



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ
ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ

**Μελέτη Συστήματος Υπενθύμισης Θεραπευτικής Αγωγής μέσω
Διαδικτυακών Υπηρεσιών.**

Μαρκέλλα Νικολοπούλου-Θεμελή

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
Υπεύθυνος
Αναγνωστόπουλος Ιωάννης
Επίκουρος Καθηγητής

Λαμία, έτος 2015



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ
ΒΙΟΙΑΤΡΙΚΗ**

**Μελέτη Συστήματος Υπενθύμισης Θεραπευτικής Αγωγής μέσω
Διαδικτυακών Υπηρεσιών.**

Μαρκέλλα Νικολοπούλου-Θεμελή

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Επιβλέπων
Αναγνωστόπουλος Ιωάννης
Επίκουρος Καθηγητής**

Λαμία, έτος 2015

Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις ⁽¹⁾, που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.
2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.
3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια
4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία:/...../2015

Ο – Η Δηλ.

(Υπογραφή)

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.

**Μελέτη Συστήματος Υπενθύμισης Θεραπευτικής Αγωγής μέσω
Διαδικτυακών Υπηρεσιών.**

Μαρκέλλα Νικολοπούλου-Θεμελή

Τριμελής Επιτροπή:

Αναγνωστόπουλος Ιωάννης, Επίκουρος Καθηγητής (επιβλέπων)

Λουκόπουλος Αθανάσιος, Λέκτορας

Πλαγιανάκος Βασίλης, Αναπληρωτής Καθηγητής

Κεφάλαιο 1. Πίνακας Περιεχομένων

Ευχαριστίες.....	1
Περίληψη	3
Abstract	3
Κεφάλαιο 1. Εισαγωγή.....	5
Κεφάλαιο 2. Συστήματα και Εφαρμογές Υπενθύμισης Ιατρικού Σκοπού	7
2.1 Ορισμός.....	7
2.2 Ιστορική Αναδρομή – Λίγα Λόγια.....	7
2.3 Προηγούμενες Έρευνες και Υλοποιήσεις.....	8
2.4 Υπάρχουσες Υλοποιήσεις.....	10
Κεφάλαιο 3. Αρχιτεκτονική.....	13
3.1 Τεχνολογίες.....	13
3.1.1 HTML - CSS.....	13
3.1.1 PHP	13
3.1.2 MySQL.....	14
3.1.3 JavaScript.....	14
3.1.4 Προσβασιμότητα	14
3.1.2 Bootstrap.....	16
3.2 Αρχιτεκτονική.....	17
3.2.1 Αναλυτική Περιγραφή Κώδικα	21
3.3 Συνοπτική Περιγραφή Χρήσης Ιστοσελίδας.....	34
3.3.1 Χρήση ως Μη Εγγεγραμμένος Χρήστης	34
3.3.2 Χρήση ως Γιατρός - Φροντιστής.....	34
3.3.3 Χρήση ως Ασθενής	35
3.3.4 Χρήση ως Διαχειριστής.....	35
3.3.5 Έξοδος Χρήστη από το σύστημα	35
3.4 Ασφάλεια	35
Κεφάλαιο 3.....	37
Κεφάλαιο 4. Διαδικτυακές Υπηρεσίες Ανοικτού Κώδικα για Εφαρμογές Ειδοποίησης/Υπενθύμισης	37
4.1 Λογισμικό Ανοικτού Κώδικα (Open Source Code).....	37
4.2 Server-side λογισμικό για υποστήριξη εφαρμογών επικοινωνίας/υπενθύμισης	37
4.2.1 AskoziaPBX.....	37
4.2.2 AsteriskPBX.....	38
4.2.3 Elastix.....	38
4.2.4 FreeSWITCH.....	38
4.2.5 FreePBX.....	39
4.2.6 GNU_Gatekeeper.....	39
4.2.7 OpenSips.....	39
4.2.8 SER.....	39
4.2.9 Yate.....	40
4.2.10 OpenVBX.....	40
4.2.11 Συγκριτικός πίνακας.....	40
Κεφάλαιο 5. Συμπεράσματα - Μελλοντικές επεκτάσεις	43
5.1 Σύγκριση με τις Υπάρχουσες Εφαρμογές.....	43

5.2 Μελλοντικές Επεκτάσεις.....	44
Παράρτημα Ι.....	47
Παράρτημα ΙΙ.....	103

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή κ. Ιωάννη Αναγνωστόπουλο, για τις πολύτιμες γνώσεις και συμβουλές που μου παρείχε κατά τη διάρκεια εκπόνησης της πτυχιακής μου εργασίας. Για την υποστήριξη και την πολύτιμη βοήθεια που μου προσέφερε κατά την υλοποίηση της. Καθώς επίσης για την καθοδήγηση στην επίλυση διαφόρων προβλημάτων, η οποία συνέβαλε καθοριστικά στην ομαλή διεκπεραίωση της εργασίας αυτής.

Περίληψη

Η διπλωματική αυτή εργασία αποτελεί μια μελέτη ενός συστήματος υπενθύμισης θεραπείας. Απευθύνεται σε άτομα που ακολουθούν θεραπευτική αγωγή και τους θεράποντες ιατρούς τους. Σκοπός της είναι να βοηθήσει τους ασθενείς, οι οποίοι πάσχουν από αδυναμία μνήμης λήψης της θεραπείας τους, η οποία μπορεί να αποβεί από επικίνδυνη, έως μοιραία. Η πρόταση που παρέχεται με την εργασία αυτή αφορά σε μία διαδικτυακή πλατφόρμα στην οποία οι γιατροί θα συνταγογραφούν τις θεραπείες στους ασθενείς και η υπενθύμιση θα παρέχεται με χρήση τηλεφωνικών κλήσεων ή μηνυμάτων. Χρησιμοποιήθηκαν τεχνολογίες ανάπτυξης του διαδικτύου. Αναλύθηκε η λειτουργία της μελέτης αυτής και άλλων εφαρμογών που ήδη κυκλοφορούν στην αγορά. Επίσης, έγινε μια σύγκριση μεταξύ των εφαρμογών αυτών και της δικής μας πρότασης. Έγινε μια αναλυτική παράθεση τεχνολογιών που προσφέρουν λειτουργικότητα τηλεφώνου μέσω του διαδικτύου. Τέλος, αναφέρθηκε πως, η μελέτη αυτή μπορεί να χρησιμοποιηθεί ως βάση για την μελλοντική δημιουργία ενός πιο ολοκληρωμένου συστήματος.

Abstract

This thesis is a study of a therapy reminder system. It is addressed at people who follow pharmaceutical treatment and their physicians. Its purpose is to help patients who suffer from forgetting their treatment due memory weakness, which may prove dangerous or even fatal. The proposal provided concerns an online platform where doctors will prescribe treatments to patients and the notification will be provided using telephone calls or messages. There were used tools that are used for the development of the web. We analyzed the operation of this study and that of other applications already on the market. Also, a comparison was made between these applications and our own proposal. Furthermore, we made a showcase of technologies offering phone functionalities via the Internet. Finally, it is stated that, this study can be used as a basis for the future creation of a more integrated system.

Κεφάλαιο 1. Εισαγωγή

Η πτυχιακή αυτή εργασία, είναι μια μελέτη, η οποία σκοπό έχει να βοηθήσει τους ασθενείς, οι οποίοι πάσχουν από αδυναμία μνήμης λήψης της θεραπείας τους. Αποτελεί μια προσπάθεια βελτίωσης της ποιότητας ζωής των ασθενών, οι οποίοι αδυνατούν για κάποιους λόγους να θυμηθούν και να πραγματοποιήσουν σωστά, τη λήψη των θεραπευτικών ουσιών, τις οποίες πρέπει να λαμβάνουν σε τακτά και συγκεκριμένα χρονικά διαστήματα. Επειδή η σωστή ή λάθος λήψη των θεραπευτικών ουσιών μπορεί να αποβεί μοιραία και καθοριστική για την πορεία-εξέλιξη της ασθένειας αλλά και τη ζωή των ασθενών, η εργασία στοχεύει στην υπενθύμιση, για τη σωστή λήψη των φαρμάκων, ώστε να αποφευχθούν άσχημες συνέπειες από τυχόν παράληψη ή λάθος λήψη τους. Αποτελεί σημαντικό παράγοντα στη θεραπεία μιας ασθένειας όχι μόνο η θεραπευτική ουσία αλλά και η σωστή λήψη αυτής. Έχουν γίνει επιστημονικές έρευνες οι οποίες το αποδεικνύουν, με συγκλονιστικά αριθμητικά αποτελέσματα, τα οποία θα αναφερθούν πιο κάτω.

Η εργασία χωρίζεται δομικά σε πέντε κεφάλαια. Το πρώτο, είναι η εισαγωγή, στην οποία θα περιγραφεί το πρόβλημα και θα γίνει μια πρόταση για το πώς μπορεί να αντιμετωπιστεί αυτό. Θα γίνει μια αναφορά στις παραμέτρους της μελέτης και στις μελλοντικές προσθήκες που θα μπορούσαν να γίνουν, σε μια προσπάθεια επέκτασης της και βελτίωσης της με τη χρήση και άλλων τεχνολογιών. Οι προσθήκες αυτές θα μπορούσαν να απασχολήσουν ερευνητές οι οποίοι ενδιαφέρονται να ασχοληθούν με το θέμα στα πλαίσια μιας επέκτασης και συμπλήρωσης του, με μεγαλύτερο εύρος επιλογών. Το δεύτερο κεφάλαιο αναφέρει παλαιότερες μελέτες που έχουν γίνει στον τομέα και ιστορικά στοιχεία για τον τομέα της τηλεϊατρικής γενικά και της τηλεφαρμακείας συγκεκριμένα. Επιπλέον παρατίθενται παραδείγματα άλλων εφαρμογών υπενθύμισης θεραπείας ή συνεδρίας με τον θεράποντα ιατρό, που έχουν υλοποιηθεί, και τα χαρακτηριστικά τους. Στο επόμενο κεφάλαιο καταγράφονται οι γλώσσες προγραμματισμού και οι τεχνολογίες που χρησιμοποιήθηκαν στην διπλωματική αυτή. Σε αυτό το κεφάλαιο επίσης, γίνεται περιγραφή της αρχιτεκτονικής της. Περιγράφεται ακόμη, η σχέση των χρηστών με την προδιαγραφή που δημιουργήθηκε, και ο σχεδιασμός της. Στο κεφάλαιο τέσσερα γίνεται μια παράθεση υπηρεσιών ανοιχτού κώδικα που προσφέρουν δυνατότητες πραγματοποίησης τηλεφωνικών κλήσεων και άλλων τηλεφωνικών υπηρεσιών μέσω διαδικτύου. Τέλος στο πέμπτο κεφάλαιο γίνεται σύγκριση, των ήδη γνωστών εφαρμογών που κυκλοφορούν στην αγορά και αναφέρθηκαν στο δεύτερο κεφάλαιο, με τη μελέτη που δημιουργήθηκε στα πλαίσια της διπλωματικής αυτής. Ακόμη αναφέρονται αναλυτικά οι μελλοντικές προσθήκες που μπορούν να γίνουν στη μελέτη. Οι προσθήκες αυτές σχετίζονται με νέες μεθόδους, οι οποίες προκύπτουν από τις ανάγκες των ασθενών, τις δυνατότητες των συστημάτων, αλλά και πιο συνδυαστικές μεθόδους ανάλυσης των στοιχείων και των δεδομένων.

Η διπλωματική αυτή αποτελεί μια πρόταση εφαρμογής, στον τομέα της τηλεϊατρικής και πιο συγκεκριμένα στο πεδίο των συστημάτων υπενθύμισης θεραπείας. Μια πρόταση διαδικτυακής πλατφόρμας υπενθύμισης θεραπείας, μέσω τηλεφώνου, σε όποιο νούμερο επιθυμεί ο χρήστης. Με τον τρόπο αυτό, θα υπενθυμίζεται σε ασθενείς με προβλήματα μνήμης, να λαμβάνουν τη δόση της θεραπείας που ακολουθούν την κατάλληλη ώρα, αποφεύγοντας έτσι σε μεγάλο βαθμό

την παράληψή της ή το ενδεχόμενο διπλής δοσολογίας. Η ιδέα προέκυψε από την αναγκαιότητα της κάλυψης ενός κενού που αντιμετωπίζουν οι μακροχρόνια πάσχοντες ασθενείς που κάνουν λήψη φαρμάκων. Στην ομάδα αυτή ανήκουν, είτε άνθρωποι μεγάλης ηλικίας, είτε νεότεροι οι οποίοι είναι μπλεγμένοι στα γρανάζια της καθημερινότητας και του σύγχρονου τρόπου ζωής, είτε μοναχικοί ασθενείς. Όλες αυτές οι κατηγορίες αντιμετωπίζουν προβλήματα στην σωστή, σε τακτά χρονικά διαστήματα, ορθολογική χρήση των φαρμάκων και άρα αποτελεσματική.

Επιλέχθηκε το τηλέφωνο σαν μέσο ειδοποίησης επειδή είναι ευρέως διαδεδομένο και εύκολο στην χρήση του από όλες τις ομάδες. Σε αντίθεση, ο ηλεκτρονικός υπολογιστής αντιμετωπίζεται ακόμη με άρνηση από μεγάλη μερίδα ατόμων που ανήκουν στις προηγούμενες ομάδες. Ειδικότερα από τα άτομα μεγάλης ηλικίας.

Σε μια εξέλιξη της, η πρόταση αυτή θα μπορούσε να συμπληρωθεί με τη δυνατότητα από μέρους του γιατρού να αλλάζει τη δοσολογία και ίσως την φαρμακευτική ουσία λήψης. Επίσης θα μπορούσαν να υπάρξουν επιπλέον τρόποι ειδοποίησης, όπως μήνυμα στον ηλεκτρονικό υπολογιστή ή μήνυμα στο κινητό κ.α.. Ένας πιο εξελιγμένος τρόπος θα μπορούσε να είναι η ειδοποίηση σε μια φορητή συσκευή η οποία θα μπορούσε να φοριέται σαν ρολόι και ίσως να έδινε το μήνυμα με φωτεινή ένδειξη.

Κατά την διαδικασία της διπλωματικής, προδιαγράφηκε μια υπηρεσία υπενθύμισης θεραπείας σε μορφή ιστοσελίδας στο διαδίκτυο. Χρησιμοποιήθηκαν γλώσσες και τεχνολογίες σχεδιασμού και ανάπτυξης του διαδικτύου. Επίσης έγινε εφαρμογή πρωτοκόλλων διευκόλυνσης πρόσβασης στο διαδίκτυο καθορισμένων από παγκόσμιους οργανισμούς.

Η όλη διαδικασία προσπάθησε να καλύψει την ανάγκη για βελτίωση των συνθηκών επιβίωσης σε ανθρώπους, των οποίων αυτές οι συνθήκες αλλά και η ίδια η ζωή εξαρτάται αποκλειστικά και μόνο από τη σωστή λήψη των φαρμάκων τους. Δεν είναι τυχαίο ότι οι οδηγίες λήψης των θεραπευτικών ουσιών περιλαμβάνουν κυρίως τον τρόπο λήψης (πόση ουσία, σε τι χρονικό διάστημα) και καθορίζονται πολύ συγκεκριμένα και αυστηρά από τον γιατρό. Αποτελεί συνθήκη αναγκαία και απαραίτητη για την πορεία – εξέλιξη της ζωής του ασθενούς. Παραβίαση της συνθήκης αυτής μπορεί να αποβεί μοιραία. Σκοπός της εργασίας αυτής είναι να βοηθήσει τους ασθενείς, οι οποίοι εκτός από την ασθένεια από την οποία πάσχουν, ξεχνούν να πάρουν σωστά τα φάρμακά τους, υπενθυμίζοντάς τους τη λήψη των φαρμάκων αυτών.

Κεφάλαιο 2. Συστήματα και Εφαρμογές Υπενθύμισης Ιατρικού Σκοπού

Στο κεφάλαιο αυτό θα γίνει μια αναφορά στην ιστορία των συστημάτων τηλεϊατρικής γενικά και πιο συγκεκριμένα στην ιστορία των συστημάτων υπενθύμισης θεραπείας. Αρχικά θα δοθεί ένας ορισμός για τα συστήματα αυτά. Μετά την ιστορική αναδρομή, θα παρατεθούν μελέτες που δείχνουν την σημαντικότητα της ύπαρξης και την επίδραση των συστημάτων υπενθύμισης θεραπείας στους ασθενείς και όλους όσους έμμεσα ή άμεσα εμπλέκονται με αυτούς. Τέλος, θα αναφερθεί, θα αναλυθεί και θα περιγραφεί η χρήση και λειτουργία διάφορων εφαρμογών υπενθύμισης θεραπείας που υπάρχουν ήδη στην αγορά.

2.1 Ορισμός

Ως «Σύστημα Υπενθύμισης Θεραπείας» ορίζεται ένα σύστημα που παρέχει έναν ή παραπάνω τρόπους για να υπενθυμίσει σε ασθενείς την θεραπευτική τους αγωγή. Τα συστήματα αυτά μπορούν να έχουν μορφή φορητών συστημάτων, εφαρμογών σε φορητές κινητές συσκευές, ξυπνητηριού με κείμενο, ή άλλη μορφή.

2.2 Ιστορική Αναδρομή – Λίγα Λόγια

Οι κοινωνικοοικονομικές μεταβολές-ανατροπές των τελευταίων δεκαετιών και οι πολιτικές των διαφόρων «αναπτυγμένων» κρατών έχουν δημιουργήσει καινούρια δεδομένα και στον χώρο της υγείας, τα οποία δεν υπήρχαν τα προηγούμενα χρόνια.

Οι κοινωνικές συνθήκες της σύγχρονης καπιταλιστικής κοινωνίας δημιούργησαν καινούριες ανάγκες, δεδομένα, απαιτήσεις, ασθένειες, οι οποίες μπορούν να ικανοποιηθούν μόνο με τα καινούρια σύγχρονα μέσα. Οι οικονομικές αλλαγές που έχουν επέλθει τα τελευταία χρόνια έχουν επηρεάσει και τις παροχές στο σύστημα υγείας.

Οι σύγχρονοι άνθρωποι ζώντας μοναχικά μέσα σε πολύβουες πολυπληθείς μεγαλουπόλεις έχουν την ανάγκη από καινούριες μορφές φροντίδας και ιατρικών παροχών.

Οι παροχές για την υγεία τα τελευταία χρόνια έχουν βελτιωθεί εντυπωσιακά. Σε αυτό συνετέλεσε και η τεράστια και ραγδαία εξέλιξη που παρατηρήθηκε στην επιστήμη της πληροφορίας και η ανάπτυξη της τεχνολογίας στον τομέα των υπολογιστών. Ακόμη η εντυπωσιακή πρόοδος στον τομέα των επικοινωνιών έχει προσφέρει και συντελέσει στον τομέα αυτό.

Η επιθυμία του σύγχρονου ανθρώπου για καλύτερη ζωή, η αύξηση του μέσου όρου ηλικίας και του προσδόκιμου ζωής έχει επιτευχθεί μέσα από τα καινούρια συστήματα υγείας τα οποία κατά κύριο λόγο βασίζονται στην προαναφερόμενη πρόοδο των νέων τεχνολογιών.

Η εκρηκτική αύξηση των χρόνιων ασθενειών, η γήρανση του πληθυσμού, η ύπαρξη όλο και περισσότερων ανθρώπων που ζουν μόνοι και πρέπει να αυτοεξυπηρετούνται, γέννησε την ανάγκη καινούριων – σύγχρονων μορφών παροχής υγείας. Η επιστήμη της

πληροφορικής ιατρικής προσπάθησε να λύσει αυτά τα «καινούρια προβλήματα» που δημιούργησε ο νέος τρόπος ζωής.

Μια πρώτη μορφή αυτής της επιστήμης είναι η τηλεϊατρική. Η επικοινωνία δηλαδή γιατρού – ασθενή, για την ιατρική κάλυψη του δεύτερου από τον πρώτο από απόσταση. Η μεταφορά ιατρικών εικόνων από απομακρυσμένες περιοχές όπου δεν υπάρχει το κατάλληλο ιατρικό προσωπικό σε νοσοκομεία και αντίστοιχους χώρους υπήρξε μια πραγματική επανάσταση. Πέρα από το πρωτοποριακό του όλου θέματος στο πλαίσιο της ιατρικής σαν επιστήμη, ήταν το θαυμαστό της ανακάλυψης αυτής στο πλαίσιο της ανθρωπιστικής της έννοιας. Πολλοί άνθρωποι σώθηκαν με αυτόν τον τρόπο.

Για πρώτη φορά αναφορά στο θέμα γίνεται το 1924 στο εξώφυλλο του περιοδικού “Radio News”. Ένας «ράδιο-γιατρός» μπορούσε να μιλά με έναν ασθενή σε πραγματικό χρόνο μέσω ραδιοκυμάτων. Η πρώτη βιβλιογραφική αναφορά σχετικά με την τηλεϊατρική παρουσιάστηκε το 1950, όπου υπήρχε μία αναλυτική περιγραφή μετάδοσης ακτινογραφιών σε τηλεφωνική γραμμή και σε μία απόσταση 44 περίπου χιλιομέτρων. Οι προσπάθειες στον τομέα αυτό συνεχίστηκαν και εξελίχθηκαν. Ήταν όμως πάντα δαπανηρές και δύσκολα εφαρμόσιμες λόγω των εκάστοτε αντίξοων συνθηκών.

Οι προσπάθειες της πληροφορικής ιατρικής αποσκοπούν ώστε οι καινούριες εφαρμογές να είναι περισσότερο απλές, λιγότερο δαπανηρές, να έχουν μεγαλύτερη και πιο ευρεία εφαρμογή. Οι ερευνητές προσπαθούν να συνδυάσουν τα νέα επιτεύγματα της τεχνολογίας των υπολογιστών και να τα προσαρμόσουν στις ανάγκες του σύγχρονου ανθρώπου. Υπάρχει μεγάλη προσπάθεια για την εφαρμογή τους στον τομέα της υγείας. Τον τομέα που συσχετίζεται με την ίδια τη ζωή.

Εδώ εμπίπτει και η συγκεκριμένη έρευνα. Προσπαθεί να βοηθήσει ανθρώπους μόνους, μεγάλης ηλικίας, ασθενείς με χρόνια νοσήματα, να έχουν όσο το δυνατό, και για όσο περισσότερο γίνεται, μια ποιότητα ζωής μαθαίνοντας να ζουν με την αρρώστια τους.

2.3 Προηγούμενες Έρευνες και Υλοποιήσεις

Η βιβλιογραφία έχει να δείξει μεγάλο πλήθος ερευνών, όσον αφορά τα συστήματα υπενθύμισης θεραπείας και τα αποτελέσματα τους στους χρήστες. Τα πορίσματα των περισσότερων από αυτές τις έρευνες, αφορούν στην εξέλιξη που είχαν οι ασθενείς, που χρησιμοποιούσαν συστήματα υπενθύμισης θεραπείας, σε διάφορες περιπτώσεις θεραπειών.

Την σημαντικότητα ύπαρξης συστημάτων υπενθύμισης θεραπείας δείχνουν οι μελέτες που πραγματοποιήθηκαν από το Ινστιτούτο Υγείας της Νέας Αγγλίας. Συγκεκριμένα, μελέτη που έκανε για την χρήση των κονδυλίων που δίνονται για την υγεία στις Ηνωμένες Πολιτείες της Αμερικής, δείχνει πως 290 δισεκατομμύρια δολάρια χάνονται σε μη ιδανικές συνταγογραφήσεις, χορηγίες φαρμάκων και διαγνώσεις, (στοιχεία που αποδεικνύουν τη σπουδαιότητα του να δίνονται τα ανάλογα-κατάλληλα φάρμακα σε κάθε περίπτωση-πάθηση). Το ποσό αυτό αντιστοιχεί στο 13% των συνολικών κονδυλίων. Επιπλέον η μη προσκόλληση στην θεραπευτική αγωγή οδηγεί σε 3.5 εκατομμύρια εισαγωγές σε νοσοκομεία και 125 χιλιάδες θανάτους το χρόνο. Αυτό αποδεικνύει, για ακόμη μια φορά και μάλιστα μέσα από επιστημονική εμπειριστατωμένη έρευνα, τη μεγάλη σημασία, όπως προαναφέραμε, της σωστής,

ορθολογικής, τακτικής, βασισμένης στις οδηγίες του θεράποντος γιατρού, λήψης των φαρμάκων από τους ασθενείς. [1]

Μια βιβλιογραφική έρευνα που αφορά στην προσκόλληση ασθενών στην θεραπεία τους, όταν έκαναν χρήση συστήματος υπενθύμισης, δείχνει πως οι συγκεκριμένοι, είχαν μια στατιστικά σημαντική αύξηση της προσκόλλησης στην καθημερινή τους θεραπεία. Οι μελέτες που υπέστησαν μεταανάλυση είχαν πραγματοποιηθεί από το 1999 μέχρι το 2009. Η αύξηση αφορά ένα 66.61% έναντι του 54.71%, 95% CI για μέσο όρο 0.8% με 22.4%. [2]

Μια τυχαιοποιημένη έρευνα των Christensen et. al., επάνω στην συμμόρφωση των ασθενών για τη λήψη της υπερτασικής θεραπείας τους, σε σχέση με την χορήγηση ή μη υπενθύμισης, έδωσε επίσης θετικά αποτελέσματα. Πιο συγκεκριμένα, όλοι οι 398 ασθενείς λάμβαναν θεραπεία καθημερινά για δώδεκα μήνες. Τους πρώτους έξι οι μισοί, των οποίων η επιλογή έγινε τυχαία, δέχονταν ηλεκτρονική ιατρική παρακολούθηση και υπενθύμιση της λήψης της θεραπείας τους, μέσω μιας συσκευής, ενώ οι υπόλοιποι μισοί ασθενείς, λάμβαναν παραδοσιακή μορφή ιατρικής παρακολούθησης. Μετά από έξι μήνες οι ομάδες αντάλλαξαν την μορφή της ιατρικής παρακολούθησης και υπενθύμισης. Οι ασθενείς που είχαν μόλις διαγνωσθεί με υπέρταση, μετά την έρευνα με τη χρήση της συγκεκριμένης συσκευής, είχαν 6% αύξηση της προσκόλλησης στην θεραπεία τους. [3]

Μια παρόμοια έρευνα που έγινε για την φυματίωση, υποστηρίζει και αυτή πως η διάγνωση και η θεραπεία υποβοηθούνται δραστικά από συστήματα υπενθύμισης θεραπείας. Η έρευνα αυτή περιελάμβανε υπενθύμιση για ραντεβού με τον θεράποντα ιατρό και όχι για την λήψη της θεραπείας. Αναλυτικότερα, η συγκεκριμένη μελέτη περιλάμβανε εννέα μελέτες, με 4654 συμμετέχοντες, που τα αποτελέσματα τους υποστήριζαν τα συμπεριλαμβανόμενα στην έρευνα κριτήρια. Από αυτές πέντε περιλάμβαναν υπενθύμιση ραντεβού για ασθενείς με ενεργό φυματίωση, δύο για προφυλακτικά μέτρα για ασθενείς με λανθάνουσα φυματίωση και τέσσερις για ασθενείς που λάμβαναν διαγνωστικά τεστ. Όλα τα αποτελέσματα έδειξαν μια αύξηση στατιστικά σημαντική στις ομάδες που λάμβαναν την υπενθύμιση, σε σχέση με αυτούς που δεν λάμβαναν. [4]

Μια ακόμα μελέτη, που ερευνά την επίδραση των συστημάτων υπενθύμισης για Αντιρετροϊκή θεραπεία, ενός εξατομικευμένου συστήματος υπενθυμίσεων κινητού τηλεφώνου, είναι αυτή των Hardy et. al.. Αυτοί έδειξαν πως μετά από μόλις έξι εβδομάδες συμμετοχής στην μελέτη, η προσκόλληση των ασθενών στην θεραπεία τους, αυξήθηκε σε αυτούς που λάμβαναν υπενθύμιση μέσω του συστήματος που σχεδίασαν. [5]

Πρέπει επίσης να αναφέρουμε και τη μελέτη των Stacey R. Finkelstein et. al. η οποία είχε σκοπό να διαπιστώσει ποιους τρόπους υπενθύμισης ραντεβού με τον θεράποντα ιατρό προτιμούν οι περισσότεροι ασθενείς. Ακόμα, επιθυμούσαν να μελετήσουν την εξοικείωση των ασθενών με διάφορα συστήματα υπενθύμισης, από διάφορους παροχείς. Τα αποτελέσματα του ερωτηματολογίου που χρησιμοποίησαν, έδειξαν πως η εξοικείωση του χρήστη με την τεχνολογία του σύντομου μηνύματος (SMS) και του ηλεκτρονικού μηνύματος (e-mail), σε συνδυασμό με την εξοικείωση του με τους αποστολείς των μηνυμάτων αυτών, δίνουν τα καλύτερα αποτελέσματα όσον αφορά την υπενθύμιση ενός ιατρικού ραντεβού. Το ερωτηματολόγιο χορηγήθηκε σε μία πρωτογενή αστική κλινική. Οι δύο παράγοντες που αναφέρθηκαν παραπάνω είναι τόσο σημαντικοί ώστε στην περίπτωση ύπαρξής τους, όλες οι υπόλοιπες

μεταβλητές όπως ηλικία και δημογραφικό γκρουπ καθίστανται αμελητέες. Μια σημαντική πρόταση που κάνουν οι μελετητές, είναι το ότι οι υπεύθυνοι που παρέχουν την υπενθύμιση θα πρέπει να διασφαλίζουν το ότι τα στοιχεία επικοινωνίας που έχουν δώσει οι χρήστες είναι σωστά, ώστε να λαμβάνουν την υπενθύμιση. Επιπλέον, προτείνουν πως σε ιδρύματα-οργανισμούς, όπως νοσοκομεία που είναι οργανωμένα μέσω σύγχρονων υπολογιστικών συστημάτων, θα είναι ιδιαίτερα εύκολη και με ιδιαίτερα χαμηλό κόστος η προσθήκη ενός συστήματος ηλεκτρονικής υπενθύμισης ραντεβού. [6]

2.4 Υπάρχουσες Υλοποιήσεις

Η εφαρμογή Care4Today Mobile Adherences προσφέρει έναν συνδυασμό μηνυμάτων σε κινητό τηλέφωνο, ενός site και μια εφαρμογή για έξυπνα κινητά τηλέφωνα. Ο χρήστης εγγράφεται δωρεάν στην πλατφόρμα στο διαδίκτυο και στην συνέχεια έχει την δυνατότητα να εισάγει όλες του τις φαρμακευτικές αγωγές. Από εκεί και πέρα η εφαρμογή του παρέχει μια πληθώρα επιλογών. Οι κυριότερες είναι η υπενθύμιση τόσο για την λήψη της θεραπείας, αλλά και για την περίπτωση που εξαντληθεί η θεραπεία και πρέπει να γίνει νέα συνταγογράφηση. Μια άλλη επιλογή που έχει ο χρήστης είναι η παρακολούθηση της λήψης της αγωγής του. Η διασύνδεση που παρέχει η εφαρμογή είναι μια σημαντική λειτουργία. Οι θεράποντες ιατροί και ο φαρμακοποιός του έχουν την δυνατότητα να ελέγχουν την λήψη της θεραπείας του και να λαμβάνουν τα κατάλληλα μέτρα. Επίσης οι συγγενείς των ασθενών μπορούν να βλέπουν το πρόγραμμα αγωγής του και να ελέγχουν εάν έγινε η σωστή λήψη της. [7]

Μια άλλη εφαρμογή υπενθύμισης είναι το MediSafe Meds & Pill Reminder. Είναι σχεδιασμένη για έξυπνες κινητές συσκευές και η υπενθύμιση γίνεται με την μορφή υπενθυμίσεων. Ο χρήστης εισάγει τις φαρμακευτικές του αγωγές και τις χρονικές περιόδους που πρέπει να τις λαμβάνει, και η εφαρμογή δημιουργεί μια ειδοποίηση την σωστή στιγμή. Στην περίπτωση που ο χρήστης αμελήσει την λήψη της αγωγής του η εφαρμογή στέλνει μια ειδοποίηση σε κοντινό του άτομο για να διασφαλίσει αυτό την λήψη της. Προσφέρεται επίσης η δυνατότητα ο χρήστης να εισάγει διάφορες ζωτικές μετρήσεις αλλά και να τις αποστείλει με mail απευθείας σε όποιον επιθυμεί. Ακόμα είναι δυνατή η αποθήκευση ιατρικών ραντεβού και γιατρών στην μνήμη της εφαρμογής. Η εφαρμογή αυτή υπάρχει για κινητές συσκευές Android και iOS. [8] [9]

Μια αντίστοιχη εφαρμογή είναι το Med Helper Pill Reminder. Και πάλι ο ίδιος ο χρήστης πρέπει να εισάγει τις θεραπείες. Μπορεί επίσης να εισάγει τους θεράποντες ιατρούς, αλλά σε αυτή την περίπτωση εάν το επιθυμεί και τους φαρμακοποιούς του. Η υπενθύμιση και πάλι παρέχεται με την μορφή ειδοποιήσεων στο κινητό τηλέφωνο. Επίσης μπορεί να εισάγει και σε αυτήν την εφαρμογή ζωτικές του μετρήσεις και ιατρικά ραντεβού. Η εφαρμογή αυτή υποστηρίζεται τόσο από συσκευές Android όσο και iOS. [10] [11]

Άλλη εφαρμογή υπενθύμισης θεραπείας σε κινητές συσκευές είναι το Dosecast - Medication Reminder. Η εφαρμογή αυτή έχει τόσο ελεύθερη όσο και επί πληρωμή έκδοση. Στην ελεύθερη υπάρχουν περιορισμένες δυνατότητες. Για παράδειγμα, δεν μπορεί να καταγράψει τις δόσεις της θεραπείας του που παρέλειψε και όταν πρέπει να επιλέξει το είδος του σκευάσματος που χορηγείται είναι κλειδωμένη μόνο η επιλογή «χάπι». Ακόμη υπάρχει η δυνατότητα να σταλούν τα στοιχεία που έχει εισάγει ο

χρήστης μέσω κάποιας εφαρμογής αποστολής μηνυμάτων. Στην περίπτωση που ο χρήστης πατήσει το αντίστοιχο κουμπί, εμφανίζεται ένα παράθυρο που τον πληροφορεί πως οι ιατρικές του πληροφορίες θα αποσταλούν σε άλλο άτομο και πως αυτό είναι πιθανά επικίνδυνο. Η εφαρμογή δίνει την δυνατότητα στο χρήστη να λάβει την θεραπεία πριν την προκαθορισμένη ώρα. Επίσης κατά την διάρκεια της αναμονής υπάρχει η δυνατότητα να αναιρεθεί η λήψη της αμέσως προηγούμενης δόσης και η παράληψη της αμέσως επόμενης. Όταν επιλεγεί η αναβολή μιας δόσης πρέπει να εισαχθεί ο χρόνος της αναβολής. Στην περίπτωση που γίνει παράληψη μιας δόσης, παραλείπονται και όλες οι επόμενες μέχρις ότου ο χρήστης πει στο σύστημα πως έλαβε μια δόση. Η εφαρμογή αυτή υπάρχει μόνο για συσκευές Android. [12]

Τέλος μια εφαρμογή υπενθύμισης ιατρικών ραντεβού είναι το RemindMe της talkSoft. Με αυτή την εφαρμογή επιβεβαιώνονται τα ραντεβού μεταξύ γιατρού και ασθενή και υπενθυμίζονται τα ραντεβού στον ασθενή. [13]

Κεφάλαιο 3. Αρχιτεκτονική

Στο κεφάλαιο αυτό θα παρατεθούν όλες οι τεχνολογίες που χρησιμοποιήθηκαν για την διπλωματική αυτή. Αρχικά, θα παρατεθούν οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν. Στην συνέχεια θα παρατεθούν όλες οι τεχνολογίες και τα πακέτα που αναπτύχθηκαν από άλλους δημιουργούς. Επίσης θα αναφερθούν τα πρωτόκολλα που ακολουθήθηκαν και εφαρμόστηκαν για την πραγματοποίηση της εργασίας.

3.1 Τεχνολογίες

3.1.1 HTML - CSS

Η HTML (HyperText Markup Language - Γλώσσα Σήμανσης Υπερκειμένου) χρησιμοποιείται στις ιστοσελίδες και είναι η κύρια γλώσσα σήμανσης τους και τα στοιχεία της είναι τα βασικά δομικά στοιχεία τους.

Η HTML αποτελείται από ετικέτες (tags), οι οποίες λειτουργούν σε ζευγάρια ανοίγματος και κλεισίματος. Με αυτές τις ετικέτες δομούνται οι ιστοσελίδες. Πολλές φορές μια ιστοσελίδα περιλαμβάνει εντολές από άλλες γλώσσες όπως η JavaScript και η PHP.

Το 1980 ο φυσικός Τιμ Μπέρνερς Λι, ο οποίος εργαζόταν στο CERN, δημιούργησε το ENQUIRE, Αυτό, μετατράπηκε το 1989 σε ένα σύστημα βασισμένο στο διαδίκτυο, το οποίο θα απαιτούσε γλώσσα υπερκειμένου. Έτσι η ανάγκη για μια τέτοια γλώσσα οδήγησε τον Μπέρνερς Λι να φτιάξει την προδιαγραφή της HTML. [14] [15]

Η CSS (Cascading Style Sheets) είναι μια γλώσσα προγραμματισμού που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ. Η μοναδική της λειτουργικότητα είναι να δίνει στιλιστικούς κανόνες για μια ιστοσελίδα, πράγμα που δεν μπορεί να κάνει η HTML. Είναι λοιπόν απαραίτητη για τη δημιουργία μιας καλοσχεδιασμένης ιστοσελίδας. Χρησιμοποιείται για τον καθορισμό της εμφάνισης ενός αρχείου γραμμένου σε κάποια γλώσσα σήμανσης όπως HTML και XHTML. Από αυτό συμπεραίνουμε πως χρησιμοποιείται για τον καθορισμό της εμφάνισης μιας ιστοσελίδας. [16] [15]

3.1.1 PHP

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο, το οποίο εμφανίζεται στον περιηγητή του χρήστη, αφού το αρχείο περάσει από έναν διακομιστή του παγκόσμιου ιστού.

Η PHP το 1994 (Rasmus Lerdorf) δημιουργήθηκε ως παράγωγο της Perl. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, υλοποιημένη σε γλώσσα C. Την ίδια χρονιά (Andi Gutmans και Zeev Suraski) η γλώσσα γράφτηκε από την αρχή, αλλά έμοιαζε πολύ με την προηγούμενη έκδοση της. Η έκδοση 3.0 είναι η πρώτη η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Ακολούθησε το 1998 η έκδοση 4 της PHP. Τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6. [17] [15]

3.1.2 MySQL

Η MySQL (Structured Query Language- Δομημένη Γλώσσα Ερωτήσεων) περιλαμβάνει τόσο ένα σύστημα σχεσιακών βάσεων δεδομένων (Relational Database Management System (RDBMS)) όσο και μια γλώσσα διατύπωσης ερωτημάτων για τις βάσεις αυτές. Ανήκει στην εταιρία Oracle και ο κώδικάς της είναι διαθέσιμος μέσω της General Public License (GNU). Έχει περισσότερες από 11 εκατομμύρια εγκαταστάσεις, μερικές από τις οποίες είναι το Flickr, το YouTube, η Wikipedia, το Google, το Facebook και το Twitter. Πέραν των εταιριών που χρησιμοποιούν την SQL υπάρχει και πληθώρα εφαρμογών που την υποστηρίζουν όπως, TYPO3, MODx, Joomla, WordPress, phpBB, MyBB και Drupal. Είναι τόσο διαδεδομένη επειδή υποστηρίζει, πολλές πλατφόρμες και τη δημιουργία ασφαλών συνεδριών (SSL) . Ακόμη είναι εύκολη στην εγκατάσταση της και έχει υψηλή διαθεσιμότητα. Επίσης έχει τη δυνατότητα εύκολης δημιουργίας εφεδρικού αντιγράφου και ανάπτυξης σε σύννεφο. [18] [15]

3.1.3 JavaScript

Η JavaScript (JS) είναι γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές, η οποία χρησιμοποιήθηκε αρχικά στην δημιουργία προγραμμάτων, εφαρμογών διαδικτύου, τα οποία εκτελούνται στην πλευρά του πελάτη. Το όνομά της θυμίζει πολύ τη Java αλλά το μόνο που κληρονομεί από αυτήν είναι οι συμβάσεις ονοματοδοσίας. Αντίθετα η υλοποίησή της επηρεάζεται στενά από τις C, Self και Scheme. Υποστηρίζει αντικειμενοστραφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού. Άλλες δημοφιλείς χρήσεις της JavaScript είναι τα έγγραφα PDF, οι εξειδικευμένοι φυλλομετρητές (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets). Τα τελευταία χρόνια έχει γίνει δημοφιλής και για την ανάπτυξη εφαρμογών διαδικτύου που εκτελούνται στην πλευρά του διακομιστή (server-side). [19] [15]

3.1.4 Προσβασιμότητα

Στην μελέτη λήφθηκαν υπόψη τα πρωτόκολλα προσβασιμότητας του w3c για διευκόλυνση της πρόσβασης για άτομα με προβλήματα όρασης. Πιο συγκεκριμένα οι κανόνες που ικανοποιήθηκαν είναι:

- Να προστίθεται ένας σύνδεσμος στο πάνω μέρος κάθε σελίδας που πηγαίνει απευθείας στην κύρια περιοχή περιεχομένου.
- Να διασφαλίζεται ότι οι πληροφορίες που διαβιβάζονται από τις διαφορές χρώματος είναι επίσης διαθέσιμες σε κείμενο.
- Να επιτρέπεται στους χρήστες να ολοκληρώσουν μια δραστηριότητα χωρίς κανένα χρονικό περιορισμό.
- Να παρέχεται ένα κουμπί υποβολής για να ξεκινήσει μια αλλαγή περιβάλλοντος.

- Να παρέχεται περιγραφή κειμένου όταν ο χρήστης παρέχει πληροφορίες που δεν είναι στη λίστα των επιτρεπόμενων τιμών.
- Να παρέχεται περιγραφή κειμένου όταν τα δεδομένα που εισάγει ο χρήστης, δεν εμπίπτουν με την απαιτούμενη μορφή ή τιμές.
- Να παρέχεται κείμενο περιγραφής των αντικειμένων, που σε διαφορετική περίπτωση ο χρήστης θα έπρεπε να στηριχθεί μόνο στις αισθητηριακές πληροφορίες για να γίνει κατανοητό το περιεχόμενο.
- Να χρησιμοποιούνται το χρώμα και το σχέδιο.
- Να χρησιμοποιούνται τα σημασιολογικά στοιχεία για τη σήμανση και δομή.
- Να επικυρώνεται η ιστοσελίδα.
- Να διαχωρίζονται οι πληροφορίες και η δομή από την παρουσίαση για να ενεργοποιούνται διαφορετικές παρουσιάσεις.
- Να χρησιμοποιείται μια τεχνολογία που έχει συνήθεις διαθέσιμους browsers που υποστηρίζουν μεγέθυνση (zoom).
- Να χρησιμοποιείται ρευστός σχεδιασμός.
- Να διασφαλίζεται ότι δεν υπάρχει καμία απώλεια περιεχομένου ή λειτουργικότητας, όταν το κείμενο αλλάζει σε μέγεθος και τα δοχεία κειμένου δεν αλλάζουν το πλάτος τους.
- Να διασφαλίζεται ότι οι πρόσθετες οπτικές ενδείξεις είναι διαθέσιμες, όταν το χρώμα του κειμένου έχει διαφορές που χρησιμοποιούνται για τη μεταφορά πληροφοριών.
- Να παρέχεται κείμενο οδηγιών στην αρχή της φόρμας ή στο σύνολο των πεδίων που περιγράφει όλα τα απαραίτητα στοιχεία.
- Να είναι προσβάσιμες όλες οι σελίδες του site από την αρχική σελίδα.
- Να ανοίγουν νέα παράθυρα και καρτέλες από μια σύνδεση, μόνο όταν είναι απαραίτητο.
- Να μην παρεμβαίνει η αναδιαμόρφωση του browser του κειμένου, όταν το παράθυρο προβολής στενεύει.
- Να δημιουργείται μια λογική σειρά πλοήγησης μέσω συνδέσμων, στοιχείων ελέγχου φορμών και αντικειμένων.
- Να παρέχεται τίτλος χρησιμοποιώντας το στοιχείο <title>.
- Να παρέχονται κουμπιά υποβολής.
- Να χρησιμοποιούνται στοιχεία επισήμανσης που να συνδέουν τις ετικέτες κειμένου με στοιχεία ελέγχου φόρμας.
- Να χρησιμοποιείται σημασιολογική σήμανση για να δειχθεί έμφαση ή ειδικό κείμενο.
- Να χρησιμοποιείται πίνακας σήμανσης για να παρουσιάσει πληροφορίες σε μορφή πίνακα.
- Να χρησιμοποιούνται διακριτικά στα στοιχεία της HTML.
- Να χρησιμοποιούνται στοιχεία συνδέσμου και εργαλεία πλοήγησης.
- Να παρέχεται κλάση στοιχείων στην αρχή της κάθε ενότητας του περιεχομένου.
- Να διασφαλίζεται ότι οι ετικέτες ανοίγματος και κλεισίματος χρησιμοποιούνται σύμφωνα με τις προδιαγραφές.
- Να διασφαλίζεται ότι οι ιστοσελίδες είναι καλά διαμορφωμένες.
- Να παρέχεται ο σκοπός ενός συνδέσμου χρησιμοποιώντας κείμενο συνδέσμου σε συνδυασμό με το στοιχείο της λίστας που περικλείει.

- Να χρησιμοποιείται ένα κουμπί με ένα στοιχείο <select> για να εκτελεστεί μια ενέργεια.
- Να χρησιμοποιείται η HTML σύμφωνα με τις προδιαγραφές.
- Να διασφαλίζεται ότι οι ιδιότητες id είναι μοναδικές σε μια ιστοσελίδα.
- Να διασφαλίζεται ότι τα στοιχεία δεν περιέχουν διπλότυπες ιδιότητες.
- Να ομαδοποιούνται σχετικές συνδέσεις χρησιμοποιώντας το στοιχείο NAV.
- Να τοποθετείται το περιεχόμενο με βάση την δομική σήμανση.
- Να χρησιμοποιείται τοis εκατό για τα μεγέθη γραμματοσειράς.
- Να χρησιμοποιείται CSS για να αλλαχθεί η παρουσίαση ενός στοιχείου διεπαφής χρήστη, όταν λαμβάνει εστίαση.
- Να υπάρχει δυνατότητα κλιμάκωσης των στοιχείων φόρμας που περιέχουν κείμενο.
- Να καθορίζεται ευθυγράμμιση είτε προς τα αριστερά ή προς τα δεξιά σε CSS.
- Να χρησιμοποιούνται σχετικές μετρήσεις για να οριστούν τα πλάτη των στηλών έτσι ώστε οι γραμμές να μπορούν να έχουν κατά μέσο όρο 80 χαρακτήρες ή λιγότερο, όταν το πρόγραμμα περιήγησης αλλάζει μέγεθος.
- Να καθοριστεί το διάστιχο σε CSS.
- Να χρησιμοποιείται CSS για να ελεγχθεί η οπτική παρουσίαση του κειμένου.
- Να καθορίζονται τα χρώματα κειμένου και φόντου του δευτερεύοντος περιεχομένου, των χαρακτηριστικών και της πλοήγησης, σε CSS, ενώ να μην προσδιορίζονται τα χρώματα φόντου και κειμένου του κυρίως περιεχομένου.
- Να χρησιμοποιούνται ποσοστιαίες τιμές σε CSS για μεγέθη <container>.
- Να επιτρέπεται στο χρήστη να παρατείνει την προθεσμία προεπιλογής.
- Να χρησιμοποιείται ένα επεκτάσιμο και πτυσσόμενο μενού για να παρακάμψει μπλοκ επαναλαμβανόμενου περιεχομένου.
- Να υπολογίζεται το μέγεθος και η θέση κατά τρόπο που κλιμακώνεται μαζί με το μέγεθος του κειμένου.
- Να μεταφερθούν οι δράσεις στο πληκτρολόγιο με την προσθήκη onclick στα κουμπιά.
- Να καθοριστεί η προεπιλεγμένη γλώσσα στην κεφαλίδα HTTP.
- Να χρησιμοποιούνται τυπικές συμβάσεις μορφοποίησης κειμένου για τις παραγράφους.
- Να χρησιμοποιούνται τυπικές συμβάσεις μορφοποίησης κειμένου για τις λίστες.
- Να χρησιμοποιούνται τυπικές συμβάσεις μορφοποίησης κειμένου για τους τίτλους.

Οι παραπάνω κανόνες ικανοποιήθηκαν με διάφορους τρόπους σε όλο το εύρος της μελέτης. [20]

3.1.2 Bootstrap

Το Bootstrap είναι μια συλλογή εργαλείων HTML, CSS και επεκτάσεων JavaScript για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Αναπτύχθηκε στο

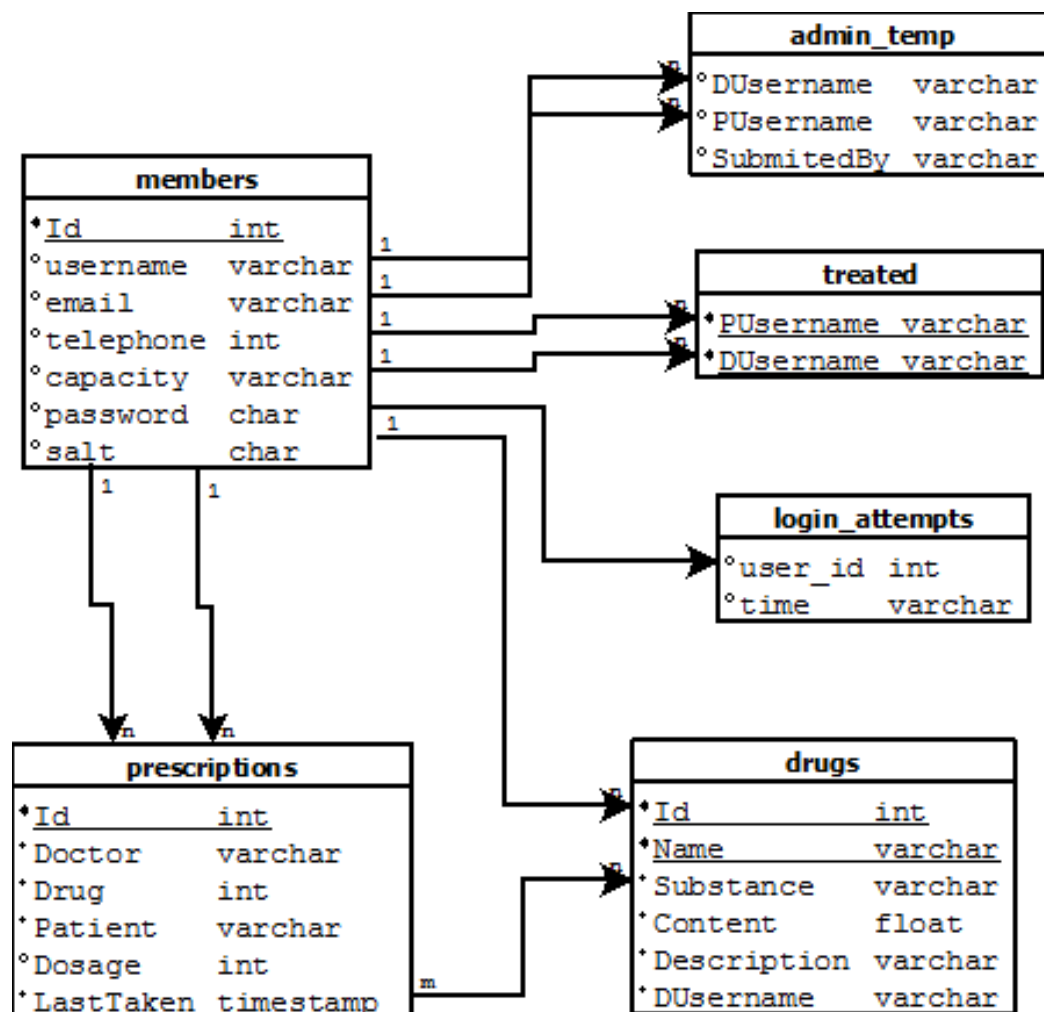
Twitter για την ενοποίηση των εργαλείων μορφοποίησης, που μέχρι εκείνη την στιγμή οδηγούσαν σε αντιφάσεις και υψηλό κόστος συντήρησης.

Η πρώτη εφαρμογή υπό πραγματικές συνθήκες έγινε κατά τη διάρκεια της πρώτης Hackweek του Twitter, όπου εκεί εξηγήθηκε η χρήση της εργαλειοθήκης και είχε τα επιθυμητά θετικά αποτελέσματα. Τον Αύγουστο του 2011 κυκλοφόρησε το Twitter Bootstrap ως λογισμικό ανοιχτού κώδικα στο GitHub. Μερικούς μήνες αργότερα, (Φεβρουάριος του 2012) ήταν το πιο δημοφιλές έργο ανάπτυξης στον ιστότοπο αυτό. [21] [22]

3.2 Αρχιτεκτονική

Η αρχιτεκτονική/σχεδιασμός ενός υπολογιστικού συστήματος αποτελεί τον τρόπο με τον οποίο δομείται και σχεδιάζεται αυτό το σύστημα. Περιλαμβάνει ένα σύνολο εργαλείων και κανόνων που πρέπει να ακολουθηθούν για τη βέλτιστη ολοκλήρωση του συστήματος. Αποτελεί πολύ σημαντικό στάδιο της υλοποίησης και υπάρχει ολόκληρη επιστήμη πάνω στην οποία βασίζεται.

Κατά την πραγματοποίηση της μελέτης, δημιουργήθηκε μια βάση δεδομένων η οποία πληρώθηκε με ψεύτικα δεδομένα (dummies). Η βάση αυτή σχεδιάστηκε για να



Διάγραμμα 1 Το σχεσιακό μοντέλο της βάσης δεδομένων

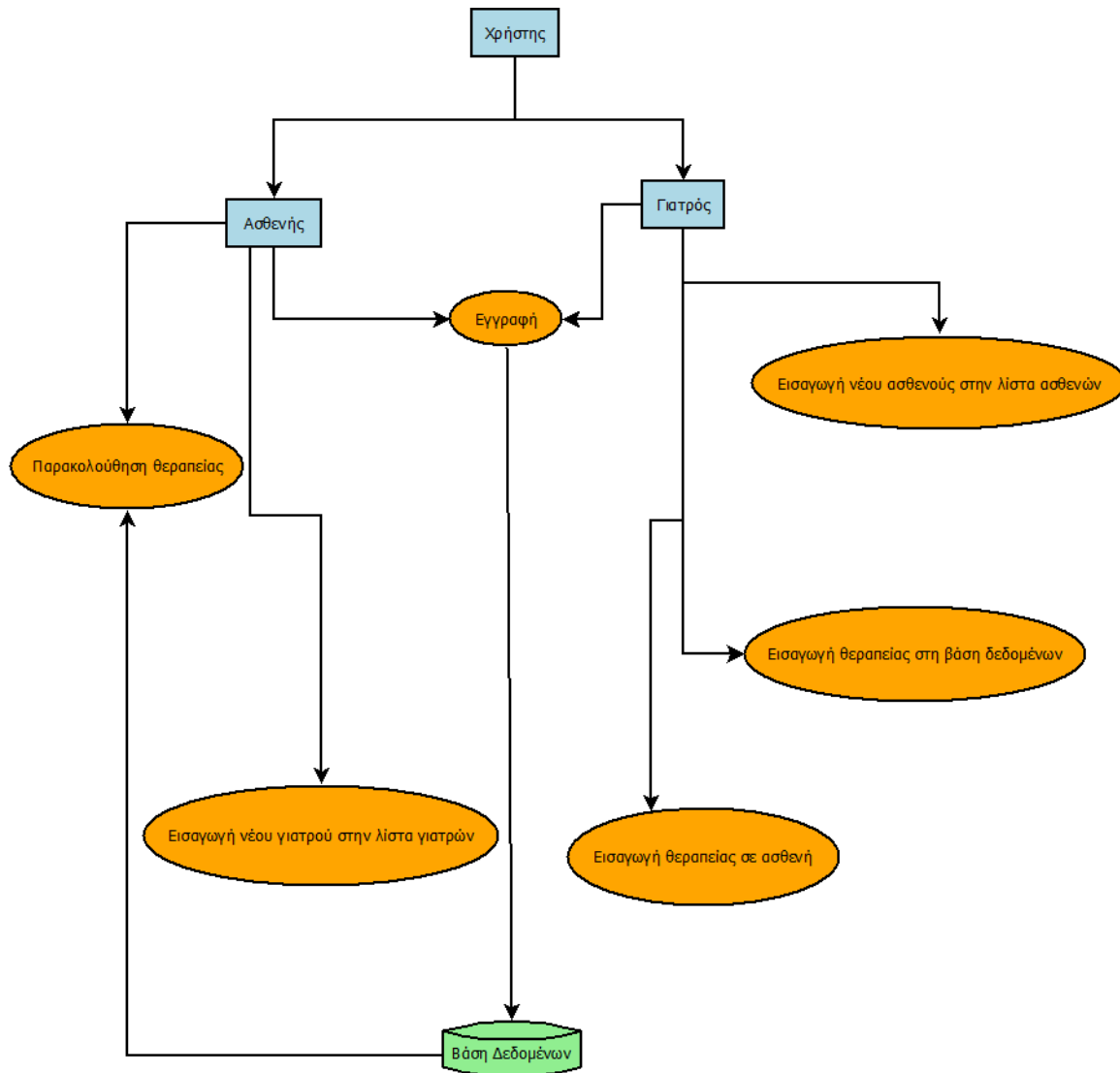
ικανοποιεί τις απαιτήσεις της μελέτης και η σχεδίασή της φαίνεται στο σχεσιακό μοντέλο του διαγράμματος 1. Ο πίνακας members είναι ο βασικός πίνακας. Στο διάγραμμα 1 φαίνονται και οι διάφορες σχέσεις που υπάρχουν μεταξύ των πινάκων.

Στον πίνακα prescriptions αποθηκεύονται τα δεδομένα για τις συνταγογραφίες που έχουν χορηγήσει οι ιατροί στους ασθενείς. Από αυτόν τον πίνακα ελέγχονται τα στοιχεία για την προβολή των επόμενων δόσεων που πρέπει να λάβει ο ασθενής.

Ο πίνακας admin_temp αποθηκεύει τα αιτήματα για σύνδεση μεταξύ των ιατρών και των ασθενών. Τα αιτήματα που κάνουν αυτά τα δύο μέλη, τα ελέγχει ο διαχειριστής του συστήματος. Όταν πραγματοποιηθεί ένα αίτημα για σύνδεση από έναν γιατρό ή έναν ασθενή αποθηκεύεται σε αυτόν τον πίνακα. Ο πίνακας αυτός δεν διαθέτει πρωτεύον κλειδί.

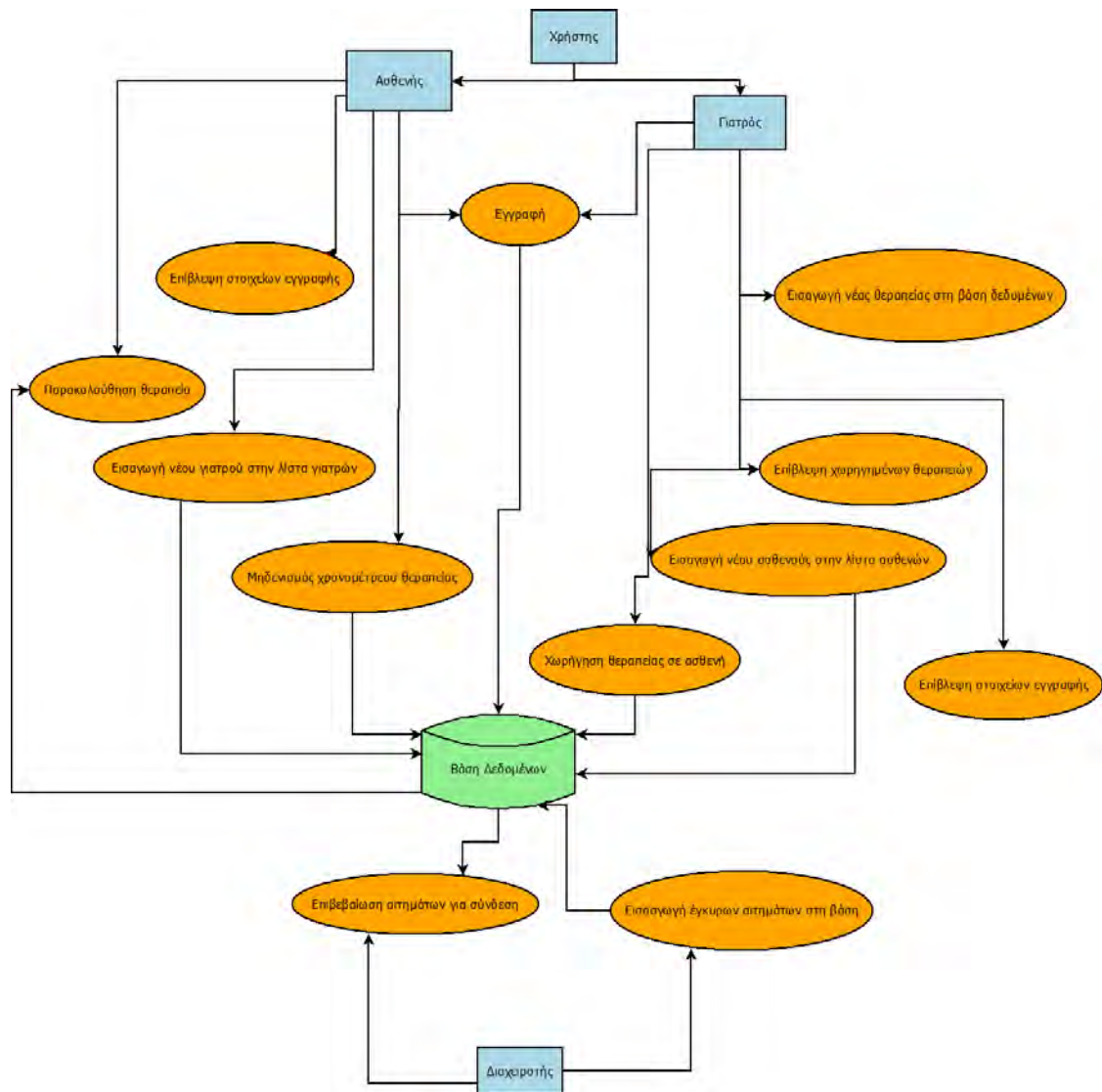
Αφού ένα αίτημα εγκριθεί από τον διαχειριστή προστίθεται στον πίνακα treated. Σε αυτόν αποθηκεύονται όλες οι εγγραφές που σχετίζονται με την σχέση μεταξύ θεραπευτή - θεραπευόμενου.

Τέλος, ο πίνακας drugs αποθηκεύει τις θεραπείες που εισάγουν οι γιατροί, για να συνταγογραφούν, καθώς και τα χαρακτηριστικά τους. Είναι υποχρεωτική η εισαγωγή του ονόματος μια θεραπείας αλλά όχι όλα τα υπόλοιπα στοιχεία. Παρόλα αυτά, είναι



Διάγραμμα 2 Διάγραμμα ροής δεδομένων επιπέδου μηδέν..

επιθυμητό να εισάγονται και αυτά καθώς θα διευκολύνουν και τον ιατρό στην μετέπειτα χρήση του συστήματος. Από την στιγμή που πρωτεύον κλειδί του πίνακα είναι το πεδίο id, υπάρχει η δυνατότητα να υπάρχουν πάνω από μία εγγραφές με το ίδιο όνομα θεραπείας. Αυτό επιτρέπει σε κάθε γιατρό να εισάγει μια θεραπεία που διαθέτει πολλές διαφορετικές περιεκτικότητες στην ενεργό ουσία, για χρήση. Επίσης, με αυτόν τον τρόπο κάθε γιατρός βλέπει μόνο τις θεραπείες που ο ίδιος έχει εισάγει και δεν προκαλείται σύγχυση, με την εμφάνιση θεραπειών διαφορετικών ειδικοτήτων σε κάθε γιατρό.



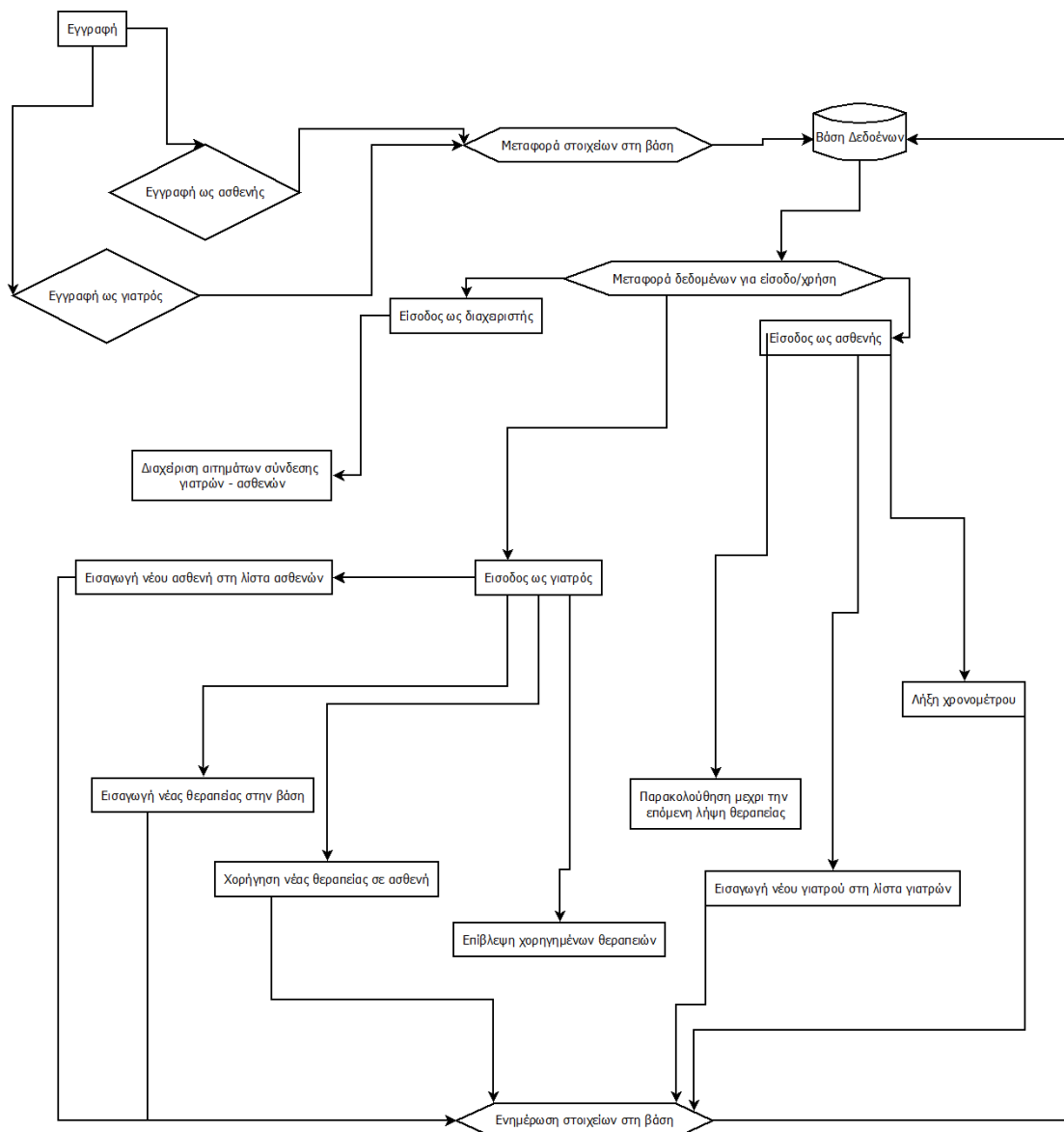
Διάγραμμα 3 Διάγραμμα ροής δεδομένων επιπέδου ένα.

Στην βάση εισάγονται οι χρήστες, οι θεραπείες καθώς και οι φαρμακευτικές αγωγές που χρησιμοποιεί κάθε γιατρός. Η πορεία που θα ακολουθήσουν τα δεδομένα, εμφανίζεται στο διάγραμμα ροής (διάγραμμα 2). Τα δεδομένα θα εισαχθούν στην βάση κατά την εγγραφή και στην συνέχεια θα ανακτηθούν από αυτήν για την είσοδο του χρήστη και τις διεργασίες που θα πραγματοποιήσει αυτός. Κάθε φορά που ο χρήστης επεξεργάζεται δεδομένα, γίνεται ενημέρωσή τους στη βάση. Το διάγραμμα 2 αποτελεί διάγραμμα ροής δεδομένων επιπέδου μηδέν. Οι πορτοκαλί ελλείψεις εικονίζουν τις

διεργασίες, τα γαλάζια ορθογώνια τους χρήστες και τα πράσινα τραπεζοειδή την αποθηκευτική μονάδα (βάση δεδομένων).

Στο διάγραμμα 3 βλέπουμε το διάγραμμα ροής δεδομένων επιπέδου ένα. Στο διάγραμμα αυτό, φαίνονται αναλυτικά όλες οι δυνατότητες που έχει ο χρήστης και πως ρέουν τα δεδομένα από και προς τη βάση δεδομένων για την πραγματοποίησή τους. Όσες λειτουργίες απαιτούν δεδομένα από την βάση, έχουν βέλος που καταλήγει σε αυτές. Ενώ αντίστροφα, όσες διεργασίες ανανεώνουν ή εισάγουν δεδομένα στην βάση, έχουν βέλος που αρχίζει από αυτές και καταλήγει στην βάση.

Σε ένα διάγραμμα ροής δεδομένων φαίνονται όλοι οι χρήστες, τα αποθηκευτικά μέσα και οι διεργασίες που μπορούν να πραγματοποιήσουν οι χρήστες αυτοί. Στο διάγραμμα επιπέδου μηδέν αναπαρίστανται τα παραπάνω με απλοϊκό τρόπο. Εμφανίζονται οι στοιχειώδεις διεργασίες και σχέσεις. Σε κάθε διάγραμμα ανωτέρου επιπέδου αυξάνει η ανάλυση των σχέσεων μεταξύ των οντοτήτων του διαγράμματος, εμφανίζονται όλες οι διεργασίες και οι μεταξύ τους σχέσεις.



Διάγραμμα 4 Διάγραμμα περιγραφής χρήσης.

Τέλος, στο διάγραμμα περιγραφής χρήσης (διάγραμμα 4), μπορούμε να δούμε τα βήματα που γίνονται κατά την χρήση της μελέτης. Στο διάγραμμα αυτό αναγράφονται όλες οι διαδικασίες που μπορεί να πραγματοποιήσει κάθε χρήστης και η σχέση που έχουν με την βάση, όπως την είδαμε στα διαγράμματα ροής δεδομένων.

Στον πίνακα 1 αναφέρονται επιγραμματικά όλες οι διαδικασίες που μπορεί να πραγματοποιήσει κάθε μια από τις κατηγορίες χρηστών, που υπάρχουν στην μελέτη.

Ρόλος	Διαδικασίες
Μη Εγγεγραμμένος Χρήστης	- Εγγραφή στο σύστημα
Γιατρός – Φροντιστής	- Επίβλεψη χορηγημένων θεραπειών. - Χορήγηση νέας θεραπείας σε ασθενή. - Εισαγωγή νέας θεραπείας στο σύστημα. - Εισαγωγή νέου ασθενή στη λίστα ασθενών του. - Επίβλεψη στοιχείων εγγραφής στο σύστημα.
Ασθενής	- Παρακολούθηση θεραπειών του. - Επίβλεψη της λίστας των θεραπόντων ιατρών του. - Εισαγωγή νέου γιατρού στην λίστα ιατρών του. - Επίβλεψη των στοιχείων εγγραφής του.
Διαχειριστής	- Επιβεβαίωση αιτημάτων από γιατρούς και ασθενείς - Εισαγωγή έγκυρων αιτημάτων στην βάση.

Πίνακας 1 Ρόλοι των χρηστών στη μελέτη και διαδικασίες που πραγματοποιούν.

Με βάση τα παραπάνω και χρησιμοποιώντας τις τεχνολογίες που περιεγράφηκαν στο προηγούμενο υπό-κεφάλαιο, πραγματοποιήθηκε η μελέτη υπενθύμισης θεραπείας για αυτή τη διπλωματική.

3.2.1 Αναλυτική Περιγραφή Κώδικα

3.2.1.1 Βάση Δεδομένων

Η βάση δεδομένων δημιουργήθηκε με το ερώτημα που υπάρχει στις γραμμές 25-27, στο αρχείο `secure_login.sql`. Με αυτόν τον κώδικα δημιουργείται μια βάση στον τομέα `localhost` με όνομα `secure_login`, χρήστη `sec_user` και κωδικό `1q2w3e4r5t`.

Στην συνέχεια δημιουργήθηκαν οι πίνακες της βάσης. Με τον κώδικα στις γραμμές 60-68 του ίδιου αρχείου γίνεται η δημιουργία του πίνακα `Drugs`. Ο πίνακας αυτός, αποθηκεύει τις φαρμακευτικές αγωγές που οι θεράποντες ιατροί έχουν εισάγει για να χορηγούν στους ασθενείς τους. Έχει τα πεδία:

- `Name`- Περιέχει το όνομα του φαρμάκου, πρωτεύον κλειδί.
- `Id`- Αύξοντας αριθμός μοναδικός για κάθε εγγραφή, πρωτεύον κλειδί.

- Substance- Αποθηκεύει την θεραπευτική ουσία που περιέχει το φάρμακο.
- Content- Αποθηκεύει την ποσότητα της ενεργού ουσίας σε mg.
- Description- Μια σύντομη περιγραφή του φαρμάκου.
- DUsername- Το username του γιατρού που εγγράφει το φάρμακο αυτό και θα μπορεί να το χορηγεί στους ασθενείς του.

Στις γραμμές 92-96 δημιουργείται ο πίνακας login_attempts, που περιέχει τις αποτυχημένες προσπάθειες εισόδου στο λογαριασμό του χρήστη και από ποιον λογαριασμό έγιναν. Έχει τα πεδία:

- user_id- Αποθηκεύει το id του χρήστη που έκανε την αποτυχημένη προσπάθεια εισόδου
- Time- Κρατάει την χρονοσφραγίδα της αποτυχημένης εισόδου.

Η δημιουργία του πίνακα Members γίνεται στις γραμμές 124-133 στο αρχείο secure_login.sql. Σε αυτόν τον πίνακα αποθηκεύονται όλοι οι χρήστες και τα στοιχεία εγγραφής τους. Έχει τα πεδία:

- Id- Αύξοντας αριθμός, μοναδικός για κάθε εγγραφή, πρωτεύον κλειδί.
- Username- Το όνομα χρήστη που επιλέγει ο χρήστης όταν εγγράφεται.
- Email- Το email που δίνει ο χρήστης όταν εγγράφεται.
- Telephone- Το τηλέφωνο που θέλει ο χρήστης να λαμβάνει την υπενθύμιση.
- Capacity- Το πεδίο αυτό παίρνει τις τιμές admin, doctor, patient και καθορίζει την ιδιότητα του κάθε χρήστη.
- Password- Περιέχει τον κωδικό πρόσβασης που έχει δώσει ο χρήστης κατά την εγγραφή του περασμένο από μια hash συνάρτηση.
- Salt- Περιέχει το γέμισμα του κωδικού, που παράγει η hash συνάρτηση, όταν κωδικοποιεί τον κωδικό πρόσβασης του χρήστη.

Η δημιουργία του πίνακα Prescriptions, που περιέχει τις φαρμακευτικές θεραπείες που έχει εγγράψει ο κάθε γιατρός σε κάθε ασθενή, γίνεται στις γραμμές 240-248 του ίδιου αρχείου. Έχει τα πεδία:

- Id- Αύξοντας αριθμός μοναδικός για κάθε εγγραφή, πρωτεύον κλειδί.
- Doctor- Το username του γιατρού που έχει κάνει την συνταγογράφηση.
- Drug- Το μοναδικό id του φαρμάκου που χορηγείται στον ασθενή.
- Patient- Το username του ασθενή που λαμβάνει την θεραπεία.
- Dosage- Την απόσταση που θα έχουν δύο διαδοχικές λήψεις της αγωγής, σε ώρες.
- LastTaken- Τη χρονοσφραγίδα της τελευταίας λήψης της θεραπείας. Κατά την πρώτη χορήγηση της θεραπείας η χρονοσφραγίδα ρυθμίζεται ως η χρονοσφραγίδα του συστήματος την συγκεκριμένη στιγμή της χορήγησης.

Στις γραμμές 272-276 δημιουργείται ο πίνακας Treated, που περιλαμβάνει την σχέση ασθενή – γιατρού. Έχει τα πεδία:

- PUsername- Το username του ασθενή που μετέχει στη σχέση, πρωτεύον κλειδί.
- DUsername- Το username του γιατρού που μετέχει στην σχέση, πρωτεύον κλειδί.

Στο αρχείο `secure_login.sql` στις γραμμές 35-40 γίνεται η δημιουργία του πίνακα `admin_temp`. Για να εισαχθεί μια έγγραφη στον πίνακα “`treated`”, θα πρέπει τόσο ο γιατρός όσο και ο ασθενής να αιτηθούν την ύπαρξή της. Στον πίνακα `Admin_temp` εισάγονται όλες οι αιτήσεις. Στην συνέχεια, ελέγχονται από τον διαχειριστή του συστήματος και σε περίπτωση που έχει γίνει αίτηση και από τα 2 μέρη εισάγεται στο σύστημα. Έχει τα πεδία:

- DUsername- Το όνομα του γιατρού που θα μετέχει στην σχέση ή έχει αιτηθεί την σχέση.
- PUsername- Το όνομα του ασθενή που θα λαμβάνει μέρος στην αίτηση ή έχει αιτηθεί την σχέση.
- SubmittedBy- Το πεδίο αυτό μπορεί να λάβει τις τιμές `patient`, στην περίπτωση που την αίτηση έχει κάνει ο εμπλεκόμενος ασθενής, ή `doctor`, στην περίπτωση που την αίτηση έχει κάνει ο εμπλεκόμενος γιατρός.

3.2.1.2 Σύνδεση με την Βάση Δεδομένων

Η σύνδεση με την βάση δεδομένων γίνεται μέσω των αρχείων `db_connect.php` και `pls-config.php`. Το πρώτο είναι το αρχείο που εισάγεται σε κάθε αρχείο που θέλει να προσπελάσει την βάση, με την γραμμή:

```
include_once 'includes/db_connect.php';
```

Ο κώδικας που περιέχει κάνει την εισαγωγή των δεδομένων που είναι αποθηκευμένα στο δεύτερο αρχείο και τα περνά ως παραμέτρους στη συνάρτηση της `php`, `mysqli`.

Το αρχείο `pls-config.php` περιλαμβάνει τα στοιχεία της βάσης και του χρήστη με τον οποίο κάνουμε είσοδο σε αυτήν και τα μετατρέπει σε παραμέτρους, ώστε να κάνει τη μελέτη πιο ασφαλή.

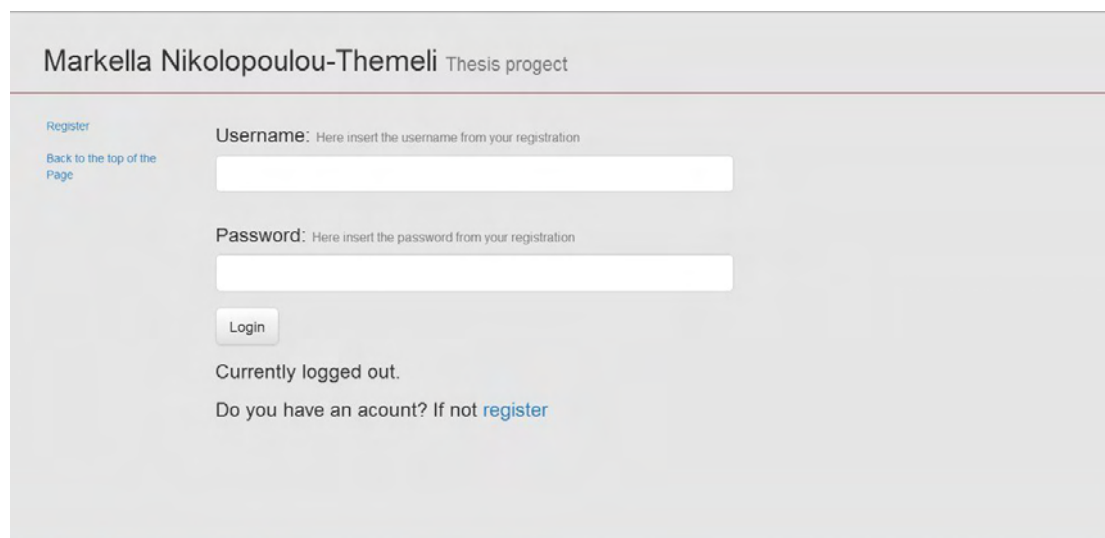
3.2.1.3 Αρχική Σελίδα

Η αρχική σελίδα βρίσκεται στο αρχείο με όνομα `index.php`. Σε αυτή θεωρήθηκε καλύτερο να περιέχεται η είσοδος χρήστη και όχι κάποια περιγραφή του site.

Στην εικόνα 1 μπορούμε να δούμε την αρχική σελίδα της μελέτης. Τα μικρότερα κείμενα δίπλα στα πεδία, όπως και όλα τα πεδία σε όλες τις σελίδες, είναι επεξηγήσεις για να ικανοποιηθεί ο κανόνας προσβασιμότητας που απαιτεί να υπάρχει επεξήγηση δίπλα σε κάθε πεδίο φόρμας. Σε αυτό το σημείο ο χρήστης εισάγει το `username` και το `password` με τα οποία έκανε την έγγραφη του, για να εισέλθει στο σύστημα.

Η λειτουργικότητα της σελίδας αυτής περιέχεται στο αρχείο `process_login.php` μέσω της συνάρτησης `login` του αρχείου `functions.php`. Και τα δύο αρχεία έχουν

αποθηκευτεί στον φάκελο `includes` για λόγους ασφαλείας. Το αρχείο `functions` ενσωματώνεται στην σελίδα με την γραμμή 3 του κώδικα.



Εικόνα 1

Το αρχείο `process_login.php` λαμβάνει τα δεδομένα που έχει εισάγει ο χρήστης στην φόρμα μέσω της συνάρτησης JavaScript `formhash` που βρίσκεται στο αρχείο `forms.js` μέσα στον φάκελο `js`. Η συνάρτηση αυτή ελέγχει και κωδικοποιεί τον κωδικό πρόσβασης που έχει δώσει ο χρήστης και τον στέλνει στην σελίδα `functions` όπου και γίνεται η είσοδος του χρήστη στο λογαριασμό του μέσω της `php` συνάρτησης `login`. Η εισαγωγή του αρχείου αυτού στην σελίδα έγινε με την γραμμή 17 του κώδικα της σελίδας `index.php`, μεταξύ των ετικετών `<head>` της σελίδας.

Αρχικά η συνάρτηση `formhash` λαμβάνει τα δεδομένα μέσω του κώδικα στις γραμμές 79-81. Αυτή δημιουργεί ένα κενό στοιχείο `element` για να αποκρύψει μέσα του τον κωδικό πρόσβασης στη γραμμή 4 του αρχείου `forms.js`, και στέλνει τα δεδομένα για επεξεργασία στη γραμμή 16 του ίδιου αρχείου.

Τα δεδομένα αυτά δέχεται και ελέγχει το αρχείο `process_login.php`. Σε αυτό το αρχείο υπάρχει ο έλεγχος της ύπαρξης του χρήστη με τα δοθέντα στοιχεία. Ο έλεγχος αυτός γίνεται με τις γραμμές 8-26 του αρχείου αυτού. Στην συνέχεια του αρχείου αυτού υπάρχει κώδικας για τον έλεγχο της ιδιότητας του χρήστη και την μετάβασή του στην σωστή σελίδα, γραμμές 28-42. Στην περίπτωση που έχει γίνει κάποιο λάθος μέσω του κώδικα οι γραμμές 45-52 του κώδικα διασφαλίζουν πως εμφανίζεται το αντίστοιχο μήνυμα λάθους.

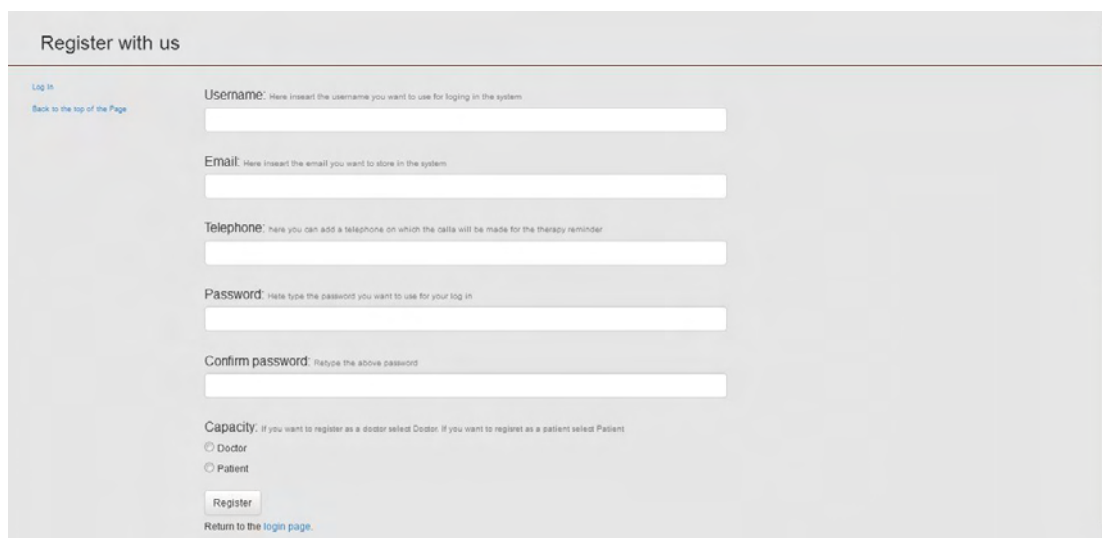
Στο αρχείο `functions.php` υπάρχουν και άλλες συναρτήσεις, που ελέγχουν εάν έχουν προσπαθήσει να προσπελάσουν τον λογαριασμό με `brute force`, ελέγχουν εάν έχει γίνει επιτυχώς το `login`, διαφορετικά εκτυπώνουν το λάθος, αρχικοποιούν την σύνοδο και κωδικοποιούν το `url` για την ασφαλέστερη μεταφορά των κωδικών μεταξύ των διαφόρων σελίδων.

Σε περίπτωση εισαγωγής λανθασμένων στοιχείων, μέσω του αρχείου `error.php` πραγματοποιείται εμφάνιση του αντίστοιχου μηνύματος λάθους. Στην συνέχεια επιτρέπεται στον χρήστη να ξαναδοκιμάσει να εισάγει τα στοιχεία του.

3.2.1.4 Σελίδα Εγγραφής

Όπως βλέπουμε στην αρχική σελίδα υπάρχει ένα link, για μετάβαση σε σελίδα εγγραφής. Ο κώδικας αυτής βρίσκεται στο αρχείο register.php.

Στην εικόνα 2 βλέπουμε την φόρμα εγγραφής χρήστη. Για την λήψη της συγκεκριμένης εικόνας έχει γίνει σμίκρυνση ώστε να χωρέσει όλη η φόρμα σε μία



The image shows a web form titled "Register with us". It contains the following elements:

- Log in** link and [Back to the top of the Page](#) link.
- Username:** "Here insert the username you want to use for logging in the system" with an input field.
- Email:** "Here insert the email you want to store in the system" with an input field.
- Telephone:** "Here you can add a telephone on which the calls will be made for the therapy reminder" with an input field.
- Password:** "Here type the password you want to use for your log in" with an input field.
- Confirm password:** "Retype the above password" with an input field.
- Capacity:** "If you want to register as a doctor select Doctor. If you want to register as a patient select Patient". It has two radio buttons: Doctor and Patient.
- Register** button.
- [Return to the login page.](#) link.

Εικόνα 2

σελίδα. Με αυτόν τον τρόπο επίσης φαίνεται πως η σελίδα δεν παραμορφώνεται κατά την σμίκρυνση ή μεγέθυνση. Αυτό ικανοποιεί την συνθήκη προσβασιμότητας για μη παραμόρφωση του περιεχομένου κατά την σμίκρυνση και μεγέθυνση. Όπως επίσης είναι ορατό στην εικόνα 2, ο χρήστης καλείται να δώσει κάποια στοιχεία για την εγγραφή του. Το username και το password είναι τα στοιχεία που θα κληθεί να δώσει κατά την είσοδό του. Επίσης θα του ζητηθεί ένα mail και ένα τηλέφωνο. Το mail χρειάζεται για να ελεγχθεί εάν υπάρχει χρήστης με το ίδιο mail στη βάση. Τέλος καλείται να επιλέξει αν θα εγγραφεί ως ασθενής ή φροντιστής-γιατρός.

Η πράξη της εγγραφής γίνεται από το αρχείο register.inc.php διαμέσου της JavaScript συνάρτησης regformhash στο αρχείο forms.js που αναφέρθηκε ανωτέρω. Η εισαγωγή του αρχείου αυτού στην σελίδα έγινε με την γραμμή 14 του κώδικα που τοποθετείται μεταξύ των ετικετών <head> της σελίδας, ενώ το register.inc.php εισάγεται με την γραμμή 2 του κώδικα. Η συνάρτηση του αρχείου register.inc.php λαμβάνει τα δεδομένα από τον regformhash αφού πρώτα αυτή ελέγξει εάν υπάρχουν όλα στην φόρμα, γραμμές 22-32 και κωδικοποιήσει τον κωδικό πρόσβασης για ασφάλεια. Η regformhash λαμβάνει τα στοιχεία της φόρμας από την γραμμή/ες 89-97 στο αρχείο register.php. Η κωδικοποίηση του κωδικού πρόσβασης γίνεται με τον ίδιο τρόπο που το κάνει και η formhash για την διαδικασία της εισόδου, με την συνάρτηση sha512 που υλοποιείται στο αρχείο sha512.js. Το αρχείο αυτό εισάγεται στην σελίδα με την γραμμή/ες 12-13 μεταξύ των ετικετών <head> της σελίδας. Στην συνέχεια, η μέθοδος του αρχείου register.inc.php, καθαρίζει τις συμβολοσειρές από μη αποδεκτούς χαρακτήρες με τις γραμμές 10-11, ελέγχει εάν υπάρχει το δοθέν username ή mail ήδη στην βάση, γραμμές 42-81, και εισάγει την εγγραφή στην βάση δεδομένων στον

πίνακα members, γραμμές 89-110. Σε περίπτωση που, υπάρχουν μη αποδεκτοί χαρακτήρες στον κωδικό, το username ή το mail υπάρχουν ήδη στην βάση, ή δεν γίνει επιτυχώς η εγγραφή, εμφανίζεται μήνυμα λάθους και επιτρέπει στον χρήστη να ξαναπροσπαθήσει να εγγραφεί.

3.2.1.5 Σελίδα Επιτυχούς Εγγραφής

Εφόσον η εγγραφή γίνει επιτυχώς, γίνεται αυτόματη μετάβαση στην σελίδα register_success.php μέσω της γραμμής 120 στο τέλος του αρχείου register.inc.php.

Registration successful!

You can now go back to the [login page](#) and log in

Εικόνα 3

Όπως φαίνεται η σελίδα αυτή (Εικόνα 3) δεν έχει υποστεί μορφοποίηση μέσω css. Εμφανίζει το μήνυμα επιτυχούς εγγραφής και προτρέπει τον χρήστη να μεταβεί στην σελίδα εισόδου.

3.2.1.6 Σελίδα Ασθενή

Ο ασθενής εισέρχεται στο λογαριασμό του με το username και το password που έχει ορίσει κατά την εγγραφή του. Το αρχείο που περιέχει τον κώδικα για την σελίδα αυτή είναι το Patient.php.

Στο επάνω μέρος εμφανίζεται ένας χαιρετισμός που τον καλωσορίζει στην σελίδα του. Στο αριστερό μέρος της σελίδας υπάρχει το μενού. Όπως παρατηρήθηκε και από τις προηγούμενες σελίδες το μενού βρίσκεται πάντα στο αριστερό μέρος της σελίδας και το κυρίως περιεχόμενο στο δεξί, κάτω από μια κεφαλίδα. Η μελέτη έχει γίνει με τέτοιο τρόπο ώστε να καλύπτει τους κανόνες προσβασιμότητας που απαιτούν οι σελίδες να έχουν κεφαλίδα και η διάταξή τους να είναι είτε στα αριστερά είτε στα δεξιά και να είναι ομοιόμορφη σε όλο το εύρος των σελίδων μιας ιστοσελίδας.

Ο χρήστης έχει μια σειρά από επιλογές που έχουν χωριστεί σε διαφορετικά sections ή σελίδες για την καλύτερη διεύθυνση του περιεχομένου σε μια λογική σειρά. Ο ασθενής μπορεί να δει την λίστα με της φαρμακευτικές του αγωγές, κάθε πόσες ώρες πρέπει να πάρει την κάθε μία, ποιος γιατρός την έχει χορηγήσει, πότε πήρε την τελευταία δόση και σε πόση ώρα θα πρέπει να πάρει την επόμενη.

Welcome Mr/Mrs markella

Therapies

Doctors

Add a new Doctor to the doctor list

Profile

Log Out

Back to the top of the Page

Therapies

In this section you can review all the prescriptions and the time left to take them

This table contains all the prescriptions, the given dosage, the prescribing doctor, the time of the last intake and the time left for the next intake

Drug	Dosage	Prescribed by:	Take again in:	Time left:
aspirin	1	drWho1	15:58:35 28-05	00:59:18
aspirin	1	drWho2	15:58:35 28-05	00:59:18
valium	3	drWho2	17:58:35 28-05	02:59:18
ponstan	3	drWho1	17:58:35 28-05	02:59:18
opatanol	7	drWho1	21:58:35 28-05	06:59:18
buscopan	1	drWho2	15:58:35 28-05	00:59:18

Εικόνα 4

Όπως φαίνεται στην εικόνα 4 τα πρώτα section της σελίδας περιέχει ένα πίνακα με όλες τις συνταγογραφήσεις που έχουν γίνει για τον συγκεκριμένο ασθενή. Ο πίνακας παρουσιάζει το όνομα της θεραπείας, την δόση σε ώρες, το username του γιατρού που την έχει χορηγήσει, την τελευταία φορά που ο ασθενής πήρε την συγκεκριμένη θεραπεία και πόση ώρα απομένει μέχρι την επόμενη προγραμματισμένη δόση. Σε αρχικά στάδια της μελέτης προβλεπόταν ο πίνακας να ανανεώνεται ανά δύο δευτερόλεπτα, για να φαίνεται η μείωση του χρόνου που απομένει μέχρι την επόμενη λήψη της θεραπείας και να αλλάζουν χρώμα οι γραμμές που οι χρόνοι τους πέφτουν κάτω από κάποιο συγκεκριμένο χρονικό όριο. Τελικά δεν έγιναν τα παραπάνω, για να συμβιβασθεί η μελέτη με δύο προϋποθέσεις προσβασιμότητας. Η μία ορίζει πως όλα τα αντικείμενα που αλλάζουν με τον χρόνο, θα πρέπει να προσφέρουν στον χρήστη την δυνατότητα να σταματήσει την εξέλιξή τους. Η δεύτερη προϋπόθεση θέτει πως θα πρέπει να υπάρχει περιγραφή κειμένου για διαφοροποιήσεις, που γίνονται με χρωματικές αλλαγές.

Κάθε φορά που γίνεται ανανέωση σελίδας, το χρονικό περιθώριο που απομένει μέχρι την επόμενη λήψη της θεραπείας, ανανεώνεται στον τρέχοντα χρόνο αναμονής, γραμμές 110-123 του αρχείου Patient.php. Όταν ο χρόνος μηδενιστεί ή γίνει μικρότερος του μηδενός, εισάγεται στην βάση δεδομένων η συγκεκριμένη χρονοσφραγίδα, τη στιγμή του μηδενισμού. Αυτό γίνεται με τον κώδικα στις γραμμές 126-140. Σε αυτές τις γραμμές ελέγχεται εάν ο υπολειπόμενος χρόνος είναι ίσος ή λιγότερος από μηδέν δευτερόλεπτα. Στην συνέχεια, με το κατάλληλο ερώτημα sql ανανεώνεται η χρονοσφραγίδα της συγκεκριμένης εγγραφής στον πίνακα της βάσης δεδομένων.

Επίσης, ο ασθενής μπορεί να δει την λίστα των γιατρών που τον έχουν υπό την επίβλεψή τους και τα στοιχεία επικοινωνίας τους.

Όπως φαίνεται από την εικόνα 5 ο πίνακας αυτού του section περιέχει τα username των θεραπόντων ιατρών και τα mail τους. Το ερώτημα που επιστρέφει τα δεδομένα προς παρουσίαση στον πίνακα είναι διατυπωμένο στις γραμμές 145-153 του αρχείου Patient.php.

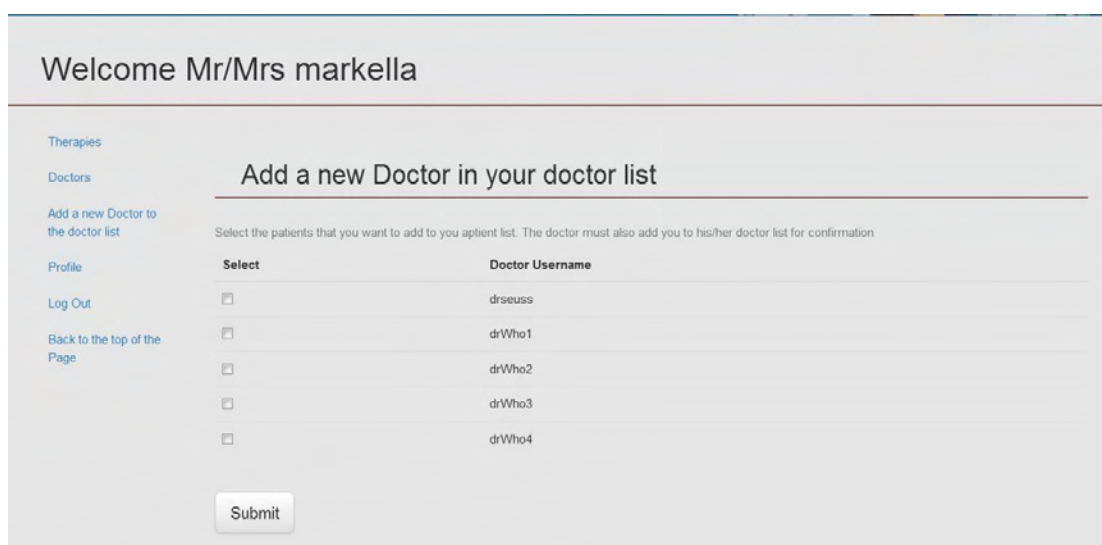


The screenshot shows a web page with the title "Doctors" and a subtitle "Here are all the doctors that have you under their care". Below the title, there is a text box stating "This table contains all the usernames and e-mail of the doctors that have you under their care". The table has two columns: "Doctor" and "e-mail".

Doctor	e-mail
drWho1	doctor@chanel.asd
drWho2	doctor2@chanel.asd

Εικόνα 5

Η επόμενη επιλογή του μενού του ασθενή, είναι η επιλογή ενός γιατρού από την λίστα των χρηστών που έχουν εγγραφεί ως γιατροί, για να εισαχθεί στην λίστα των θεραπόντων ιατρών του. Αυτό πραγματοποιείται σε διαφορετική σελίδα για λόγους χώρου. Το αρχείο αυτό είναι το add_doctor.php.



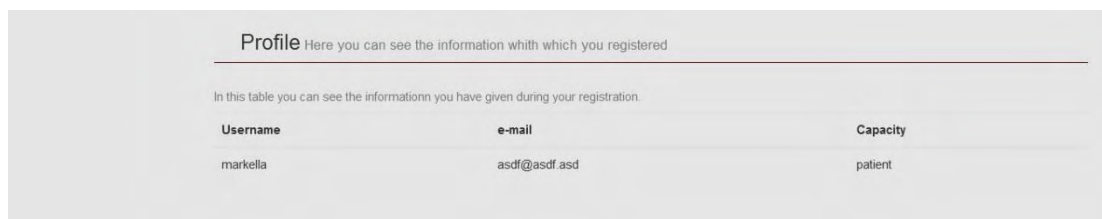
The screenshot shows a web page with the title "Welcome Mr/Mrs markella". On the left side, there is a navigation menu with links for "Therapies", "Doctors", "Add a new Doctor to the doctor list", "Profile", "Log Out", and "Back to the top of the Page". The main content area is titled "Add a new Doctor in your doctor list" and contains a form with a "Submit" button. The form has a table with two columns: "Select" and "Doctor Username".

Select	Doctor Username
<input type="checkbox"/>	drseuss
<input type="checkbox"/>	drWho1
<input type="checkbox"/>	drWho2
<input type="checkbox"/>	drWho3
<input type="checkbox"/>	drWho4

Εικόνα 6

Στην εικόνα 6 βλέπουμε την σελίδα που γίνεται το αίτημα για είσοδο νέου γιατρού στην λίστα των θεραπόντων ιατρών ενός ασθενή. Το αίτημα αυτό εισάγεται στον πίνακα admin_temp που ελέγχει ο διαχειριστής, ως μια εγγραφή που περιλαμβάνει το username του συνδεδεμένου ασθενή, το username του γιατρού που επέλεξε και την λέξη patient για να γνωρίζει ο διαχειριστής πως το αίτημα εισήλθε από την πλευρά του ασθενή, στις γραμμές 102-121 του κώδικα, στο αρχείο add_doctor.php. Αυτή τη στιγμή η εισαγωγή γίνεται για έναν μόνο γιατρό τη φορά. Τέλος, μπορεί να δει τα στοιχεία με τα οποία έχει κάνει εγγραφή.

Η παρουσίαση των δεδομένων αυτών όπως φαίνονται στην εικόνα 7 γίνεται με τον κώδικα των γραμμών 179-206.



The screenshot shows a 'Profile' section with a table of registration information. The table has three columns: Username, e-mail, and Capacity. The data row shows 'markella', 'asdf@asdf.asd', and 'patient'.

Username	e-mail	Capacity
markella	asdf@asdf.asd	patient

Εικόνα 7

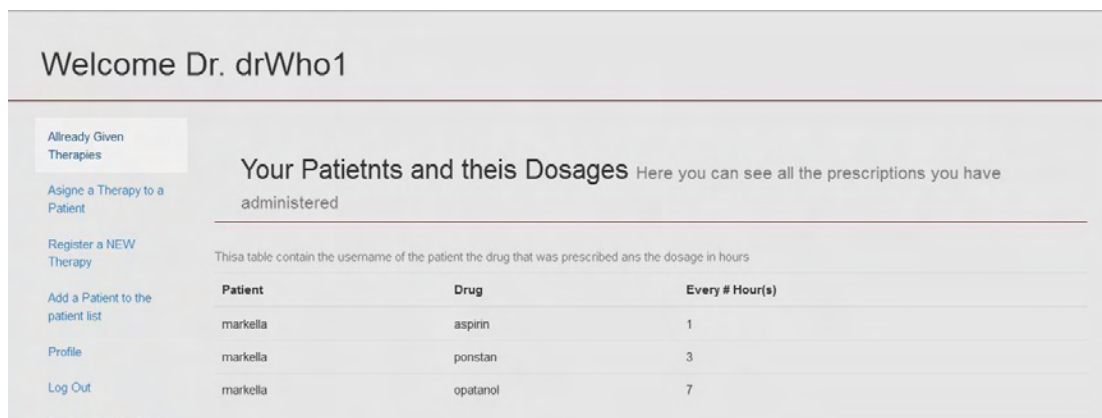
Στην σελίδα αυτή η λειτουργικότητα έχει πραγματοποιηθεί με ενσωματωμένο κώδικα PHP και ερωτήματα SQL.

3.2.1.7 Σελίδα Γιατρού

Όταν ένας χρήστης που έχει εγγραφεί ως γιατρός κάνει είσοδο στο λογαριασμό του φορτώνεται η σελίδα doctor.php. Σε αυτή τη σελίδα ο χρήστης έχει μια σειρά από δυνατότητες. Όπως και στην σελίδα του ασθενή έτσι και εδώ στο επάνω μέρος εμφανίζεται ένας χαιρετισμός που τον καλωσορίζει στην σελίδα του και στο αριστερό μέρος της σελίδας υπάρχει το μενού το οποίο περιέχει μια λίστα από επιλογές.

Στο πρώτο μέρος της σελίδα του μπορεί να δει όλες τις συνταγογραφήσεις που έχει χορηγήσει στους ασθενείς του (Εικόνα 8).

Στο τμήμα αυτό της σελίδας ο χρήστης μπορεί να δει τους ασθενείς στους οποίους



The screenshot shows a 'Welcome Dr. drWho1' header and a table titled 'Your Patients and their Dosages'. The table lists three patients with their respective drugs and dosages.

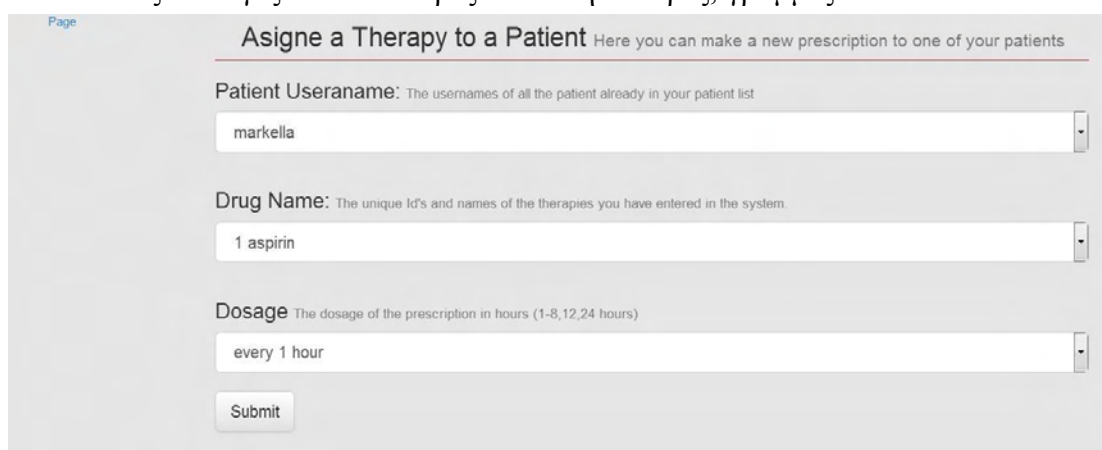
Patient	Drug	Every # Hour(s)
markella	aspirin	1
markella	ponstan	3
markella	opatanol	7

Εικόνα 8

έχει χορηγήσει θεραπεία, ποια είναι αυτή και με ποια δόσολογία. Αυτό γίνεται με τις γραμμές 76-105 του κώδικα της σελίδας αυτής.

Ακόμα μπορεί να χορηγήσει μια νέα θεραπεία σε έναν ασθενή που υπάρχει ήδη στην λίστα ασθενών του (Εικόνα 9).

Στο πρώτο πεδίο εμφανίζονται όλοι οι ασθενείς που έχει ο συγκεκριμένος γιατρός υπό την επίβλεψή του. Εάν με κάποιον ασθενή δεν έχει δημιουργήσει αυτή την σχέση δεν θα εμφανιστεί στην λίστα αυτή. Στο δεύτερο πεδίο εμφανίζονται όλες οι φαρμακευτικές αγωγές που έχει εισάγει ο γιατρός αυτός στο σύστημα. Στο τελευταίο πεδίο καλείται να εισάγει κάθε πόσες ώρες θα λαμβάνει ο ασθενής την θεραπεία αυτή. Οι δοθείσες επιλογές είναι 1-8 ώρες και 12 ή 24 ώρες, γραμμές 141-188 του κώδικα.



Page

Assign a Therapy to a Patient

Here you can make a new prescription to one of your patients

Patient Username: The usernames of all the patient already in your patient list

markella

Drug Name: The unique Id's and names of the therapies you have entered in the system.

1 aspirin

Dosage The dosage of the prescription in hours (1-8,12,24 hours)

every 1 hour

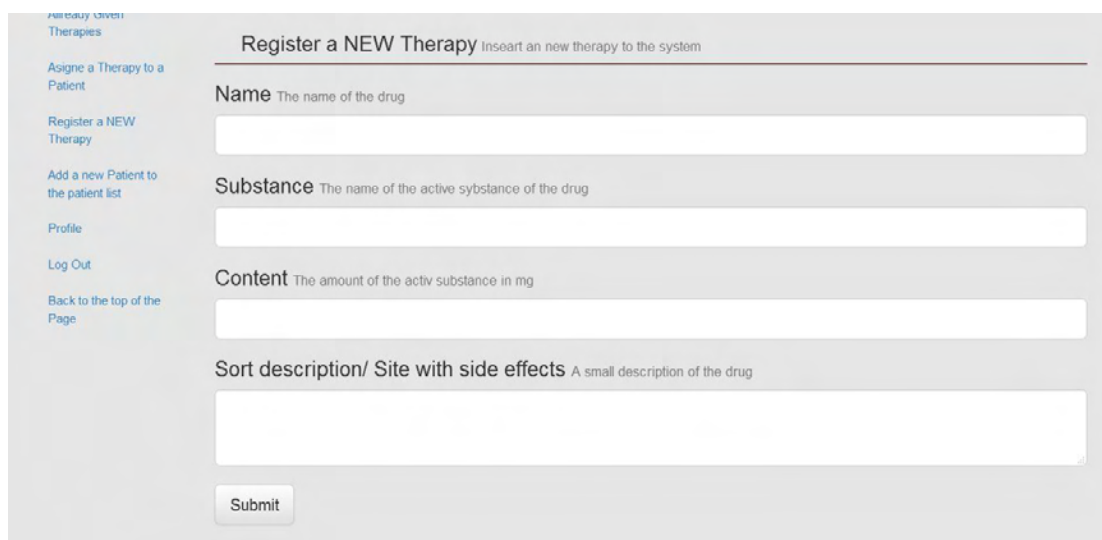
Submit

Εικόνα 9

Από την στιγμή που ο γιατρός εισάγει την θεραπεία αρχίζει αμέσως η πρώτη αντίστροφη μέτρηση. Η εισαγωγή των στοιχείων γίνεται στον πίνακα drugs της βάσης δεδομένων με τις γραμμές 189-212

Η επόμενη δυνατότητα που έχει ένας γιατρός, είναι να εισάγει μια νέα θεραπεία, που επιθυμεί να συνταγογραφεί.

Ο χρήστης καλείται να συμπληρώσει τα πεδία που φαίνονται στην εικόνα 10. Το



Therapies

Register a NEW Therapy

Inseart an new therapy to the system

Name The name of the drug

Substance The name of the active sybstance of the drug

Content The amount of the activ substance in mg

Sort description/ Site with side effects A small description of the drug

Submit

Εικόνα 10

πρώτο πεδίο ζητά να συμπληρωθεί το όνομα της θεραπείας που θα εισαχθεί στην βάση. Το δεύτερο και το τρίτο θα πρέπει να συμπληρωθούν με το όνομα της δραστικής ουσίας και την ποσότητά της σε mg στο φάρμακο αυτό αντίστοιχα. Αυτά τα δύο πεδία

συμπεριλήφθηκαν καθώς ένα φάρμακο μπορεί να έχει διαφορετικές περιεκτικότητες στην ποσότητα της δραστικής ουσίας που περιέχει. Τέλος, ο χρήστης πρέπει να συμπληρώσει μια σύντομη περιγραφή της θεραπείας. Όλα τα παραπάνω πεδία φαίνονται στις γραμμές 79-92 του αρχείου new_therapy.php. Στην συνέχεια γίνεται εισαγωγή των στοιχείων αυτών στην βάση δεδομένων, με τον κώδικα των γραμμών 93-114.

Όπως και ο ασθενής έτσι και ο γιατρός, μπορεί να επιλέξει κάποιον από την λίστα χρηστών που έχουν εγγραφεί ως ασθενείς, για να τον εισάγει στην λίστα των ασθενών του, με τον κώδικα του αρχείου add_patient.php.

Η σελίδα αυτή λειτουργεί όπως και στον ασθενή και φαίνεται στην εικόνα 11. Ο

The screenshot shows a web interface for adding a new patient to a list. The main heading is "Add a new Patent in your patient list". Below the heading, there is a sub-heading: "Select the patients that you want to add to you aptient list. The patient must also add you to his/her doctor list for confirmation".

Select	Patient Username
<input type="checkbox"/>	test_user
<input type="checkbox"/>	markella
<input type="checkbox"/>	user
<input type="checkbox"/>	sheldon
<input type="checkbox"/>	penny
<input type="checkbox"/>	army
<input type="checkbox"/>	bernadet

At the bottom of the form, there is a "Submit" button.

Εικόνα 11

γιατρός επιλέγει τον ασθενή που επιθυμεί να εισάγει στην λίστα. Το αίτημα εισάγεται στον πίνακα του διαχειριστή. Ο διαχειριστής ελέγχει τα αιτήματα και τα εισάγει στη βάση. Τα αιτήματα μπορούν να γίνονται ένα την φορά.

Οι δυο παραπάνω επιλογές έχουν τοποθετηθεί σε διαφορετικές σελίδες γιατί η html δεν μπορεί να υποστηρίξει πολλαπλές φόρμες σε μια σελίδα. Το πρόβλημα που αντιμετωπίζει είναι η ύπαρξη πολλαπλών ετικετών <form> οπότε στην περίπτωση που ζητηθεί η επικύρωση μιας από τις φόρμες αδυνατεί να επιλέξει από ποια από όλες θα πρέπει να τραβήξει τα δεδομένα.

Τέλος, ο χρήστης μπορεί να δει τα δεδομένα με τα οποία έκανε εγγραφή, στο section Profile της σελίδας του (Εικόνα 12).

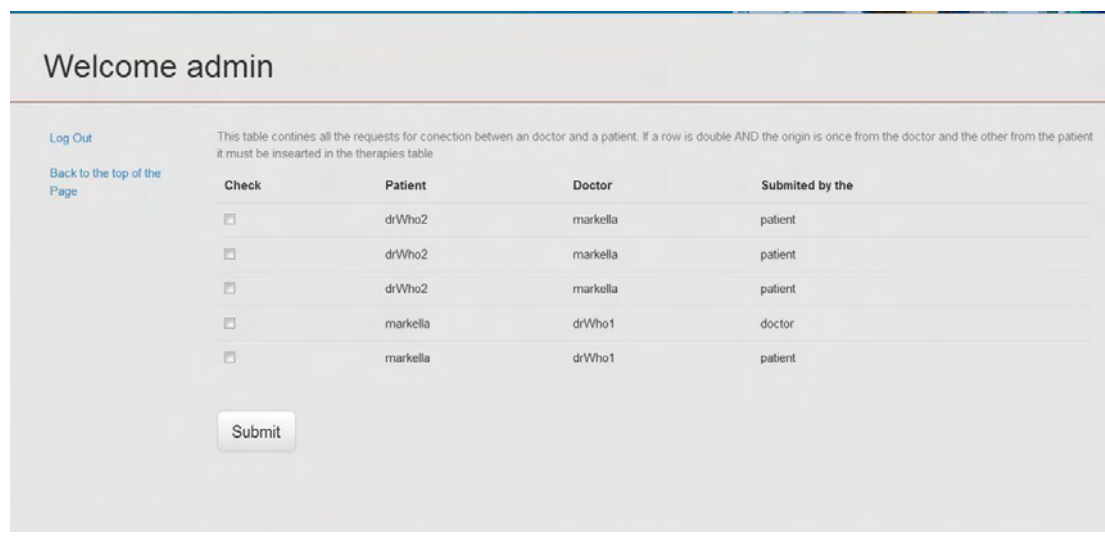
The screenshot shows the "Profile" section of a user's account. The heading is "Profile Here you can see the information you gave upon registration". Below the heading, there is a sub-heading: "Here you can see the information you gave upon registration".

Username	e-mail	Capacity
drWho1	doctor@chanel.asd	doctor

Εικόνα 12

3.2.1.8 Σελίδα Διαχειριστή

Η είσοδος στον λογαριασμό του διαχειριστή γίνεται όπως και στις άλλες δύο περιπτώσεις με το username και το password του διαχειριστή. Σε αυτή την περίπτωση όμως, φορτώνεται η σελίδα admin.php που περιλαμβάνει το περιβάλλον χρήστη του



Εικόνα 13

διαχειριστή.

Στην εικόνα 13 βλέπουμε την σελίδα του διαχειριστή. Όπως και στις προηγούμενες δύο περιπτώσεις η σελίδα ακολουθεί την ίδια διαμόρφωση με το μενού στο αριστερό μέρος και το καλωσόρισμα στην κεφαλίδα.

Στο κυρίως σώμα της σελίδας βλέπουμε τον πίνακα με τα αιτήματα από τους γιατρούς και τους ασθενείς για σύνδεση των δύο. Η εισαγωγή στον πίνακα treated από τον οποία τραβάει δεδομένα όλη η υπόλοιπη μελέτη γίνεται μόνο αν το αίτημα υπάρχει δύο φορές και η μία από τις δύο είναι καταχωρημένη από τον γιατρό ενώ η άλλη από τον ασθενή. Η παρουσίαση του πίνακα αυτού γίνεται με τον κώδικα 73-86.

Check	Patient	Doctor	Submitted by the
<input type="checkbox"/>	drWho2	markella	patient
<input type="checkbox"/>	drWho2	markella	patient
<input type="checkbox"/>	drWho2	markella	patient
<input type="checkbox"/>	markella	drWho1	doctor
<input type="checkbox"/>	markella	drWho1	patient
<input type="checkbox"/>	markella	drWho4	doctor
<input type="checkbox"/>	markella	drWho4	patient
<input type="checkbox"/>	sheldon	drseuss	doctor
<input type="checkbox"/>	sheldon	drseuss	patient
<input type="checkbox"/>	sheldon	drWho2	patient
<input type="checkbox"/>	sheldon	drWho2	patient

Εικόνα 14

Για παράδειγμα, στην συγκεκριμένη περίπτωση της εικόνας 14, τα 3 πρώτα αιτήματα είναι λανθασμένα και δεν λαμβάνονται υπόψιν. Οι γραμμές 4, 5 παρουσιάζουν ένα έγκυρο αίτημα για σύνδεση του γιατρού drWho4 με τον ασθενή markella, όπως επίσης και οι γραμμές 6, 7 και 8, 9 παρουσιάζουν έγκυρα ερωτήματα. Αυτό συμβαίνει επειδή εμφανίζεται το ίδιο ερώτημα δύο φορές και το ένα προέρχεται από τον εμπλεκόμενο ασθενή ενώ το άλλο από τον εμπλεκόμενο γιατρό. Αντίθετα οι δύο τελευταίες σειρές παρουσιάζουν ένα ερώτημα το οποίο δεν θα πρέπει να εισαχθεί στην βάση. Αυτό συμβαίνει επειδή το αίτημα να μην υπάρχει δύο φορές, αλλά και οι δύο προέρχονται από τον εμπλεκόμενο ασθενή.

3.2.1.9 Αρχείο Μορφοποίησης

Το αρχείο μορφοποίησης περιέχει όλους τους κανόνες css για την μορφοποίηση των σελίδων. Το αρχείο αυτό γράφτηκε συμπληρωματικά με τους κανόνες css που κληρονομήθηκαν όταν εφαρμόστηκε το Bootstrap. Περιλαμβάνονται στο αρχείο style.css μέσα στο φάκελο css.

Πιο συγκεκριμένα, έχει γραφεί κώδικας για την μορφοποίηση της κεφαλίδας κάθε σελίδας, και για την μορφοποίηση του γενικού σώματος των σελίδων.

Οι κανόνες που έχουν γραφεί για την μορφοποίηση του σώματος, γραμμές 2-6, διασφαλίζουν πως το φόντο δεν θα επαναληφθεί, πως θα έχει την απόχρωση του γκρι με δεκαεξαδικό κωδικό #E5E5E5 και πως το κάτω του όριο θα έχει περιθώριο 30 πίξελ από το επόμενο στοιχείο για την περίπτωση που υπάρχει κάποιο υποσέλιδο.

Οι κανόνες που έχουν γραφεί για την μορφοποίηση της κεφαλίδας βρίσκονται στις γραμμές 8-11. Η μορφοποίηση αυτή σημαίνει πως η κεφαλίδα θα έχει ως κάτω όριο μια γραμμή πάχους τριών πίξελ και χρώματος με δεκαεξαδικό κωδικό #40000 και πως θα αφήνει περιθώριο από το αριστερό όριο της οθόνης, το 3% αυτής σε όλες τις συσκευές.

Οι κανόνες αυτοί έχουν γραφτεί με τέτοιο τρόπο ώστε να παρακάμπτουν τους αρχικούς κανόνες του Bootstrap.

Το αρχείο αυτό ενσωματώθηκε σε όλες τις σελίδες με την εισαγωγή της γραμμής

```
<link href="css/style.css" rel="stylesheet">
```

μεταξύ των ετικετών <head>.

3.2.1.10 Έξοδος από το σύστημα

Η έξοδος γίνεται μέσω της σελίδας logout.php που καλείται κάθε φορά που πατιέται η επιλογή Log out. Η πράξη της εξόδου γίνεται με τον κώδικα της μεθόδου που είναι αποθηκευμένη στο αρχείο logout.php.

3.3 Συνοπτική Περιγραφή Χρήσης Ιστοσελίδας

3.3.1 Χρήση ως Μη Εγγεγραμμένος Χρήστης

Στην περίπτωση που εισέλθει στην σελίδα ένας μη εγγεγραμμένος χρήστης, έχει την δυνατότητα να μεταβεί στην σελίδα “register” και να κάνει εγγραφή στο σύστημα. Εκεί θα κληθεί να δώσει κάποια στοιχεία του για την εγγραφή. Αυτά είναι:

- Username- Το username που επιθυμεί να εισάγει για την εγγραφή του και την είσοδό του.
- Email- Το email που επιθυμεί να εισάγει για την εγγραφή του και την είσοδό του.
- Password- Τον κωδικό πρόσβασης που επιθυμεί να εισάγει για την εγγραφή του και την είσοδό του.
- Confirm Password- Ο χρήστης καλείται να εισάγει τον κωδικό πρόσβασής του 2 φορές για λόγους ασφαλείας.
- Capacity- Το πεδίο αυτό έχει 2 επιλογές “Doctor” και “Patient”. Οι ρόλοι αυτοί καθορίζουν την ιδιότητα και τις δυνατότητες που θα έχει ο χρήστης στη μελέτη.

3.3.2 Χρήση ως Γιατρός - Φροντιστής

Ως «Φροντιστής» ορίζεται ο χρήστης που έχει υπό την περίθαλψή του το λιγότερο έναν ασθενή. Αυτή την ιδιότητα την αποκτά ο χρήστης όταν κατά την εγγραφή επιλέξει το “Doctor” πεδίο στην επιλογή “Capacity”.

Ο φροντιστής εισέρχεται στο λογαριασμό του με το username και το password που έχει ορίσει κατά την εγγραφή του. Στην σελίδα που φορτώνει στην συνέχεια υπάρχει μια λίστα από επιλογές.

- Μπορεί να δει την λίστα των θεραπειών που έχει αναθέσει στους ασθενείς του.
- Μπορεί να χορηγήσει μια νέα θεραπεία, από αυτές που ο ίδιος έχει εισάγει στο σύστημα, σε έναν από τους ασθενείς της λίστας ασθενών του.
- Μπορεί να εισάγει μια νέα θεραπεία.
- Μπορεί επιλέξει ένα νέο ασθενή από την λίστα των χρηστών που έχουν εγγραφεί ως ασθενείς .
- Μπορεί να δει τα στοιχεία με τα οποία έχει εγγραφεί.

Σε αυτό το σημείο πρέπει να σημειώσουμε, πως από την στιγμή που ένας φροντιστής επιλέξει έναν ασθενή για εισαγωγή στη λίστα ασθενών του, η εγγραφή αυτή εισέρχεται σε έναν πίνακα που θα πρέπει να ελεγχθεί από τον διαχειριστή. Στον ίδιο πίνακα εισέρχονται και οι αιτήσεις που κάνει κάθε ασθενής για την εισαγωγή ενός φροντιστή στους θεράποντες του. Στην συνέχεια ο διαχειριστής ελέγχει εάν μια αίτηση έχει γίνει και από τα 2 μέλη και εάν έχει γίνει, την εισάγει στη βάση δεδομένων.

3.3.3 Χρήση ως Ασθενής

Ως «Ασθενής» ορίζεται αυτός που είναι υπό την επίβλεψη τουλάχιστον ενός φροντιστή. Αυτή την ιδιότητα την αποκτά ο χρήστης όταν κατά την εγγραφή επιλέξει το “Patient” πεδίο στην επιλογή “Capacity”.

Ο ασθενής εισέρχεται στο λογαριασμό του με το username και το password που έχει ορίσει κατά την εγγραφή του. Στην σελίδα που φορτώνει στην συνέχεια υπάρχει μια λίστα από επιλογές.

- Μπορεί να δει την λίστα με της φαρμακευτικές του αγωγές, κάθε πόσες ώρες πρέπει να πάρει την κάθε μία, ποιος γιατρός την έχει χορηγήσει, πότε πήρε την τελευταία δόση και σε πόση ώρα θα πρέπει να πάρει την επόμενη.
- Μπορεί να δει την λίστα των γιατρών που τον έχουν υπό την επίβλεψή τους και τα στοιχεία επικοινωνίας τους.
- Μπορεί να επιλέξει έναν νέο γιατρό από την λίστα των χρηστών που έχουν εγγραφεί ως γιατροί, για να εισαχθεί στην λίστα των θεραπόντων ιατρών του.
- Επίσης μπορεί να δει τα στοιχεία με τα οποία έχει κάνει εγγραφή.

3.3.4 Χρήση ως Διαχειριστής

Ως «Διαχειριστής» ορίζεται ο χρήστης που διαχειρίζεται την μελέτη. Η ιδιότητα «Διαχειριστής» έχει μπει χειροκίνητα σε μια εγγραφή του πίνακα χρηστών και δεν μπορεί να την επιλέξει ένας χρήστης την στιγμή που εγγράφεται στο σύστημα. Η μόνη ιδιότητα του διαχειριστή στην μέχρι τώρα μελέτη, είναι να ελέγχει την λίστα των αιτήσεων που έχουν γίνει από τους υπόλοιπους χρήστες για σύνδεση γιατρού-φροντιστή. Στην περίπτωση που έχει γίνει αίτηση και από τα δύο μέρη η εγγραφή εισάγεται στον πίνακα “treated” της βάσης δεδομένων.

3.3.5 Έξοδος Χρήστη από το σύστημα

Ο χρήστης μπορεί να κάνει έξοδο από τον λογαριασμό του, από όλες τις σελίδες πατώντας την επιλογή log out που υπάρχει σαν προτελευταία επιλογή στο μενού των σελίδων του χρήστη, αλλά και σε περίπτωση που βρεθεί στην αρχική σελίδα μπορεί να αποσυνδεθεί πατώντας το log out, κάτω από τα πεδία εισόδου. Η επιλογή αυτή εμφανίζεται στο σημείο που κανονικά αναγράφεται το register, όταν ένας συνδεδεμένος χρήστης βρεθεί στην αρχική σελίδα.

3.4 Ασφάλεια

Μια μελέτη που περιέχει και διαχειρίζεται προσωπικά και ευαίσθητα δεδομένα όπως αυτή που προτάθηκε σε αυτή τη διπλωματική, πρέπει να περιέχει αυξημένο βαθμό ασφαλείας για την διαφύλαξή τους. Ιδιαίτερα τα ιατρικά δεδομένα αποτελούν μια από τις πλέον ευαίσθητες κατηγορίες προσωπικών δεδομένων. Γι’ αυτό το λόγο όλη η μελέτη σχεδιάστηκε με μεγάλη προσοχή όσον αφορά την προστασία αυτών των δεδομένων.

Αρχικά, θα πρέπει να αναφερθεί πως όλα τα αρχεία που διαχειρίζονται ή προσπελούν κωδικούς ή μεθόδους κρυπτογράφησης είναι αποθηκευμένα σε ειδικό φάκελο ξεχωριστά από τα υπόλοιπα αρχεία. Σελίδες που χρειάζεται να προσπελάσουν κωδικούς πρόσβασης θα πρέπει να καλέσουν ένα από τα προαναφερθέντα αρχεία. Κατά την δημιουργία του κώδικα πρόσβασης στη βάση δεδομένων, δημιουργήθηκαν ειδικές συναρτήσεις που την προστατεύουν από sql injection επιθέσεις. Για να συμβεί αυτό, όλες οι μεταβλητές περνούν από την μια συνάρτηση στην άλλη και από το ένα αρχείο στο άλλο με την μορφή παραμέτρων, ώστε να μην μπορούν να τροποποιηθούν από κάποιον επιτιθέμενο.

Για την διασφάλιση της αποτυχίας οποιασδήποτε επίθεσης εγκαθιδρύθηκε SSL συνεδρία. Δημιουργήθηκε self signed πιστοποιητικό και κλειδί και εγκαθιδρύθηκε εικονικός υποδοχέας στην κρυπτογραφημένη πύλη 443 της μελέτης. Ένας εικονικός διακομιστής είναι μια υποδομή λογισμικού που προσφέρει την δυνατότητα στον χρήστη να φιλοξενεί πολλαπλές διευθύνσεις ιστοσελίδων σε έναν σέρβερ. Κάθε διεύθυνση είναι ανεξάρτητη από τις υπόλοιπες και έχει τα δικά της χαρακτηριστικά και ιδιότητες. Η πύλη 443, η οποία προσφέρεται από πρωτόκολλο επικοινωνίας TCP-IP και χρησιμοποιήθηκε στη μελέτη για την φιλοξενία της ιστοσελίδας που προτάθηκε, είναι η πύλη που επιτρέπει την ασφαλή περιήγηση στο περιεχόμενο του σέρβερ. Ένα αυτόπογεγραμμένο πιστοποιητικό είναι ένα πιστοποιητικό που δημιουργείται από την αρχή που πιστοποιεί. Για την κωδικοποίηση των κωδικών πρόσβασης χρησιμοποιήθηκε η συνάρτηση hash sha512 (Secure Hash Algorithm) που υπάρχει στο αρχείο sha512.js. Ο αλγόριθμος αυτός ανήκει στην οικογένεια αλγορίθμων sha-2, και είναι πανομοιότυπος με τον sha256, με μόνη διαφοροποίηση στον αριθμό των γύρων κωδικοποίησης. Μετά την χρήση μιας συνάρτησης hash δύο διαφορετικές εισοδοί θα πρέπει πάντα να έχουν διαφορετικές εξόδους. Για την επιβεβαίωση της πιστότητας του αποτελέσματος της συνάρτησης γίνεται σύγκριση της τιμής του με μια προκαθορισμένη αναμενόμενη τιμή εξόδου. Η sha512 δημιουργεί μια σύνοψη μήκους 512 bits της εισόδου που της δίνεται, πραγματοποιώντας 80 γύρους κωδικοποίησης με τις πράξεις And, Xor, Rot, Add (mod 264), Or, Shr. [23]

Κεφάλαιο 4. Διαδικτυακές Υπηρεσίες Ανοικτού Κώδικα για Εφαρμογές Ειδοποίησης/Υπενθύμισης

Σε αυτό το κεφάλαιο θα γίνει μια παράθεση υπηρεσιών ανοιχτού κώδικα που προσφέρουν δυνατότητες πραγματοποίησης τηλεφωνικών κλήσεων και άλλων τηλεφωνικών υπηρεσιών μέσω διαδικτύου. Τέλος, παρατίθεται ένας πίνακας με τις δυνατότητες επικοινωνίας που προσφέρει η κάθε υπηρεσία και την πιθανότητα χρήση τους στη μελέτη.

4.1 Λογισμικό Ανοικτού Κώδικα (Open Source Code)

Ως ανοιχτός κώδικας αναφέρεται κάθε υπολογιστικό πρόγραμμα του οποίου ο πηγαίος κώδικας είναι ελεύθερος για χρήση και επεξεργασία από την κοινότητα των προγραμματιστών. Οι προγραμματιστές που συνεισφέρουν στην δημιουργία ή επεξεργασία του κώδικα αυτού ονομάζονται συνεισφέροντες (υποστηρικτές), και όσοι απλά χρησιμοποιούν τον κώδικα αυτό ονομάζονται μη συνεισφέροντες. [24]

Λογισμικό ανοιχτού κώδικα είναι οποιοδήποτε πρόγραμμα του οποίου ο πηγαίος κώδικας διατίθεται ελεύθερα στο διαδίκτυο, χωρίς όμως αυτό να σημαίνει απαραίτητα πως και το ίδιο το πρόγραμμα είναι ελεύθερο ή δωρεάν. [25]

4.2 Server-side λογισμικό για υποστήριξη εφαρμογών επικοινωνίας/υπενθύμισης

4.2.1 AskoziaPBX

Το AskoziaPBX (private branch exchange) είναι το κύριο προϊόν της εταιρίας Askozia. Είναι ένα λογισμικό ανοιχτού κώδικα, που βασίζεται στις τεχνολογίες m0n0wall και Asterisk. Διατίθεται υπό μια αυστηρή BSD (Berkeley Software Distribution) η οποία επιτρέπει την τροποποίηση του πηγαίου κώδικα του προϊόντος αλλά απαιτεί αδειδότηση για να μεταπωληθεί, και οι τελευταίες εκδόσεις του διατίθενται μόνο υπό εμπορική άδεια. Από την εταιρία προσφέρεται μια ελαφριά έκδοση, για την πραγματοποίηση ελέγχων, που διαθέτει όλα τα προϊόντα, αλλά περιορίζεται στις 2 μόνο ταυτόχρονες κλήσεις. Το AskoziaPBX υποστηρίζει τεχνολογίες που επιτρέπουν σε γραμμές αναλογικού τηλεφώνου να πραγματοποιούν κλήσεις μέσω του διαδικτύου. Επίσης προσφέρονται δυνατότητες τηλεφωνητή, τηλεφωνικής συνδιάσκεψης, συνέχισης κλήσης από άλλη συσκευή, προώθησης κλήσεων, μετατροπής fax σε ηλεκτρονικό μήνυμα και μετατροπής φωνής σε κείμενο. Τέλος, τα μηνύματα και οι υπηρεσίες είναι πλήρως διαθέσιμες στα Αγγλικά, Γερμανικά, Ιταλικά, Ισπανικά, Γαλλικά, Ολλανδικά και μερικώς διαθέσιμες στα Κινέζικα, Ιαπωνικά, Πολωνικά, Δανέζικα και Σουηδικά. [26]

4.2.2 AsteriskPBX

Το AsteriskPBX είναι λογισμικό, που επιτρέπει στα συνδεδεμένα τηλέφωνα να πραγματοποιήσουν μεταξύ τους κλήσεις. Προσφέρει επίσης και άλλες υπηρεσίες όπως τηλεφωνητή, τηλεφωνική συνδιάσκεψη, δυναμικές τηλεφωνικές απαντήσεις (τηλεφωνικό μενού) και αυτόματη διανομή κλήσεων. Οι χρήστες μπορούν να δημιουργήσουν μόνοι τους κώδικα σε C ή με το πρωτόκολλο που προσφέρει το ίδιο το Asterisk. Προσφέρει διασύνδεση τηλεφώνων μέσω του διαδικτύου, που μπορεί να προεκταθεί στην σύνδεση μεταξύ ηλεκτρονικών τηλεφώνων και του δημόσιου τηλεφωνικού δικτύου μεταγωγής. Προσφέρεται με αδειοδότηση ελεύθερου κώδικα και είναι διαθέσιμο παγκοσμίως, υποστηρίζοντας Αμερικάνικα Αγγλικά, Γαλλικά, Περσικά (Φαρσί) και Μεξικάνικα Ισπανικά, επίσης προσφέρει Αυστραλιανά Αγγλικά για μερικές από τις λειτουργίες του. Οι γλώσσες εμπλουτίζονται με την κατάθεση ενημερώσεων από τους προγραμματιστές της κοινότητας ανοιχτού κώδικα. Έχει χρησιμοποιηθεί για την δημιουργία πολλών εμπορικών προϊόντων για πολλές εταιρίες. [27]

4.2.3 Elastix

Είναι λογισμικό που τρέχει στον σέρβερ του συστήματος που επιθυμεί να το χρησιμοποιήσει και προσφέρει κλήσεις μεταξύ ηλεκτρονικών τηλεφώνων, αποστολή email, αποστολή σύντομων ηλεκτρονικών μηνυμάτων και δυνατότητα συνδιάσκεψης. Η δημιουργία του βασίστηκε σε άλλα προϊόντα ανοιχτού κώδικα, όπως τα Asterisk, FreePBX, HylaFAX, Openfire και Postfix που χρησιμοποιήθηκαν γιατί προσέφεραν μια από τις λειτουργίες που επιθυμούσε να ενσωματώσει το Elastix. Είναι ελεύθερο λογισμικό που προσφέρεται με γενική δημόσια αδειοδότηση GNU. Ακόμα είναι η πρώτη εταιρία που προσέφερε έξυπνη πραγματοποίηση κλήσης. Η έξυπνη πραγματοποίηση κλήσης αρχίζει να καλεί ένα πλήθος από τηλεφωνικά νούμερα και όταν μια από τις κλήσεις αυτές απαντηθεί την συνδέει με έναν από τους τηλεφωνητές που πολλές φορές αναφέρονται και ως πράκτορες. [28]

4.2.4 FreeSWITCH

Το FreeSWITCH είναι τηλεπικοινωνιακό λογισμικό ανοιχτού κώδικα που χρησιμοποιείται για την δημιουργία προϊόντων που μεταδίδουν φωνή και μηνύματα. Κυκλοφορεί με αδειοδότηση από το ίδρυμα Mozilla και με μια άδεια ελεύθερου λογισμικού. Το κύριο προϊόν του, η βιβλιοθήκη libfreeswitch, μπορεί να λειτουργήσει μόνη της όσο και ως μέρος άλλων υλοποιήσεων. Για την δημιουργία του χρησιμοποιήθηκαν μεταξύ άλλων το Apache Portable Runtime, SQLite, PCRE, Sofia-SIP, libspx, mod_spandsp, libSRTP. Το FreeSWITCH προσφέρει πλήθος δυνατοτήτων. Μερικές από τις οποίες είναι, τηλεφωνική συνδιάσκεψη, δυναμική φωνητική απάντηση, σύνθεση φωνής και αναγνώριση φωνής, πρωτόκολλα για κλήση μεταξύ γραμμών του δημόσιου τηλεφωνικού δικτύου μεταγωγής και ψηφιακών αλλά και αναλογικών γραμμών, χρήση του GoogleTalk, πρωτόκολλα για Voice over IP κλήσεις συμπεριλαμβανομένου της δημιουργίας/εγκαθίδρυσης κλήσης. Πέραν των έτοιμων εφαρμογών προσφέρεται δυνατότητα στους προγραμματιστές να

δημιουργήσουν τα δικά τους εργαλεία γράφοντας σε ένα πλήθος από γλώσσες προγραμματισμού. Πέραν της μετάδοσης ήχου το FreeSWITCH επιτρέπει την διάδοση και βίντεο. [29]

4.2.5 FreePBX

Είναι ένα γραφικό περιβάλλον χρήστη που διαχειρίζεται το AsteriskPBX. Διατίθεται με μια GNU γενική δημοσία αδειοδότηση. Ακόμη, χρησιμοποιείται σαν κεντρικό πρόγραμμα στην υλοποίηση πολλών άλλων εφαρμογών. Είναι διαθέσιμο σε Αγγλικά, Βουλγαρικά, Κινέζικα, Γερμανικά, Γαλλικά, Εβραϊκά, Ουγγρικά, Ιταλικά, Πορτογαλικά, Βραζιλιάνικα Πορτογαλικά, Ρωσικά, Σουηδικά, Ισπανικά και Ιαπωνικά. Μπορεί να εγκατασταθεί ως μέρος του προκαθορισμένου πακέτου Distro που περιλαμβάνει το λειτουργικό του συστήματος και άλλες προδιαγραφές. Οι κατασκευαστές του εκτιμούν πως έχει εγκατασταθεί 5.000.000 φορές και έχει χρησιμοποιηθεί σε 500.000 ενεργές εφαρμογές PBX. [30]

4.2.6 GNU_Gatekeeper

Είναι ένα προϊόν που χρησιμοποιεί έναν H.323 Gatekeeper για τον έλεγχο της αποδοχής κλήσεων. Όλη η υλοποίηση βασίζεται σε μια στοίβα OpenH323 ή H323Plus. Ένας ελεγκτής πύλης διασφαλίζει τόσο την μετατροπή μιας διεύθυνσης κλήσης σε IP όσο και άλλες λειτουργίες όπως εισαγωγές ελέγχου, διακίνηση κλήσεων, υπηρεσίες αυθεντικοποίησης και λογιστικών. Είναι διαθέσιμη για πλήθος λειτουργικών συστημάτων υπό μια GNU γενική δημόσια αδειοδότηση, πράγμα που διασφαλίζει την δυνατότητα των δημιουργών της να προσφέρουν πρόσβαση κατευθείαν στις βιβλιοθήκες OpenH323 και OpenSSL που χρησιμοποιεί. Προσφέρεται από το 1999 αλλά μόνο στην Αγγλική γλώσσα. Κάποιες από τις υπηρεσίες που προσφέρει είναι, υποστήριξη μεγάλου εύρους λειτουργικών, προσπάθεια επανάκλησης σε περίπτωση σφάλματος γραμμής, δρομολόγηση εξωτερικών κλήσεων. [31]

4.2.7 OpenSips

Είναι ένα ελεύθερο λογισμικό πρωτόκολλου έναρξης συνεδρίας με χρήση διακομιστή. Επίσης, προσφέρει λειτουργικότητα επιπέδου εφαρμογής, για μετάδοση βίντεο, φωνής και σύντομων μηνυμάτων. Χρησιμοποιείται σε πλήθος εφαρμογές που προσφέρουν διασύνδεση VoIP. Μερικές από τις δυνατότητες που προσφέρει είναι, διαχειριστή διακομιστή SIP, διακομιστή ανακατεύθυνσης SIP, διακομιστή εφαρμογής SIP, δίαυλο επικοινωνίας μεταξύ SIP και XAMPP για την αποστολή σύντομων μηνυμάτων IM και SMS σε κανάλι αμφίδρομης κατεύθυνσης, διαχειριστή φόρτου και αποστολέα SIP και front end επικοινωνία. [32]

4.2.8 SER

Το SER, ή αλλιώς SIP Express Router, διατίθεται με μια αδειοδότηση ελεύθερου λογισμικού GNU και μπορεί να λειτουργήσει ως καταχωρητής διακομιστής,

διαμεσολαβητής διακομιστής ή ανακατευθυντής διακομιστής σε SIP. Έχει τη δυνατότητα να διαμορφωθεί για να εξυπηρετήσει εξειδικευμένους ρόλους, για παράδειγμα εξισορρόπηση φορτίου ή SIP front-end για διακομιστές εφαρμογών. Προσφέρει πλήρη υποστήριξη της λειτουργικότητας RFC 3261, ένα πλήθος από βάσεις δεδομένων και λειτουργιών κλήσης. Πλέον αποτελεί κεντρικό κομμάτι για πολλά λειτουργικά συστήματα και τις διάφορες εκδοχές τους. Είναι γραμμένο σε C και υποστηρίζει τόσο IPv4 όσο και IPv6. Τέλος έχει πολύ μικρό μέγεθος εγκατάστασης, με βασική εγκατάσταση στα 300kb και όλα του τα εργαλεία να φτάνουν το ανώτερο στα 630kb. [33]

4.2.9 Yate

Είναι ελεύθερο τηλεπικοινωνιακό λογισμικό ανοιχτού κώδικα που υποστηρίζει την μετάδοση βίντεο, φωνής, δεδομένων και σύντομων μηνυμάτων. Είναι φτιαγμένο με την C++ και επιτρέπει την δημιουργία επεκτάσεων με γλώσσες όπως Perl, Python και PHP. Η αρχιτεκτονική του βασίζεται σε 4 τμήματα που δημιουργούν ένα σύστημα διακίνησης μηνυμάτων. Τον Ιανουάριο του 2014 δημοσιεύτηκε η τελευταία έκδοση 5.0. Τα τμήματα αυτά είναι ο πυρήνας, η μηχανή μηνυμάτων, η μηχανή τηλεφωνίας και τα στοιχεία Yate. Ο τρόπος με τον οποίο είναι δομημένο προσφέρει την δυνατότητα μίξης των συστημάτων του με άλλα πρωτόκολλα κειμενογράφου ή βάσης δεδομένων, πράγμα που διευκολύνει την εγγραφή κώδικα με οποιαδήποτε άλλη γλώσσα. [34]

4.2.10 OpenVBX

Είναι λογισμικό ανοιχτού κώδικα που προσφέρει την δυνατότητα στους χρήστες να δημιουργήσουν εφαρμογές για τηλεφωνικές κλήσεις και αποστολή σύντομων μηνυμάτων. Είναι σχεδιασμένο ώστε να είναι επεκτάσιμο από τους χρήστες και εύκολο στην χρήση. Προσφέρει τόσο ένα περιβάλλον χρήστη όσο και ένα API που μπορεί να χρησιμοποιηθεί για να δημιουργήσει νέες λειτουργικότητες για το τελικό προϊόν που επιθυμεί να χρησιμοποιήσει ο χρήστης. Ακόμα, επιτρέπει στους χρήστες του να μεταπωλούν το τελικό τους προϊόν στους πελάτες τους. Είναι διαθέσιμο με μια δημόσια αδειοδότηση του ιδρύματος Mozilla, αλλά δεν έχει εμπορική τεχνική υποστήριξη πελατών. Εγκαθίσταται και λειτουργεί στον διακομιστή του χρήστη. Και έχει ως προαπαιτούμενο την ύπαρξη τελευταίων εκδόσεων της PHP και MySQL. Είναι εφαρμογή PHP που προσφέρει γραφικό περιβάλλον χρήστη, βάση δεδομένων και λειτουργικότητα ολοκλήρωσης και ελέγχου κλήσεων. [35]

4.2.11 Συγκριτικός πίνακας

Στον πίνακα 2 μπορούμε να δούμε μια σύγκριση των προαναφερθέντων λογισμικών. Ο πίνακας είναι χωρισμένος σε τέσσερις στήλες. Η πρώτη στήλη αναφέρει τα ονόματα των υπηρεσιών προς σύγκριση. Στην δεύτερη στήλη παρουσιάζεται η δυνατότητα πραγματοποίησης κλήσης για όσες από αυτές την προσφέρουν και στην τρίτη η δυνατότητα αποστολής μηνύματος. Τέλος, η τέταρτη στήλη παρουσιάζει, την πιθανότητα χρήσης της υπηρεσίας αυτής στην μελέτη που προτάθηκε με την

διπλωματική αυτή εργασία. Οι μορφές μηνυμάτων που μπορούν να σταλούν από τα διάφορα λογισμικά είναι σύντομο διαδικτυακό μήνυμα (IM) και σύντομο μήνυμα κειμένου (SMS).

Υπηρεσία	Πραγματοποίηση κλήση	Αποστολή μηνύματος	Πρόταση για την μελέτη αυτή
AskoziaPBX	•		
AsteriskPBX	•		•
Elastix	•	•	•
FreeSWITCH	•		•
FreePBX			
GNU_Gatekeeper	•		
OpenSips	•	•	
SER	•		
Yate	•	•	•
OpenVBX	•	•	•

Πίνακας 2 Σύγκριση διαδικτυακών κέντρων.

Κεφάλαιο 5. Συμπεράσματα - Μελλοντικές επεκτάσεις

Στο κεφάλαιο αυτό θα γίνει μια σύγκριση της μελέτης που πραγματοποιήσαμε με τις υπάρχουσες εφαρμογές που κυκλοφορούν στην αγορά. Επίσης θα αναφερθούν οι μελλοντικές επεκτάσεις που μπορούν να γίνουν στην μελέτη αυτή.

5.1 Σύγκριση με τις Υπάρχουσες Εφαρμογές

Συμπερασματικά, η πτυχιακή αυτή εργασία έχει διαφορές αλλά και ομοιότητες με τις ήδη υπάρχουσες εφαρμογές της αγοράς. Η πλειονότητα των εφαρμογών που υπάρχουν αυτή τη στιγμή στην αγορά βασίζονται σε συστήματα έξυπνων κινητών συσκευών. Αντίθετα η μελέτη που πραγματοποιήσαμε βασίζεται σε μια διαδικτυακή πλατφόρμα. Αυτό της δίνει το πλεονέκτημα να απευθύνεται σε μεγαλύτερο κοινό. Επίσης, ενώ οι υπόλοιπες εφαρμογές έχουν ως μορφή υπενθύμισης παραλλαγές της ειδοποίησης των κινητών συσκευών, η δική μας προδιαγραφή προτείνει την κλήση σε τηλέφωνο, κινητό ή σταθερό, ένα μέσω ευρύτερα διαδεδομένο από τις έξυπνες κινητές συσκευές. Η κυριότερη έλλειψη που έχει η δική μας πρόταση, σε σχέση με τις υπάρχουσες υλοποιήσεις, είναι η διασφάλιση πως ο χρήστης θα λάβει την θεραπεία του όταν πραγματοποιηθεί η υπενθύμιση.

Σε σχέση με την εφαρμογή care4today η δική μας έχει πολλές ελλείψεις. Το σύστημα αυτό διαθέτει τόσο διαδικτυακό τμήμα όσο και εφαρμογή. Οι γιατροί μπορούν να ελέγχουν την λήψη των θεραπειών των ασθενών, όπως συμβαίνει και στη δική μας πρόταση. Επίσης, τα συγγενικά πρόσωπα των χρηστών μπορούν να τους παροτρύνουν να λαμβάνουν τη θεραπεία τους και να επιβεβαιώνουν την λήψη της, πράγμα στο οποίο η δική μας μελέτη υστερεί. Ακόμη, όταν κοντεύει να τελειώσει το σκεύασμα ο φαρμακοποιός μπορεί να ειδοποιήσει για λήψη νέας συνταγογράφησης. Ένα άλλο πλεονέκτημα της συγκεκριμένης εφαρμογής είναι η διατήρηση διαφόρων ζωτικών μετρήσεων τις οποίες μπορεί ο γιατρός να συμβουλευτεί για να αλλάξει την θεραπεία, να ενημερώσει το σύστημα και σε συνεργασία με τον φαρμακοποιό του ασθενούς να τον ενημερώσουν για την αλλαγή της θεραπείας του. Συνολικά λοιπόν η εφαρμογή αυτή είναι πιο ολοκληρωμένη από την δική μας πρόταση.

Συγκρίνοντας την μελέτη μας με την εφαρμογή MediSafe Meds & Pill Reminder βρίσκουμε και πάλι πολλές ελλείψεις, στη δική μας. Το κύριο προτέρημα της είναι η διασφάλιση πως ο ασθενής θα λάβει την θεραπεία του. Κατά την διάρκεια της υπενθύμισης η εφαρμογή θα ανοίξει στη συσκευή, ακόμη και όταν είναι κλειστή κάνοντάς την έτσι δύσκολο να αγνοηθεί – ξεχαστεί. Εάν παρόλα αυτά ο ασθενής την αγνοήσει η εφαρμογή θα ειδοποιήσει το άτομο που έχει οριστεί ως υπεύθυνος να του θυμίζει την θεραπεία του. Ακόμη, όπως και η προηγούμενη εφαρμογή, αποθηκεύει ζωτικές μετρήσεις που μπορούν να σταλούν απευθείας σε κάποιο γιατρό.

Η τρίτη εφαρμογή που αναφέρεται παραπάνω και κάνουμε σύγκριση είναι η Med Helper Pill Reminder. Η εφαρμογή αυτή αν και έχει πολλές δυνατότητες είναι πολύ δύσκολη στην χρήση και δυσνόητη. Επίσης, οι χρωματικοί συνδυασμοί που χρησιμοποιεί καθιστούν μερικά μέρη της σχεδόν αδύνατο να διαβαστούν. Ακόμη το μενού της είναι δυσνόητο, εάν κάποιος δεν διαβάσει το εγχειρίδιο χρήσης και τα παραδείγματα χρήσης. Η δική μας μελέτη αποφεύγει τέτοιου είδους προβλήματα χρησιμοποιώντας και εφαρμόζοντας τους κανόνες προσβασιμότητας του w3c. Παρόλα

αυτά, η Med Helper Pill Reminder, προσφέρει επαρκή υπενθύμιση με την μορφή ειδοποίησης, και δίνει την δυνατότητα αποθήκευσης των στοιχείων των θεραπόντων ιατρών και του φαρμακοποιού του χρήστη. Επίσης, μπορεί ο χρήστης να καταγράψει τις ζωτικές του μετρήσεις και να τις μοιραστεί με τους γιατρούς του.

Τέλος, η εφαρμογή Dosecast - Medication Reminder αποτελεί ένα απλό σύστημα υπενθύμισης θεραπείας. Ο χρήστης θα πρέπει να εισάγει μόνος του τις φαρμακευτικές του αγωγές όπως και στις παραπάνω περιπτώσεις και στην συνέχεια θα μπορεί να λαμβάνει υπενθυμίσεις για την λήψη τους, καθώς και να δει την επόμενη ώρα που θα πρέπει να κάνει λήψη. Στην περίπτωση όμως που έχει εγκαταστήσει και χρησιμοποιεί την ελεύθερη έκδοση της εφαρμογής, μπορεί να κατηγοριοποιήσει το σκεύασμα της αγωγής μόνο ως χάπι και δεν έχει πρόσβαση σε άλλες επιλογές. Επίσης, δεν υπάρχει πρόσβαση στα στοιχεία του γιατρού που χορήγησε την κάθε θεραπεία. Ακόμα, η πληθώρα των στοιχείων και ο τρόπος με τον οποία ζητούνται κάνουν το σύστημα αρκετά δύσχρηστο.

5.2 Μελλοντικές Επεκτάσεις

Η μελέτη αυτή θα μπορούσε να περιλαμβάνει πολλά περισσότερα χαρακτηριστικά, αλλά στο πλαίσιο αυτής της διπλωματικής δημιουργήθηκαν οι βάσεις της.

Μια από τις προσθήκες που θα μπορούσαν να γίνουν, είναι επιπλέον τρόποι υπενθύμισης της θεραπείας. Μια από τις απλούστερες ιδέες υπενθύμισης, αποτελεί το αναδυόμενο παράθυρο στον περιηγητή του χρήστη και μια άλλη, η αποστολή σύντομου μηνύματος ή mail. Μια ακόμη μορφή υπενθύμισης, που θα μπορούσε να γίνει, με χρήση υλικού αυτή τη φορά, είναι η χρήση συστήματος διακοπής σήματος τηλεόρασης για ασθενείς που απασχολούνται πολλές ώρες με αυτό το μέσο ψυχαγωγίας. Η μελέτη, με την βοήθεια του κατάλληλου υλικού, θα διακόπτει την λήψη σήματος στην τηλεόραση του χρήστη και θα του προβάλλει το κατάλληλο μήνυμα υπενθύμισης της θεραπείας που πρέπει να λάβει την συγκεκριμένη στιγμή.

Άλλου τύπου προσθήκες που προβλέπονται είναι η υλοποίηση της αυτόματης ανανέωσης του πίνακα που περιέχει τις θεραπείες του χρήστη, ώστε να δίνει την αίσθηση αντίστροφης μέτρησης. Για να παραμείνει η ιστοσελίδα συμβατή με τους κανόνες προσβασιμότητας, μαζί με την αυτόματη ανανέωση του πίνακα θα προστεθεί και δομή που θα δίνει την δυνατότητα στον χρήστη να σταματάει την αντίστροφη μέτρηση. Επιθυμητό είναι επίσης να ικανοποιηθούν όλοι οι κανόνες προσβασιμότητας που έχουν τεθεί από τον οργανισμό w3c. Άλλες βλέψεις επίσης είναι η υλοποίηση εφαρμογής για έξυπνα κινητά τηλέφωνα και πιθανότατα η δημιουργία κάποιας φορητής συσκευής που θα αλλάζει χρώμα κατά το πέρασ του χρόνου. Ακόμα, μια άλλη προσθήκη θα μπορούσε να είναι η εισαγωγή ιατρικών παιχνιδιών για την βελτίωση της μνήμης των ασθενών. Τα παιχνίδια αυτά θα βοηθούν «με ευχάριστο τρόπο» τον χρήστη-ασθενή να εξασκεί τον εαυτό του ώστε να θυμάται να παίρνει τα φάρμακα του. Θα απευθύνεται κυρίως σε ασθενείς οι οποίοι είναι στα πρώτα στάδια απώλειας της μνήμης αλλά και σε όλους τους υπόλοιπους που δυσκολεύονται να θυμηθούν την λήψη της θεραπείας τους, και χρειάζονται τρόπους εξάσκησης για τη βελτίωση και την διατήρηση της, αλλά και ψυχολογική υποστήριξη για την αντιμετώπιση της δύσκολης ομολογουμένως αυτής κατάστασης.

Ένας άλλος τομέας που μπορεί να κινηθεί η μελέτη είναι η επιβεβαίωση πως ο χρήστης πήρε την θεραπεία του. Αυτό μπορεί να γίνει με πολλούς τρόπους. Ένας από αυτούς είναι η φωνητική επιβεβαίωση μέσω τηλεφώνου. Πέντε λεπτά μετά την κλήση, που θα γίνεται για την υπενθύμιση, θα γίνεται μια δεύτερη κλήση, που θα ζητά από τον χρήστη να απαντήσει ναι ή όχι στο αν πήρε την θεραπεία του. Στην περίπτωση θετικής απάντησης θα γίνεται ενημέρωση και μηδενισμός του χρόνου και επανεκκίνηση της αντίστροφης μέτρησης, ενώ σε αντίθετη περίπτωση θα ξαναγίνεται κλήση σε πέντε λεπτά. Ένας άλλος απλοϊκός τρόπος είναι η χειροκίνητη ενημέρωση της βάσης δεδομένων, μέσω της ιστοσελίδας. Ένας ακόμα είναι η υπενθύμιση μέσω ενός τρίτου ατόμου που θα έχει καταχωρηθεί σαν υπεύθυνος για τον χρήστη αυτό. Το άτομο αυτό αφού επικοινωνήσει με τον ασθενή και ελέγξει αν πήρε τη δόση και θα ενημερώνει για τη λήψη.

Κάθε θεραπεία θα έχει ένα πεδίο το οποίο θα εμφανίζεται μετά τον μηδενισμό του χρονομέτρου και θα είναι επιλέξιμο. Όταν επιλεγεί αυτό το πεδίο η βάση θα ενημερώνεται πως ο χρήστης έλαβε την θεραπεία του. Επιπλέον, ο γιατρός θα μπορεί να δηλώνει εάν την συγκεκριμένη θεραπεία ο ασθενής θα μπορεί να την λάβει πριν την προκαθορισμένη ώρα ή όχι, έτσι ο ασθενής θα μπορεί να λαμβάνει την θεραπεία ακόμη και πριν την λήξη του χρονομέτρου και να ενημερώνει τη βάση με τον παραπάνω τρόπο. Στην περίπτωση δημιουργίας φορητής συσκευής είναι δυνατή η ενσωμάτωση ενός Near field communication (NFC) αποστολέα στην συσκευή. Σε συνδυασμό με την τοποθέτηση ενός δέκτη κοντά στο σημείο που αποθηκεύει ο χρήστης τις θεραπείες του, θα υπάρχει αυτόματη ενημέρωση, όταν ο χρήστης λαμβάνει την θεραπεία.

Με την ενσωμάτωση όμως ελέγχου για την λήψη της θεραπείας από τους ασθενείς, δημιουργούνται νέα ηθικά διλήμματα που πρέπει να απαντηθούν. Τα προσωπικά δεδομένα των ασθενών που αφορούν στην υγεία, είναι επικίνδυνο να διακινούνται χωρίς ασφάλεια στο διαδίκτυο και άρα να είναι εύκολο να υποκλαπούν. Τα ιατρικά προσωπικά δεδομένα αποτελούν τα πιο ευαίσθητα προσωπικά δεδομένα ενός ατόμου και άρα οι χρήστες πρέπει να διακατέχονται από ευαισθησία και ιδιαίτερη προσοχή κατά την παραχώρησή τους.

Παράρτημα I

Παρουσίαση Κώδικα

I. Κώδικας HTML – PHP

i. db_connect.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.2 Σύνδεση με την Βάση Δεδομένων, στην σελίδα 22. Ο ρόλος του κώδικα αυτού, είναι να προσφέρει πρόσβαση στην βάση δεδομένων, στα αρχεία που το καλούν.

```
1 <?php
2 include_once 'psl-config.php'; // As functions.php is not
3 included
4 $mysqli = new mysqli(HOST, USER, PASSWORD, DATABASE);
```

ii. pls-config.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.2 Σύνδεση με την Βάση Δεδομένων, στην σελίδα 22. Ο ρόλος του κώδικα αυτού, είναι να μετατρέψει τα στοιχεία εισόδου στην βάση δεδομένων σε παραμέτρους, ώστε να τις χρησιμοποιήσει το αρχείο db_connect.php.

```
1 <?php
2 /**
3  * These are the database login details
4  */
5 define("HOST", "localhost"); // The host you want to
6 connect to.
7 define("USER", "sec_user"); // The database username.
8 define("PASSWORD", "1q2w3e4r5t"); // The database password.
9 define("DATABASE", "secure_login"); // The database name.
10
11 define("CAN_REGISTER", "any");
12 define("DEFAULT_ROLE", "member");
13
14 define("SECURE", FALSE); // FOR DEVELOPMENT ONLY!!!!
15 ?>
```

iii. logout.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.10 Έξοδος από το σύστημα, στην σελίδα 33. Ο ρόλος του κώδικα αυτού, είναι να εκτελεί την διαδικασία της αποσύνδεσης ενός χρήστη από τον λογαριασμό του.

```
1 <?php
```

```

2  include_once 'functions.php';
3  session_start();
4
5  // Unset all session values
6  $_SESSION = array();
7
8  // get session parameters
9  $params = session_get_cookie_params();
10
11 // Delete the actual cookie.
12 setcookie(session_name(),
13           '', time() - 42000,
14           $params["path"],
15           $params["domain"],
16           $params["secure"],
17           $params["httponly"]);
18
19 // Destroy session
20 session_destroy();
21 header('Location: ../index.php');

```

iv. error.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.3 Αρχική Σελίδα, στην σελίδα 23. Το αρχείο αυτό, επεξεργάζεται και εμφανίζει τα μηνύματα λάθους, σε περίπτωση εισαγωγής λανθασμένων στοιχείων εισόδου χρήστη ή άλλων περιπτώσεων.

```

1  <?php
2  $error = filter_input(INPUT_GET, 'err', $filter =
3  FILTER_SANITIZE_STRING);
4
5  if (! $error) {
6      $error = 'Oops! An unknown error happened.';
7  }
8  ?>
9  <!DOCTYPE html>
10 <html>
11     <head>
12         <meta charset="UTF-8">
13         <title>Secure Login: Error</title>
14         <link rel="stylesheet" href="styles/main.css" />
15     </head>
16     <body>
17         <h1>There was a problem</h1>
18         <p class="error"><?php echo $error; ?></p>
19     </body>
20 </html>

```


v. index.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.3 Αρχική Σελίδα, στην σελίδα 23. Στο αρχείο αυτό, είναι αποθηκευμένος ο κώδικας για την αρχική σελίδα της μελέτης. Περιέχει την φόρμα εισόδου και μεταβιβάζει τα στοιχεία εισόδου στο αρχείο επεξεργασίας για την πραγματοποίηση της πράξης εισόδου.

```
1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';
4 sec_session_start();
5
6 if (login_check($mysqli) == true) {
7     $logged = 'in';
8 } else {
9     $logged = 'out';
10 }
11 ?>
12 <!DOCTYPE html>
13 <html lang="en">
14 <head>
15     <title>Thesis Project </title>
16     <script type="text/JavaScript"
17 src="js/sha512.js"></script>
18     <script type="text/JavaScript" src="js/forms.js"></script>
19     <meta name="viewport" content="width=device-width,
20 initial-scale=1"><!-- defining responsivnes in mobile devices
21 -->
22     <meta charset="utf-8"><!-- defining the character set for
23 encoding -->
24     <link href="css/bootstrap.css" rel="stylesheet">
25     <link href="css/bootstrap.min.css" rel="stylesheet">
26     <link href="css/bootstrap-theme.css" rel="stylesheet">
27     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
28     <link href="css/style.css" rel="stylesheet"> <!-- styling
29 link -->
30 </head>
31 <body>
32     <div id="index">
33         <div class="page-header"> <!-- page header: preferred
34 because there was nothing to put in the navibar -->
35             <h1>Markella Nikolopoulou-Themeli
36                 <small>Thesis project</small>
37             </h1>
38         </div>
39         <div class="container-fluid" >
40             <div class=" col-md-2" role="complementary">
41                 <nav class="bs-docs-sidebar hidden-print
42 hidden-xs hidden-sm">
43                     <ul class="nav bs-docs-sidenav">
44                         <li class="">
```

```

45             <?php
46 echo '<a href="register.php">Register</a>'; ?>
47     </li>
48     <li>
49         <a href="#">Back to the top of the Page</a>
50     </li>
51 </ul>
52 </nav>
53 </div>
54 <div class="col-md-6">
55     <?php
56         if (isset($_GET['error']))
57             echo '<h3 class="error">Error Logging In!</h3>';
58     }
59     ?>
60 <form action="includes/process_login.php" method="post"
61 name="login_form">
62 <h3>Username: <small>Here insert the username from your
63 registration</small></h3> <input type="text" name="username"
64 class="form-control input-lg" />
65 <br>
66 <h3>Password: <small>Here insert the password from your
67 registration</small></h3> <input type="password"
68 name="password" id="password" class="form-control input-lg" />
69 <br>
70 <input type="button" class="btn btn-default btn-lg"
71 value="Login" onclick="formhash(this.form,
72 this.form.password);" />
73 </form>
74 <?php
75 if (login_check($mysqli) == true) {
76 echo '<h3>Currently logged ' . $logged . ' as ' .
77 htmlentities($_SESSION['username']) . '</h3>';
78 echo '<h3>Do you want to change user? <a
79 href="includes/logout.php">Log out</a></h3>';
80 } else {
81 echo '<h3>Currently logged ' . $logged . '</h3>';
82 echo "<h3>Do you have an account? If not <a
83 href='register.php'>register</a></h3>";
84 }
85 ?>
86 </div>
87 </div>
88 </div>
89 </body>
90 </html>

```

vi. process_login.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.3 Αρχική Σελίδα, στην σελίδα 23. Ο ρόλος του κώδικα αυτού, είναι να επεξεργαστεί να στοιχεία που έχει

δώσει ο χρήστης για την είσοδό του στο σύστημα και να πραγματοποιήσει την πράξη της εισόδου.

```
1 <?php
2 include_once 'db_connect.php';
3 include_once 'functions.php';
4
5 sec_session_start(); // Our custom secure way of starting a
6 PHP session.
7
8 if (isset($_POST['username'], $_POST['p'])) {
9     $username = $_POST['username'];
10    $password = $_POST['p']; // The hashed password.
11
12    if (login($username, $password, $mysqli) == true) {
13        // Login success
14        if ($stmt = $mysqli->prepare("SELECT capacity
15                                    FROM members
16                                    WHERE username = ?
17                                    LIMIT 1"))
18            {
19            $stmt->bind_param('s', $username); // Bind
20 "$username" to parameter.
21 $stmt->execute(); // Execute the prepared
22 query.
23 $stmt->store_result();
24
25 $stmt->bind_result($capacity);
26 $stmt->fetch();
27
28 if ($capacity == "patient")
29 {
30     // echo $capacity;
31     header('Location: ../Patient.php');
32 }
33 elseif ($capacity == "doctor")
34 {
35     //echo $capacity;
36     header('Location: ../Doctor.php');
37 }
38 elseif ($capacity == "admin")
39 {
40     //echo $capacity;
41     header('Location: ../Admin.php');
42 }
43
44 }
45 } else {
46     // Login failed
47     header('Location: ../index.php?error=1');
48 }
```

```

49 } else {
50     // The correct POST variables were not sent to this page.
51     echo 'Invalid Request';
52 }

```

vii. functions.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.3 Αρχική Σελίδα, στην σελίδα 23. Στο αρχείο αυτό, περιέχονται διάφορες συναρτήσεις για την πράξη εισόδου στο σύστημα και την δημιουργία ασφαλούς συνεδρίας.

```

1  <?php
2  include_once 'psl-config.php';
3
4  function sec_session_start() {
5      $session_name = 'sec_session_id';    // Set a custom
6  session name
7      $secure = SECURE;
8      // This stops JavaScript being able to access the session
9  id.
10     $httponly = true;
11     // Forces sessions to only use cookies.
12     if (ini_set('session.use_only_cookies', 1) === FALSE) {
13         header("Location: ../error.php?err=Could not initiate
14 a safe session (ini_set)");
15         exit();
16     }
17     // Gets current cookies params.
18     $cookieParams = session_get_cookie_params();
19     session_set_cookie_params($cookieParams["lifetime"],
20         $cookieParams["path"],
21         $cookieParams["domain"],
22         $secure,
23         $httponly);
24     // Sets the session name to the one set above.
25     session_name($session_name);
26     session_start();    // Start the PHP session
27     session_regenerate_id(true);    // regenerated the
28 session, delete the old one.
29 }
30
31 function login($username, $password, $mysqli) {
32     // Using prepared statements means that SQL injection is
33 not possible.
34     if ($stmt = $mysqli->prepare("SELECT id, username,
35 password, salt
36                                     FROM members
37                                     WHERE username = ?
38                                     LIMIT 1")) {

```

```

39         $stmt->bind_param('s', $username); // Bind
40 "$username" to parameter.
41         $stmt->execute(); // Execute the prepared query.
42         $stmt->store_result();
43
44         // get variables from result.
45         $stmt->bind_result($user_id, $username, $db_password,
46 $salt);
47         $stmt->fetch();
48
49         // hash the password with the unique salt.
50         $password = hash('sha512', $password . $salt);
51         if ($stmt->num_rows == 1)
52         {
53             // If the user exists we check if the account is
54 locked
55             // from too many login attempts
56
57             if (checkbrute($user_id, $mysqli) == true)
58             {
59                 // Account is locked
60                 // Send an email to user saying their account
61 is locked
62                 return false;
63             }
64
65             else
66             {
67                 // Check if the password in the database
68 matches
69                 // the password the user submitted.
70                 if ($db_password == $password)
71                 {
72                     // Password is correct!
73                     // Get the user-agent string of the user.
74                     $user_browser =
75 $_SERVER['HTTP_USER_AGENT'];
76                     // XSS protection as we might print this
77 value
78                     $user_id = preg_replace("/^[^0-9]+/", "",
79 $user_id);
80                     $_SESSION['user_id'] = $user_id;
81                     // XSS protection as we might print this
82 value
83                     $username = preg_replace("/[^a-zA-Z0-9_\-
84 ]+/", "", $username);
85                     $_SESSION['username'] = $username;
86                     $_SESSION['login_string'] = hash('sha512',
87 $password . $user_browser);
88                     // Login successful.
89                     return true;
90                 }

```

```

91         else
92         {
93             // Password is not correct
94             // We record this attempt in the database
95             $now = time();
96             $mysqli->query("INSERT INTO
97 login_attempts(user_id, time)
98                                     VALUES ('$user_id',
99 '$now')");
100             return false;
101         }
102     }
103     // *
104 }
105 else
106 {
107     // No user exists.
108     return false;
109 }
110 }
111 }
112
113 function checkbrute($user_id, $mysqli) {
114     // Get timestamp of current time
115     $now = time();
116
117     // All login attempts are counted from the past 2 hours.
118     $valid_attempts = $now - (2 * 60 * 60);
119
120     if ($stmt = $mysqli->prepare("SELECT time
121                                     FROM login_attempts
122                                     WHERE user_id = ?
123                                     AND time >
124 '$valid_attempts'")) {
125         $stmt->bind_param('i', $user_id);
126
127         // Execute the prepared query.
128         $stmt->execute();
129         $stmt->store_result();
130
131         // If there have been more than 5 failed logins
132         if ($stmt->num_rows > 5) {
133             return true;
134         } else {
135             return false;
136         }
137     }
138 }
139
140 function login_check($mysqli) {
141     // Check if all session variables are set
142     if (isset($_SESSION['user_id'],

```

```

143         $_SESSION['username'],
144         $_SESSION['login_string'])) {
145
146     $user_id = $_SESSION['user_id'];
147     $login_string = $_SESSION['login_string'];
148     $username = $_SESSION['username'];
149
150     // Get the user-agent string of the user.
151     $user_browser = $_SERVER['HTTP_USER_AGENT'];
152
153     if ($stmt = $mysqli->prepare("SELECT password
154                                 FROM members
155                                 WHERE id = ? LIMIT 1"))
156 {
157     // Bind "$user_id" to parameter.
158     $stmt->bind_param('i', $user_id);
159     $stmt->execute(); // Execute the prepared query.
160     $stmt->store_result();
161
162     if ($stmt->num_rows == 1) {
163         // If the user exists get variables from
164 result.
165         $stmt->bind_result($password);
166         $stmt->fetch();
167         $login_check = hash('sha512', $password .
168 $user_browser);
169
170         if ($login_check == $login_string) {
171             // Logged In!!!!
172             return true;
173         } else {
174             // Not logged in
175             return false;
176         }
177     } else {
178         // Not logged in
179         return false;
180     }
181 } else {
182     // Not logged in
183     return false;
184 }
185 } else {
186     // Not logged in
187     return false;
188 }
189 }
190
191 function esc_url($url) {
192
193     if ('' == $url) {
194         return $url;

```

```

195     }
196
197     $url = preg_replace('|[^\a-z0-9-
198 ~+_.?#=!&;,/:%@$|*\'()\x80-\xff]|i', '', $url);
199
200     $strip = array('%0d', '%0a', '%0D', '%0A');
201     $url = (string) $url;
202
203     $count = 1;
204     while ($count) {
205         $url = str_replace($strip, '', $url, $count);
206     }
207
208     $url = str_replace(';://', ':://', $url);
209
210     $url = htmlentities($url);
211
212     $url = str_replace('&', '&#038;', $url);
213     $url = str_replace('"', '&#039;', $url);
214
215     if ($url[0] !== '/') {
216         // We're only interested in relative links from
217         $_SERVER['PHP_SELF']
218         return '';
219     } else {
220         return $url;
221     }
222 }

```

viii. Register.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.4 Σελίδα Εγγραφής, στις σελίδες 24 και 25. Το αρχείο αυτό, περιέχει την φόρμα στην οποία ο χρήστης εισάγει τα στοιχεία του για να πραγματοποιήσει την εγγραφή του.

```

1 <?php
2 include_once 'includes/register.inc.php';
3 include_once 'includes/functions.php';
4 ?>
5 <!DOCTYPE html>
6 <html>
7 <head>
8     <meta name="viewport" content="width=device-width,
9     initial-scale=1">
10    <meta charset="UTF-8">
11    <title>Registration Form</title>
12    <script type="text/JavaScript"
13    src="js/sha512.js"></script>
14    <script type="text/JavaScript" src="js/forms.js"></script>
15    <link href="css/bootstrap.css" rel="stylesheet">

```



```

16     <link href="css/bootstrap.min.css" rel="stylesheet">
17     <link href="css/bootstrap-theme.css" rel="stylesheet">
18     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
19     <link href="css/style.css" rel="stylesheet">
20 </head>
21 <body>
22     <div class="container-fluid" id="register">
23     </div>
24     <div class="page-header"> <!-- page header: preferred
25 because there was nothing to put in the navbar -->
26         <h1>Register with us</h1>
27
28     </div>
29
30     <div class="container-fluid" >
31         <div class=" col-md-2" role="complementary">
32             <nav class="bs-docs-sidebar hidden-print
33 hidden-xs hidden-sm">
34                 <ul class="nav bs-docs-sidenav">
35
36                     <li class="">
37                         <?php echo '<a
38 href="index.php">Log In</a>'; ?>
39                     </li>
40
41                     <li>
42                         <a href="#">Back to the top of the
43 Page</a>
44                     </li>
45                 </ul>
46             </nav>
47         </div>
48         <div class="col-md-6">
49             <!-- Registration form to be output if the POST
50 variables are not
51 set or if the registration script caused an error. -->
52
53             <?php
54             if (!empty($error_msg)) {
55                 echo $error_msg;
56             }
57             ?>
58             <form action="<?php echo
59 esc_url($_SERVER['PHP_SELF']); ?>"
60                 method="post"
61                 name="registration_form">
62
63                 <h3>Username: <small>Here insert the username you
64 want to use for logging in the system</small></h3> <input
65 type='text' name='username' id='username' class="form-control
66 input-lg" /><br>

```

```

67         <h3>Email: <small>Here insert the email you want
68 to store in the system</small></h3> <input type="text"
69 name="email" id="email" class="form-control input-lg" /><br>
70         <h3>Telephone: <small>Here you can add a telephone
71 on which the calla will be made for the therapy
72 reminder</small></h3> <input type="text" name="tel" id="tel"
73 class="form-control input-lg" /><br>
74         <h3>Password: <small>Hete type the password you
75 want to use for your log in</small></h3> <input
76 type="password" name="password" id="password" class="form-
77 control input-lg" /><br>
78         <h3>Confirm password: <small>Retype the above
79 password </small></h3> <input type="password"
80 name="confirmpwd" id="confirmpwd" class="form-control input-
81 lg" /><br>
82         <h3>Capacity: <small>If you want to register as a
83 doctor select Doctor. If you want to register as a patient
84 select Patient</small></h3> <h4> <input type="radio"
85 name="capacity" id="capacity" value="doctor" /> Doctor</h4>
86         <h4> <input type="radio" name="capacity"
87 id="capacity" value="patient" /> Patient</h4>
88         <br>
89         <input type="button" class="btn btn-default btn-
90 lg" value="Register" onclick="return regformhash(this.form,
91 this.form.username,
92 this.form.email,
93 this. form.tel,
94 this.form.password,
95 this.form.confirmpwd,
96 this.form.capacity);"
97 />
98     </form>
99     <h4>Return to the <a href="index.php">login
100 page</a>.</h4>
101     </div>
102 </div>
103
104 </body>
105 </html>

```

ix. register.inc.php

Ο κώδικας αυτός, αναφέρεται στα υπό-κεφάλαια 3.2.1.4 Σελίδα Εγγραφής και 3.2.1.5 Σελίδα Επιτυχούς Εγγραφής, στις σελίδες 24 και 25. Ο ρόλος του κώδικα αυτού, είναι ο έλεγχος των στοιχείων με τα οποία επιθυμεί ένας χρήστης να εγγραφεί και η εισαγωγή τους στη βάση δεδομένων.

```

1 <?php
2 include_once 'db_connect.php';
3 include_once 'psl-config.php';

```

```

4
5 $error_msg = "";
6
7 if (isset($_POST['username'], $_POST['email'], $_POST['tel'],
8 $_POST['p'], $_POST['capacity'])) {
9     // Sanitize and validate the data passed in
10    $username = filter_input(INPUT_POST, 'username',
11 FILTER_SANITIZE_STRING);
12    $email = filter_input(INPUT_POST, 'email',
13 FILTER_SANITIZE_EMAIL);
14    $email = filter_var($email, FILTER_VALIDATE_EMAIL);
15    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
16        // Not a valid email
17        $error_msg .= '<p class="error">The email address you
18 entered is not valid</p>';
19    }
20
21    $telephone = filter_input(INPUT_POST, 'tel',
22 FILTER_SANITIZE_STRING);
23    $password = filter_input(INPUT_POST, 'p',
24 FILTER_SANITIZE_STRING);
25    if (strlen($password) != 128) {
26        // The hashed pwd should be 128 characters long.
27        // If it's not, something really odd has happened
28        $error_msg .= '<p class="error">Invalid password
29 configuration.</p>';
30    }
31
32    // Username validity and password validity have been
33 checked client side.
34    // This should be adequate as nobody gains any
35 advantage from
36 // breaking these rules.
37 //
38
39    $capacity = filter_input(INPUT_POST, 'capacity',
40 FILTER_SANITIZE_STRING);
41
42    $prep_stmt = "SELECT id FROM members WHERE email = ? LIMIT
43 1";
44    $stmt = $mysqli->prepare($prep_stmt);
45
46    // check existing email
47    if ($stmt) {
48        $stmt->bind_param('s', $email);
49        $stmt->execute();
50        $stmt->store_result();
51
52        if ($stmt->num_rows == 1) {
53            // A user with this email address already exists
54            $error_msg .= '<p class="error">A user with this
55 email address already exists.</p>';

```

```

56             $stmt->close();
57         }
58             $stmt->close();
59     } else {
60         $error_msg .= '<p class="error">Database error Line
61 39</p>';
62             $stmt->close();
63     }
64
65     // check existing username
66     $prep_stmt = "SELECT id FROM members WHERE username = ?
67 LIMIT 1";
68     $stmt = $mysqli->prepare($prep_stmt);
69
70     if ($stmt) {
71         $stmt->bind_param('s', $username);
72         $stmt->execute();
73         $stmt->store_result();
74
75             if ($stmt->num_rows == 1) {
76                 // A user with this username already
77 exists
78                 $error_msg .= '<p class="error">A user
79 with this username already exists</p>';
80                 $stmt->close();
81             }
82             $stmt->close();
83     } else {
84         $error_msg .= '<p class="error">Database error
85 line 55</p>';
86         $stmt->close();
87     }
88
89     // TODO:
90     // We'll also have to account for the situation where the
91 user doesn't have
92 // rights to do registration, by checking what type of
93 user is attempting to
94 // perform the operation.
95
96     if (empty($error_msg)) {
97         // Create a random salt
98         //$random_salt = hash('sha512',
99 uniqid(openssl_random_pseudo_bytes(16), TRUE)); // Did not
100 work
101         $random_salt = hash('sha512', uniqid(mt_rand(1,
102 mt_getrandmax()), true));
103
104         // Create salted password
105         $password = hash('sha512', $password . $random_salt);
106
107         // Insert the new user into the database

```

```

108         if ($insert_stmt = $mysqli->prepare("INSERT INTO
109 members (username, email, telephone, capacity, password, salt)
110 VALUES (?, ?, ?, ?, ?, ?)") {
111             $insert_stmt->bind_param('ssssss', $username,
112 $email, $telephone, $capacity, $password, $random_salt);
113             //$insert_stmt->execute(); // Execute the prepared
114 query.
115             if (! $insert_stmt->execute()) {
116                 header('Location:
117 ../error.php?err=Registration failure: INSERT');
118             }
119         }
120         header('Location: ./register_success.php');
121     }
122 }

```

x. register_success.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.5 Σελίδα Επιτυχούς Εγγραφής, στην σελίδα 25. Το αρχείο αυτό, περιέχει τον κώδικα για την σελίδα που εμφανίζεται μετά την επιτυχημένη εγγραφή ενός νέου χρήστη

```

1 <!DOCTYPE html>
2 <html>
3     <head>
4         <meta charset="UTF-8">
5         <title>Secure Login: Registration Success</title>
6         <link rel="stylesheet" href="styles/main.css" />
7     </head>
8     <body>
9         <h1>Registration successful!</h1>
10        <p>You can now go back to the <a
11 href="index.php">login page</a> and log in</p>
12    </body>
13 </html>

```

xi. Patient.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.6 Σελίδα Ασθενή, στις σελίδες 25, 26 και 27. Στο αρχείο αυτό, υπάρχει ο κώδικας για την σελίδα που εμφανίζεται στον ασθενή χρήστη και οι δυνατότητες που του προσφέρονται από την μελέτη. Δεν περιέχεται ο κώδικας για την σελίδα με την οποία πραγματοποιείται το αίτημα για την διασύνδεση με νέο γιατρό.

```

1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';

```

```

4
5  sec_session_start();
6
7  ?>
8
9  <!DOCTYPE html>
10 <html>
11 <head>
12     <meta charset="UTF-8">
13     <!--     <meta http-equiv="refresh" content="1"> -->
14     <title>Patient's page</title>
15     <script type="text/JavaScript"
16 src="js/sha512.js"></script>
17     <script type="text/JavaScript" src="js/forms.js"></script>
18     <meta name="viewport" content="width=device-width,
19 initial-scale=1"><!-- defining responsivnes in mobile devices
20 -->
21     <link href="css/bootstrap.css" rel="stylesheet">
22     <link href="css/bootstrap.min.css" rel="stylesheet">
23     <link href="css/bootstrap-theme.css" rel="stylesheet">
24     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
25     <link href="css/style.css" rel="stylesheet"> <!-- styling
26 link -->
27     <script type="text/javascript">
28     update();
29     setInterval('update()',1000);
30     document.getElementById('x').innerHTML=1;
31     </script>
32
33 </head>
34 <body>
35     <div id="patient">
36         <div class="page-header"> <!-- page header: preferred
37 because there was nothing to put in the navibar -->
38             <h1>Welcome Mr/Mrs <?php echo
39 htmlentities($_SESSION['username']); ?>
40             </h1>
41         </div>
42         <div class="container-fluid">
43             <div class=" col-md-2" role="complementary">
44                 <nav class="bs-docs-sidebar hidden-print hidden-xs
45 hidden-sm">
46                     <ul class="nav bs-docs-sidenav">
47                         <li>
48                             <a href="#therapies">Therapies</a>
49                         </li>
50                         <li>
51                             <a href="#doctors">Doctors</a>
52                         </li>
53                         <li>
54                             <a href="add_doctor.php">Add a new
55 Doctor to the doctor list</a>

```

```

56         </li>
57         <li>
58             <a href="#profile">Profile</a>
59         </li>
60         <li class="">
61             <?php echo '<a
62 href="includes/logout.php">Log Out</a>' ?>
63         </li>
64         <li>
65             <a href="#">Back to the top of the
66 Page</a>
67         </li>
68     </ul>
69 </nav>
70
71 </div>
72 <div class="col-md-10">
73     <div class="bs-docs-section">
74         <h3 id="therapies" class="page-
75 header">Therapies<small> In this section you can review all
76 the prescriptions and the time left to take them</small></h3>
77
78         <table class="table">
79             <thead>
80                 <caption>This table contains all the
81 prescriptions, the given dosage, the prescribing doctor, the
82 time of the last intake and the time left for the next
83 intake.</caption>
84                 <tr>
85                     <th>Drug</th>
86                     <th>Take every (hour(s))</th>
87                     <th>Prescribed by:</th>
88                     <th>Take again in:</th>
89                     <th>Time left:</th>
90                 </tr>
91             </thead>
92             <tbody>
93
94                 <?php
95                 $stmt = "SELECT prescriptions.Id,
96 prescriptions.Doctor, prescriptions.Drug,
97 prescriptions.Dosage, prescriptions.LastTaken, drugs.Name
98                 FROM `prescriptions`, `drugs`
99                 WHERE prescriptions.Drug = drugs.Id
100                AND Patient =
101                '".$_SESSION['username']."'";
102                 $result = mysqli_query($mysqli,$stmt);
103                 $pa = array();
104                 while ($x = mysqli_fetch_array($result)){
105                     $pa[] = $x;
106                 }
107                 foreach ($pa as $i){

```

```

108
109
110             $nextTakeIn =
111 strtotime($i['LastTaken']) + ($i['Dosage'] - 1) * 3600;
112             $id = $i['Id'];
113             $time = time() + 3600;
114             $timeLeft =
115 (strtotime($i['LastTaken']) + (($i['Dosage'] - 1) * 3600)) -
116 (time() + 3600);
117
118             $t = "success";
119
120             printf("<tr id='x' class='st'><td
121 id='y'>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td></tr
122 >\n",$i['Name'],$i['Dosage'],$i['Doctor'], date('H:i:s d-m',
123 $nextTakeIn), date('H:i:s',$timeLeft));
124
125
126             if ($nextTakeIn <= strtotime("+0
127 seconds")) {
128                 echo "You should have taken your
129 ".$i['Name']." in ".date('H:i:s d-m', $nextTakeIn)."<br>";
130                 $sql = "UPDATE prescriptions SET
131 prescriptions.LastTaken = CURRENT_TIMESTAMP WHERE
132 prescriptions.Id = '". $i['Id']. "'";
133                 $stmt3 = $mysqli->prepare($sql);
134                 $stmt3->execute();
135             }
136         }
137     ?>
138 </tbody>
139 </table>
140 </div>
141 <div class="bs-docs-section">
142     <h3 id="doctors" class="page-header">Doctors
143 <small>Here are all the doctors that have you under their
144 care</small></h3>
145     <?php
146     $stmt1 = "SELECT `username` , `email` , `telephone`
147 FROM `treated` , `members`
148 WHERE treated.DUsername = members.username
149 AND PUsername = '". $_SESSION['username']. "'";
150     $result1 = mysqli_query($mysqli,$stmt1);
151     while ($i = mysqli_fetch_array($result1))
152         $p[] = $i;
153     ?>
154
155     <table class="table">
156         <thead>
157             <caption>This table contains all the
158 usernames and e-mail of the doctors that have you under their
159 care</caption>

```



```

160         <tr>
161             <th>Doctor</th>
162             <th>e-mail</th>
163         </tr>
164     </thead>
165     <tbody>
166         <?php
167         foreach ($p as $z) {
168             printf("<tr> <td>%s</td> <td>%s</td>
169 </tr>\n", $z['username'], $z['email']);
170         }
171         ?>
172     </tbody>
173 </table>
174 </div>
175 <div class="bs-docs-section">
176     <h3 id="profile" class="page-
177 header">Profile<small> Here you can see the information with
178 which you registered</small></h3>
179     <?php
180     $stmt2 = "SELECT *
181     FROM `members`
182     WHERE username = '".$_SESSION['username']."'";
183     $result = mysqli_query($mysqli, $stmt2);
184     $i = mysqli_fetch_array($result);
185     ?>
186     <table class="table">
187         <thead>
188             <caption>In this table you can see the
189 information you have given during your registration.</caption>
190             <tr>
191                 <th>Username</th>
192                 <th>e-mail</th>
193                 <th>Capacity</th>
194             </tr>
195         </thead>
196         <tbody>
197             <tr>
198                 <td> <?php echo $i['username']; ?>
199 </td>
200                 <td> <?php echo $i['email']; ?> </td>
201                 <td> <?php echo $i['capacity']; ?>
202 </td>
203             </tr>
204         </tbody>
205     </table>
206 </div>
207 </div>
208 </div>
209 </div>
210 <script type="text/javascript">
211

```

```

212 update();
213 setInterval('update()',1000);
214 document.getElementById('x').innerHTML=1;
215
216 </script>
217
218
219 </body>
220 </html>

```

xii. add_doctor.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.6 Σελίδα Ασθενή, στην σελίδα 27. Το αρχείο αυτό, περιέχει τον κώδικα της σελίδας για την πραγματοποίηση των αιτημάτων από τον ασθενή για σύνδεση με νέο γιατρό.

```

1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';
4 // include_once 'includes/Doctor.inc.php';
5
6 sec_session_start();
7 ?>
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta charset="UTF-8">
12     <title>Doactors's patient list</title>
13     <script type="text/JavaScript"
14     src="js/sha512.js"></script>
15     <script type="text/JavaScript" src="js/forms.js"></script>
16     <meta name="viewport" content="width=device-width,
17     initial-scale=1"><!-- defining responsivnes in mobile devices
18     -->
19     <link href="css/bootstrap.css" rel="stylesheet">
20     <link href="css/bootstrap.min.css" rel="stylesheet">
21     <link href="css/bootstrap-theme.css" rel="stylesheet">
22     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
23     <link href="css/style.css" rel="stylesheet"> <!-- styling
24     link -->
25 </head>
26 <body>
27     <div id="patient">
28         <div class="page-header"> <!-- page header: preferred
29     because there was nothing to put in the navbar -->
30         <h1>Welcome Mr/Mrs <?php echo
31     htmlentities($_SESSION['username']); ?>
32         </h1>
33     </div>
34     <div class="container-fluid">

```

```

35         <div class=" col-md-2" role="complementary">
36             <nav class="bs-docs-sidebar hidden-print hidden-xs
37 hidden-sm">
38                 <ul class="nav bs-docs-sidenav">
39                     <li>
40                         <a
41 href="Patient.php#therapies">Therapies</a>
42                     </li>
43                     <li>
44                         <a
45 href="Patient.php#doctors">Doctors</a>
46                     </li>
47                     <li>
48                         <a href="#">Add a new Doctor to the
49 doctor list</a>
50                     </li>
51                     <li>
52                         <a
53 href="Patient.php#profile">Profile</a>
54                     </li>
55                     <li class="">
56                         <?php echo '<a
57 href="includes/logout.php">Log Out</a>' ?>
58                     </li>
59                     <li>
60                         <a href="#">Back to the top of the
61 Page</a>
62                     </li>
63                 </ul>
64             </nav>
65         </div>
66
67         <div class="col-md-10">
68             <div class="bs-docs-section">
69                 <h2 id="present" class="page-header">Add a new
70 Doctor in your doctor list</h2>
71                 <form action= "#" method= "POST" >
72                     <table class="table">
73                         <caption>Select the patients that you want
74 to add to your a doctor list. The doctor must also add you to
75 his/her doctor list for confirmation </caption>
76                         <thead>
77                             <tr>
78                                 <th>Select</th>
79                                 <th>Doctor Username</th>
80                             </tr>
81                         </thead>
82                         <tbody>
83                             <?php
84                                 $stmt = "SELECT username
85 FROM `members`

```

```

87         WHERE members.capacity = 'doctor';
88         $result = mysqli_query($mysqli,$stmt);
89         while ($i = mysqli_fetch_array($result))
90             $p[] = $i;
91
92         foreach($p as $doctor){
93
94             $line = $doctor['username'];
95
96         printf("<tr><td>%s</td><td>%s</td></tr>\n", '<input
97         type="checkbox" name="doctor" value=".' . $doctor['username'] . '>',
98         $doctor['username']);
99             }
100
101         if (isset($_POST['doctor'])) {
102             $sql= "INSERT INTO admin_temp (PUsername,
103             DUsername, SubmittedBy)
104             VALUES ('".$_SESSION['username']."',
105             '".$_POST['doctor']."', 'patient')";
106
107             $ret = mysqli_query($mysqli, $sql);
108
109             if (!$ret)
110             {
111                 die('Error: ' .
112                 mysqli_error($mysqli));
113             }
114             else {
115                 echo "1 record added";
116             }
117         }
118         ?>
119     </tbody>
120
121 </table>
122 <br>
123 <input type="Submit" class="btn btn-default btn-lg"
124 value="Submit" />
125
126 </form>
127 </div>
128 </div>
129 </div>
130 </div>
131 </body>
132 </html>

```

xiii. Doctor.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.7 Σελίδα Γιατρού, στην σελίδα 28. Στο αρχείο αυτό, υπάρχει ο κώδικας για την σελίδα που εμφανίζεται στον

γιατρό χρήστη και οι δυνατότητες που του προσφέρονται από την μελέτη. Δεν περιέχεται ο κώδικας, για την σελίδα με την οποία πραγματοποιείται το αίτημα για τη διασύνδεση με νέο ασθενή και για τη σελίδα εισαγωγής νέας θεραπείας στην βάση δεδομένων.

```
1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';
4 // include_once 'includes/Doctor.inc.php';
5
6 sec_session_start();
7 ?>
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta charset="UTF-8">
12     <title>Doctor's page</title>
13     <script type="text/JavaScript"
14 src="js/sha512.js"></script>
15     <script type="text/JavaScript" src="js/forms.js"></script>
16     <meta name="viewport" content="width=device-width,
17 initial-scale=1"><!-- defining responsivnes in mobile devices
18 -->
19     <link href="css/bootstrap.css" rel="stylesheet">
20     <link href="css/bootstrap.min.css" rel="stylesheet">
21     <link href="css/bootstrap-theme.css" rel="stylesheet">
22     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
23     <link href="css/style.css" rel="stylesheet"> <!-- styling
24 link -->
25 </head>
26 <body>
27     <div id="patient">
28         <div class="page-header"> <!-- page header: preferred
29 because there was nothing to put in the navibar -->
30         <h1>Welcome Dr. <?php echo
31 htmlentities($_SESSION['username']); ?>
32         </h1>
33     </div>
34     <div class="container-fluid">
35         <div class=" col-md-2" role="complementary">
36             <nav class="bs-docs-sidebar hidden-print
37 hidden-xs hidden-sm">
38                 <ul class="nav bs-docs-sidenav">
39                     <li>
40                         <a href="#present">Already Given
41 Therapies</a>
42                     </li>
43                     <li>
44                         <a href="#asignment">Assign a
45 Therapy to a Patient</a>
```

```

46         </li>
47         <li>
48             <a href="new_therapy.php">Register
49 a NEW Therapy</a>
50         </li>
51         <li>
52             <a href="add_patient.php">Add a
53 Patient to the patient list</a>
54         </li>
55         <li>
56             <a href="#profile">Profile</a>
57         </li>
58         <li class="">
59             <?php echo '<a
60 href="includes/logout.php">Log Out</a>' ?>
61         </li>
62
63         <li>
64             <a href="#">Back to the top of the
65 Page</a>
66         </li>
67     </ul>
68 </nav>
69 </div>
70
71     <div class="col-md-10">
72         <div class="bs-docs-section">
73             <h2 id="present" class="page-header">Your
74 Patients and their Dosages <small>Here you can see all the
75 prescriptions you have administered</small></h2>
76             <?php
77             $stmt = "SELECT *
78 FROM `prescriptions`, `drugs`
79 WHERE prescriptions.Drug = drugs.Id
80 AND Doctor = '".$_SESSION['username']."'";
81 $result = mysqli_query($mysqli,$stmt);
82 while ($i = mysqli_fetch_array($result))
83     $p[] = $i;
84             ?>
85             <table class="table">
86                 <thead>
87                     <caption>This table contains the
88 user name of the patient the drug that was prescribed ans the
89 dosage in hours</caption>
90                     <tr>
91                         <th>Patient</th>
92                         <th>Drug</th>
93                         <th>Every # Hour(s)</th>
94                     </tr>
95                 </thead>
96                 <tbody>
97                     <?php

```

```

98         foreach ($p as $i) {
99
100     printf("<tr><td>%s</td><td>%s</td><td>%s</td></tr>\n",$i['Pati
101     ent'],$i['Name'],$i['Dosage']);
102         }
103         ?>
104         </tbody>
105     </table>
106 </div>
107
108     <div class="bs-docs-section">
109         <h2 id="asignment" class="page-
110     header">Assign a Therapy to a Patient <small>Here you can make
111     a new prescription to one of your patients</small></h2>
112
113         <form action= "#" method= "POST" >
114
115             <?php
116                 $sql1 = "SELECT PUsername
117                 FROM `treated`
118                 WHERE DUsername =
119     '".$_SESSION['username']."'";
120                 $results1 = mysqli_query($mysqli,
121     $sql1);
122                 while ($i =
123     mysqli_fetch_array($results1))
124                     $patients[] = $i;
125                 ?>
126
127                 <h3>Patient Username: <small>The
128     usernames of all the patient already in your patient
129     list</small></h3>
130                 <select name='patient' id='patient'
131     class="form-control input-lg">
132                     <?php
133                         foreach($patients as $patient)
134                             printf("<option>%s</option>",
135     $patient['PUsername']);
136                     ?>
137                 </select>
138                 <br>
139
140                 <?php
141                 $sql2 = "SELECT *
142                 FROM Drugs
143                 WHERE DUsername =
144     '".$_SESSION['username']."'";
145                 $results2 = mysqli_query($mysqli,
146     $sql2);
147                 while ($j =
148     mysqli_fetch_array($results2))
149                     $drugs[] = $j;

```

```

150         ?>
151
152         <h3>Drug Name: <small>The unique Id's
153 and names of the therapies you have entered in the
154 system.</small></h3>
155         <select name='drug' id='drug'
156 class="form-control input-lg">
157             <?php
158                 foreach($drugs as $drug)
159                     printf("<option> %s
160 %s</option>", $drug['Id'], $drug['Name']);
161             ?>
162         </select>
163         <br>
164         <h3>Dosage <small>The dosage of the
165 prescription in hours (1-8,12,24 hours)</small></h3>
166         <select name='dosage' class="form-
167 control input-lg">
168             <option value='1'>every 1
169 hour</option>
170             <option value='2'>every 2
171 hours</option>
172             <option value='3'>every 3
173 hours</option>
174             <option value='4'>every 4
175 hours</option>
176             <option value='5'>every 5
177 hours</option>
178             <option value='6'>every 6
179 hours</option>
180             <option value='7'>every 7
181 hours</option>
182             <option value='8'>every 8
183 hours</option>
184             <option value='12'>every 12
185 hours</option>
186             <option value='12'>every 24
187 hours</option>
188         </select>
189         <?php
190             if (isset($_POST['patient'])) {
191                 $sql= "INSERT INTO prescriptions
192 (Doctor, Drug, Patient, Dosage, LastTaken)
193                 VALUES
194 ('".$_SESSION['username']."', '".$_POST['drug']."',
195 '".$_POST['patient']."', '".$_POST['dosage']."',
196 CURRENT_TIMESTAMP)";
197
198                 $ret = mysqli_query($mysqli,
199 $sql);
200
201                 if (!$ret)

```



```

202         {
203             die('Error: ' .
204 mysqli_error($mysqli));
205         }
206         else {
207             echo "1 record added";
208         }
209     }
210     ?>
211     <br>
212     <input type="Submit" class="btn btn-
213 default btn-lg" value="Submit" />
214     </form>
215 </div>
216
217     <div class="bs-docs-section">
218         <h3 id="profile" class="page-
219 header">Profile <small>Here you can see the information you
220 gave upon registration</small></h3>
221         <?php
222         $stmt2 = "SELECT *
223 FROM `members`
224 WHERE username =
225 '".$_SESSION['username']."'";
226         $result = mysqli_query($mysqli, $stmt2);
227         $i = mysqli_fetch_array($result);
228         ?>
229         <table class="table">
230             <thead>
231                 <caption>Here you can see the
232 information you gave upon registration</caption>
233                 <tr>
234                     <th>Username</th>
235                     <th>e-mail</th>
236                     <th>Capacity</th>
237                 </tr>
238             </thead>
239             <tbody>
240                 <td> <?php echo $i['username']; ?>
241 </td>
242                 <td> <?php echo $i['email']; ?>
243 </td>
244                 <td> <?php echo $i['capacity']; ?>
245 </td>
246             </tbody>
247         </table>
248     </div>
249 </div>
250 </div>
251 </div>
252 </body>
253 </html>

```

xiv. new_therapy.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.7 Σελίδα Γιατρού, στην σελίδα 30. Το αρχείο αυτό, περιέχει τον κώδικα της σελίδας για την εισαγωγή νέας θεραπείας από τον γιατρό στην βάση δεδομένων.

```
1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';
4 // include_once 'includes/Doctor.inc.php';
5
6 sec_session_start();
7 ?>
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta charset="UTF-8">
12     <title>Add a new therapy</title>
13     <script type="text/JavaScript"
14 src="js/sha512.js"></script>
15     <script type="text/JavaScript" src="js/forms.js"></script>
16     <meta name="viewport" content="width=device-width,
17 initial-scale=1"><!-- defining responsivnes in mobile devices
18 -->
19     <link href="css/bootstrap.css" rel="stylesheet">
20     <link href="css/bootstrap.min.css" rel="stylesheet">
21     <link href="css/bootstrap-theme.css" rel="stylesheet">
22     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
23     <link href="css/style.css" rel="stylesheet"> <!-- styling
24 link -->
25 </head>
26 <body>
27     <div id="patient">
28         <div class="page-header"> <!-- page header: preferred
29 because there was nothing to put in the navbar -->
30             <h1>Welcome Dr. <?php echo
31 htmlentities($_SESSION['username']); ?>
32             </h1>
33         </div>
34         <div class="container-fluid">
35             <div class=" col-md-2" role="complementary">
36                 <nav class="bs-docs-sidebar hidden-print
37 hidden-xs hidden-sm">
38                     <ul class="nav bs-docs-sidenav">
39                         <li>
40                             <a
41 href="Doctor.php#present">Already Given Therapies</a>
42                             </li>
43                         <li>
44                             <a
45 href="Doctor.php#asignment">Assign a Therapy to a Patient</a>
```

```

46         </li>
47         <li>
48             <a
49 href="Doctor.php#therapy">Register a NEW Therapy</a>
50         </li>
51         <li>
52             <a href="add_patient.php">Add a
53 new Patient to the patient list</a>
54         </li>
55         <li>
56             <a
57 href="Doctor.php#profile">Profile</a>
58         </li>
59         <li class="">
60             <?php echo '<a
61 href="includes/logout.php">Log Out</a>' ?>
62         </li>
63
64         <li>
65             <a href="#">Back to the top of the
66 Page</a>
67         </li>
68     </ul>
69 </nav>
70 </div>
71
72     <div class="col-md-10">
73         <div class="bs-docs-section">
74             <h3 id="therapy" class="page-
75 header">Register a NEW Therapy<small> Insert a new therapy to
76 the system</small></h3>
77             <form action= "#" method= "POST" >
78
79                 <h3>Name <small>The name of the
80 drug</small></h3> <input type="text" name="name" id="name"
81 class="form-control input-lg"/>
82                 <h3>Substance <small>The name of the
83 active substance of the drug</small></h3> <input type="text"
84 name="substance" id="substance" class="form-control input-
85 lg"/>
86                 <h3>Content <small>The amount of the
87 active substance in mg</small></h3> <input type="text"
88 name="content" id="content" class="form-control input-lg"/>
89                 <h3>Sort description/ Site with side
90 effects <small>A small description of the drug </small></h3>
91 <textarea name="effects" id="effects" class="form-control
92 input-lg"></textarea>
93                 <?php
94                     if (isset($_POST['name'])) {
95                         $sql= "INSERT INTO drugs (Name,
96 Substance, Content, Description, DUsername)

```

```

97             VALUES ('"$_POST['name']."',
98 '"$_POST['substance']."', '"$_POST['content']."',
99 '"$_POST['effects']."', '"$_SESSION['username']."'");
100
101             $ret = mysqli_query($mysqli,
102 $sql);
103
104             if (!$ret)
105             {
106                 die('Error: ' .
107 mysqli_error($mysqli));
108             }
109             else {
110                 echo "1 record added";
111             }
112         }
113     ?>
114     <br>
115     <input type="Submit" class="btn btn-
116 default btn-lg" value="Submit" />
117     </form>
118
119     </div>
120 </div>
121 </div>
122 </div>
123 </div>
124 </body>
125 </html>

```

xv. add_patient.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.7 Σελίδα Γιατρού, στην σελίδα 30. Το αρχείο αυτό, περιέχει τον κώδικα της σελίδας για την πραγματοποίηση των αιτημάτων από τον γιατρό για σύνδεση με νέο ασθενή.

```

1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';
4 // include_once 'includes/Doctor.inc.php';
5
6 sec_session_start();
7 ?>
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta charset="UTF-8">
12     <title>Doctors's patient list</title>
13     <script type="text/JavaScript"
14 src="js/sha512.js"></script>

```

```

15     <script type="text/JavaScript" src="js/forms.js"></script>
16     <meta name="viewport" content="width=device-width,
17 initial-scale=1"><!-- defining responsivnes in mobile devices
18 -->
19     <link href="css/bootstrap.css" rel="stylesheet">
20     <link href="css/bootstrap.min.css" rel="stylesheet">
21     <link href="css/bootstrap-theme.css" rel="stylesheet">
22     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
23     <link href="css/style.css" rel="stylesheet"> <!-- styling
24 link -->
25 </head>
26 <body>
27     <div id="patient">
28         <div class="page-header"> <!-- page header: preferred
29 because there was nothing to put in the navibar -->
30             <h1>Welcome Dr. <?php echo
31 htmlentities($_SESSION['username']); ?>
32             </h1>
33         </div>
34         <div class="container-fluid">
35             <div class=" col-md-2" role="complementary">
36                 <nav class="bs-docs-sidebar hidden-print
37 hidden-xs hidden-sm">
38                     <ul class="nav bs-docs-sidenav">
39                         <li>
40                             <a
41 href="Doctor.php#present">Allready Given Therapies</a>
42                             </li>
43                             <li>
44                             <a
45 href="Doctor.php#asignment">Asigne a Therapy to a Patient</a>
46                             </li>
47                             <li>
48                                 <a href="new_therapy.php">Register
49 a NEW Therapy</a>
50                             </li>
51                             <li>
52                                 <a href="#">Add a new Patient to
53 the patient list</a>
54                             </li>
55                             <li>
56                                 <a
57 href="Doctor.php#profile">Profile</a>
58                             </li>
59                             <li class="">
60                                 <?php echo ' <a
61 href="includes/logout.php">Log Out</a>' ?>
62                                 </li>
63                             <li>
64                                 <a href="#">Back to the top of the
65 Page</a>
66

```

```

67         </li>
68     </ul>
69 </nav>
70 </div>
71
72     <div class="col-md-10">
73         <div class="bs-docs-section">
74             <h2 id="present" class="page-header">Add a
75 new Patent in your patient list</h2>
76             <form action= "#" method= "POST" >
77                 <table class="table">
78                     <thead>
79                         <caption>Select the patients
80 that you want to add to you aptient list. The patient must
81 also add you to his/her doctor list for confirmation</caption>
82                     <tr>
83                         <th>Select</th>
84                         <th>Patient Username</th>
85                     </tr>
86                 </thead>
87                 <tbody>
88                     <?php
89                         $stmt = "SELECT username
90 FROM `members`
91 WHERE members.capacity =
92 'patient'";
93                         $result =
94 mysqli_query($mysqli,$stmt);
95                         while ($i =
96 mysqli_fetch_array($result))
97                             $p[] = $i;
98
99                         foreach($p as $patient){
100
101                             $line = $patient['username'];
102
103 printf("<tr><td>%s</td><td>%s</td></tr>\n", '<input
104 type="checkbox" name="patient"
105 value=".'$patient['username'].'>', $patient['username']);
106                             }
107
108                             if (isset($_POST['patient'])) {
109                                 $sql= "INSERT INTO admin_temp
110 (DUsername, PUsername, SubmittedBy)
111                                 VALUES
112 ('".$_SESSION['username']."', '".$_POST['patient']."',
113 'doctor')";
114
115                                 $ret = mysqli_query($mysqli,
116 $sql);
117
118                                 if (!$ret)

```

```

119         {
120             die('Error: ' .
121 mysqli_error($mysqli));
122         }
123         else {
124             echo "1 record added";
125         }
126     }
127 }
128 }
129 ?>
130 </tbody>
131
132 </table>
133 <br>
134 <input type="Submit" class="btn btn-default
135 btn-lg" value="Submit" />
136
137 </form>
138 </div>
139 </div>
140 </div>
141 </div>
142 </body>

```

xvi. Admin.php

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.8 Σελίδα Διαχειριστή, στην σελίδα 31. Το αρχείο αυτό, περιέχει τον κώδικα της σελίδας του διαχειριστή.

```

1 <?php
2 include_once 'includes/db_connect.php';
3 include_once 'includes/functions.php';
4
5 sec_session_start();
6 ?>
7
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta charset="UTF-8">
12     <!-- <meta http-equiv="refresh" content="1"> -->
13     <title>Admin's page</title>
14     <script type="text/JavaScript"
15 src="js/sha512.js"></script>
16     <script type="text/JavaScript" src="js/forms.js"></script>
17     <meta name="viewport" content="width=device-width,
18 initial-scale=1"><!-- defining responsivnes in mobile devices
19 -->
20     <link href="css/bootstrap.css" rel="stylesheet">

```

```

21     <link href="css/bootstrap.min.css" rel="stylesheet">
22     <link href="css/bootstrap-theme.css" rel="stylesheet">
23     <link href="css/bootstrap-theme.min.css" rel="stylesheet">
24     <link href="css/style.css" rel="stylesheet"> <!-- styling
25 link -->
26 </head>
27 <body>
28     <div id="admin">
29         <div class="page-header"> <!-- page header: preferred
30 because there was nothing to put in the navbar -->
31             <h1>Welcome <?php echo
32 htmlentities($_SESSION['username']); ?>
33             </h1>
34         </div>
35         <div class="container-fluid">
36             <div class="col-md-2" role="complementary">
37                 <nav class="bs-docs-sidebar hidden-print
38 hidden-xs hidden-sm">
39                     <ul class="nav bs-docs-sidenav">
40
41                         <li class="">
42                             <?php echo '<a
43 href="includes/logout.php">Log Out</a>' ?>
44                             </li>
45
46                         <li>
47                             <a href="#">Back to the top of the
48 Page</a>
49                             </li>
50                     </ul>
51                 </nav>
52             </div>
53             <div class="col-md-10">
54                 <!-- <h5>If a row is double there will be the
55 opportunity to insert it in the therapies table </h5> -->
56
57                 <div class="bs-docs-section">
58
59                     <table class="table">
60                         <caption>This table contains all the
61 requests for connection between a doctor and a patient.
62                         If a row is double AND the origin is
63 once from the doctor and the other from the patient it must be
64 inserted in the therapies table </caption>
65                         <thead>
66                             <tr>
67                                 <th>Check</th>
68                                 <th>Patient</th>
69                                 <th>Doctor</th>
70                                 <th>Submitted by the</th>
71                             </tr>
72                         </thead>

```



```

73         <tbody>
74
75             <?php
76                 $stmt = "SELECT *
77                 FROM `admin_temp`
78                 ORDER BY PUsername, DUsername";
79                 $result =
80 mysqli_query($mysqli,$stmt);
81                 while ($i =
82 mysqli_fetch_array($result))
83                     $p[] = $i;
84                 foreach ($p as $i) {
85
86 printf("<tr><td>%s</td><td>%s</td><td>%s</td><td>%s</td></tr>\n",
87 '<input type="checkbox" name="entry"
88 value="('.$i['DUsername'].'.$i['PUsername'].')">' ,
89 $i['PUsername'], $i['DUsername'], $i['SubmittedBy']);
90                 }
91                 if (isset($_POST['entry'])) {
92
93                     $sql= "INSERT INTO treated
94 (PUsername, DUsername)
95                     VALUES
96 ('".$_POST['PUsername']."', '".$_POST['DUsername']."')";
97
98                     $ret = mysqli_query($mysqli,
99 $sql);
100
101                     if (!$ret)
102                     {
103                         die('Error: ' .
104 mysqli_error($mysqli));
105                     }
106                     else {
107                         echo "1 record added";
108
109
110                     }
111                 }
112             ?>
113         </tbody>
114     </table>
115     <br>
116     <input type="Submit" class="btn btn-
117 default btn-lg" value="Submit" />
118
119 </div>
120
121 </div>
122
123
124

```

```
125         </body>
126     </html>
```

II. Κώδικας CSS

i. style.css

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.9 Αρχείο Μορφοποίησης, στις σελίδες 32 και 33. Ο ρόλος του κώδικα αυτού, είναι η μορφοποίηση των ιστοσελίδων στις οποίες εισάγεται - καλείται.

```
1  /* general styling */
2  body {
3      background-repeat: no-repeat;
4      background: #E5E5E5; /* Standard syntax */
5      padding-bottom: 30px;
6  }
7
8  .page-header {
9      border-bottom: 1px solid #400000;
10     padding-left: 3%
11 }
```

III. Κώδικας SQL

i. secure_login.sql

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.1 Βάση Δεδομένων, στις σελίδες 20, 21 και 22. Το αρχείο αυτό, περιέχει τον κώδικα για την δημιουργία της βάσης δεδομένων και την δημιουργία και πλήρωση των πινάκων της.

```
1  -- phpMyAdmin SQL Dump
2  -- version 4.2.11
3  -- http://www.phpmyadmin.net
4  --
5  -- Host: 127.0.0.1
6  -- Generation Time: May 26, 2015 at 06:06 PM
7  -- Server version: 5.6.21
8  -- PHP Version: 5.6.3
9
10 SET FOREIGN_KEY_CHECKS=0;
11 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
12
13
14 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
15 */;
16 /*!40101 SET
17 @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
```

```

18  /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
19  */;
20  /*!40101 SET NAMES utf8 */;
21
22  --
23  -- Database: `secure_login`
24  --
25  CREATE DATABASE IF NOT EXISTS `secure_login` DEFAULT CHARACTER
26  SET latin1 COLLATE latin1_swedish_ci;
27  USE `secure_login`;
28
29  -- -----
30
31  --
32  -- Table structure for table `admin_temp`
33  --
34
35  DROP TABLE IF EXISTS `admin_temp`;
36  CREATE TABLE IF NOT EXISTS `admin_temp` (
37    `DUsername` varchar(30) NOT NULL,
38    `PUsername` varchar(30) NOT NULL,
39    `SubmittedBy` varchar(10) NOT NULL
40  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
41
42  --
43  -- Dumping data for table `admin_temp`
44  --
45
46  INSERT INTO `admin_temp` (`DUsername`, `PUsername`,
47  `SubmittedBy`) VALUES
48  ('drWho1', 'markella', 'doctor'),
49  ('markella', 'drWho2', 'patient'),
50  ('markella', 'drWho2', 'patient'),
51  ('markella', 'drWho2', 'patient'),
52  ('drWho1', 'markella', 'patient');
53
54  -- -----
55
56  --
57  -- Table structure for table `drugs`
58  --
59
60  DROP TABLE IF EXISTS `drugs`;
61  CREATE TABLE IF NOT EXISTS `drugs` (
62    `Name` varchar(20) NOT NULL DEFAULT '',
63    `Id` int(11) NOT NULL,
64    `Substance` varchar(20) DEFAULT '100',
65    `Content` float DEFAULT '200',
66    `Description` varchar(20) DEFAULT NULL,
67    `DUsername` varchar(30) NOT NULL,
68  ) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=latin1;
69

```

```

70 --
71 -- Dumping data for table `drugs`
72 --
73
74 INSERT INTO `drugs` (`Name`, `Id`, `Substance`, `Content`,
75 `Description`, `DUsername`, `Take`) VALUES
76 ('aspirin', 1, '100', 200, NULL, 'drWho1', NULL),
77 ('depon', 2, '100', 200, NULL, 'drWho1', NULL),
78 ('valium', 3, '100', 200, NULL, 'drWho1', NULL),
79 ('ponstan', 4, '100', 200, NULL, 'drWho1', NULL),
80 ('buscopan', 5, '', 0, '', 'drWho1', ''),
81 ('aspirin', 6, '100', 200, NULL, 'drWho2', NULL),
82 ('ponstan', 7, '100', 200, NULL, 'drWho2', NULL),
83 ('buscopan', 8, '', 0, '', 'drWho2', ''),
84 ('opatanol', 11, '', 0, 'eyedrops', 'drWho1', NULL);
85
86 -----
87
88 --
89 -- Table structure for table `login_attempts`
90 --
91
92 DROP TABLE IF EXISTS `login_attempts`;
93 CREATE TABLE IF NOT EXISTS `login_attempts` (
94   `user_id` int(11) NOT NULL,
95   `time` varchar(30) NOT NULL
96 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
97
98 --
99 -- Dumping data for table `login_attempts`
100 --
101
102 INSERT INTO `login_attempts` (`user_id`, `time`) VALUES
103 (1, '1424803629'),
104 (1, '1424803649'),
105 (2, '1424809489'),
106 (2, '1426173583'),
107 (5, '1426179372'),
108 (2, '1426459710'),
109 (2, '1426459720'),
110 (5, '1426785238'),
111 (2, '1427117981'),
112 (15, '1430744724'),
113 (15, '1430744736'),
114 (8, '1430745055'),
115 (5, '1430911242'),
116 (5, '1432481058');
117
118 -----
119
120 --
121 -- Table structure for table `members`

```

```

122 --
123
124 DROP TABLE IF EXISTS `members`;
125 CREATE TABLE IF NOT EXISTS `members` (
126   `id` int(11) NOT NULL,
127   `username` varchar(30) NOT NULL,
128   `email` varchar(50) NOT NULL,
129   `telephone` int(20) NOT NULL,
130   `capacity` varchar(10) NOT NULL,
131   `password` char(128) NOT NULL,
132   `salt` char(128) NOT NULL
133 ) ENGINE=InnoDB AUTO_INCREMENT=20 DEFAULT CHARSET=latin1;
134
135 --
136 -- Dumping data for table `members`
137 --
138
139 INSERT INTO `members` (`id`, `username`, `email`, `telephone`,
140 `capacity`, `password`, `salt`) VALUES
141 (1, 'test_user', 'test@example.com', 0, 'patient',
142 '00807432eae173f652f2064bdca1b61b290b52d40e429a7d295d76a71084a
143 a96c0233b82f1feac45529e0726559645acaed6f3ae58a286b9f075916ebf6
144 6cacc',
145 'f9aab579fc1b41ed0c44fe4ecdbfcd4cb99b9023abb241a6db833288f4ee
146 a3c02f76e0d35204a8695077dcf81932aa59006423976224be0390395bae15
147 2d4ef'),
148 (2, 'markella', 'asdf@asdf.asd', 2147483647, 'patient',
149 'b6887e8a6ff70470827f448810a846111cc5ac6c1c422f5383dd9c8dcf782
150 33dee04c55cde7b85aaf04ff91fe17c14c18bfa1110570ce986e4a0de52566
151 d646c',
152 'f38084c35ffb1d4ee3313a10a62110f5c5a75cb5e31571f81284ed60c2b69
153 9c4b167dc391a12a7b1dbb72e28846e88399ad26a3c26bebc52ca06d7ac9f7
154 7150c'),
155 (3, 'user', 'user@user.usr', 0, 'patient',
156 '51a843a5b02a12dd20243d3752cf68a0c0e13f8a31b7bab03df83992a0888
157 27ab7833cde1d56f11246dd6419bccad41e3683ad770b96a35a2f078fafdbd
158 75e06',
159 '9c7ba559fe5d92c8c9a066afe0f4df1da5e1ce1a04605243399cb7897a740
160 a56b8110784e02ecbf8eb6a3e6ca978a3c42fbef0f37fa523dbfa840ca986e
161 5713e'),
162 (4, 'drseuss', 'drseuss@user.usr', 0, 'doctor',
163 '01cb38c87345ba7b789b69db0959d33761e4a4777f90815eb9a24062d611f
164 b81498e10cd48c2768fb48370b62918094696ab14ae61bda39b597460fb048
165 83190',
166 '376a9180db6f098db3d63a94c68a0f4013dc5ec984b6f1908cf2356fabff0
167 81dafe95b542f1a44e9eed7d14e24d5300819b18952498db820a361a89488d
168 d548d'),
169 (5, 'drWho1', 'doctor@chanel.asd', 0, 'doctor',
170 '8db379f820a3e46837de2a97e8b48ddc6e2365d0e7dbfa71f51a0f372aa0c
171 974e413e96e73ac3d066ac93f1e0c206014c5b1c095bd790cab65d8a9c349f
172 ef662',
173 '95d047e53f07e11a6436983ba747b04a5b0c7b269ba1f16cd5c4cff1bb9b0

```

174 083d19a4907e61aa4d7bca1f4e49ef9549581f2411ef85fc0610f02dd7699d
175 dee4b'),
176 (8, 'drWho2', 'doctor2@chanel.asd', 0, 'doctor',
177 'cb8dceb5316ab84d107fa28f4034400078aebfc1e86ed3390ac70e95067e3
178 1e0789be0e39d35a04e71d701d1e15da0024c6cc167f1757baeb0efb56c50f
179 344cc',
180 '5fa02a7ae4d8e9271c47bd8d30df69df99da705fad5f8a5700fe3823fbc29
181 472a0bff95b145703061622e24327069222d7ab068ed4c6c9a6f77a8cd2f69
182 e3d6c'),
183 (9, 'drWho3', 'doctor3@chanel.asd', 0, 'doctor',
184 'ee6065128fd598cf3093600390236425940b263f71e4fc7936adb3311b168
185 9cd30af01e03d1d2caa4eacd7cb3f045d5439cb71c353d7eb605ff3d73c730
186 2343d',
187 '9caf538d1e3e216884e92296b00f131c47f9117376b9c6693472142056309
188 8863a377295bb6f6104267b4057db2f4b171aa22eb5e2c96eea23626a84e79
189 757d0'),
190 (13, 'drWho4', 'doctor4@chanel.asd', 0, 'doctor',
191 '80c33b53e4a0ce6db9c0ca5901883ddbff6e9b7f3ed603e30310c91006b3e
192 40f8906407bed2ebc765b7d43712aa6a181e21d66f5ad6593701cabfab9260
193 c02c0',
194 '296d9691aed5e94d9153761142b8e7b6448e0e860b1ea0be81f2ad65e11c0
195 33481ae7d21d88a81255f89058647742493e285ba1d73141c33fc985395d98
196 74807'),
197 (14, 'sheldon', 'drSheldonCooper@cbs.tv', 0, 'patient',
198 'f9b5695b7cdaf114e8fc6a60567218f74726337bea8887289a77ecee913d6
199 9a799d4f9e55b5417262e8c4a9a395bcf0511d4c4333583431faa8eae0470e
200 08ed1',
201 '8acce05a4d5278e7fa8e1b6216b76f7d1a57aa643b177b4891595cd8b6593
202 e14f3b5fd0568efd8ceb98cb00951ba32c89cc7dbef6b18ef4ff0a78d6882e
203 5c176'),
204 (16, 'admin', 'admin@admin.co', 0, 'admin',
205 '677c39134b7be6542e4c1065f0e8a630e86b6afb6901662b32ec83e0871b3
206 adac12fe7018bf28a6d12ea465a06058bb9f386b233a268d89ad0848ee515f
207 88993',
208 '765d222b5d89b1724b17ee7378c7ff7b5c65086adeb7a4e8cb7dd0a915c87
209 4ef324ce8bd2fa7dca41d8722eb3c18883f9e382e0828fbc1e9b6167ee3c8c
210 36c03'),
211 (17, 'penny', 'penny@abc.tv', 0, 'patient',
212 'fe5ff95ce820723edc63af02cb4b4069e6e87912bf211e7b1b3b002fa8969
213 e811db69624e87ff7ec3a4ef78636267b3b75437b3ae18d4e96f2ace281d46
214 e1aa8',
215 '692694ae690bfa54f4f4bda582713c37a75b22168ad12092675cbd5da5c54
216 9ecb32a5bb0b7e7e15e45efe68b98ff59f5a8096bb38b78c062f5cb690d871
217 e42e6'),
218 (18, 'amy', 'amy@abc.tv', 1234567890, 'patient',
219 '9016db7a9b56d6ba4eaf874257625fa7dc340db30aedde86b5ec57d332a47
220 152654fe544f0d996394704c0bc77657309888bb84f51522b0ea4479875bbd
221 681c4',
222 '89c1dc3ea587689ff75a27a61821f49269b59fba3b7df1747f624132348a2
223 c5afe4a5c6e5efc742fa81feb0d151502d7399f88013a0159e5452dfd7cd0d
224 c95ec'),

```

225 (19, 'bernadet', 'berny@abc.tv', 1234567890, 'patient',
226 'e8dd9727ad26e7974d21a54e72bc57a9a900eb7fca8642f2a81072f6ee8a3
227 8265e626e422b0e6f7c6826d25df0792afdf5b31b1df7dd8a2830be7c19f30
228 80014',
229 '2793bd7686107616d7f98b96f578ac8276df27dbb4871568e894c99e6d1f4
230 3186333eeb667a370f26a3adbe87326db08d66b581ae156e838fafcaf1a359
231 02a64');
232
233 -- -----
234
235 --
236 -- Table structure for table `prescriptions`
237 --
238
239 DROP TABLE IF EXISTS `prescriptions`;
240 CREATE TABLE IF NOT EXISTS `prescriptions` (
241 `Id` int(11) NOT NULL,
242 `Doctor` varchar(20) DEFAULT NULL,
243 `Drug` int(11) DEFAULT NULL,
244 `Patient` varchar(20) DEFAULT NULL,
245 `Dosage` int(11) DEFAULT NULL,
246 `LastTaken` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
247 UPDATE CURRENT_TIMESTAMP
248 ) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=latin1;
249
250 --
251 -- Dumping data for table `prescriptions`
252 --
253
254 INSERT INTO `prescriptions` (`Id`, `Doctor`, `Drug`,
255 `Patient`, `Dosage`, `LastTaken`) VALUES
256 (1, 'drWho1', 1, 'markella', 1, '2015-05-26 15:56:35'),
257 (2, 'drWho2', 2, 'sheldon', 2, '2015-05-26 15:56:35'),
258 (4, 'drWho2', 1, 'markella', 1, '2015-05-26 15:56:35'),
259 (5, 'drWho2', 3, 'markella', 3, '2015-05-26 15:56:35'),
260 (6, 'drWho1', 0, 'opatanol', 6, '2015-05-26 15:56:35'),
261 (7, 'drWho1', 0, 'markella', 6, '2015-05-26 15:56:35'),
262 (10, 'drWho1', 4, 'markella', 3, '2015-05-26 15:56:35'),
263 (11, 'drWho1', 11, 'markella', 7, '2015-05-26 15:56:35'),
264 (12, 'drWho2', 8, 'markella', 1, '2015-05-26 15:56:35');
265
266 -- -----
267
268 --
269 -- Table structure for table `treated`
270 --
271
272 DROP TABLE IF EXISTS `treated`;
273 CREATE TABLE IF NOT EXISTS `treated` (
274 `PUsername` varchar(20) NOT NULL,
275 `DUsername` varchar(20) NOT NULL
276 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

277
278 --
279 -- Dumping data for table `treated`
280 --
281
282 INSERT INTO `treated` (`PUsername`, `DUsername`) VALUES
283 ('markella', 'drWho1'),
284 ('markella', 'drWho2'),
285 ('sheldon', 'drWho2');
286
287 --
288 -- Indexes for dumped tables
289 --
290
291 --
292 -- Indexes for table `drugs`
293 --
294 ALTER TABLE `drugs`
295   ADD PRIMARY KEY (`Id`,`Name`);
296
297 --
298 -- Indexes for table `members`
299 --
300 ALTER TABLE `members`
301   ADD PRIMARY KEY (`id`), ADD FULLTEXT KEY `capacity`
302   (`capacity`);
303
304 --
305 -- Indexes for table `prescriptions`
306 --
307 ALTER TABLE `prescriptions`
308   ADD PRIMARY KEY (`Id`);
309
310 --
311 -- Indexes for table `treated`
312 --
313 ALTER TABLE `treated`
314   ADD PRIMARY KEY (`PUsername`, `DUsername`);
315
316 --
317 -- AUTO_INCREMENT for dumped tables
318 --
319
320 --
321 -- AUTO_INCREMENT for table `drugs`
322 --
323 ALTER TABLE `drugs`
324 MODIFY `Id` int(11) NOT NULL AUTO_INCREMENT,AUTO_INCREMENT=12;
325 --
326 -- AUTO_INCREMENT for table `members`
327 --
328 ALTER TABLE `members`

```



```

329 MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,AUTO_INCREMENT=20;
330 --
331 -- AUTO_INCREMENT for table `prescriptions`
332 --
333 ALTER TABLE `prescriptions`
334 MODIFY `Id` int(11) NOT NULL
335 AUTO_INCREMENT,AUTO_INCREMENT=13;SET FOREIGN_KEY_CHECKS=1;
336
337 /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT
338 */;
339 /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS
340 */;
341 /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION
342 */;

```

IV. Κώδικας JavaScript

i. forms.js

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.2.1.3 Αρχική Σελίδα, στις σελίδες 23 και 24. Ο ρόλος του κώδικα αυτού, είναι η επεξεργασία της φόρμας εγγραφής και εισόδου και ο έλεγχος των στοιχείων που εισάγονται σε αυτές, κυρίως των κωδικών πρόσβασης.

```

1 function formhash (form, password) {
2     // Create a new element input, this will be our hashed
3     password field.
4     var p = document.createElement ("input");
5
6     // Add the new element to our form.
7     form.appendChild (p);
8     p.name = "p";
9     p.type = "hidden";
10    p.value = hex_sha512 (password.value);
11
12    // Make sure the plaintext password doesn't get sent.
13    password.value = "";
14
15    // Finally submit the form.
16    form.submit ();
17 }
18
19 function regformhash (form, uid, email, telephone, password,
20 conf, capacity) {
21     // Check each field has a value
22     if (uid.value == '' ||
23         email.value == '' ||
24         telephone.value == '' ||
25         password.value == '' ||
26         conf.value == '' ||
27         capacity.value == '') {

```

```

28
29     alert ('You must provide all the requested details.
30 Please try again');
31     return false;
32 }
33
34 // Check the username
35
36 re = /^w+$/;
37 if (!re.test (form.username.value)) {
38     alert ("Username must contain only letters, numbers
39 and underscores. Please try again");
40     form.username.focus ();
41     return false;
42 }
43
44 // Check that the password is sufficiently long (min 6
45 chars)
46 // The check is duplicated below, but this is included to
47 give more
48 // specific guidance to the user
49 if (password.value.length < 6) {
50     alert ('Passwords must be at least 6 characters long.
51 Please try again');
52     form.password.focus ();
53     return false;
54 }
55
56 // At least one number, one lowercase and one uppercase
57 letter
58 // At least six characters
59
60 var re = /(?=.*\d) (?=.*[a-z]) (?=.*[A-Z]).{6,}/;
61 if (!re.test (password.value)) {
62     alert ('Passwords must contain at least one number,
63 one lowercase and one uppercase letter. Please try again');
64     return false;
65 }
66
67 // Check password and confirmation are the same
68 if (password.value != conf.value) {
69     alert ('Your password and confirmation do not match.
70 Please try again');
71     form.password.focus ();
72     return false;
73 }
74
75 // Create a new element input, this will be our hashed
76 password field.
77 var p = document.createElement ("input");
78
79 // Add the new element to our form.

```

```

80     form.appendChild (p);
81     p.name = "p";
82     p.type = "hidden";
83     p.value = hex_sha512 (password.value);
84
85     // Make sure the plaintext password doesn't get sent.
86     password.value = "";
87     conf.value = "";
88
89     // Finally submit the form.
90     form.submit ();
91     return true;
92 }

```

ii. sha512.js

Ο κώδικας αυτός, αναφέρεται στο υπό-κεφάλαιο 3.4 Ασφάλεια, στις σελίδες 35 και 25. Το αρχείο αυτό, περιέχει τον κρυπτογραφικό αλγόριθμο sha512.

```

1  /*
2   * A JavaScript implementation of the Secure Hash Algorithm,
3   * SHA-512, as defined
4   * in FIPS 180-2
5   * Version 2.2 Copyright Anonymous Contributor, Paul Johnston
6   * 2000 - 2009.
7   * Other contributors: Greg Holt, Andrew Kepert, Ydnar,
8   * Lostinet
9   * Distributed under the BSD License
10  * See http://pajhome.org.uk/crypt/md5 for details.
11  */
12
13 /*
14  * Configurable variables. You may need to tweak these to be
15  * compatible with
16  * the server-side, but the defaults work in most cases.
17  */
18 var hexcase = 0; /* hex output format. 0 - lowercase; 1 -
19 uppercase      */
20 var b64pad  = ""; /* base-64 pad character. "=" for strict RFC
21 compliance    */
22
23 /*
24  * These are the functions you'll usually want to call
25  * They take string arguments and return either hex or base-64
26  * encoded strings
27  */
28 function hex_sha512(s)    { return
29 rstr2hex(rstr_sha512(str2rstr_utf8(s))); }
30 function b64_sha512(s)   { return
31 rstr2b64(rstr_sha512(str2rstr_utf8(s))); }

```

```

32 function any_sha512(s, e) { return
33 rstr2any(rstr_sha512(str2rstr_utf8(s)), e);}
34 function hex_hmac_sha512(k, d)
35   { return rstr2hex(rstr_hmac_sha512(str2rstr_utf8(k),
36 str2rstr_utf8(d))); }
37 function b64_hmac_sha512(k, d)
38   { return rstr2b64(rstr_hmac_sha512(str2rstr_utf8(k),
39 str2rstr_utf8(d))); }
40 function any_hmac_sha512(k, d, e)
41   { return rstr2any(rstr_hmac_sha512(str2rstr_utf8(k),
42 str2rstr_utf8(d)), e);}
43
44 /*
45  * Perform a simple self-test to see if the VM is working
46  */
47 function sha512_vm_test()
48 {
49   return hex_sha512("abc").toLowerCase() ==
50
51 "ddaf35a193617abacc417349ae20413112e6fa4e89a97ea20a9eeee64b55d
52 39a" +
53
54 "2192992a274fcl1a836ba3c23a3feebbd454d4423643ce80e2a9ac94fa54ca
55 49f";
56 }
57
58 /*
59  * Calculate the SHA-512 of a raw string
60  */
61 function rstr_sha512(s)
62 {
63   return binb2rstr(binb_sha512(rstr2binb(s), s.length * 8));
64 }
65
66 /*
67  * Calculate the HMAC-SHA-512 of a key and some data (raw
68 strings)
69  */
70 function rstr_hmac_sha512(key, data)
71 {
72   var bkey = rstr2binb(key);
73   if(bkey.length > 32) bkey = binb_sha512(bkey, key.length *
74 8);
75
76   var ipad = Array(32), opad = Array(32);
77   for(var i = 0; i < 32; i++)
78     {
79       ipad[i] = bkey[i] ^ 0x36363636;
80       opad[i] = bkey[i] ^ 0x5C5C5C5C;
81     }
82

```

```

83     var hash = binb_sha512(ipad.concat(rstr2binb(data)), 1024 +
84 data.length * 8);
85     return binb2rstr(binb_sha512(opad.concat(hash), 1024 +
86 512));
87 }
88
89 /*
90  * Convert a raw string to a hex string
91  */
92 function rstr2hex(input)
93 {
94     try { hexcase } catch(e) { hexcase=0; }
95     var hex_tab = hexcase ? "0123456789ABCDEF" :
96 "0123456789abcdef";
97     var output = "";
98     var x;
99     for(var i = 0; i < input.length; i++)
100    {
101        x = input.charCodeAt(i);
102        output += hex_tab.charAt((x >>> 4) & 0x0F)
103            + hex_tab.charAt( x          & 0x0F);
104    }
105    return output;
106 }
107
108 /*
109  * Convert a raw string to a base-64 string
110  */
111 function rstr2b64(input)
112 {
113     try { b64pad } catch(e) { b64pad=''; }
114     var tab =
115 "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz012345678
116 9+/" ;
117     var output = "";
118     var len = input.length;
119     for(var i = 0; i < len; i += 3)
120     {
121         var triplet = (input.charCodeAt(i) << 16)
122             | (i + 1 < len ? input.charCodeAt(i+1) << 8 :
123 0)
124             | (i + 2 < len ? input.charCodeAt(i+2)          :
125 0);
126         for(var j = 0; j < 4; j++)
127         {
128             if(i * 8 + j * 6 > input.length * 8) output += b64pad;
129             else output += tab.charAt((triplet >>> 6*(3-j)) & 0x3F);
130         }
131     }
132     return output;
133 }
134

```

```

135  /*
136   * Convert a raw string to an arbitrary string encoding
137   */
138  function rstr2any(input, encoding)
139  {
140      var divisor = encoding.length;
141      var i, j, q, x, quotient;
142
143      /* Convert to an array of 16-bit big-endian values, forming
144      the dividend */
145      var dividend = Array(Math.ceil(input.length / 2));
146      for(i = 0; i < dividend.length; i++)
147      {
148          dividend[i] = (input.charCodeAt(i * 2) << 8) |
149          input.charCodeAt(i * 2 + 1);
150      }
151
152      /*
153       * Repeatedly perform a long division. The binary array
154       forms the dividend,
155       * the length of the encoding is the divisor. Once computed,
156       the quotient
157       * forms the dividend for the next step. All remainders are
158       stored for later
159       * use.
160       */
161      var full_length = Math.ceil(input.length * 8 /
162                                (Math.log(encoding.length)
163 / Math.log(2)));
164      var remainders = Array(full_length);
165      for(j = 0; j < full_length; j++)
166      {
167          quotient = Array();
168          x = 0;
169          for(i = 0; i < dividend.length; i++)
170          {
171              x = (x << 16) + dividend[i];
172              q = Math.floor(x / divisor);
173              x -= q * divisor;
174              if(quotient.length > 0 || q > 0)
175                  quotient[quotient.length] = q;
176          }
177          remainders[j] = x;
178          dividend = quotient;
179      }
180
181      /* Convert the remainders to the output string */
182      var output = "";
183      for(i = remainders.length - 1; i >= 0; i--)
184          output += encoding.charAt(remainders[i]);
185
186      return output;

```

```

187 }
188
189 /*
190  * Encode a string as utf-8.
191  * For efficiency, this assumes the input is valid utf-16.
192  */
193 function str2rstr_utf8(input)
194 {
195     var output = "";
196     var i = -1;
197     var x, y;
198
199     while(++i < input.length)
200     {
201         /* Decode utf-16 surrogate pairs */
202         x = input.charCodeAt(i);
203         y = i + 1 < input.length ? input.charCodeAt(i + 1) : 0;
204         if(0xD800 <= x && x <= 0xDBFF && 0xDC00 <= y && y <=
205 0xDFFF)
206         {
207             x = 0x10000 + ((x & 0x03FF) << 10) + (y & 0x03FF);
208             i++;
209         }
210
211         /* Encode output as utf-8 */
212         if(x <= 0x7F)
213             output += String.fromCharCode(x);
214         else if(x <= 0x7FF)
215             output += String.fromCharCode(0xC0 | ((x >>> 6) &
216 0x1F),
217                                     0x80 | (x
218 0x3F));
219         else if(x <= 0xFFFF)
220             output += String.fromCharCode(0xE0 | ((x >>> 12) &
221 0x0F),
222                                     0x80 | ((x >>> 6) &
223 0x3F),
224                                     0x80 | (x
225 0x3F));
226         else if(x <= 0x1FFFFFF)
227             output += String.fromCharCode(0xF0 | ((x >>> 18) &
228 0x07),
229                                     0x80 | ((x >>> 12) &
230 0x3F),
231                                     0x80 | ((x >>> 6) &
232 0x3F),
233                                     0x80 | (x
234 0x3F));
235     }
236     return output;
237 }
238

```

```

239  /*
240  * Encode a string as utf-16
241  */
242  function str2rstr_utf16le(input)
243  {
244      var output = "";
245      for(var i = 0; i < input.length; i++)
246          output += String.fromCharCode( input.charCodeAt(i)
247      & 0xFF,
248                                     (input.charCodeAt(i) >>> 8)
249      & 0xFF);
250      return output;
251  }
252
253  function str2rstr_utf16be(input)
254  {
255      var output = "";
256      for(var i = 0; i < input.length; i++)
257          output += String.fromCharCode((input.charCodeAt(i) >>> 8)
258      & 0xFF,
259                                     input.charCodeAt(i)
260      & 0xFF);
261      return output;
262  }
263
264  /*
265  * Convert a raw string to an array of big-endian words
266  * Characters >255 have their high-byte silently ignored.
267  */
268  function rstr2binb(input)
269  {
270      var output = Array(input.length >> 2);
271      for(var i = 0; i < output.length; i++)
272          output[i] = 0;
273      for(var i = 0; i < input.length * 8; i += 8)
274          output[i>>5] |= (input.charCodeAt(i / 8) & 0xFF) << (24 -
275  i % 32);
276      return output;
277  }
278
279  /*
280  * Convert an array of big-endian words to a string
281  */
282  function binb2rstr(input)
283  {
284      var output = "";
285      for(var i = 0; i < input.length * 32; i += 8)
286          output += String.fromCharCode((input[i>>5] >>> (24 - i %
287  32)) & 0xFF);
288      return output;
289  }
290

```



```

291  /*
292   * Calculate the SHA-512 of an array of big-endian dwords, and
293   a bit length
294   */
295  var sha512_k;
296  function binb_sha512(x, len)
297  {
298     if(sha512_k == undefined)
299     {
300         //SHA512 constants
301         sha512_k = new Array(
302 new int64(0x428a2f98, -685199838), new int64(0x71374491,
303 0x23ef65cd),
304 new int64(-1245643825, -330482897), new int64(-373957723, -
305 2121671748),
306 new int64(0x3956c25b, -213338824), new int64(0x59f111f1, -
307 1241133031),
308 new int64(-1841331548, -1357295717), new int64(-1424204075, -
309 630357736),
310 new int64(-670586216, -1560083902), new int64(0x12835b01,
311 0x45706fbe),
312 new int64(0x243185be, 0x4ee4b28c), new int64(0x550c7dc3, -
313 704662302),
314 new int64(0x72be5d74, -226784913), new int64(-2132889090,
315 0x3b1696b1),
316 new int64(-1680079193, 0x25c71235), new int64(-1046744716, -
317 815192428),
318 new int64(-459576895, -1628353838), new int64(-272742522,
319 0x384f25e3),
320 new int64(0xfc19dc6, -1953704523), new int64(0x240calcc,
321 0x77ac9c65),
322 new int64(0x2de92c6f, 0x592b0275), new int64(0x4a7484aa,
323 0x6ea6e483),
324 new int64(0x5cb0a9dc, -1119749164), new int64(0x76f988da, -
325 2096016459),
326 new int64(-1740746414, -295247957), new int64(-1473132947,
327 0x2db43210),
328 new int64(-1341970488, -1728372417), new int64(-1084653625, -
329 1091629340),
330 new int64(-958395405, 0x3da88fc2), new int64(-710438585, -
331 1828018395),
332 new int64(0x6ca6351, -536640913), new int64(0x14292967,
333 0xa0e6e70),
334 new int64(0x27b70a85, 0x46d22ffc), new int64(0x2e1b2138,
335 0x5c26c926),
336 new int64(0x4d2c6dfc, 0x5ac42aed), new int64(0x53380d13, -
337 1651133473),
338 new int64(0x650a7354, -1951439906), new int64(0x766a0abb,
339 0x3c77b2a8),
340 new int64(-2117940946, 0x47edae6), new int64(-1838011259,
341 0x1482353b),

```

```

342 new int64(-1564481375, 0x4cf10364), new int64(-1474664885, -
343 1136513023),
344 new int64(-1035236496, -789014639), new int64(-949202525,
345 0x654be30),
346 new int64(-778901479, -688958952), new int64(-694614492,
347 0x5565a910),
348 new int64(-200395387, 0x5771202a), new int64(0x106aa070,
349 0x32bbd1b8),
350 new int64(0x19a4c116, -1194143544), new int64(0x1e376c08,
351 0x5141ab53),
352 new int64(0x2748774c, -544281703), new int64(0x34b0bcb5, -
353 509917016),
354 new int64(0x391c0cb3, -976659869), new int64(0x4ed8aa4a, -
355 482243893),
356 new int64(0x5b9cca4f, 0x7763e373), new int64(0x682e6ff3, -
357 692930397),
358 new int64(0x748f82ee, 0x5defb2fc), new int64(0x78a5636f,
359 0x43172f60),
360 new int64(-2067236844, -1578062990), new int64(-1933114872,
361 0x1a6439ec),
362 new int64(-1866530822, 0x23631e28), new int64(-1538233109, -
363 561857047),
364 new int64(-1090935817, -1295615723), new int64(-965641998, -
365 479046869),
366 new int64(-903397682, -366583396), new int64(-779700025,
367 0x21c0c207),
368 new int64(-354779690, -840897762), new int64(-176337025, -
369 294727304),
370 new int64(0x6f067aa, 0x72176fba), new int64(0xa637dc5, -
371 1563912026),
372 new int64(0x113f9804, -1090974290), new int64(0x1b710b35,
373 0x131c471b),
374 new int64(0x28db77f5, 0x23047d84), new int64(0x32caab7b,
375 0x40c72493),
376 new int64(0x3c9ebe0a, 0x15c9bebc), new int64(0x431d67c4, -
377 1676669620),
378 new int64(0x4cc5d4be, -885112138), new int64(0x597f299c, -
379 60457430),
380 new int64(0x5fcb6fab, 0x3ad6faec), new int64(0x6c44198c,
381 0x4a475817));
382 }
383
384 //Initial hash values
385 var H = new Array(
386 new int64(0x6a09e667, -205731576),
387 new int64(-1150833019, -2067093701),
388 new int64(0x3c6ef372, -23791573),
389 new int64(-1521486534, 0x5f1d36f1),
390 new int64(0x510e527f, -1377402159),
391 new int64(-1694144372, 0x2b3e6c1f),
392 new int64(0x1f83d9ab, -79577749),
393 new int64(0x5be0cd19, 0x137e2179));

```

```

394
395     var T1 = new int64(0, 0),
396         T2 = new int64(0, 0),
397         a = new int64(0,0),
398         b = new int64(0,0),
399         c = new int64(0,0),
400         d = new int64(0,0),
401         e = new int64(0,0),
402         f = new int64(0,0),
403         g = new int64(0,0),
404         h = new int64(0,0),
405         //Temporary variables not specified by the document
406         s0 = new int64(0, 0),
407         s1 = new int64(0, 0),
408         Ch = new int64(0, 0),
409         Maj = new int64(0, 0),
410         r1 = new int64(0, 0),
411         r2 = new int64(0, 0),
412         r3 = new int64(0, 0);
413     var j, i;
414     var W = new Array(80);
415     for(i=0; i<80; i++)
416         W[i] = new int64(0, 0);
417
418     // append padding to the source string. The format is
419     described in the FIPS.
420     x[len >> 5] |= 0x80 << (24 - (len & 0x1f));
421     x[((len + 128 >> 10)<< 5) + 31] = len;
422
423     for(i = 0; i<x.length; i+=32) //32 dwords is the block size
424     {
425         int64copy(a, H[0]);
426         int64copy(b, H[1]);
427         int64copy(c, H[2]);
428         int64copy(d, H[3]);
429         int64copy(e, H[4]);
430         int64copy(f, H[5]);
431         int64copy(g, H[6]);
432         int64copy(h, H[7]);
433
434         for(j=0; j<16; j++)
435         {
436             W[j].h = x[i + 2*j];
437             W[j].l = x[i + 2*j + 1];
438         }
439
440         for(j=16; j<80; j++)
441         {
442             //sigma1
443             int64rrot(r1, W[j-2], 19);
444             int64revrrot(r2, W[j-2], 29);
445             int64shr(r3, W[j-2], 6);

```

```

446     s1.l = r1.l ^ r2.l ^ r3.l;
447     s1.h = r1.h ^ r2.h ^ r3.h;
448     //sigma0
449     int64rrot(r1, W[j-15], 1);
450     int64rrot(r2, W[j-15], 8);
451     int64shr(r3, W[j-15], 7);
452     s0.l = r1.l ^ r2.l ^ r3.l;
453     s0.h = r1.h ^ r2.h ^ r3.h;
454
455     int64add4(W[j], s1, W[j-7], s0, W[j-16]);
456 }
457
458 for(j = 0; j < 80; j++)
459 {
460     //Ch
461     Ch.l = (e.l & f.l) ^ (~e.l & g.l);
462     Ch.h = (e.h & f.h) ^ (~e.h & g.h);
463
464     //Sigma1
465     int64rrot(r1, e, 14);
466     int64rrot(r2, e, 18);
467     int64revrrot(r3, e, 9);
468     s1.l = r1.l ^ r2.l ^ r3.l;
469     s1.h = r1.h ^ r2.h ^ r3.h;
470
471     //Sigma0
472     int64rrot(r1, a, 28);
473     int64revrrot(r2, a, 2);
474     int64revrrot(r3, a, 7);
475     s0.l = r1.l ^ r2.l ^ r3.l;
476     s0.h = r1.h ^ r2.h ^ r3.h;
477
478     //Maj
479     Maj.l = (a.l & b.l) ^ (a.l & c.l) ^ (b.l & c.l);
480     Maj.h = (a.h & b.h) ^ (a.h & c.h) ^ (b.h & c.h);
481
482     int64add5(T1, h, s1, Ch, sha512_k[j], W[j]);
483     int64add(T2, s0, Maj);
484
485     int64copy(h, g);
486     int64copy(g, f);
487     int64copy(f, e);
488     int64add(e, d, T1);
489     int64copy(d, c);
490     int64copy(c, b);
491     int64copy(b, a);
492     int64add(a, T1, T2);
493 }
494 int64add(H[0], H[0], a);
495 int64add(H[1], H[1], b);
496 int64add(H[2], H[2], c);
497 int64add(H[3], H[3], d);

```

```

498     int64add(H[4], H[4], e);
499     int64add(H[5], H[5], f);
500     int64add(H[6], H[6], g);
501     int64add(H[7], H[7], h);
502 }
503
504 //represent the hash as an array of 32-bit dwords
505 var hash = new Array(16);
506 for(i=0; i<8; i++)
507 {
508     hash[2*i] = H[i].h;
509     hash[2*i + 1] = H[i].l;
510 }
511 return hash;
512 }
513
514 //A constructor for 64-bit numbers
515 function int64(h, l)
516 {
517     this.h = h;
518     this.l = l;
519     //this.toString = int64toString;
520 }
521
522 //Copies src into dst, assuming both are 64-bit numbers
523 function int64copy(dst, src)
524 {
525     dst.h = src.h;
526     dst.l = src.l;
527 }
528
529 //Right-rotates a 64-bit number by shift
530 //Won't handle cases of shift>=32
531 //The function revrrot() is for that
532 function int64rrot(dst, x, shift)
533 {
534     dst.l = (x.l >>> shift) | (x.h << (32-shift));
535     dst.h = (x.h >>> shift) | (x.l << (32-shift));
536 }
537
538 //Reverses the dwords of the source and then rotates right by
539 shift.
540 //This is equivalent to rotation by 32+shift
541 function int64revrrot(dst, x, shift)
542 {
543     dst.l = (x.h >>> shift) | (x.l << (32-shift));
544     dst.h = (x.l >>> shift) | (x.h << (32-shift));
545 }
546
547 //Bitwise-shifts right a 64-bit number by shift
548 //Won't handle shift>=32, but it's never needed in SHA512
549 function int64shr(dst, x, shift)

```

```

550 {
551     dst.l = (x.l >>> shift) | (x.h << (32-shift));
552     dst.h = (x.h >>> shift);
553 }
554
555 //Adds two 64-bit numbers
556 //Like the original implementation, does not rely on 32-bit
557 operations
558 function int64add(dst, x, y)
559 {
560     var w0 = (x.l & 0xffff) + (y.l & 0xffff);
561     var w1 = (x.l >>> 16) + (y.l >>> 16) + (w0 >>> 16);
562     var w2 = (x.h & 0xffff) + (y.h & 0xffff) + (w1 >>> 16);
563     var w3 = (x.h >>> 16) + (y.h >>> 16) + (w2 >>> 16);
564     dst.l = (w0 & 0xffff) | (w1 << 16);
565     dst.h = (w2 & 0xffff) | (w3 << 16);
566 }
567
568 //Same, except with 4 addends. Works faster than adding them
569 one by one.
570 function int64add4(dst, a, b, c, d)
571 {
572     var w0 = (a.l & 0xffff) + (b.l & 0xffff) + (c.l & 0xffff) +
573 (d.l & 0xffff);
574     var w1 = (a.l >>> 16) + (b.l >>> 16) + (c.l >>> 16) + (d.l
575 >>> 16) + (w0 >>> 16);
576     var w2 = (a.h & 0xffff) + (b.h & 0xffff) + (c.h & 0xffff) +
577 (d.h & 0xffff) + (w1 >>> 16);
578     var w3 = (a.h >>> 16) + (b.h >>> 16) + (c.h >>> 16) + (d.h
579 >>> 16) + (w2 >>> 16);
580     dst.l = (w0 & 0xffff) | (w1 << 16);
581     dst.h = (w2 & 0xffff) | (w3 << 16);
582 }
583
584 //Same, except with 5 addends
585 function int64add5(dst, a, b, c, d, e)
586 {
587     var w0 = (a.l & 0xffff) + (b.l & 0xffff) + (c.l & 0xffff) +
588 (d.l & 0xffff) + (e.l & 0xffff);
589     var w1 = (a.l >>> 16) + (b.l >>> 16) + (c.l >>> 16) + (d.l
590 >>> 16) + (e.l >>> 16) + (w0 >>> 16);
591     var w2 = (a.h & 0xffff) + (b.h & 0xffff) + (c.h & 0xffff) +
592 (d.h & 0xffff) + (e.h & 0xffff) + (w1 >>> 16);
593     var w3 = (a.h >>> 16) + (b.h >>> 16) + (c.h >>> 16) + (d.h
594 >>> 16) + (e.h >>> 16) + (w2 >>> 16);
595     dst.l = (w0 & 0xffff) | (w1 << 16);
596     dst.h = (w2 & 0xffff) | (w3 << 16);
597 }

```

Παράρτημα II

Βιβλιογραφία

- [1] Institute New England Healthcare, "Thinking Outside the Pillbox," August 2009.
- [2] S. Fenerty and et.al., "The effect of reminder systems on patients' adherence to treatment," *Patient Preference and Adherence*, vol. 6, February 2012.
- [3] A. Christensen and et.al., "The impact of an electronic monitoring and reminder device on patient compliance with antihypertensive therapy: a randomized controlled trial.," 2010 Jan.
- [4] Q. Liu and et.al., "Reminder systems to improve patient adherence to tuberculosis clinic appointments for diagnosis and treatment.," 2014 Nov.
- [5] H. Hardy and et.al., "Randomized Controlled Trial of a Personalized Cellular Phone Reminder System to Enhance Adherence to Antiretroviral Therapy".
- [6] S. R. Finkelstein and et.al., "Appointment Reminder Systems and Patient Preferences: Patient Technology Usage and Familiarity with Other Service Providers as Predictive".
- [7] Janssen Pharmaceutical Companies, "care4today," [Online]. Available: <http://www.care4today.com/mhm/>.
- [8] I. MediSafe, "Medisafe Medication Reminder and Pill Organizer," [Online]. Available: <https://itunes.apple.com/us/app/medisafe-medication-reminder/id573916946?mt=8>.
- [9] I. MediSafe, "MediSafe Meds & Pill Reminder," [Online]. Available: <https://play.google.com/store/apps/details?id=com.medisafe.android.client&hl=en>.
- [10] I. Earth Flare, "Med Helper – Pill Reminder and Medication Tracker," [Online]. Available: <https://itunes.apple.com/us/app/med-helper-pill-reminder-medication/id438926478?mt=8>.
- [11] EarthFlare, "Med Helper Pill Reminder," [Online]. Available: <https://play.google.com/store/apps/details?id=com.earthflare.android.medhelper.lite&hl=en>.
- [12] L. Montuno Software, "Dosecast - Medication Reminder," [Online]. Available: <https://play.google.com/store/apps/details?id=com.montunosoftware.dosecast&hl=en>.
- [13] TalkSoft, "Automated Appointment Reminder System," [Online]. Available: <http://www.talksoftonline.com/automated-appointment-reminder-systems.shtml>.
- [14] Wikipedia, "HTML," [Online]. Available: <http://el.wikipedia.org/wiki/HTML>.
- [15] R. Nixon, Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 (Learning Php, Mysql, Javascript, Css & Html5), O'REILLY.
- [16] Wikipedia, "CSS," [Online]. Available: <http://el.wikipedia.org/wiki/CSS>.

- [17] Wikipedia, "PHP," [Online]. Available: <http://el.wikipedia.org/wiki/PHP>.
- [18] Wikipedia, "MySQL," [Online]. Available: <http://el.wikipedia.org/wiki/MySQL>.
- [19] Wikipedia, "JavaScript," [Online]. Available: <http://el.wikipedia.org/wiki/JavaScript>.
- [20] w3.org, "Techniques for WCAG 2.0," [Online]. Available: <http://www.w3.org/TR/WCAG20-TECHS/>.
- [21] T. Bootstrap, "Bootstrap," [Online]. Available: <http://getbootstrap.com/>.
- [22] Wikipedia, "Bootstrap," [Online]. Available: <http://en.wikipedia.org/wiki/Bootstrap>.
- [23] Wikipedia, "SHA-2," [Online]. Available: <https://en.wikipedia.org/wiki/SHA-2>.
- [24] Wikipedia. [Online]. Available: https://el.wikipedia.org/wiki/%CE%91%CE%BD%CE%BF%CE%B9%CE%BA%CF%84%CF%8C%CF%82_%CE%9A%CF%8.
- [25] Wikipedia. [Online]. Available: https://el.wikipedia.org/wiki/%CE%9B%CE%BF%CE%B3%CE%B9%CF%83%CE%BC%CE%B9%CE%BA%CF%8C_%CE%B1%CE%BD%CE%BF%CE%B9%CE%BA%CF%84%CE%BF%CF%8D_%CE%BA%CF%8E%CE%B4%CE%B9%CE%BA%CE%B1.
- [26] Wikipedia, "AskoziaPBX," [Online]. Available: <https://en.wikipedia.org/wiki/AskoziaPBX>.
- [27] Wikipedia, "AsteriskPBX," [Online]. Available: https://en.wikipedia.org/wiki/Asterisk_%28PBX%29.
- [28] Wikipedia, "Elastix," [Online]. Available: <https://en.wikipedia.org/wiki/Elastix>.
- [29] Wikipedia, "FreeSWITCH," [Online]. Available: <https://en.wikipedia.org/wiki/FreeSWITCH>.
- [30] Wikipedia, "FreePBX," [Online]. Available: <https://en.wikipedia.org/wiki/FreePBX>.
- [31] Wikipedia, "GNU_Gatekeeper," [Online]. Available: https://en.wikipedia.org/wiki/GNU_Gatekeeper.
- [32] opensips, "opensips," [Online]. Available: <http://www.opensips.org/>.
- [33] iptel.org, "SER," [Online]. Available: <http://www.iptel.org/ser/>.
- [34] Wikipedia, "Yate," [Online]. Available: https://en.wikipedia.org/wiki/Yate_%28telephony_engine%29.
- [35] OpenVBX, "OpenVBX," [Online]. Available: <http://www.openvbx.org/>.

