείμενο μέσα στην σελίδα οι εικόνες, να καθορίζεις συνδέσμους προς άλλες σελίδες και γενικά να φτιάξεις την δομή μιας σελίδας.

Η HTML σαν γλώσσα μορφοποίησης έχει πέντε εκδόσεις όπου η HTML5 είναι και η τελευταία έκδοση. Αυτή η έκδοση σε σχέση με τις υπόλοιπες προσφέρει περισσότερες δυνατότητες και επιλογές για την δημιουργία ιστοσελίδων κυρίως όσο αναφορά τις φόρμες και τα πεδία (email, password, phone). Με την HTML5 η ανάπτυξη μιας εφαρμογής γίνεται πιο εύκολη και αυτό γιατί πλέον ενσωματώνει όλες τις προσθήκες που έχουν πιο παλιές εκδόσεις που θα έπρεπε να χρησιμοποιηθούν plugins.

5.1.4 JavaScript

Η JavaScript είναι μια scripting γλώσσα η οποία είναι υπεύθυνη για την διαδραστικότητα που προσφέρει στις σελίδες που είναι γραμμένες με HTML. Θεωρείται από τις ερμηνευτικές γλώσσες προγραμματισμού και αυτό γιατί δεν περνάει από την διαδικασία της σύνταξης. Επίσης η συγκεκριμένη γλώσσα τρέχει στον περιηγητή και όχι στην μεριά του server τουλάχιστον τις περισσότερες φορές. Αρκετοί νομίζουν ότι η JavaScript και η Java είναι παρόμοιες γλώσσες όμως στην πραγματικότητα δεν ισχύει κάτι τέτοιο αφού είναι τελείως διαφορετικές ακόμη και στην χρήση τους.

Όπως αναφέραμε και πιο πάνω είναι μια scripting γλώσσα με πολλές δυνατότητες. Μπορούμε με αυτή να αλλάξουμε ένα περιεχόμενο ενός HTML στοιχείου ή ακόμα πατώντας κάπου ο χρήστης να μπορεί δυναμικά να λαμβάνει τα αντίστοιχα αποτελέσματα. Επίσης μπορούμε να δημιουργήσουμε cookies[21] για την αποθήκευση των δεδομένων στον υπολογιστή του επισκέπτη.

Ο λόγος που θεωρείται δυναμική γλώσσα είναι τόσο για την σύνταξη που είναι απλή και εύκολη ώστε να την μάθει κανένας αλλά και στον τρόπο της εκτέλεσης της από τον περιηγητή που είναι runtime. Επίσης είναι αντικειμενοστρεφής γλώσσα και κάθε συνάρτηση που φτιάχνεται σε αυτή αποτελεί και ένα αντικείμενο με ιδιότητες και μεθόδους.

Η JavaScript θεωρείται πλέον η πρώτη επιλογή των προγραμματιστών που ασχολούνται με την ανάπτυξη εφαρμογών διαδικτύου και αυτό γιατί υπάρχουν πολλοί λόγοι όπως:

- Η επεξεργασία γίνεται από την πλευρά του πελάτη
- Είναι απλή και εύκολη στην εκμάθησή της
- Δεν απαιτεί ειδικό μεταγλωττιστή παρά μόνο ένας περιηγητής
- Υπάρχουν περιηγητές με ενσωματωμένη JavaScript
- Είναι αρκετά γρήγορη για τον τελικό χρήστη

Φυσικά και στην δική μας περίπτωση η web εφαρμογή που αναπτύξαμε είναι γραμμένη μόνο σε JavaScript αφού και το framework το οποίο χρησιμοποιήθηκε είναι φτιαγμένο για αυτήν την γλώσσα προγραμματισμού. Χρησιμοποιήθηκαν επίσης και πολλές έτοιμες βιβλιοθήκες JavaScript για να πετύχουμε το επιθυμητό αποτέλεσμα. Για παράδειγμα έχουν χρησιμοποιηθεί

τα `vue-resource`[22] για την εκτέλεση `request` με το `backend` όπως επίσης και η χρήση των `vue-charts`[23] που χρησιμοποιήθηκαν για την απεικόνιση διαγραμμάτων.

5.1.5 CSS (Cascading Style Sheets)

Εικόνα 5.1.3



Όπως έχουμε γράψει και στον τίτλο της υποενότητας το CSS σημαίνει Cascading Style Sheets. Είναι αυτό που καθορίζει την εμφάνιση των στοιχείων HTML όλης της σελίδας. Υποστηρίζεται από την HTML4 και μετά και ήρθε για να λύσει διάφορα προβλήματα μορφοποίησης και κυρίως τον κόπο και τον χρόνο που χρειαζόντουσαν οι προγραμματιστές μέχρι τότε. Έτσι όλη η εμφάνιση της σελίδας καθορίζεται από ένα αρχείο πλέον CSS που ενσωματώνει τον απαραίτητο κώδικα μορφοποίησης. Η εμφάνιση είναι πιο εύκολη να αλλάξει αρκεί να πειράξουμε μόνο ένα αρχείο.

Στην δική μας περίπτωση φυσικά και χρειάστηκε να γράψουμε CSS κώδικα για να φτιάξουμε την εμφάνιση της σελίδας και για να ομαδοποιήσουμε στοιχεία τα οποία θα πρέπει να έχουν την ίδια εμφάνιση. Το `framework` που χρησιμοποιήσαμε δίνει την δυνατότητα στον χρήστη να γράψει τον CSS κώδικα στο ίδιο αρχείο με την HTML και την JavaScript, αυτό όμως δεν αποτελεί κανένα πρόβλημα εφόσον το Vue αναπτύχθηκε για κάτι τέτοιο και μάλιστα θεωρείται ευκολία από την μεριά των προγραμματιστών.

5.1.6 Vuetify framework

Εικόνα 5.1.4



Το Vuetify είναι ένα `framework` το οποίο αναφέρεται στα `component`. Πιο συγκεκριμένα έχει να κάνει με την συμπεριφορά των στοιχείων, τα εφέ και την εμφάνισή τους. Μπορεί να χρησιμοποιηθεί είτε σαν `tag` στην HTML είτε σαν βιβλιοθήκη. Στην δική μας περίπτωση το

χρησιμοποιήσαμε σαν βιβλιοθήκη στο Vue και αυτό γιατί υποστηρίζεται από το framework που χρησιμοποιήσαμε. Το συγκεκριμένο εργαλείο μας δίνει το material design[24] και αυτό είναι πολύ σημαντικό όταν μιλάμε για ένα mobile application. Τα εικονίδια τα οποία χρησιμοποιεί είναι τα εικονίδια που χρησιμοποιεί και το android στις εφαρμογές του. Υπάρχουν ακόμη και έτοιμα templates που μπορούν να χρησιμοποιηθούν όπως και να φτιάξει το δικό του με custom στοιχεία εμφάνισης για την HTML. Ένα αντίστοιχο framework είναι και το Bootstrap[25] το οποίο διαθέτει και αυτό material design με τα εικονίδια που χρησιμοποιεί το android. Ο λόγος που σε αυτήν την περίπτωση χρησιμοποιήσαμε το Vuetify είναι ξεκάθαρα γιατί και ο λόγος ύπαρξης του είναι η ανάπτυξη mobile web application με material design.

5.1.7 Webstorm

Εικόνα 5.1.5



Για την υλοποίηση και την ανάπτυξη της εφαρμογής χρησιμοποιήσαμε το Webstorm[26] το οποίο αποτελεί ένα JavaScript IDE με πολλές δυνατότητες και στην πιο συγκεκριμένα στην δική μας περίπτωση θα λέγαμε ότι είναι η καλύτερη επιλογή. Πιο κάτω θα δούμε μερικές δυνατότητες που έχει και χρησιμοποιήσαμε για την ανάπτυξη της εφαρμογής:

- Γρήγορη πλοήγηση μεταξύ φακέλων και αρχείων
- Γρήγορη αναζήτηση συναρτήσεων και κώδικα
- Auto complete για πιο εύκολη και ταχύτερη γραφή
- Διατίθεται σε όλες τις πλατφόρμες αν και εμείς το χρησιμοποιήσαμε σε Linux
- Υπάρχουν αρκετά plugins που υποστηρίζει τόσο για την ανάπτυξη του κώδικα αλλά και για την εμφάνιση του editor στα χρώματα και στον τρόπο γραφής
- Υποστηρίζει το Node.js και αυτό είναι πολύ σημαντικό γιατί δεν χρειάζεται να έχουμε επιπλέον παράθυρα ανοιχτά
- Υποστηρίζει και τα τρία JavaScript framework (Angular , React , Vue)
- Υπάρχει ταυτόχρονη επεξεργασία για τα αρχεία και για τον κώδικα που χρειάζεται αλλαγές ή διόρθωση

- Και μια από τις πιο σημαντικές δυνατότητες είναι το debug το οποίο γίνεται με πολύ εύκολο και αναλυτικό τρόπο για τον προγραμματιστή με ένα πλήρες οδηγό για τα σφάλματα που υπάρχουν είτε αυτά είναι λογικής είτε συντακτικά

5.2 Τεχνολογίες που χρησιμοποιήθηκαν για την εφαρμογή Android

Για την ανάπτυξη της εφαρμογής Android θα χρησιμοποιήσουμε το Android Studio[27] που είναι ένα ολοκληρωμένο προγραμματιστικό περιβάλλον IDE και χρησιμοποιείται για την ανάπτυξη εφαρμογών Android. Η γλώσσα η οποία θα χρησιμοποιήσουμε είναι η Java[28] και XML[29] για την εμφάνιση και την δομή στην οθόνη της κινητής συσκευής. Εδώ θα πρέπει να αναφέρουμε ότι ο κώδικας που γράψαμε στο Android είναι για να προσαρμόσουμε την web εφαρμογή που έχουμε αναλύσει πιο πάνω και να τρέξει στην κινητή συσκευή.

5.2.1 Το Android Studio

Εικόνα 5.2.1



Η πλατφόρμα Android Studio είναι ένα ολοκληρωμένο προγραμματιστικό περιβάλλον IDE το οποίο παρέχει στους προγραμματιστές Android την ανάπτυξη εφαρμογών με λογισμικό ανοιχτού κώδικα. Η συγκεκριμένη πλατφόρμα έχει αναπτυχθεί από την εταιρία JetBrains IntelliJ IDEA[30] και δεν χρειάζεται κάποια άδεια ή αγορά για την χρήση της. Το προγραμματιστικό εργαλείο παρέχει στον χρήστη όλα τα πλεονεκτήματα που αναφέραμε και πιο πάνω για το WebStorm αφού και τα δύο είναι της ίδιας εταιρίας.

Ο κώδικας που έχουμε γράψει δεν αφορά την λειτουργικότητα της εφαρμογής μιας και η εφαρμογή είναι απολύτως έτοιμη για χρήση. Σε αυτήν την περίπτωση αυτό που έχουμε κάνει είναι να προσαρμόσουμε το web application και να το ενσωματώσουμε σε ένα εκτελέσιμο apk για κινητή συσκευή Android. Για να το πετύχουμε αυτό χρησιμοποιήσαμε ένα WebView[31] για να δείξουμε την web εφαρμογή που φτιάξαμε στην οθόνη της συσκευής. Εδώ έχουμε γίνει κάποιες παραμετροποιήσεις όπως ότι έχουμε απενεργοποιήσει το back button της συσκευής γιατί στην συγκεκριμένη περίπτωση δεν μας χρησιμεύει κάπου στην πλοήγηση. Επίσης έχει γραφτεί κώδικας για να επιτρέπουμε στην εφαρμογή να έχει πρόσβαση στον αποθηκευτικό χώρο και να μπορεί ο χρήστης να ανεβάζει τις φωτογραφίες προφίλ.

5.2.2 Java

Την γλώσσα προγραμματισμού Java όπως ήδη έχουμε αναφέρει και πιο πάνω την χρησιμοποιήσαμε για την υλοποίηση της εφαρμογής σε Android. Είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού όπου ο κώδικας που γράφεται μετατρέπεται σε Bytecode και χρησιμοποιεί το JVM (Java Virtual Machine)[32] για να τρέξει πάνω σε αυτό. Αποτελεί από τις πιο γνωστές γλώσσες προγραμματισμού και αυτό γιατί οι προγραμματιστές έχουν την δυνατότητα να γράφουν μόνο μια φορά κώδικα και μέσω του JVM να μπορεί να τρέξει σε όλα τα λειτουργικά. Τα τελευταία χρόνια την χρησιμοποιούμε και στις εφαρμογές που φτιάχνουμε για το Android όπως κάναμε και στην δική μας περίπτωση.

5.2.3 XML

HXML (eXtensible Markup Language) είναι μια γλώσσα σήμανσης η οποία μοιάζει πολύ με την γλώσσα μορφοποίησης HTML που έχουμε αναφέρει πιο πάνω. Αυτή η γλώσσα σήμανσης παρέχει ένα σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση κειμένων στο διαδίκτυο και όχι μόνο αφού στην συγκεκριμένη περίπτωση την χρησιμοποιούμε για το layout της Android εφαρμογής. Έχει σχεδιαστεί για να χρησιμοποιείται στην αναπαράσταση δομών δεδομένων στις υπηρεσίες ιστού, δίνοντας απλότητα και χρηστικότητα στην γραφή. Στην δική μας περίπτωση και γενικά στην ανάπτυξη μιας εφαρμογής Android μας χρησιμεύει για να προσδιορίσουμε με μεγαλύτερη ακρίβεια την θέση και γενικότερα την εμφάνιση των στοιχείων που αποτυπώνονται πάνω στην οθόνη. Εμείς αυτό που είχαμε να κάνουμε είναι να προσαρμόσουμε το Webview να είναι σε όλη την οθόνη και σαφώς όταν αλλάζει ο προσανατολισμός να εμφανίζονται τα στοιχεία όπως πρέπει.

5.3 Τεχνολογίες που χρησιμοποιήθηκαν στο Backend

Οι τεχνολογίες που χρησιμοποιήσαμε για το backend είναι η γλώσσα Ruby, Ruby on Rails και για την βάση δεδομένων χρησιμοποιήσαμε την MySQL. Εδώ θα πρέπει να αναφέρουμε ότι η επικοινωνία μεταξύ του frontend και του backend πραγματοποιείται μέσω API τα οποία είναι φτιαγμένα στην γλώσσα προγραμματισμού Ruby. Τα δεδομένα τα οποία ανταλλάσσουν οι δύο αυτοί server γίνεται μέσω JSON[33] αντικειμένων και αυτό που στέλνει αλλά και αυτό που λαμβάνει η εφαρμογή, και έτσι το διαχειρίζεται με κατάλληλο τρόπο. Επίσης όσα αναφορά την επικοινωνία των δύο server αντιμετωπίσαμε ένα πρόβλημα το οποίο λύθηκε με την χρήση proxy για την ανακατεύθυνση της ip διεύθυνσης.

5.3.1 Ruby

Εικόνα 5.3.1



Η Ruby είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού γενικής χρήσης η οποία έχει χαρακτηριστικά με την γλώσσα Perl[34] και κάποιες επιρροές από την Smalltalk. Σχεδιάστηκε στην Ιαπωνία το 1990 από τον Yukihiro Matz Mastumoto ενώ η πρώτη έκδοση κυκλοφόρησε τον Δεκέμβριο του 1994. Τρέχει σε όλες τις πλατφόρμες και μάλιστα σε μερικές από αυτές είναι ήδη εγκατεστημένη. Είναι αρκετά απλή και ένας ο οποίος ξέρει από προγραμματισμό μπορεί να την μάθει σε σύντομο χρονικό διάστημα. Είναι δωρεάν και δεν χρειάζεται να πληρώσεις άδειες για το λογισμικό που χρησιμοποιείς αρκεί μόνο να εγκαταστήσεις τα απαραίτητα.

5.3.2 Rails

Εικόνα 5.3.2



Το Rails είναι ένα εργαλείο το οποίο δημιουργήθηκε για να υποστηρίξει την γλώσσα προγραμματισμού Ruby. Αρχικά αναπτύχθηκε από τον David Heinemeier Hansson περίπου το 2005. Η ανάπτυξη web εφαρμογών σε Ruby έγινε πιο εύκολη και αυτό γιατί το Rails περιλαμβάνει αρκετές βιβλιοθήκες (gems) που μπορούν να βοηθήσουν του προγραμματιστές. Τα τελευταία χρόνια όλο και περισσότερες εφαρμογές χρησιμοποιούν την γλώσσα Ruby και το Rails όπως είναι οι παρακάτω ιστοσελίδες:

- Github
- Themeforest
- Slideshare
- Scribd
- Shopify

- Twitter

5.3.3 JSON

Το JSON (JavaScript Object Notation) είναι ένα πρότυπο ανταλλαγής δεδομένων αρκετά εύκολο στην ανάγνωση και στην γραφή του. Επίσης είναι αρκετά εύκολο ακόμη και για τι μηχανές να διαβαστεί και να το παράγουν. Είναι τελείως ανεξάρτητο από άλλες γλώσσες προγραμματισμού αν και είναι βασισμένο πάνω σε ένα υποσύνολο της γλώσσα JavaScript και χρησιμοποιεί αρκετές πρακτικές αυτής. Είναι με λίγα λόγια μια ιδανική γλώσσα για μεταφορά δεδομένων. Στην δική μας περίπτωση η web εφαρμογή στέλνει και δέχεται δεδομένα μέσω JSON αντικειμένων. Υπάρχουν δύο δομές για να δημιουργήσουμε ένα JSON:

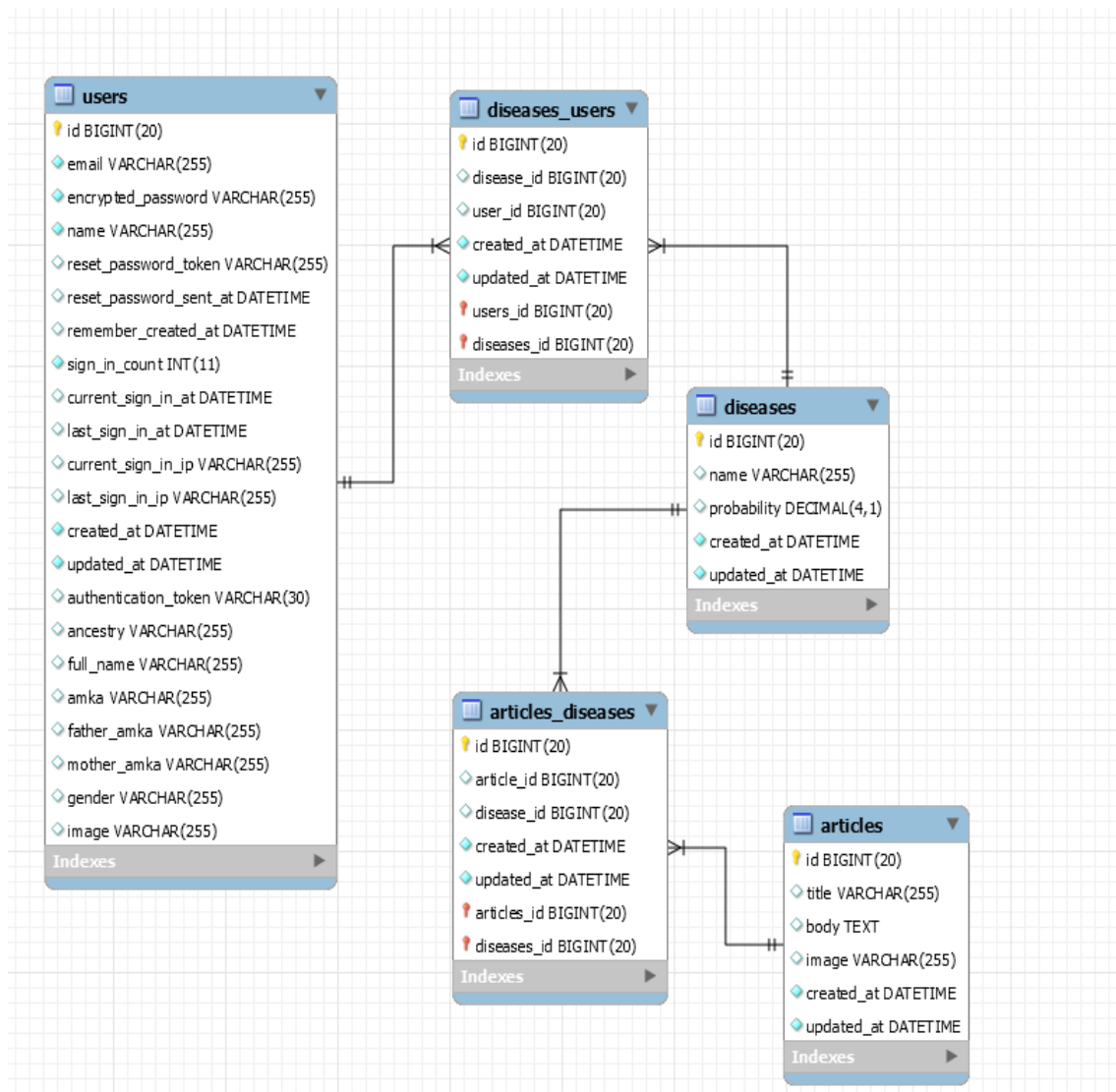
- Συλλογή από ζεύγη key και value. Αυτό είναι για τις περισσότερες γλώσσες προγραμματισμού το JSON Object και στην δική μας περίπτωση η δομή η οποία χρησιμοποιήσαμε.
- Ταξινόμηση λίστας τιμών. Αυτό για τις περισσότερες γλώσσες προγραμματισμού αναγνωρίζεται ως ένας πίνακας με δεδομένα.

5.3.4 MySQL

Η βάση δεδομένων είναι η συλλογή δεδομένων τα οποία αποθηκεύονται με έναν συγκεκριμένο τρόπο ώστε να είναι εύκολη η διαχείριση τους. Για να το πετύχουμε αυτό , χρησιμοποιούμε ένα σύστημα διαχείρισης βάσεων δεδομένων (DataBase Management Systems - DBMS)το οποίο διαθέτει κατάλληλα εργαλεία για την διαχείριση των αποθηκευμένων δεδομένων.

Για την προσπέλαση των δομών δεδομένων μέσω του συστήματος DBMS χρησιμοποιούμε την γλώσσα MySQL η οποία μας επιτρέπει δίνοντας εντολές στο DBMS να έχουμε μια εύκολη διαχείριση στα δεδομένα μας , είτε αυτό πρόκειται για το κτίσιμο της βάσης δεδομένων, είτε για την ανάγνωση, αντιγραφή , διαγραφή , προσθήκη δεδομένων. Εδώ πρέπει να αναφέρουμε ότι μια τέτοια βάση πρόκειται για σχεσιακή βάση δεδομένων όπως είναι και η δική μας.

Η σχεσιακή βάση δεδομένων είναι ο τρόπος ο οποίος αποθηκεύονται τα δεδομένα και οι σχέσεις οι οποίες αποκτούν για την καλύτερη διαχείριση αλλά και για την αποφυγή επανάληψης δεδομένων , το οποίο μας κοστίζει σε χρόνο κατά την προσπέλαση των δεδομένων αλλά και σε χώρο. Η δική μας βάση δεδομένων αποτελείται από πέντε πίνακες και θα την δούμε πιο αναλυτικά στην συνέχεια.



Στην εικόνα 5.3.4 βλέπουμε το σχήμα της βάσης το οποίο χρησιμοποιήσαμε για την εφαρμογή μας. Αποτελείται από πέντε πίνακες:

- Users
- Diseases
- Diseases User
- Articles
- Article Diseases

Users

Σε αυτόν τον πίνακα έχουμε όλες τις απαραίτητες πληροφορίες και δεδομένα που χρειαζόμαστε και συνδέονται άμεσα με τον χρήστη όπως :

- Id. Το id είναι μοναδικό για κάθε χρήστη και είναι αυτό που θα μας βοηθήσει να συνδέσουμε τους δύο πίνακες μεταξύ τους, users και diseases users
- Email. Εδώ να αποθηκεύεται το email του χρήστη το οποίο θα είναι και αυτό μοναδικό και απαραίτητο κατά την εγγραφή του στο σύστημα
- Encrypted password. Αυτός θα είναι ο κωδικός που θα βάζει ο χρήστης κατά την είσοδο στην εφαρμογή. Είναι απαραίτητος και μπορεί να αποτελείται από γράμματα , αριθμούς ή σύμβολα αρκεί να μην είναι μικρότερος από έξι χαρακτήρες.
- Authentication token. Αυτό αποτελεί ένα μοναδικό χαρακτηριστικό για κάθε χρήστη το οποίο δημιουργείται αυτόματα κατά την σύνδεση του χρήστη στην εφαρμογή και διαγράφεται αυτόματα κατά την έξοδο του. Με αυτό το χαρακτηριστικό ο server μπορεί να γνωρίζει αν τα στοιχεία του χρήστη είναι έγκυρα. Αν είναι έγκυρα τότε ο server δημιουργεί το authentication token το οποίο η εφαρμογή το κρατάει και σε περίπτωση που ο χρήστης κάνει κάποιο request στην βάση (όπως για παράδειγμα να προσθέσει κάποια πάθηση) γίνεται ταυτοποίηση με το email του και το authentication token. Όταν ο χρήστης πραγματοποιήσει έξοδο από την εφαρμογή (το οποίο είναι και αυτό ένα request στην βάση) τότε ο server το διαγράφει. Επίσης η ίδια διαδικασία που περιγράφηκε πιο πάνω πραγματοποιείται και σε περίπτωση που ο χρήστης κάνει εγγραφή στο σύστημα.
- Ancestry. Εδώ αποθηκεύετε η κληρονομικότητα που μπορεί να έχει κάποιος χρήστης σε σχέση με κάποιον άλλον χρήστη. Η κληρονομικότητα προκύπτει από το ΑΜΚΑ του κάθε χρήστη , στην βάση όμως αυτό που αποθηκεύετε είναι το id των άλλων χρηστών με τους οποίους υπάρχει σχέση. Δεν αποτελεί στοιχείο το οποίο ο χρήστης πρέπει να δώσει υποχρεωτικά σε αντίθεση με αυτά που αναφέραμε πιο πάνω , όταν όμως δηλωθεί , ο χρήστης θα μπορεί να βλέπει τις πιθανότητες που έχει να αποκτήσει μια πάθηση σε σχέση με την κληρονομικότητα.
- Full name. Υπάρχει ένα πεδίο κατά την εγγραφή στο οποίο ο χρήστης θα πρέπει να δώσει το όνομα και το επίθετό του. Αυτό το πεδίο είναι υποχρεωτικό για κάθε χρήστη και δεν μπορεί να περιέχει αριθμούς ή σύμβολα.
- ΑΜΚΑ. Στο ίδιο πεδίο που αναφέραμε πιο πάνω όπου ο χρήστης θα κάνει την εγγραφή στο σύστημα θα υπάρχει και ένα πεδίο που θα δηλώνει το ΑΜΚΑ του. Αποτελεί και αυτό υποχρεωτικό πεδίο για κάθε χρήστη αποτελούμενο μόνο από έντεκα αριθμούς. Επίσης είναι μοναδικός αριθμός και δεν μπορούν δύο χρήστες να έχουν δηλώσει το ίδιο ΑΜΚΑ.
- Father amka. Εδώ θα αποθηκεύετε το ΑΜΚΑ του πατέρα ώστε να προκύψει η συγγένεια. Αυτό δηλώνεται μία φορά και δεν μπορεί ο χρήστης να το ξανά αλλάξει
- Mother amka. Εδώ θα αποθηκεύεται το ΑΜΚΑ της μητέρας για να προκύψει η συγγένεια που αναφέραμε και πιο πάνω. Τα δύο συγγενικά ΑΜΚΑ δεν μπορούν να είναι ίδια. Επίσης θα πρέπει και αυτά να αποτελούνται μόνο από έντεκα αριθμούς.
- Gender. Εδώ θα αποθηκεύετε το φύλο του χρήστη όπου θα έχει δύο επιλογές κατά την εγγραφή male ή female.
- Image. Εδώ θα αποθηκεύετε η φωτογραφία προφίλ του κάθε χρήστη όπου δεν ανήκει στα υποχρεωτικά πεδία και θα είναι σε μορφή url

Diseases

Σε αυτόν τον πίνακα θα αποθηκεύουμε τις πληροφορίες για τις παθήσεις που θα υπάρχουν αποθηκευμένες στην βάση. Ο χρήστης δεν μπορεί να διαγράψει ή να προσθέσει κάτι σε αυτόν τον πίνακα. Τα πεδία του πίνακα είναι τα εξής:

- Id. Το id είναι μοναδικό για κάθε πάθηση και απαραίτητο στοιχείο όπου μας βοηθά να έχουμε δημιουργήσει σχέσεις με άλλους πίνακες που θα αναλύσουμε πιο κάτω
- Name. Εδώ θα αποθηκεύεται το όνομα της κάθε πάθησης το οποίο θα είναι μοναδικό αφού δεν υπάρχει και λόγος να υπάρχει άσκοπα δύο φορές μέσα στην βάση.

- Probability. Σε αυτό το πεδίο θα αποθηκεύετε η πιθανότητα που θα έχει η πάθηση σε σχέση με την κληρονομικότητα που θα δηλώνει ο χρήστης

Diseases Users

Αυτός ο πίνακας έχει προκύψει από την σχέση που έχουν οι πίνακες users και diseases. Αυτοί οι δύο πίνακες έχουν σχέση πολλά προς πολλά. Δηλαδή ένας χρήστης μπορεί να έχει πολλές παθήσεις όπως και μία πάθηση μπορεί να την έχουν πολλοί χρήστες. Οπότε σε αυτόν τον πίνακα έχουμε αποθηκευμένα δύο ξένα κλειδιά που αναφέρονται στους άλλους δύο πίνακες.

Articles

Σε αυτόν τον πίνακα θα αποθηκεύουμε όλες τις πληροφορίες από τα άρθρα που θα είναι αποθηκευμένα στην βάση , οι οποίες είναι οι εξής:

- Id. Το id θα είναι μοναδικό χαρακτηριστικό για κάθε άρθρο όπως είδη έχουμε αναφέρει και πιο πάνω.
- Title. Εδώ θα αποθηκεύσουμε τον τίτλο του κάθε άρθρου ο οποίος θα είναι και αυτός μοναδικός αφού δεν υπάρχει λόγος να υπάρχουν δύο άρθρα με τον ίδιο τίτλο.
- Body. Σε αυτό το χαρακτηριστικό του πίνακα θα είναι αποθηκευμένο το κυρίως σώμα του άρθρου.

Articles Diseases

Αυτός ο πίνακας έχει προκύψει από την σχέση που έχουν οι πίνακες articles και diseases. Αυτοί οι δύο πίνακες έχουν σχέση πολλά προς πολλά. Ένα άρθρο μπορεί να αναφέρετε σε πάνω από μια πάθηση όπως και μία πάθηση μπορεί να αναφέρεται σε πάνω από ένα άρθρο. Οπότε σε αυτόν τον πίνακα έχουμε αποθηκευμένα δύο ξένα κλειδιά που αναφέρονται στους άλλους δύο πίνακες. Μέσα από την σχέση που έχουν οι δύο πίνακες μπορούμε να εμφανίζουμε τα προτεινόμενα άρθρα για κάθε χρήστη που προκύπτουν από τις παθήσεις που έχει δηλώσει ο ίδιος ο χρήστης

Όλα τα υπόλοιπα τα οποία δεν τα έχουμε αναφέρει είναι πληροφορίες σχετικά με την κίνηση και τον λογαριασμό του κάθε χρήστη. Αναφέρονται δηλαδή στο πότε πραγματοποιήσε σύνδεση στην εφαρμογή , πόσες φορές έχει συνδεθεί, πότε δημιουργήθηκε ο λογαριασμός του κτλ. Όλα αυτά προκύπτουν από μια βιβλιοθήκη (gems) η οποία μαζί με την βάση την οποία θέλουμε να φτιάξουμε, μας δημιούργησε και όλα αυτά τα πεδία.

ΚΕΦΑΛΑΙΟ 6. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η web εφαρμογή που αναπτύξαμε και αναλύσαμε πιο πάνω με βάση τους στόχους και τις απαιτήσεις, μας βοήθησε να γνωρίσουμε καλύτερα ένα καινούργιο framework, το Vue.js. Μέσα από αυτό είδαμε πόσο φιλικό και απαραίτητο μπορεί να είναι ένα περιβάλλον προγραμματισμού για την ανάπτυξη μιας web εφαρμογής σαν την δική μας. Επίσης πόσο εύκολο είναι για έναν προγραμματιστή που δημιουργεί μια web εφαρμογή να μπορεί να την κάνει συμβατή και σε άλλες πλατφόρμες γράφοντας μόνο μια φορά κώδικα. Το θέμα της συμβατότητας είναι πολύ σημαντικό στις μέρες μας αφού με κάτι τέτοιο μια εταιρία μπορεί να εξοικονόμησε χρόνο και χρήμα. Είδαμε επίσης πως με τα κατάλληλα εργαλεία οι εφαρμογές που μπορούμε να δημιουργήσουμε είναι εξίσου αντάξιες όχι μόνο στην εμφάνιση αλλά και στην λειτουργικότητα, με τις εφαρμογές που είναι φτιαγμένες σε Android.

Υπάρχουν αρκετές προσθήκες που θα μπορούσαν να υλοποιηθούν ώστε να πετύχουμε καλύτερη εμπειρία χρήσης για τον χρήστη. Θα μπορούσε ο χρήστης να αποθηκεύει την θεραπεία του και να λαμβάνει ειδοποίηση από την εφαρμογή για την ώρα που θα πρέπει να την λάβει. Όπως και να αποθηκεύει τα ραντεβού με του γιατρούς που πρόκειται να επισκεφτεί ή να υπάρχει μια φόρμα επικοινωνίας όπου ο χρήστης θα μπορεί να επικοινωνεί με τον γιατρό. Κάτι πολύ χρήσιμο ακόμη θα ήταν και η αποθήκευση ιατρικών εξετάσεων όπως για παράδειγμα μια ακτινογραφία. Είναι δυνατότητες που υπάρχουν είδη σε ανάλογες εφαρμογές και θα μπορούσαν να μπουν πολύ εύκολα και στην δική μας web εφαρμογή.

Όσο αναφορά τα άρθρα ο χρήστης θα μπορούσε να μοιράζεται κάποια άρθρα με άλλους χρήστες και σε περίπτωση που οι χρήστες δεν είναι συνδεδεμένοι στην εφαρμογή να έρχεται ειδοποίηση για κάτι τέτοιο. Επίσης μια έξτρα λειτουργία η οποία θα βοηθούσε πολύ στην ανάγνωση των άρθρων, είναι να υπάρχει η επιλογή των αγαπημένων, όπου ο χρήστης θα μπορεί να αποθηκεύει άρθρα στα αγαπημένα και να τα διαβάζει όποτε θέλει.

Οι επεκτάσεις και προσθήκες που αναφέραμε πιο πάνω μπορούν να γίνουν πάνω στην υπάρχουσα εργασία με τις κατάλληλες παραμετροποιήσεις στον κώδικα. Είναι πολύ σημαντικό να αναφέρουμε ότι σε τέτοιες εφαρμογές που αναφέρονται σε ευαίσθητα προσωπικά δεδομένα όπως είναι οι παθήσεις που μπορεί να έχει κάποιος ή, η κληρονομικότητα αλλά και οποιαδήποτε από τις βελτιώσεις που προτείναμε πιο πάνω, θα πρέπει να έχει σαν πρωταρχικό στόχο της ασφάλεια αυτών των προσωπικών δεδομένων και κατά συνέπεια την κωδικοποίηση αυτών.

ΑΝΑΦΟΡΕΣ

1. Αρχική σελίδα του μαθήματος Ιατρική Πληροφορική
<https://gunet2.cs.unipi.gr/courses/TME140/>
2. Medical Records Android Application
https://play.google.com/store/apps/details?id=vladimir.yerokhin.medicalrecord&hl=en_US
3. My medical history Android Application https://download.cnet.com/My-Medical-History-E-HEALTH/3000-2129_4-77507960.html
4. Bootstrap <https://getbootstrap.com/>
5. Vue Resource , NPM <https://www.npmjs.com/package/vue-resource>
6. Vue Charts <https://vue-chartjs.org/#/home>
7. JetBrains IntelliJ IDEA <https://www.jetbrains.com/>
8. Perl <https://www.perl.org/>

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Ελληνική Εταιρία Διαδικτύου <https://www.eled.gr/>
2. Android official page <https://www.android.com/>
3. Vue.js Introduction and Documentation <https://vuejs.org/v2/guide/index.html>
4. The complete Android Splash screen guide <https://android.jelise.eu/the-complete-android-splash-screen-guide-c7db82bce565>
5. The Ultimate Guide to JavaScript Framework <https://javascriptreport.com/the-ultimate-guide-to-javascript-frameworks/>
6. React JavaScript framework <https://reactjs.org/>
7. Angular JavaScript framework <https://angular.io/>
8. JavaScript programming language <https://www.javascript.com/>
9. HTML5 Introduction , W3schools https://www.w3schools.com/html/html5_intro.asp
10. CSS Introduction , W3school https://www.w3schools.com/css/css_intro.asp
11. Vuetify Material Design Component Framework <https://vuetifyjs.com/en/>
12. Node.js web server About <https://nodejs.org/en/about/>
13. Ruby programming language <https://www.ruby-lang.org/en/>
14. Ruby on Rails Guides <http://guides.rubyonrails.org/>
15. MySQL official documentation <https://dev.mysql.com/doc/>
16. What is the DOM , Chris Coyier <https://css-tricks.com/dom/>
17. What is the Application Programming Interface https://www.ip.gr/el/dictionary/378-API_Application_Programming_Interface
18. A brief SGML tutorial <https://www.w3.org/TR/WD-html40-970708/intro/sgmltut.html>
19. HTTP Cookies, MDN Web Doc <https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies>
20. Material Design for Android, Developers <https://developer.android.com/guide/topics/ui/look-and-feel/>
21. WebStorm , IntelliJ IDEA , <https://www.jetbrains.com/webstorm/>
22. Android Studio , Developers <https://developer.android.com/studio/intro/>
23. Java language <https://www.cnet.com/forums/discussions/what-is-java-and-can-i-do-without-it/>
24. XML , W3school <https://www.w3schools.com/xml/default.asp>
25. Webview <https://developer.android.com/reference/android/webkit/WebView>
26. JVM <https://www.javatpoint.com/internal-details-of-jvm>
27. JSON Introduction , W3school https://www.w3schools.com/js/js_json_intro.asp

ΠΑΡΑΡΤΗΜΑΤΑ

Ο κώδικας της web εφαρμογής:

https://github.com/vaggeliskits/vue.js_medical_app

Ο κώδικας της Android εφαρμογής:

https://github.com/vaggeliskits/Android_medical_app

Ο κώδικας στο Rails:

https://github.com/vaggeliskits/rails_medical_app