

ΤΕΙ ΠΕΛΟΠΟΝΝΗΣΟΥ, ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ (Έδρα: Σπάρτη)

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

“Δημιουργία πλατφόρμας καταγραφής καταλόγου Ιατρών με δυνατότητες αναλυτικής
εύρεσης και αναζήτησης σύμφωνα με πληθώρα δεδομένων και φίλτρων”

Κιντής Θεοδόσιος

A.M.: 2012013

Επιβλέπων καθηγητής:

Πουλόπουλος Βασίλειος

Σπάρτη

Ακαδημαϊκό Έτος 2017-2018

ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ ΚΑΙ ΑΝΑΛΗΨΗΣ ΠΡΟΣΩΠΙΚΗΣ ΕΥΘΥΝΗΣ

"Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ενυπογράφως ότι είμαι αποκλειστικός συγγραφέας της παρούσας Πτυχιακής Εργασίας, για την ολοκλήρωση της οποίας κάθε βοήθεια είναι πλήρως αναγνωρισμένη και αναφέρεται λεπτομερώς στην εργασία αυτή. Έχω αναφέρει πλήρως και με σαφείς αναφορές, όλες τις πηγές χρήσης δεδομένων, απόψεων, θέσεων και προτάσεων, ιδεών και λεκτικών αναφορών, είτε κατά κυριολεξία είτε βάση επιστημονικής παράφρασης. Αναλαμβάνω την προσωπική και ατομική ευθύνη ότι σε περίπτωση αποτυχίας στην υλοποίηση των ανωτέρω δηλωθέντων στοιχείων, είμαι υπόλογος έναντι λογοκλοπής, γεγονός που σημαίνει αποτυχία στην Πτυχιακή μου Εργασία και κατά συνέπεια αποτυχία απόκτησης του Τίτλου Σπουδών, πέραν των λοιπών συνεπειών του νόμου περί πνευματικών δικαιωμάτων. Δηλώνω, συνεπώς, ότι αυτή η Πτυχιακή Εργασία προετοιμάστηκε και ολοκληρώθηκε από εμένα προσωπικά και αποκλειστικά και ότι, αναλαμβάνω πλήρως όλες τις συνέπειες του νόμου στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δε μου ανήκει διότι είναι προϊόν λογοκλοπής άλλης πνευματικής ιδιοκτησίας."

Όνομα και Επώνυμο Συγγραφέα (Με κεφαλαία):

Υπογραφή (Ολογράφως, χωρίς μονογραφή):

Ημερομηνία (Ημέρα – Μήνας – Έτος):

ΠΕΡΙΛΗΨΗ

Η «Πλατφόρμα Εύρεσης Ιατρών» είναι μία ιστοσελίδα στην οποία οποιοσδήποτε ιατρός μπορεί να καταχωρεί τα στοιχεία του με σκοπό την προβολή στο internet και οποιοσδήποτε χρήστης μπορεί να αναζητά ιατρό της αρεσκείας του.

Αυτή η εφαρμογή αναπτύχθηκε με την χρήση της γλώσσας προγραμματισμού PHP η οποία βοήθησε να μετατραπεί το site από στατικό σε δυναμικό. Συνάμα, χρησιμοποιήθηκε η HTML, όπου έγινε χρήση των ετικετών (tags) της για να συμπεριληφθούν τα κείμενα που φαίνονται στο site και άλλα αρχεία τα οποία μπορούν καλεστούν αντίστοιχα. Κυρίαρχο ρόλο είχαν η CSS και η Bootstrap όπου είχαν ως αποτέλεσμα να προσφέρουν ένα όμορφο στυλιστικό αποτέλεσμα αλλά και να κάνουν την ιστοσελίδα να είναι (responsive) δηλαδή να προσαρμόζεται σε κάθε διάσταση οθόνης. Επιπλέον, η γλώσσα βάσεων δεδομένων Mysql βοήθησε μέσα από το PHPMyadmin το οποίο είναι εργαλείο βάσεων δεδομένων να δημιουργηθεί η βάση δεδομένων με όλους τους πίνακες.

Μολονότι τα επίπεδα των χρηστών αυτής της εφαρμογή είναι τρία (χρήστης-επισκέπτης, χρήστης-μέλος, και διαχειριστής) όπου κάθε ένας χρήστης έχει διαφορετικά δικαιώματα με αποτέλεσμα να έχουν διαφορετικές λειτουργίες, όλοι οι χρήστες μπορούν να κάνουν αναζήτηση ιατρού. Στην κορυφή της πυραμίδας βρίσκεται ο διαχειριστής που έχει απεριόριστα δικαιώματα, για παράδειγμα να προσθέτει και διαγράφει ιατρεία ασφαλίσεις και ειδικότητες στους ιατρούς.

Λαμβάνοντας υπόψιν τα προαναφερθέντα δημιουργήθηκε μια φιλική προς τον χρήστη εφαρμογή η οποία είναι εύχρηστη και ασφαλής για τους χρήστες της. Παράλληλα, εξασφαλίζεται η ποιότητα και η αυθεντικότητα των στοιχείων του ιατρού αλλά και της ύπαρξής του, εφόσον ο διαχειριστής έρχεται σε τηλεφωνική επικοινωνία με κάθε ιατρό.

Πιο αναλυτικά:

Στο **πρώτο κεφάλαιο**, γίνεται μια αναφορά στο internet και στις διαδικτυακές εφαρμογές, αλλά και στην συγκεκριμένη εφαρμογή όπου αναλύονται οι προδιαγραφές, τα πλεονεκτήματα και ο σκοπός της.

Στο **δεύτερο κεφάλαιο**, αναλύονται οι γλώσσες προγραμματισμού και βάσεων δεδομένων (HTML, CSS, PHP, Bootstrap, και MySQL) και τα εργαλεία (PHPStorm και PhpMyAdmin) που χρησιμοποιήθηκαν για την ανάπτυξη αυτού του διαδικτυακού τύπου.

Στο **τρίτο κεφάλαιο**, περιγράφονται αναλυτικά οι χρήστες (επισκέπτης, μέλος και διαχειριστής) και τα χαρακτηριστικά τους.

Το **τέταρτο κεφάλαιο** αφορά αποκλειστικά την αρχιτεκτονική της βάσης δεδομένων, όπου γίνεται ανάλυση των πινάκων που την απαρτίζουν.

Στο **πέμπτο** αναλύονται οι απαιτήσεις του συστήματος και καταγράφεται διεξοδικά περιγραφή του αποτελέσματος χρήσης του συστήματος.

Στο **έκτο κεφάλαιο** επεξηγείται η υλοποίηση του συστήματος όπου αναπτύσσονται όλα τα αρχεία που απαρτίζουν την ιστοσελίδα.

Τέλος, στο **έβδομο κεφάλαιο** επεξηγούνται τα συμπεράσματα και οι στόχοι της εφαρμογής, αλλά και οι μελλοντικοί στόχοι.

Ευχαριστίες

Στο σημείο αυτό, θα ήθελα να ευχαριστήσω τον κύριο Πουλόπουλο Βασίλειο καθηγητή του τμήματος Μηχανικών Πληροφορικής και Μηχανικό Η/Υ και Πληροφορικής, MSc, PhD για την εμπιστοσύνη που μου έδειξε συμπεριλαμβάνοντάς με στην ομάδα των φοιτητών των οποίων εποπτεύει τις πτυχιακές εργασίες, για την ειλικρινή προθυμία του να συνεργαστεί και να δώσει τις απαραίτητες επιστημονικές κατευθύνσεις καθ' όλη τη διάρκεια εκπόνησης της μελέτης, για την άμεση ανταπόκριση του στα αιτήματα και τις απορίες μου καθώς και για το κίνητρο που μου προσέφερε ώστε να εξελιχτώ γνωστικά και να μελετήσω εις βάθος το συγκεκριμένο θέμα.

Επιπλέον, θα ήθελα να ευχαριστήσω τους πρώην συνεργάτες μου στην εταιρία digital up Διονύση και Γιάννη που μου έδωσαν τεχνογνωσία στην κατασκευή των sites, μου έλυσαν τυχόν απορίες και κυρίως μου έδωσαν το κίνητρο να προσπαθώ μόνος μου να λύνω τα προγραμματιστικά προβλήματα.

Τέλος, δεν θα μπορούσα να παραλείψω να ευχαριστήσω την κοπέλα μου Πέγκυ η οποία γνωρίζοντας πως ο μαθησιακός τύπος μου είναι ο διαπροσωπικός, ήταν δίπλα μου και με στήριζε.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΙΣΑΓΩΓΗ.....	9
Κεφάλαιο 1 - INTERNET ΚΑΙ ΕΦΑΡΜΟΓΕΣ.....	10
1.1 Γενικά.....	11
1.2 Η ανάπτυξη διαδικτυακών εφαρμογών σήμερα.....	11
1.3 Πλατφόρμα Εύρεσης Ιατρών.....	12
1.4 Προδιαγραφές μιας Πλατφόρμας Εύρεσης Ιατρών.....	12
1.5 Πλεονεκτήματα ενός τέτοιου τύπου.....	13
1.6 Σκοπός.....	13
Κεφάλαιο 2 – SOFTWARE ΚΑΙ ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΔΥΝΑΜΙΚΩΝ ΙΣΤΟΣΕΛΙΔΩΝ.....	14
2.1 Γενικά.....	15
2.2 HTML, CSS, PHP, Bootstrap, MYSQL.....	15
2.3 Η HTML.....	15
2.3.1 Εισαγωγή στην HTML.....	15
2.3.2 Πλεονεκτήματα της HTML.....	16
2.4 Η CSS.....	17
2.4.1 Εισαγωγή στην CSS.....	17
2.4.2 Ποια πλεονεκτήματα έχει η χρήση CSS έναντι της μορφοποίησης μέσω HTML attributes;.....	17
2.5 Η PHP.....	18
2.5.1 Εισαγωγή στην PHP.....	18
2.5.2 Τι μπορεί να κάνει η PHP;.....	19
2.5.3 Πλεονεκτήματα της PHP.....	20
2.6 Το Bootstrap.....	21
2.6.1 Εισαγωγή στο Bootstrap.....	21
2.6.2 Πλεονεκτήματα του Bootstrap.....	24
2.7 Η MySQL.....	25
2.7.1 Εισαγωγή στην MySQL.....	25
2.8 PHPStorm.....	26

2.9 PhpMyAdmin.....	27
2.9.1 Εισαγωγή στο PhpMyAdmin	27
2.9.2 Χαρακτηριστικά PhpMyAdmin	27
Κεφάλαιο 3 - ΛΕΙΤΟΥΡΓΙΕΣ ΚΑΙ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΧΡΗΣΤΩΝ.....	29
3.1 Γενικά.....	30
3.2 Υπηρεσίες που παρέχει η εφαρμογή.....	30
3.2.1 Λειτουργίες Χρήστη-Επισκέπτη.....	30
3.2.2 Λειτουργίες Χρήστη-Μέλους.....	30
3.2.3 Λειτουργίες Διαχειριστή.....	31
3.3 Χαρακτηριστικά Χρηστών.....	31
Κεφάλαιο 4 – ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ.....	33
4.1 Η αρχιτεκτονική της Βάσης Δεδομένων.....	34
4.1.1 Γενικά.....	34
4.1.2 Πίνακας ιατρού (iatros).....	35
4.1.3 Πίνακας λογαριασμού (login).....	36
4.1.4 Πίνακες ειδικότητας, διεύθυνσης και ασφάλισης (eidikotita, diefthinsi, asfalisi).....	36
4.1.5 Πίνακες όλων των ειδικοτήτων, διευθύνσεων και ασφαλίσεων (allresidencies, allcities, alltameia).....	36
Κεφάλαιο 5 – ΑΠΑΙΤΗΣΕΙΣ ΣΥΣΤΗΜΑΤΟΣ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑ ΧΡΗΣΗΣ ΣΥΣΤΗΜΑΤΟΣ.....	37
5.1 Γενικά	38
5.2 Ανάλυση των απαιτήσεων.....	38
5.3 Περιγραφή του αποτελέσματος χρήσης του συστήματος.....	38
Κεφάλαιο 6 – ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ.....	46
6.1 Γενικά.....	47
6.2 Αρχείο functions.php.....	47
6.3 Αρχείο mainpage.php.....	48
6.4 Αρχείο register.php.....	49
6.5 Αρχείο login.php.....	50
6.6 Αρχείο edit.php	51

6.7 Αρχείο submit_data.php.....	52
6.8 Αρχείο admin.php.....	53
6.9 Αρχείο add_iatreio.php.....	57
6.10 Αρχείο add_eidikotita.php.....	58
6.11 Αρχείο add_asfalisi.php.....	58
6.12 Αρχείο delete_iatreio.php.....	59
6.13 Αρχείο delete_eidikotita.php.....	59
6.14 Αρχείο delete_asfalisi.php.....	60
6.15 Αρχείο session.php.....	60
6.16 Αρχείο logout.php.....	61
6.17 Αρχείο search.php.....	61
Κεφάλαιο 7 – ΣΥΜΠΕΡΑΣΜΑΤΑ.....	66
7.1 Γενικά.....	67
7.2 Συμπεράσματα.....	67
7.3 Στόχοι της Εφαρμογής.....	68
7.4 Μελλοντικοί Στόχοι.....	68
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	70

ΕΙΣΑΓΩΓΗ

Στην σημερινή εποχή το διαδίκτυο παίζει καθοριστικό ρόλο στην ζωή μας δεδομένου ότι διευκολύνει, παρέχει άμεσα πληροφορίες ανεξαρτήτως χρόνου και τόπου. Επιπλέον, η ιατρική περίθαλψη ανέκαθεν αποτελούσε και αποτελεί ανάγκη για τον οποιονδήποτε άνθρωπο και η αναζήτηση ιατρού αποτελούσε μια περίπλοκη και χρονοβόρα διαδικασία. Πλέον, η ανάγκη αυτή καλύπτεται με την χρήση του internet. Έτσι λοιπόν, κατασκευάσαμε μία πλατφόρμα εύρεσης ιατρών προκειμένου να καλυφθεί η ανάγκη αυτή του κάθε ασθενή να βρίσκει άμεσα μέσω internet τον ιατρό που χρειάζεται.

Πιο συγκεκριμένα, με την παρούσα εργασία καλύψαμε όλες αυτές τις ανάγκες, δημιουργώντας μια πλατφόρμα καταγραφής καταλόγου Ιατρών με δυνατότητες αναλυτικής εύρεσης και αναζήτησης σύμφωνα με πληθώρα δεδομένων και φίλτρων. Ταυτόχρονα, δίνεται η ευκαιρία σε οποιονδήποτε ιατρό να καταχωρήσει τα στοιχεία του και να διευρύνει το εύρος των αποτελεσμάτων της αναζήτησης, αλλά και να προβάλει την επαγγελματική του δραστηριότητα.

Σε όλα τα κεφάλαια παρουσιάζεται η δημιουργία του δυναμικού αυτού site. Αρχικά, δομήσαμε μια βάση δεδομένων η οποία απαρτίζεται από όλους τους πίνακες βάσεων δεδομένων. Στη συνέχεια, κατασκευάστηκε το δυναμικό site, το οποίο συνεργάζεται με τη βάση δεδομένων και αντλεί στοιχεία από τη βάση δεδομένων. Αυτά τα βήματα έγιναν προκειμένου να γίνει η εφαρμογή λειτουργική και έτοιμη προς χρήση.

Κεφάλαιο 1

INTERNET ΚΑΙ ΕΦΑΡΜΟΓΕΣ

1.1 Γενικά

Στο πρώτο κεφάλαιο της παρούσας πτυχιακής εργασίας θα γίνει μια αναδρομή στην ιστορία του internet και τη αναγκαιότητα του στη σύγχρονη εποχή.

Επιπλέον, θα γίνει αναλυτικότερη αναφορά στις πλατφόρμες και πιο συγκεκριμένα στην πλατφόρμα εύρεσης γιατρών και τα οφέλη της σε γιατρούς και ασθενείς. Τέλος, θα προσδιοριστεί ο σκοπός κατασκευής και χρήσης μιας τέτοιας πλατφόρμας.

1.2 Η ανάπτυξη διαδικτυακών εφαρμογών σήμερα

Το διαδίκτυο αποτελεί πλέον μέρος της καθημερινότητας ενός μεγάλου μέρους του πληθυσμού κυρίως του δυτικού κόσμου. Αποτελεί εργαλείο δουλειάς για άτομα σχετιζόμενα ή μη με την επιστήμη της πληροφορικής. Επιπλέον, αποτελεί μέσο επικοινωνίας μεταξύ των ανθρώπων, μέσο διάδοσης του πολιτισμού, πώλησης προϊόντων και υπηρεσιών και σαφώς μια διαρκώς ανανεώσιμη πηγή πληροφόρησης και ενημέρωσης. Επομένως, η εύρεση οποιασδήποτε υπηρεσίας, πληροφορίας ακόμη και ατόμων γίνεται πλέον εύκολη, άμεση και γρήγορη χάρη στο διαδίκτυο. Να σημειωθεί πως εκτός από την ταχύτητα και το εύρος των αναγκών που μπορεί να καλύψει το διαδίκτυο, αυτό γίνεται με τρόπο εύκολο από κάθε χρήστη, δηλαδή δε χρειάζεται ιδιαίτερη εκπαίδευση η χρήση του, ενώ παράλληλα η περιήγηση στο διαδίκτυο είναι ευχάριστη και πιο ενδιαφέρουσα δεδομένου ότι πολλές φορές δεν περιέχει μόνο κείμενα αλλά εικόνες και βίντεο.

Για την ανάπτυξη διαδικτυακών εφαρμογών κυρίαρχο ρόλο έχει η Java και άλλες συναφείς γλώσσες προγραμματισμού. Συνάμα, χρησιμοποιώντας και άλλες τεχνολογίες διαδικτύου όπως είναι η PHP, HTML, CSS, Javascript είμαστε ικανοί να αναπτύσσουμε ιστοσελίδες και πλατφόρμες με πολλές λειτουργικότητες. Αναλυτικότερα, με την HTML χτίζεται η δομή της σελίδας. Παράλληλα, χρησιμοποιώντας τεχνολογίες βάσεων δεδομένων, όπως είναι η MySQL, μπορούμε να αυξήσουμε τις λειτουργίες αυτές, αφού μας δίνεται η δυνατότητα να καταγράψουμε και να αποθηκεύσουμε πληθώρα όγκου δεδομένων και να τον ανακαλέσουμε οποιαδήποτε στιγμή επιθυμούμε με την PHP.

Ταυτόχρονα, κάνοντας χρήση με γλώσσες όπως η JAVASCRIPT και η CSS αλλά και

frameworks όπως είναι η bootstrap, έχουμε την πολυτέλεια να παρέχουμε στον χρήστη ένα όμορφο και οικείο αποτέλεσμα.

Όλες αυτές οι τεχνολογίες και πολλές άλλες φτιάχτηκαν από τον άνθρωπο για τον άνθρωπο για να εξυπηρετήσουν κάποιες από τις ανάγκες του. Ανάγκες όπως το ηλεκτρονικό εμπόριο(e-commerce), τηλεμάθηση (e-learning), αλλά και όπως αυτή της αναζήτησης. Στην παρούσα πτυχιακή εργασία θα μας απασχολήσει αυτή της αναζήτησης ιατρών.

1.3 Πλατφόρμα Εύρεσης Ιατρών

Το διαδίκτυο αποτελεί το πιο σύγχρονο, άμεσο, έγκυρο και ανέξοδο μέσο προώθησης υπηρεσιών, προϊόντων. Όσον αφορά στην εύρεση ιατρών, η διαδικασία μπορεί να είναι μια εξαιρετική ευκαιρία τόσο για τους ιατρούς οι οποίοι μπορούν να προβληθούν μέσω της σελίδας – πλατφόρμας, δεδομένου ότι σε λίγο χρόνο με μηδενικό κόστος μπορούν να γνωστοποιήσουν τις υπηρεσίες τους στο ευρύ κοινό, όσο και για τους χρήστες οι οποίοι μπορούν άμεσα και βάζοντας τα κατάλληλα φίλτρα να βρίσκουν ακριβώς αυτό που ψάχνουν γλιτώνοντας περιττή ταλαιπωρία και χρόνο. Συνεπώς, η πλατφόρμα θα αποσκοπεί στην ορθότερη και στοχευμένη επιλογή γιατρού για την κάλυψη κάθε εξατομικευμένης ιατρικής ανάγκης με μηδενικό κόστος και ταλαιπωρία.

1.4 Προδιαγραφές ενός Δικτυακού Τύπου Εύρεσης Ιατρού

Παρακάτω αναφέρονται τα βασικότερα χαρακτηριστικά της εφαρμογής, τόσο από την πλευρά του ιατρού όσο και από την πλευρά χρηστών.

- Πρόσβαση μέσω internet, χρησιμοποιώντας απλώς κάποιον browser
- Δυνατότητα log in, εγγραφής και καταχώρησης στοιχείων από τους ιατρούς
- Αναζήτηση με φίλτρα όπως είναι η ειδικότητα, η περιοχή και η ασφάλιση
- Λίστα όλων των ειδικοτήτων των γιατρών και των διαθέσιμων ασφαλειών
- Ευκολία στη χρήση τόσο από τους γιατρούς όσο και για τους χρήστες

- Εμφάνιση αποτελεσμάτων αναζήτησης με όλα τα απαραίτητα στοιχεία των ιατρών
- Δυνατότητα αποσύνδεσης από το σύστημα

1.5 Πλεονεκτήματα ενός τέτοιου τύπου

Η λειτουργία του διαδικτύου ως εργαλείου εύρεσης πληροφοριών μας δίνει μια μεγάλη ποικιλία νέων δυνατοτήτων στο χώρο της αναζήτησης ιατρών και στον τρόπο λειτουργίας τέτοιων δικτυακών τόπων.

Τα βασικά πλεονεκτήματα της χρήσης ενός δικτυακού τύπου εύρεσης ιατρών είναι :

- Παγκόσμια πρόσβαση
- Προβολή προφίλ ιατρών
- Μηδενικό κόστος
- Γρήγορη αναζήτηση
- Ευκολία χρήσης
- Αναζήτηση ιατρού σε πανελλαδικό επίπεδο
- Εύρεση ιατρών βάσει επιθυμητών φίλτρων

1.6 Σκοπός

Ένας από τους στόχους της πτυχιακής αυτής εργασίας είναι η δημιουργία μιας διαδικτυακής πλατφόρμας εύρεσης ιατρών με φίλτρα, με σκοπό την διευκόλυνση εύρεσης του κατάλληλου επαγγελματία. Μέσω αυτής της δημιουργίας, γίνεται ανάπτυξη ποιοτικού κώδικα και βάσης δεδομένων. Επιπλέον εφαρμόζονται θεωρητικές γνώσεις, κατανοείται η μεθοδολογία κατασκευής δυναμικών ιστοσελίδων, γίνεται εκμάθηση τρόπου προσέγγισης προβλημάτων, αναζήτηση κατάλληλης λύσης και προσαρμογή της στον κώδικά μας και παρακολούθησης διαρκών εξελίξεων σχετικών με τις γλώσσες προγραμματισμού και βάσεων δεδομένων όπως είναι η PHP και η MySQL.

Κεφάλαιο 2

SOFTWARE ΚΑΙ ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ

ΔΥΝΑΜΙΚΩΝ ΙΣΤΟΣΕΛΙΔΩΝ

2.1 Γενικά

Ο σκοπός αυτού του κεφαλαίου είναι η παρουσίαση όλων των εργαλείων που χρησιμοποιήθηκαν για την ανάπτυξη της πλατφόρμας. Τα εργαλεία αφορούν γλώσσες προγραμματισμού και βάσεις δεδομένων και τα περιβάλλοντα τα οποία χρησιμοποιήθηκαν για την ανάπτυξή τους. Στην συνέχεια, όλα αυτά που μας βοήθησαν για την ανάπτυξη αυτής της πλατφόρμας θα αναλυθούν ένα προς ένα.

2.2 HTML, CSS, PHP, Bootstrap, MYSQL

Η εφαρμογή υλοποιήθηκε με τη χρήση των HTML, CSS, PHP, Bootstrap και MYSQL και παρακάτω γίνεται ανάπτυξη σε αυτά. Τα κριτήρια επιλογής αυτών των εργαλείων είναι τα εξής:

- Τα πέντε προϊόντα είναι ανοιχτού κώδικα.
- Η μια δεν παρεμποδίζει τη λειτουργία της άλλης, επομένως είναι πλήρως λειτουργικά μεταξύ τους.
- Συνδυαστικά οι παραπάνω γλώσσες οδηγούν σε μια δυναμική σελίδα με όμορφο στυλιστικό αποτέλεσμα, ασφάλεια, αξιοπιστία και ταχύτητα.

2.3 Η HTML

2.3.1 Εισαγωγή στην HTML

«Η γλώσσα HTML (HyperText Markup Language) είναι ο τρόπος με τον οποίο ο browser απεικονίζει τα έγγραφα. Με τον όρο έγγραφα εννοούμε κάθε είδους αρχείου, όπως αρχείο κειμένου, εικόνες, animation και βίντεο. Τα αρχεία της HTML είναι απλά αρχεία κειμένου στα οποία υπάρχουν ειδικοί κώδικες, οι οποίοι ονομάζονται ετικέτες (tags), που δηλώνουν στον browser τον τρόπο με τον οποίο θα εμφανιστούν τα αρχεία αυτά.

Οι ετικέτες έχουν την παρακάτω γενική δομή:

<όνομα_ετικέτας>κείμενο</όνομα_ετικέτας>

Με το όνομα_ετικέτας δηλώνεται στον browser η ενέργεια που θα εκτελεστεί στο κείμενο. Για να δηλώσουμε το σημείο που θα σταματήσει να εκτελείτε η ενέργεια αυτή χρησιμοποιούμε την αντίστοιχη ετικέτα κλεισίματος </όνομα_ετικέτας>. Υπάρχουν περιπτώσεις ετικετών HTML που δεν έχουν ετικέτα κλεισίματος, π.χ. όταν θέλουμε να εισάγουμε μία εικόνα απλά δηλώνουμε την ενέργεια αυτή με την αντίστοιχη HTML ετικέτα – η ετικέτα κλεισίματος στην περίπτωση αυτή δεν έχει κανένα νόημα. Αντίθετα, όταν θέλουμε να κάνουμε ένα κείμενο να έχει έντονη γραφή τότε πρέπει να συμπεριλάβουμε το κείμενο εντός της αντίστοιχης HTML ετικέτας.

Στο όνομα ετικέτας δε γίνεται διάκριση ανάμεσα σε πεζούς και κεφαλαίους χαρακτήρες. Οποιοσδήποτε συνδιασμός είναι αποδεκτός Π.χ. οι ετικέτες , ή είναι ισοδύναμες.»[4]

Αξίζει να τονιστεί ότι η HTML είναι όλο το περιεχόμενο της ιστοσελίδας. Επίσης, τα αρχεία HTML έχουν κατάληξη .html ή .htm. Γραμμές html μπορούμε να συναντήσουμε και μέσα σε .php και .phtml αρχεία.

2.3.2 Πλεονεκτήματα της HTML

- Η HTML είναι ευρέως γνωστή
- Έχει μεγάλη δεξαμενή πληροφοριών (documentation) στο internet σε μορφή κειμένου για τους λεκτικούς και σε μορφή βίντεο για τους κιναισθητικούς και τους ακουστικούς τύπους
- Είναι απλή στην χρήση της
- Εύκολη στην κατανόηση της
- Το όνομα των περισσότερων ετικετών έχει το αρχικό γράμμα ή και περισσότερα γράμματα της λέξης που δηλώνει την δουλειά που κάνει π.χ. το <p> αφορά παράγραφο (paragraph), το αφορά εικόνα και την πηγή της (image source)
- Μπορούμε να εισάγουμε κείμενα, εικόνες, βίντεο, ήχο, πίνακες, λίστες εύκολα με την χρήση των κατάλληλων ετικετών

2.4 Η CSS

2.4.1 Εισαγωγή στην CSS

«Το CSS σημαίνει Cascading Style Sheets. Το CSS περιγράφει τον τρόπο εμφάνισης των στοιχείων HTML σε οθόνη, χαρτί ή σε άλλα μέσα. Το CSS εξοικονομεί πολλή δουλειά. Μπορεί να ελέγχει τη διάταξη πολλών ιστοσελίδων ταυτόχρονα. Τα εξωτερικά φύλλα στυλ αποθηκεύονται σε αρχεία CSS.» [1]

Η CSS είναι μία γλώσσα που χρησιμοποιείται για την εμφάνιση της ιστοσελίδας. Με άλλα λόγια, είναι υπεύθυνη για το στυλιστικό αποτέλεσμα της ιστοσελίδας αφού μέσω αυτής διαμορφώνεται το περιεχόμενο της ιστοσελίδας, δηλαδή ότι έχει προκύψει από την HTML. Μπορούμε να αλλάζουμε τα χρώματα, την γραμματοσειρά, τα κενά, την θέση κ.ά. του περιεχομένου. Άρα, η χρήση της κρίνεται απαραίτητη αφού μας παρέχεται η δυνατότητα να δίνουμε μια όμορφη όψη στην ιστοσελίδα μας.

2.4.2 Ποια πλεονεκτήματα έχει η χρήση CSS έναντι της μορφοποίησης μέσω HTML attributes;

- «Πολύ μεγαλύτερη ευελιξία. Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασική HTML.
- Ευκολότερη συντήρηση των ιστοσελίδων. Η εμφάνιση ενός ολόκληρου site μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS. Έτσι, κάθε αλλαγή στο στυλ της ιστοσελίδας μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, αντί για την επεξεργασία πολλών σημείων σε κάθε σελίδα που υπάρχει στο site. ☐
- Μικρότερο μέγεθος αρχείου, δεδομένου ότι ο κάθε κανόνας μορφοποίησης γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.
- Καλύτερο SEO (Search engine optimization). Οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα σε περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση στο περιεχόμενο σκέτο, οπότε είναι πολύ ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν (indexing).

- Γρηγορότερες σελίδες. Όταν χρησιμοποιούμε εξωτερικό αρχείο CSS, ο browser την πρώτη φορά που θα φορτώσει κάποια σελίδα του site μας το αποθηκεύει στην cache, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που κατεβάζει ο χρήστης του κάποια άλλη σελίδα του site.» [5]

2.5 Η PHP

2.5.1 Εισαγωγή στην PHP

«Το PHP (αναδρομικό ακρωνύμιο για την PHP: Hypertext Preprocessor) είναι μια ευρέως χρησιμοποιούμενη γλώσσα λογισμικού γενικής χρήσης ανοιχτού κώδικα που είναι ιδιαίτερα κατάλληλη για ανάπτυξη ιστοσελίδων και μπορεί να ενσωματωθεί σε HTML.

Παραδείγματος χάρη:

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Example</title>
  </head>
  <body>

    <?php
      echo "Hi, I'm a PHP script!";
    ?>

  </body>
</html>
```

Αντί για πολλές εντολές για την εξαγωγή HTML (όπως φαίνεται στην C ή Perl), οι σελίδες PHP περιέχουν HTML με ενσωματωμένο κώδικα που κάνει "κάτι" (σε αυτή την περίπτωση, η έξοδος "Γεια σου, είμαι PHP script!"). Ο κώδικας PHP περικλείεται σε ειδικές οδηγίες έναρξης και λήξης <? php και ?> Που σας επιτρέπουν να μεταβείτε και να βγείτε

από τη "λειτουργία PHP".

Αυτό που διακρίνει την PHP από κάτι σαν JavaScript από την πλευρά του πελάτη είναι ότι ο κώδικας εκτελείται στον εξυπηρετητή, δημιουργώντας HTML που στη συνέχεια αποστέλλεται στον πελάτη. Ο πελάτης θα λάβει τα αποτελέσματα της εκτέλεσης του σεναρίου, αλλά δεν θα ξέρει τι είναι ο υποκείμενος κώδικας. Μπορείτε ακόμη να διαμορφώσετε τον διακομιστή ιστού σας για να επεξεργαστείτε όλα τα αρχεία HTML με την PHP, και τότε πραγματικά δεν υπάρχει κανένας τρόπος ώστε οι χρήστες να μπορούν να πουν τι έχεις στο μανίκι σου.

Τα καλύτερα πράγματα στη χρήση της PHP είναι ότι είναι εξαιρετικά απλή για έναν νεοφερμένο, αλλά προσφέρει πολλά προηγμένα χαρακτηριστικά για έναν επαγγελματία προγραμματιστή. Μην φοβάστε να διαβάσετε τη μακρά λίστα των χαρακτηριστικών της PHP. Μπορείτε να μεταβείτε σε ένα μικρό χρονικό διάστημα και να αρχίσετε να γράφετε απλά σενάρια σε λίγες ώρες.»[2]

2.5.2 Τι μπορεί να κάνει η PHP;

«Οτιδήποτε. Η PHP επικεντρώνεται κυρίως σε δέσμες ενεργειών από πλευράς διακομιστή, ώστε να μπορείτε να κάνετε οτιδήποτε μπορεί να κάνει οποιοδήποτε άλλο πρόγραμμα CGI, όπως συλλογή δεδομένων φόρμας, δημιουργία δυναμικού περιεχομένου σελίδας ή αποστολή και λήψη cookies. Αλλά η PHP μπορεί να κάνει πολύ περισσότερα.

- Σάρωση διακομιστή. Αυτό είναι το πιο παραδοσιακό και κύριο πεδίο στόχου για την PHP. Χρειάζεστε τρία πράγματα για να κάνετε αυτή την εργασία: τον επεξεργαστή PHP (CGI ή μονάδα διακομιστή), έναν διακομιστή ιστού και ένα πρόγραμμα περιήγησης ιστού. Πρέπει να εκτελέσετε τον διακομιστή ιστού, με μια συνδεδεμένη εγκατάσταση PHP. Μπορείτε να αποκτήσετε πρόσβαση στην έξοδο του προγράμματος PHP με ένα πρόγραμμα περιήγησης στο Web, βλέποντας τη σελίδα PHP μέσω του διακομιστή. Όλα αυτά μπορούν να τρέξουν στο μηχάνημα του σπιτιού σας, αν πειραματίζετε απλώς στον προγραμματισμό της PHP.
- Γραμματοσειρά γραμμής εντολών(Command line script). Μπορείτε να δημιουργήσετε script PHP για να το εκτελέσετε χωρίς διακομιστή ή πρόγραμμα περιήγησης.

Χρειάζεστε μόνο τον αναλυτή PHP για να το χρησιμοποιήσετε με αυτόν τον τρόπο. Αυτός ο τύπος χρήσης είναι ιδανικός για σενάρια τα οποία εκτελούνται τακτικά χρησιμοποιώντας cron (on * nix ή Linux) ή το Task Scheduler (σε Windows). Αυτά τα σενάρια μπορούν επίσης να χρησιμοποιηθούν για απλές εργασίες επεξεργασίας κειμένου.

- Γράφοντας εφαρμογές γραφείου. Η PHP δεν είναι ίσως η καλύτερη γλώσσα για να δημιουργήσετε μια εφαρμογή γραφείου με γραφικό περιβάλλον εργασίας, αλλά αν γνωρίζετε πολύ καλά την PHP και θέλετε να χρησιμοποιήσετε μερικές προηγμένες λειτουργίες PHP στις εφαρμογές σας από την πλευρά του πελάτη, μπορείτε επίσης να χρησιμοποιήσετε το PHP-GTK γράψτε τέτοια προγράμματα. Έχετε επίσης τη δυνατότητα να γράφετε εφαρμογές πολλαπλών πλατφορμών με αυτόν τον τρόπο. Το PHP-GTK είναι μια επέκταση στην PHP, η οποία δεν είναι διαθέσιμη στην κύρια διανομή.»[2]

2.5.3 Πλεονεκτήματα της PHP

- Είναι λογισμικό ανοιχτού κώδικα (open source).
- Η PHP τρέχει σε όλες τις πλατφόρμες όπως Windows, Linux, Unix, Mac OS X κ.λπ.
- Είναι συμβατή με σχεδόν όλους τους server που χρησιμοποιούνται σήμερα (Apache, IIS, κ.λπ.)
- Υποστηρίζει ένα ευρύ φάσμα από Βάσεις Δεδομένων, είναι δωρεάν, είναι εύκολη να τη μάθει κανείς και τρέχει αποτελεσματικά στους servers.
- Υψηλή απόδοση.
- Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, κ.ά.).
- Ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες Web διαδικασίες.
- Χαμηλό κόστος.
- Ευκολία μάθησης και χρήσης.
- Μεταφερσιμότητα.
- Διαθεσιμότητα του πηγαίου κώδικα.

- Αρκετά frameworks με καλό documentation

2.6 To Bootstrap

2.6.1 Εισαγωγή στο Bootstrap

Το bootstrap είναι ένα web design framework. Είναι πάρα πολύ χρήσιμο framework που μας βοηθάει να κάνουμε το site μας responsive. Χρησιμοποιεί κλάσεις όπου η κάθε μια έχει το δικό της χαρακτηριστικό. Έχει στηριχθεί στα media queries της CSS αλλά με δικά της breakpoints.

«Το framework αυτό μας επιτρέπει να δημιουργήσουμε γρήγορα την web εφαρμογή μας χωρίς να χρειάζεται να ανησυχούμε για το σχεδιασμό του (το Bootstrap είναι δωρεάν) ειδικά στο στάδιο της ανάπτυξης του site μας . Το framework περιλαμβάνει εκατοντάδες στοιχεία που μπορούμε να χρησιμοποιήσουμε στην σελίδα μας ,τα οποία και ως επί το πλείστον είναι responsive , περιλαμβάνοντας grid system, buttons, forms, panels, navigation bars και πολλά ακόμα.»[6]

«Το Bootstrap είναι μια συλλογή εργαλείων ανοιχτού κώδικα (Ελεύθερο λογισμικό) για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Περιέχει HTML και CSS για τις μορφές τυπογραφίας, κουμπιά πλοήγησης και άλλων στοιχείων του περιβάλλοντος, καθώς και προαιρετικές επεκτάσεις JavaScript. Είναι το πιο δημοφιλές πρόγραμμα στο GitHub και έχει χρησιμοποιηθεί από τη NASA και το MSNBC, μεταξύ άλλων.

Προέλευση:

Το Bootstrap αναπτύχθηκε από τον Mark Otto και τον Jacob Thornton στο Twitter ως ένα πλαίσιο για την ενθάρρυνση της συνέπειας στα εσωτερικά εργαλεία. Πριν το Bootstrap, διάφορες βιβλιοθήκες χρησιμοποιήθηκαν για την ανάπτυξη της διεπαφής, η οποία οδήγησε σε αντιφάσεις και υψηλή φορολογική επιβάρυνση συντήρησης. Σύμφωνα με τον Twitter developer Mark Otto, για την αντιμετώπιση αυτών των προκλήσεων : "... πήρα μαζί μία σούπερ μικρή ομάδα προγραμματιστών για να σχεδιάσει και να κατασκευάσει ένα νέο εργαλείο εσωτερικής και είδα μια ευκαιρία να κάνουμε κάτι περισσότερο. Μέσω αυτής της διαδικασίας, είδαμε τους εαυτούς μας να χτίζουν κάτι πολύ

πιο σημαντικό από οποιοδήποτε άλλο εσωτερικό εργαλείο. Μήνες αργότερα, καταλήξαμε σε μια πρώιμη έκδοση του Bootstrap ως έναν τρόπο για να καταγράψουμε και να μοιραστούμε κοινά πρότυπα σχεδιασμού και περιουσιακών στοιχείων εντός της εταιρείας ».

Η πρώτη εγκατάσταση υπό πραγματικές συνθήκες συνέβη κατά τη διάρκεια της πρώτης Hackweek του Twitter ». Ο Mark Otto έδειξε σε κάποιους συναδέλφους πώς να επιταχύνουν την ανάπτυξη του έργου τους με τη βοήθεια της εργαλειοθήκης. Ως αποτέλεσμα, δεκάδες ομάδες έχουν μετακινηθεί στο πλαίσιο.

Χαρακτηριστικά:

Το Bootstrap έχει σχετικά ελλιπή υποστήριξη για HTML5 και CSS, αλλά είναι συμβατό με όλους τους φυλλομετρητές (browsers). Βασικές πληροφορίες συμβατότητας των ιστοσελίδων ή εφαρμογές είναι διαθέσιμες για όλες τις συσκευές και τα προγράμματα περιήγησης. Υπάρχει μια έννοια της μερικής συμβατότητας που κάνει τα βασικά στοιχεία μιας ιστοσελίδας που διατίθενται για όλες τις συσκευές και τα προγράμματα περιήγησης. Για παράδειγμα, οι ιδιότητες πάνω στις οποίες θεσπίστηκε το CSS3 για στρογγυλεμένες γωνίες, κλίσεις και σκιές, χρησιμοποιούνται από το Bootstrap παρά την έλλειψη υποστήριξης από μεγάλα προγράμματα περιήγησης στο Web. Αυτά επεκτείνουν τη λειτουργικότητα του πακέτου εργαλείων, αλλά δεν απαιτούνται για τη χρήση του.

Από την έκδοση 2.0 υποστηρίζει επίσης ανταποκρίσιμο σχεδιασμό (responsive design). Αυτό σημαίνει ότι η διάταξη των ιστοσελίδων προσαρμόζεται δυναμικά, λαμβάνοντας υπόψη τα χαρακτηριστικά της συσκευής που χρησιμοποιείται (PC, tablet, κινητό τηλέφωνο).

Το Bootstrap είναι ανοικτού κώδικα και είναι διαθέσιμο στο GitHub. Οι προγραμματιστές ενθαρρύνονται να συμμετέχουν στο έργο και να κάνουν τη δική τους συνεισφορά στην πλατφόρμα.

Δομή και λειτουργία:

Το Bootstrap είναι σπονδυλωτό και αποτελείται ουσιαστικά από μια σειρά στυλ(stylsheets) που εφαρμόζουν τα διάφορα συστατικά του πακέτου εργαλείων. Ένα στυλ που ονομάζεται bootstrap.less περιλαμβάνει τα συστατικά stylesheets. Οι προγραμματιστές μπορούν να προσαρμόσουν το αρχείο Bootstrap, επιλέγοντας τα

στοιχεία που θέλουν να χρησιμοποιήσουν στο έργο τους.

Προσαρμογές είναι δυνατές σε περιορισμένη έκταση μέσω ενός κεντρικού στυλ διαμόρφωσης. Η χρήση γλώσσας στυλ επιτρέπει τη χρήση για μεταβλητές, λειτουργίες και φορείς (operators), ένθετους επιλογείς, γνωστά και ως μείγματα mixin.

Από την έκδοση 2.0, η διαμόρφωση του Bootstrap έχει επίσης μία ειδική επιλογή "Προσαρμογή" στην τεκμηρίωση (documentation). Επιπλέον, ο σχεδιαστής του έργου επιλέγει σε μια φόρμα τα επιθυμητά συστατικά και τα προσαρμόζει, εάν είναι αναγκαίο, σε τιμές διαφόρων εναλλακτικών λύσεων για τις ανάγκες του. Στη συνέχεια δημιουργείται ένα πακέτο που περιλαμβάνει ήδη το προ-χτισμένο CSS στυλ.

Σύστημα πλέγματος(Grid System) και ανταποκρίσιμος σχεδιασμός(responsive design).

Το Bootstrap έρχεται σταθερό με 940 pixel πλάτος. Εναλλακτικά, ο προγραμματιστής μπορεί να χρησιμοποιήσει μία μεταβλητού πλάτους διάταξη. Και για τις δύο περιπτώσεις, η εργαλειοθήκη έχει τέσσερις παραλλαγές για χρήση των διαφόρων ψηφισμάτων και τύπους συσκευών κινητά τηλέφωνα, πορτρέτα και το τοπία, ταμπλέτες και υπολογιστές με χαμηλή και υψηλή ανάλυση. Κάθε παραλλαγή ρυθμίζει το πλάτος των στηλών.

Η κατανόηση του CSS στυλ:

Το Bootstrap παρέχει ένα σύνολο στυλ που παρέχουν βασικούς ορισμούς στυλ για όλα τα βασικά στοιχεία HTML. Αυτά παρέχουν ενιαία, σύγχρονη εμφάνιση για πίνακες, μορφοποίηση κειμένου, καθώς και στοιχεία μιας φόρμας.

Επαναχρησιμοποιήσιμα συστατικά:

Εκτός από τα βασικά HTML στοιχεία, το Bootstrap περιέχει και άλλα στοιχεία περιβάλλοντος που χρησιμοποιούνται συχνά. Αυτά περιλαμβάνουν κουμπιά με προηγμένα χαρακτηριστικά (π.χ. ομαδοποίηση κουμπιών ή drop -down επιλογή, οριζόντιες και κάθετες καρτέλες, πλοήγηση, σελιδοποίηση, κ.λπ.), ετικέτες, προηγμένες τυπογραφικές δυνατότητες, εικονίδια, προειδοποιητικά μηνύματα και μια γραμμή προόδου.

JavaScript στοιχεία:

Το Bootstrap έρχεται με πολλά συστατικά JavaScript σε μια μορφή jQuery plugin.

Παρέχουν πρόσθετη διεπαφή χρήστη με στοιχεία όπως παράθυρα διαλόγου, επεξηγήσεις, και καρουσέλ. Μπορούν επίσης να επεκτείνουν τη λειτουργικότητα ορισμένων υφιστάμενων στοιχείων της διασύνδεσης, όπως για παράδειγμα μια αυτόματη πλήρη λειτουργία για πεδία εισαγωγής. Στην έκδοση 2.0, υποστηρίζονται τα ακόλουθα JavaScript plugins: Modal, Αναπτυσσόμενο, Scrollspy, Tab, Tooltip, Popover, Alert, Button, Collapse, Carousel και Typeahead.

Διατίθεται επίσης μια υλοποίηση του Bootstrap Twitter που χρησιμοποιεί το Dojo Toolkit. Λέγεται Dojo Bootstrap και είναι στα plugins του Bootstrap Twitter. Χρησιμοποιεί 100% κώδικα Dojo και έχει υποστήριξη για AMD (Asynchronous Module Definition).

Χρήση:

Για να χρησιμοποιηθεί το Bootstrap σε μια σελίδα HTML, ο σχεδιαστής του έργου κάνει λήψη του στυλ CSS Bootstrap και περιλαμβάνει μια σύνδεση στο αρχείο HTML. .»[3]

2.6.2 Πλεονεκτήματα της Bootstrap

- Είναι responsive

Κυρίαρχος ρόλος ύπαρξης, αλλά και χρήσης της από εμάς είναι το γεγονός ότι είναι responsive.

- User Friendly και User Experience

Το γεγονός ότι είναι responsive, έχει ως απότοκο να κάνει το site να είναι User Friendly. Συνάμα, με αυτό τον τρόπο βελτιώνεται η εμπειρία του χρήστη (User Experience) στο site.

- Έτοιμες κλάσεις

Το Bootstrap έχει έτοιμες κλάσεις που μπορούμε να χρησιμοποιήσουμε. Έτσι, μπορούμε και γράφουμε πιο εύκολα κώδικα, γλιτώνουμε χρόνο αλλά και μέγεθος στα αρχεία αφού η HTML και η CSS έχουν λιγότερες γραμμές κώδικα

- Πληθώρα χρήσεων

Έχει πάρα πολλές κλάσεις που μπορούμε να χρησιμοποιήσουμε είτε HTML, CSS ακόμη και javascript

- Εύκολη εγκατάσταση

Υπάρχουν δύο τρόποι χρήσης της. Από την μία μεριά μπορούμε να επωφεληθούμε από

τις κλάσεις της χρησιμοποιώντας μέσω cdn. Από την άλλη μεριά, μπορούμε να κατεβάσουμε τα αρχεία της και να τα αποθηκεύσουμε στο project μας

- Είναι δωρεάν

Χωρίς κανένα απολύτως κόστος μπορούμε να κάνουμε χρήση της Bootstrap κάτι που την καθιστά ακόμα καλύτερη και πιο δημοφιλή

2.7 Η MySQL

2.7.1 Εισαγωγή στην MySQL

«Μια Βάση Δεδομένων (Database) είναι ένας οργανωμένος τρόπος αποθήκευσης πληροφοριών και πρόσβασης τους με πολλούς τρόπους με διάφορα προγράμματα. Μια βάση δεδομένων είναι κάτι παραπάνω από μια απλή συλλογή αποθηκευμένων στοιχείων. Ένας άλλος ορισμός είναι ότι μια βάση δεδομένων είναι ένα ολοκληρωμένο σύστημα που αποτελείται από δεδομένα (data) και από το κατάλληλο λογισμικό (software), τα οποία χρησιμοποιώντας το υλικό (hardware) βοηθούν στην ενημέρωση και πληροφόρηση των χρηστών (users). Το Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS, DataBase Management System) είναι ένα σύνολο από προγράμματα που επιτρέπουν τον χειρισμό των δεδομένων μιας ή περισσότερων βάσεων δεδομένων που ανήκουν στο ίδιο σύστημα. Το DBMS περιέχει κάποια εργαλεία γενικής χρήσης για να μπορούμε να δημιουργούμε και να χειριζόμαστε τα δεδομένα. Ένα πρόγραμμα που διαχειρίζεται βάσεις δεδομένων αποκαλείται Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS, DataBase Management System) και με την βοήθειά του μπορούμε να αποθηκεύσουμε, προσθέσουμε, τροποποιήσουμε, εμφανίσουμε ή και να διαγράψουμε τα αποθηκευμένα δεδομένα.

Τα δεδομένα που υπάρχουν στις βάσεις δεδομένων πρέπει να είναι:

- Ολοκληρωμένα (Integrated), δηλαδή τα δεδομένα πρέπει να είναι αποθηκευμένα σε ομοιόμορφα οργανωμένα σύνολα αρχείων όπου δεν πρέπει να υπάρχει επανάληψη ή πλεονασμός (redundancy) των ίδιων στοιχείων.
- Καταμεριζόμενα (Shared), δηλαδή να μπορούν περισσότεροι του ενός χρήστες να βλέπουν και να μοιράζονται τα ίδια δεδομένα την ίδια χρονική στιγμή.

Στόχοι και χρησιμότητα βάσης δεδομένων:

- Ο περιορισμός της πολλαπλής αποθήκευσης των ίδιων στοιχείων (redundancy).
 - Ο καταμερισμός (sharing) των ίδιων στοιχείων σ' όλους τους χρήστες.
 - Η ομοιομορφία (uniformity) στον χειρισμό και την αναπαράσταση των δεδομένων.
 - Η επιβολή κανόνων ασφαλείας (security).
 - Η διατήρηση της ακεραιότητας (integrity) και της αξιοπιστίας (reliability) των δεδομένων.
 - Η ανεξαρτησία των δεδομένων (data independence) και των προγραμμάτων από τον φυσικό τρόπο αποθήκευσης των δεδομένων.»[7]
-
- «Η MySQL είναι ένα σύστημα βάσης δεδομένων που χρησιμοποιείται στο διαδίκτυο
 - MySQL είναι ένα σύστημα βάσης δεδομένων που τρέχει σε ένα διακομιστή
 - Η MySQL είναι ιδανική τόσο για μικρές όσο και για μεγάλες εφαρμογές
 - Η MySQL είναι πολύ γρήγορη, αξιόπιστη και εύκολη στη χρήση
 - Η MySQL χρησιμοποιεί τυπική SQL
 - MySQL συγκεντρώνεται σε διάφορες πλατφόρμες
 - Η MySQL είναι ελεύθερη για λήψη και χρήση
 - Η MySQL αναπτύσσεται, διανέμεται και υποστηρίζεται από την Oracle Corporation
 - MySQL ονομάστηκε από την κόρη του συνιδρυτή Monty Widenius: My»[1]

2.8 PHPStorm

Το PHPStorm είναι ένα πάρα πολύ χρήσιμο εργαλείο για την ανάπτυξη κώδικα και όχι μόνο. Πιο συγκεκριμένα, σε αυτό το εργαλείο μπορεί να αναπτυχθεί κώδικας σε μια πληθώρα γλώσσων προγραμματισμού μιας και υποστηρίζει την δημιουργία αρκετών αρχείων όπως τα: .html, .css, .php και άλλα. Έχοντας κάνει ενσωμάτωση της γλώσσας PHP, το auto complete εμφανίζει και διορθώνει όλες τις συναρτήσεις που χρησιμοποιούμε. Επιπροσθέτως, σημαντικό είναι το γεγονός ότι το συγκεκριμένο πρόγραμμα κάνει αυτόματα save με οποιαδήποτε αλλαγή χαρακτήρα, επομένως δεν χρειάζεται να το

κάνουμε εμείς κάτι το οποίο μας γλυτώνει χρόνο και κόπο αλλά και κερδίζουμε την επιβεβαίωση ότι οι αλλαγές έχουν αποθηκευτεί. Συνάμα, αποφεύγονται τυχόν απώλειες δεδομένων για περιπτώσεις όπου θα μπορούσε να κλείσει αναπάντεχα ο υπολογιστής. Επίσης, μπορούμε να κάνουμε προβολή όλων των πινάκων της βάσης δεδομένων το οποίο βοηθάει στην εξοικονόμηση χρόνου.

2.9 PHPMyAdmin

2.9.1 Εισαγωγή στο PhpMyAdmin

«Το phpMyAdmin είναι ένα εργαλείο ελεύθερου λογισμικού γραμμένο στην PHP, το οποίο προορίζεται να χειριστεί τη διαχείριση της MySQL μέσω του Διαδικτύου. Το phpMyAdmin υποστηρίζει ένα ευρύ φάσμα λειτουργιών στις υπηρεσίες MySQL και MariaDB. Συχνά χρησιμοποιούμενες λειτουργίες (διαχείριση βάσεων δεδομένων, πίνακες, στήλες, σχέσεις, ευρετήρια, χρήστες, δικαιώματα κ.λπ.) μπορούν να εκτελεστούν μέσω του περιβάλλοντος χρήστη, ενώ έχετε ακόμα τη δυνατότητα να εκτελέσετε άμεσα οποιαδήποτε δήλωση SQL.» [8]

2.9.2 Χαρακτηριστικά PhpMyAdmin

- «Δυναμική διαδικτυακή διασύνδεση
- Υποστήριξη για τις περισσότερες λειτουργίες MySQL:
 - Να περιηγηθείτε και να αποθέστε βάσεις δεδομένων, πίνακες, προβολές, πεδία και ευρετήρια
 - να δημιουργήσετε, να αντιγράψετε, να ρίξετε, να μετονομάσετε και να αλλάξετε βάσεις δεδομένων, πίνακες, πεδία και ευρετήρια
 - διακομιστή συντήρησης, βάσεις δεδομένων και πίνακες, με προτάσεις σχετικά με τη διαμόρφωση του διακομιστή
 - να εκτελεί, να επεξεργάζεται και να επισημαίνει οποιαδήποτε δήλωση SQL, ακόμη και παρτίδες ερωτημάτων

- να διαχειρίζεστε τους λογαριασμούς και τα προνόμια των χρηστών της MySQL
- διαχείριση αποθηκευμένων διαδικασιών και ενεργοποιήσεων
- Εισαγωγή δεδομένων από CSV και SQL
- Εξαγωγή δεδομένων σε διάφορες μορφές: CSV, SQL, XML, PDF, ISO / IEC 26300 - Κείμενο και υπολογιστικό φύλλο OpenDocument, Word, LATEX και άλλοι
- Διαχείριση πολλών διακομιστών
- Δημιουργία γραφικών της διάταξης της βάσης δεδομένων σας σε διάφορες μορφές
- Δημιουργία πολύπλοκων ερωτημάτων χρησιμοποιώντας Query-by-example (QBE)
- Αναζήτηση σε παγκόσμιο επίπεδο σε μια βάση δεδομένων ή σε ένα υποσύνολο της
- Μετασχηματισμός αποθηκευμένων δεδομένων σε οποιαδήποτε μορφή χρησιμοποιώντας ένα σύνολο προκαθορισμένων λειτουργιών, όπως την εμφάνιση δεδομένων BLOB ως εικόνα ή λήψης» [8]

Κεφάλαιο 3

ΛΕΙΤΟΥΡΓΙΕΣ ΚΑΙ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΧΡΗΣΤΩΝ

3.1 Γενικά

Στο κεφάλαιο αυτό γίνεται :

- συνοπτική περιγραφή όλων των λειτουργιών και υπηρεσιών της εφαρμογής,
- ανάλυση των χαρακτηριστικών των χρηστών

3.2 Υπηρεσίες που παρέχει η εφαρμογή

Η κύρια υπηρεσία αυτής της εφαρμογής είναι η αναζήτηση ιατρού με επιλογή κριτηρίων που επιθυμεί ο κάθε χρήστης.

Ταυτόχρονα, η δεύτερη υπηρεσία αφορά τους γιατρούς, οι οποίοι μέσα από την πλατφόρμα αυτή μπορούν και προβάλλονται επαγγελματικά.

Υπάρχουν 3 κατηγορίες χρηστών οι οποίες θα αναλυθούν στην συνέχεια.

3.2.1 Λειτουργίες Χρήστη-Επισκέπτη

Η εφαρμογή παρέχει σε έναν απλό χρήστη-επισκέπτη τις εξής υπηρεσίες, οι οποίες είναι η αναζήτηση και η εγγραφή στο σύστημα.

Η αναζήτηση είναι διαθέσιμη και στα τρία επίπεδα χρηστών (χρήστης-επισκέπτης, εγγεγραμμένος χρήστης, διαχειριστής) και είναι η κύρια λειτουργία της εφαρμογής. Ο χρήστης μπορεί να αναζητήσει τον ιατρό που επιθυμεί με βάση την ειδικότητα, την πόλη και την ασφάλιση.

Η εγγραφή αφορά μόνο τους ιατρούς όπου μπορούν να μετατραπούν από απλοί επισκέπτες σε εγγεγραμμένοι χρήστες, ώστε να καταχωρήσουν τα στοιχεία τους και να μπορούν να συμπεριλαμβάνονται στα κριτήρια-φίλτρα αναζήτησης.

3.2.2 Λειτουργίες Χρήστη-Μέλους

Ο χρήστης-μέλος, δηλαδή ο ιατρός, έχει τις λειτουργίες που έχει ο χρήστης-επισκέπτης, πιο συγκεκριμένα μπορεί να κάνει αναζήτηση και εγγραφή. Επιπλέον, για να μπορέσει

ο εκάστοτε ιατρός να μπει στον λογαριασμό του, απαιτείται η σύνδεση (log in). Στην συνέχεια γίνεται ανακατεύθυνση στην φόρμα με όλες τις πληροφορίες που έχει εισάγει κατά την εγγραφή και μπορεί να τις δει. Στη συνέχεια, μπορεί να εισάγει τα επαγγελματικά του στοιχεία. Πιο συγκεκριμένα, μπορεί να εισάγει την πόλη, όπου βρίσκεται το ιατρείο και το τηλέφωνο ιατρείου και επίσης μία ή περισσότερες ειδικότητες και ασφαλίσεις. Μια άλλη λειτουργία είναι η αποσύνδεση από το σύστημα, όπου αποσυνδέεται ο χρήστης- μέλος από το σύστημα, και επανέρχεται ως απλός επισκέπτης στην αρχική σελίδα.

3.2.3 Λειτουργίες Διαχειριστή

Ο Διαχειριστής βρίσκεται στην κορυφή της πυραμίδας των χρηστών της εφαρμογής και του αναλογούν όλα τα δικαιώματα που έχει ο χρήστης-μέλος και ο απλός χρήστης επισκέπτης της σελίδας.

Επιπλέον ο Διαχειριστής έχει πρόσβαση στην Βάση Δεδομένων (ΒΔ) μέσω μίας σελίδας που μόνο αυτός μπορεί να εισέλθει με αποτέλεσμα να μπορεί να κάνει:

- Προβολή όλων των στοιχείων όλων των ιατρών και των μοναδικών κωδικών τους
- Προσθήκη ιατρείου. Το σύστημα έχει φτιαχτεί με τέτοιο τρόπο ώστε να επαληθεύεται η ύπαρξη των ιατρών και η ορθότητα των στοιχείων τους. Έτσι, κατά την εγγραφή ενός ιατρού ο διαχειριστής καλεί στο τηλέφωνο που έχει εισάγει ο ιατρός και σε περίπτωση που έχει δεύτερο ιατρείο, προσθέτει στην βάση το τηλέφωνο και την περιοχή του δεύτερου ιατρείου. Να σημειωθεί πως μπορεί να εισάγει όσα ιατρεία (πόλεις και τηλέφωνα) επιθυμεί ο ιατρός.
- Προσθήκη περισσότερων ειδικοτήτων και ασφαλίσεων
- Διαγραφή ιατρείων, ειδικοτήτων και ασφαλίσεων

3.3 Χαρακτηριστικά Χρηστών

Όσον αφορά τους επισκέπτες και τους εγγεγραμμένους χρήστες, να σημειωθεί πως δεν απαιτούνται εξειδικευμένες γνώσεις με τον υπολογιστή. Συνάμα, ο Διαχειριστής δεν

χρειάζεται να γνωρίζει από βάσεις δεδομένων, διότι μπορεί απλά και εύκολα να προβάλει, να επεξεργαστεί και να διαγράψει τα στοιχεία των ιατρών μέσα από μία σελίδα της εφαρμογής.

Ο απλός επισκέπτης, που είναι αυτός στον οποίο απευθύνεται κατά κύριο λόγο η εφαρμογή, μπορεί με ευκολία να αναζητήσει το γιατρό που επιθυμεί με κριτήρια που εκείνος θέτει, διότι η σελίδα είναι πολύ απλή, περιεκτική και φιλική και αυτό επειδή δεν περιέχει περιττή πληροφορία. Ο χρήστης – μέλος, δηλαδή ο ιατρός, είναι επίσης πολύ σημαντικός παράγοντας για την εφαρμογή, αφού τα αποτελέσματα της αναζήτησης είναι οι εγγεγραμμένοι και επικυρωμένοι ιατροί. Τέλος, ο διαχειριστής είναι εκείνος ο οποίος επικυρώνει την ύπαρξη των γιατρών και την ορθότητα των στοιχείων τους. Άρα είναι υπεύθυνος για την έγκυρη και εύρυθμη λειτουργία της εφαρμογής.

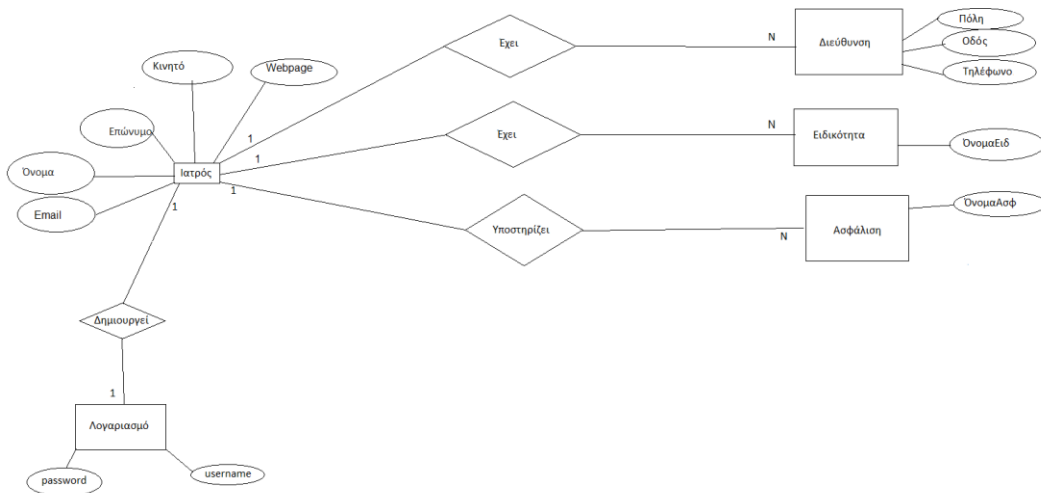
Κεφάλαιο 4

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

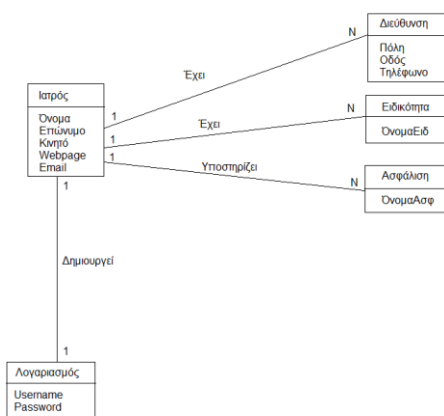
4.1 Η αρχιτεκτονική της Βάσης Δεδομένων

4.1.1 Γενικά

Η δομή της Βάσης δεδομένων περιγράφεται από τα παρακάτω δύο σχήματα (διάγραμμα οντοτήτων-συσχετίσεων ER και διάγραμμα κλάσεων).



Εικόνα 1: Διάγραμμα Οντοτήτων-Συσχετίσεων ER



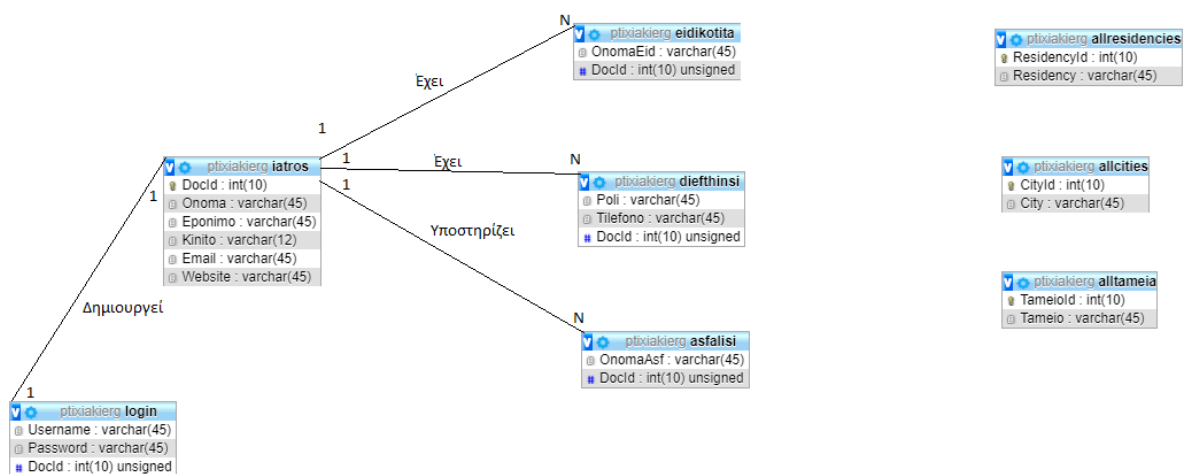
Εικόνα 2: Διάγραμμα Κλάσεων

Και στα δύο παραπάνω σχήματα φαίνεται η σχέση του γιατρού με τους άλλους πίνακες. Με άλλα λόγια, φαίνονται οι σχέσεις 1 προς 1 (ένα προς ένα) και 1 προς N (ένα προς πολλά).

Πιο συγκεκριμένα, ένας ιατρός μπορεί:

- να έχει πολλές διευθύνσεις (1 προς N)
- να έχει πολλές ειδικότητες (1 προς N)
- να υποστηρίζει πολλές ασφαλίσεις (1 προς N)
- δημιουργεί ένα λογαριασμό (1 προς 1)

Κατόπιν, ακολουθεί το διάγραμμα με όλους τους πίνακες, οι οποίοι θα αναλυθούν στην συνέχεια. Το DocId είναι τύπου integer (ακέραιος αριθμός) και είναι αύξων αριθμός (auto increment) ξεκινώντας από το 1 όπου η πρώτη εγγραφή θα έχει τον αριθμό 1, η δεύτερη τον αριθμό 2 κ.ο.κ. Επίσης, όπου υπάρχει varchar 45 σημαίνει πως μπορούν να εισαχθούν μέχρι 45 χαρακτήρες.



Εικόνα 3: Διάγραμμα Πινάκων

4.1.2 Πίνακας ιατρού (iatros)

Υπάρχει ο πίνακας iatros όπου είναι τα πρώτα στοιχεία που καταχωρεί ο ιατρός

κατά την εγγραφή του. Αυτά είναι: Όνομα, Επώνυμο, Κινητό, Email και Website.

Παράλληλα καταχωρείται και ο κωδικός DocId όπου είναι μοναδικός και είναι πρωτεύον κλειδί σε αυτόν τον πίνακα.

4.1.3 Πίνακας λογαριασμού (login)

Ο πίνακας login αφορά πάλι στοιχεία που καταχωρούνται κατά την εγγραφή, αλλά αποθηκεύονται σε ξεχωριστό πίνακα, διότι όταν ο χρήστης κάνει login ταυτοποιείται η ορθότητα του username, του password και του DocId. Εάν είναι σωστά τα στοιχεία τότε γίνεται επιτυχής η σύνδεση. Εδώ, το DocId είναι ξένο κλειδί.

4.1.4 Πίνακες ειδικότητας, διεύθυνσης και ασφάλισης (eidikotita, diefthinsi, asfalisi)

Οι πίνακες eidikotita, diefthinsi, asfalisi αφορούν τα αντίστοιχα πεδία και είναι ξεχωριστοί πίνακες από τον πίνακα iatros, διότι ένας ιατρός μπορεί να έχει περισσότερες από μία ειδικότητες, ιατρεία και ασφαλίσεις.

Ο πίνακας eidikotita έχει τα ονόματα των ειδικοτήτων των ιατρών, ο πίνακας diefthinsi περιέχει τις πόλεις και τα τηλέφωνα των ιατρείων και ο πίνακας ασφάλιση περιέχει τα ονόματα των υποστηριζόμενων ασφαλίσεων του εκάστοτε ιατρού.

4.1.5 Πίνακες όλων των ειδικοτήτων, διευθύνσεων και ασφαλίσεων (allresidencies, allcities, alltameia)

Οι πίνακες allresidencies, allcities, alltameia αφορούν όλες τις ειδικότητες, τις πόλεις και τις ασφαλίσεις αντίστοιχα, που μπορεί να καταχωρήσει ο ιατρός κατά την επεξεργασία των στοιχείων του. Ο καθένας πίνακας έχει ένα μοναδικό Id.

Κεφάλαιο 5

ΑΠΑΙΤΗΣΕΙΣ ΣΥΣΤΗΜΑΤΟΣ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑ

ΧΡΗΣΗΣ ΣΥΣΤΗΜΑΤΟΣ

5.1 Γενικά

Κάθε πληροφοριακό σύστημα, έτσι και αυτό, έχει κάποιες απαιτήσεις τόσο από τους χρήστες, όσο και από το software και το hardware. Σε αυτό ,λοιπόν, το κεφάλαιο θα αναλυθούν οι απαιτήσεις του συγκεκριμένου συστήματος, καθώς και θα περιγραφεί το αποτέλεσμα χρήσης του.

5.2 Ανάλυση των απαιτήσεων

- Απαιτήσεις από τους χρήστες:

Πολύ σημαντικό ρόλο για αυτή την εφαρμογή παίζει το γεγονός ότι δεν απαιτεί εξειδικευμένες γνώσεις υπολογιστή. Ένας απλός εξοικειωμένος χρήστης με τον υπολογιστή μπορεί να περιηγηθεί στην πλατφόρμα αυτή. Αυτό συμβαίνει διότι η εφαρμογή μας χαρακτηρίζεται από την απλότητα στην χρήση της κάτι που την χαρακτηρίζει ως User-Friendly και έτσι βελτιώνεται η εμπειρία του χρήστη (User Experience).

- Απαιτήσεις από το software:

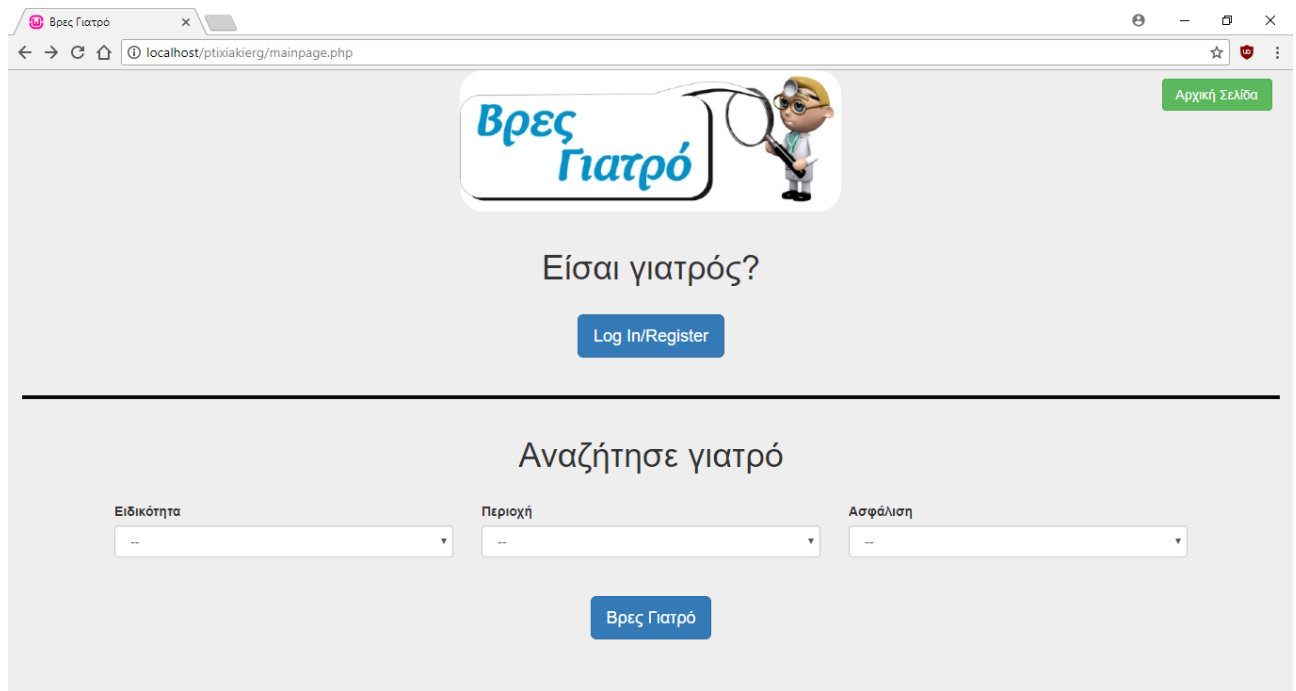
η εφαρμογή έχει απαίτηση σε software να έχεις έναν περιηγητή (browser) π.χ. Mozilla Firefox, Google Chrome, Opera κλπ.

5.3 Περιγραφή του αποτελέσματος χρήσης του συστήματος

Αρχικά, σε όλες τις σελίδες υπάρχει header με το logo της εφαρμογής στο κέντρο όπου πατώντας το γίνεται σύνδεση στην αρχική σελίδα και το κουμπί της αρχικής σελίδας πάνω δεξιά.

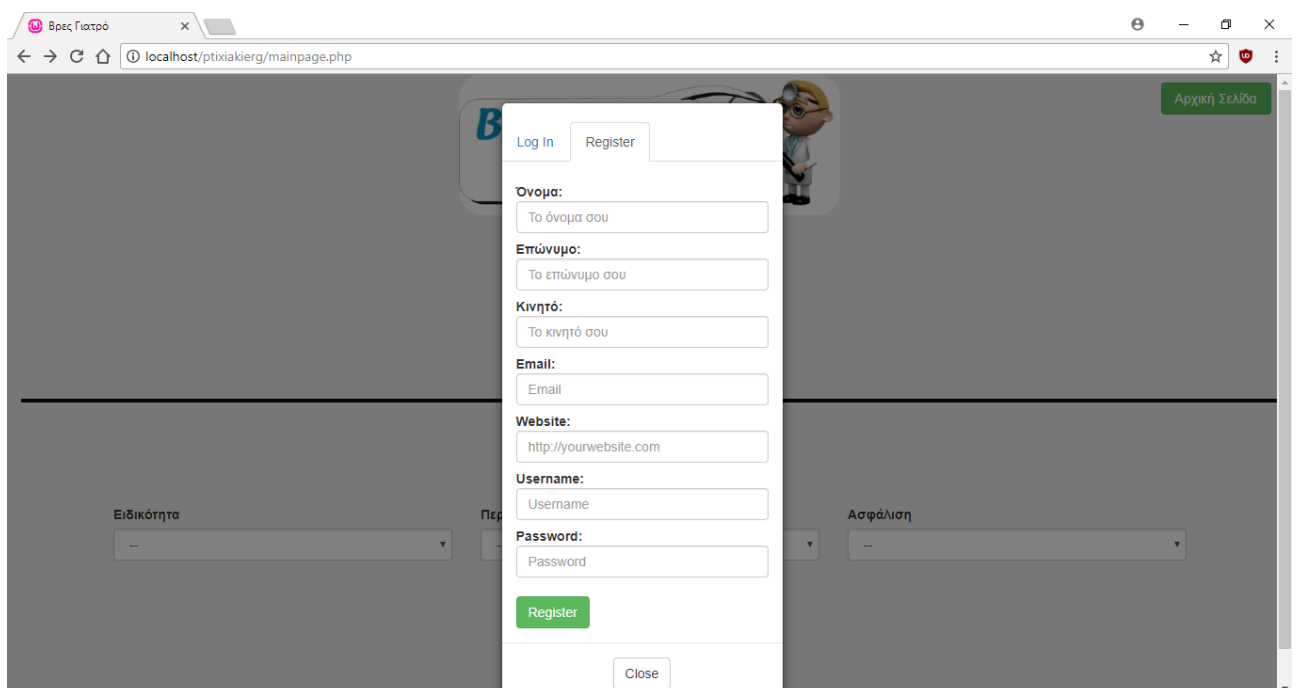
- Αρχική Σελίδα

Η Αρχική σελίδα όπου είναι η πιο σημαντική σελίδα μιας και είναι η πρώτη όψη που θα αντικρύσει ο χρήστης μπαίνοντας στην εφαρμογή, περιέχει όλες τις λειτουργίες που μπορούν να κάνουν οι χρήστες όπου θα αναλυθούν στην συνέχεια.



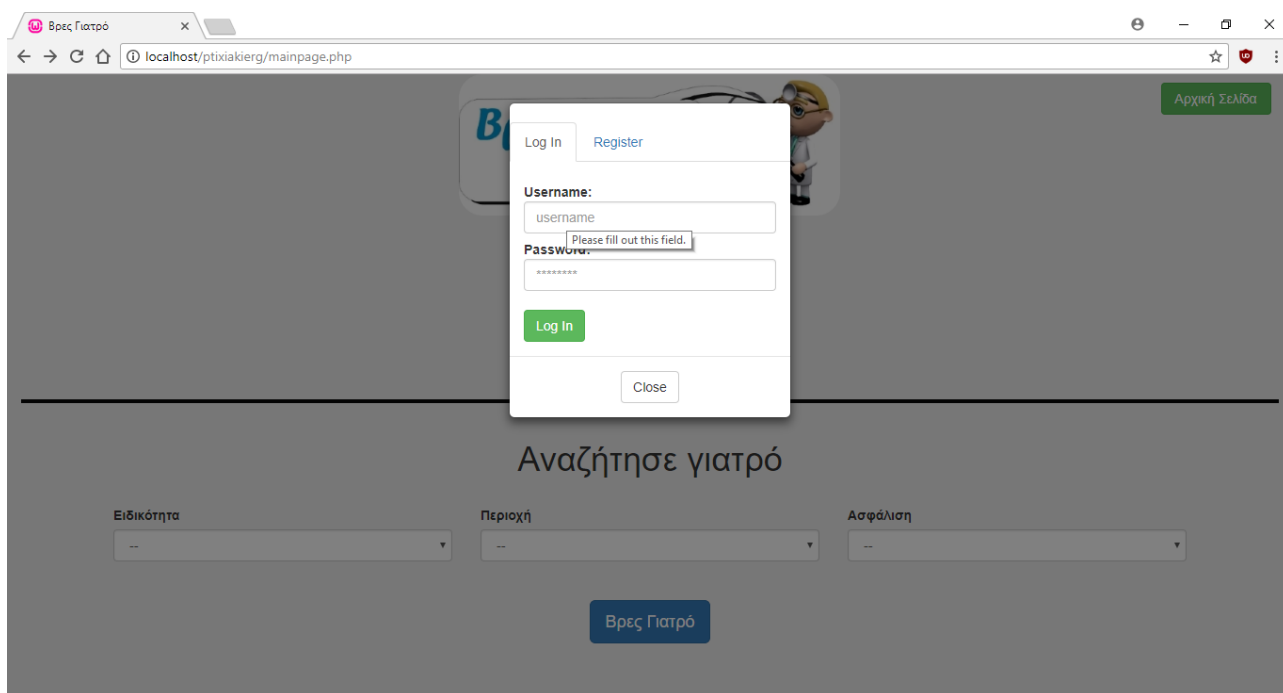
- Εγγραφή ιατρού

Στην αρχική σελίδα, δίνεται η δυνατότητα σε κάθε ιατρό να κάνει εγγραφή στο σύστημα και να δημιουργήσει έναν λογαριασμό, πατώντας το κουμπί Log In/Register. Ο κάθε ιατρός καταχωρεί τα στοιχεία του για την δημιουργία του λογαριασμού του όπου είναι: όνομα, επώνυμο, κινητό, email, website, username και password τα οποία θα αποθηκεύονται στην ΒΔ. Παράλληλα, στην φόρμα εγγραφής, ελέγχεται αν όλα τα υποχρεωτικά πεδία είναι συμπληρωμένα και όσον αφορά το email ελέγχεται αν είναι της μορφής [x@x.x](#)



- Σύνδεση ιατρού

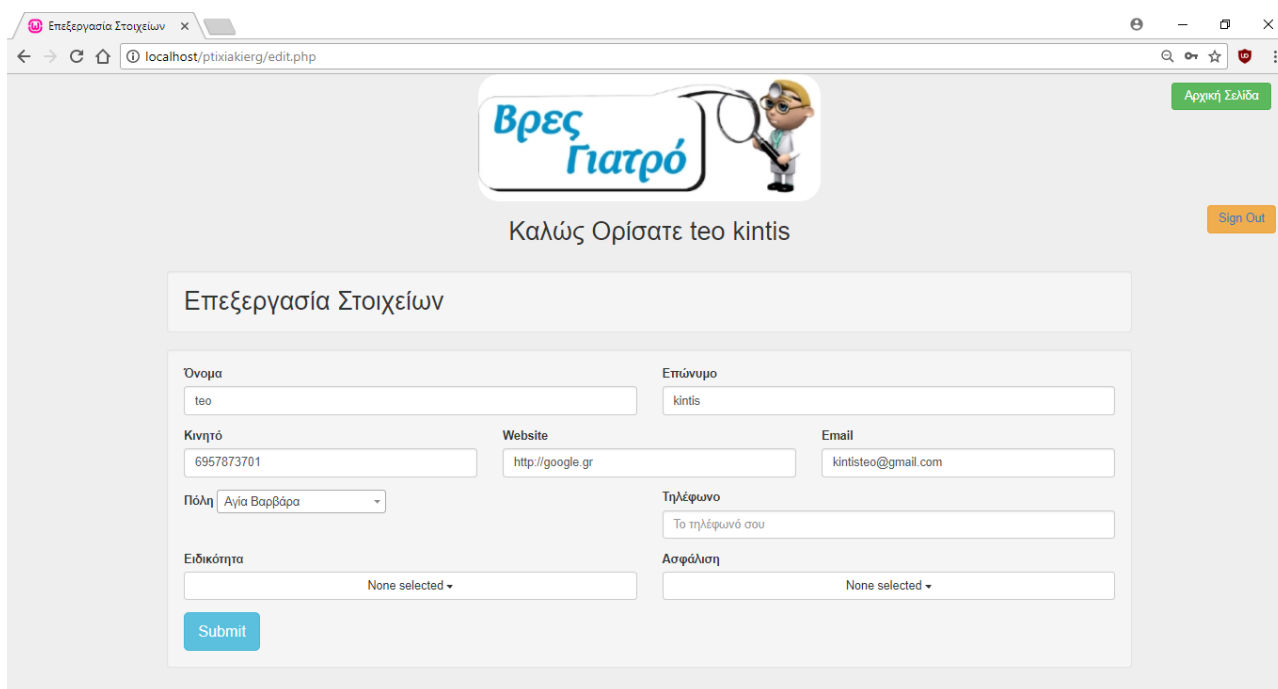
Στην αρχική σελίδα, μπορεί κάθε εγγεγραμμένος ιατρός να συνδέεται στον λογαριασμό του, πατώντας το κουμπί Log In/Register και δίνοντας το Username και το Password, τα οποία είναι υποχρεωτικά πεδία στην φόρμα, που είχε δώσει κατά την εγγραφή. Στην συνέχεια, θα μπορεί να επεξεργαστεί το προφίλ του.



- Επεξεργασία προφίλ ιατρού:

Εδώ εμφανίζονται τα στοιχεία που είχε καταχωρήσει κατά την εγγραφή. Επίσης, μπορεί να εισάγει και τα υπόλοιπα στοιχεία που απαρτίζουν το προφίλ του όπου είναι: Ειδικότητα, Ασφάλιση, Πόλη, Τηλέφωνο. Ένας ιατρός μπορεί να έχει πολλαπλές ειδικότητες και ασφαλίσεις έτσι μπορεί να διαλέξει περισσότερες από μία ειδικότητες και ασφαλίσεις. Όμως, είναι σκόπιμο που δεν δίνεται η δυνατότητα στον ιατρό να προσθέσει πάνω από μία περιοχή και πάνω από ένα τηλέφωνο. Μόλις ο ιατρός καταχωρήσει τα στοιχεία αυτά, θα γίνει η εγγραφή στην βάση. Η σκέψη είναι πως μόλις ο διαχειριστής δει την εγγραφή αυτή στην βάση, θα τηλεφωνήσει στον ιατρό με σκοπό να γίνει η επιβεβαίωση της ύπαρξης του ανθρώπου αυτού αλλά και η εγκυρότητα των στοιχείων του. Σε περίπτωση που ο ιατρός έχει πάνω από ένα ιατρεία, τότε μετά την τηλεφωνική

επικοινωνία του διαχειριστή με τον ιατρό, ο διαχειριστής θα καταχωρήσει στην βάση τα υπόλοιπα ιατρεία, δηλαδή την περιοχή και το τηλέφωνο του ιατροείου.

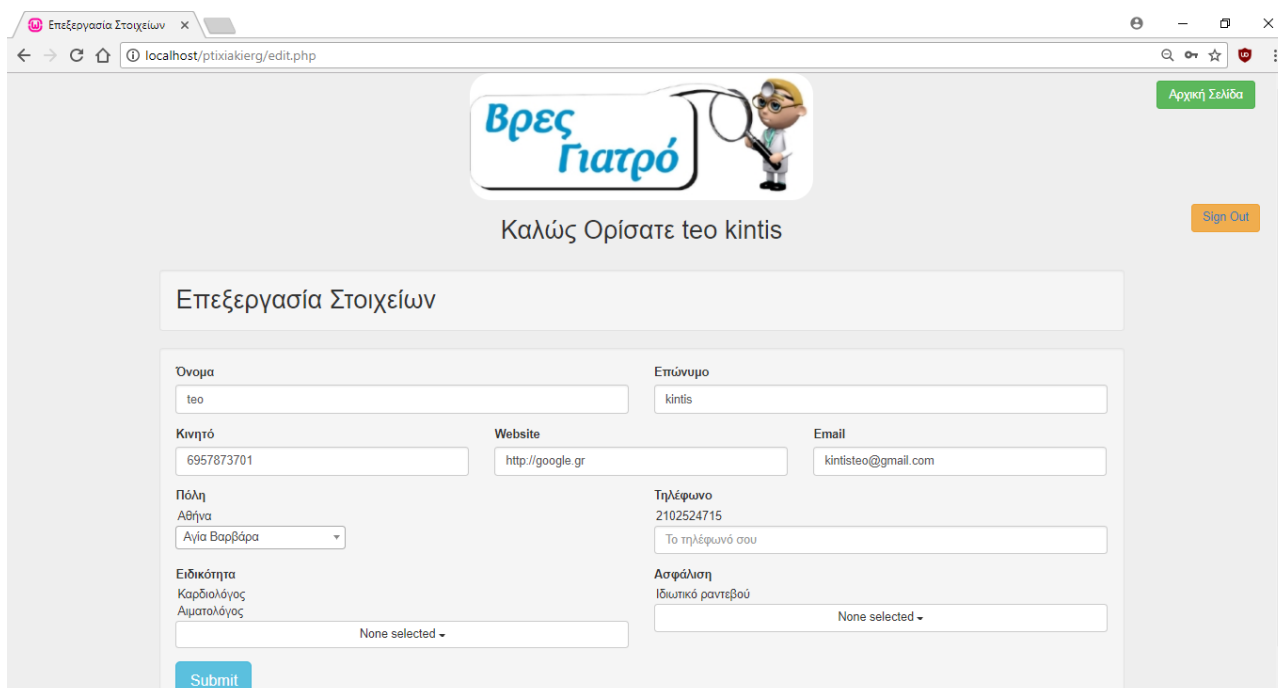


The screenshot shows a web browser window with the URL localhost/ptxiakierng/edit.php. The page features a logo for 'Bres Giatro' with a cartoon doctor character. Below the logo, the text 'Καλώς Ορίσατε teo kintis' is displayed. A form titled 'Επεξεργασία Στοιχείων' (Edit Data) is shown with the following fields:

Όνομα	teo	Επώνυμο	kintis
Κινητό	6957873701	Website	http://google.gr
		Email	kintisteo@gmail.com
Πόλη	Αγία Βαρβάρα	Τηλέφωνο	Το τηλέφωνό σου
Ειδικότητα	None selected	Ασφάλιση	None selected

A 'Submit' button is located at the bottom left of the form.

Να σημειωθεί πως όταν ο ιατρός που έχει καταχωρήσει τα υπόλοιπα στοιχεία του (ειδικότητα, ασφάλιση, πόλη και τηλέφωνο) συνδεθεί πάλι, θα μπορεί να βλέπει αυτά τα στοιχεία.



This screenshot shows the same web browser window as the previous one, but with updated data in the registration form:

Όνομα	teo	Επώνυμο	kintis
Κινητό	6957873701	Website	http://google.gr
		Email	kintisteo@gmail.com
Πόλη	Αθήνα	Τηλέφωνο	2102524715
	Αγία Βαρβάρα		Το τηλέφωνό σου
Ειδικότητα	Καρδιολόγος	Ασφάλιση	Ιδιωτικό ραντεβού
	Αιματολόγος		None selected

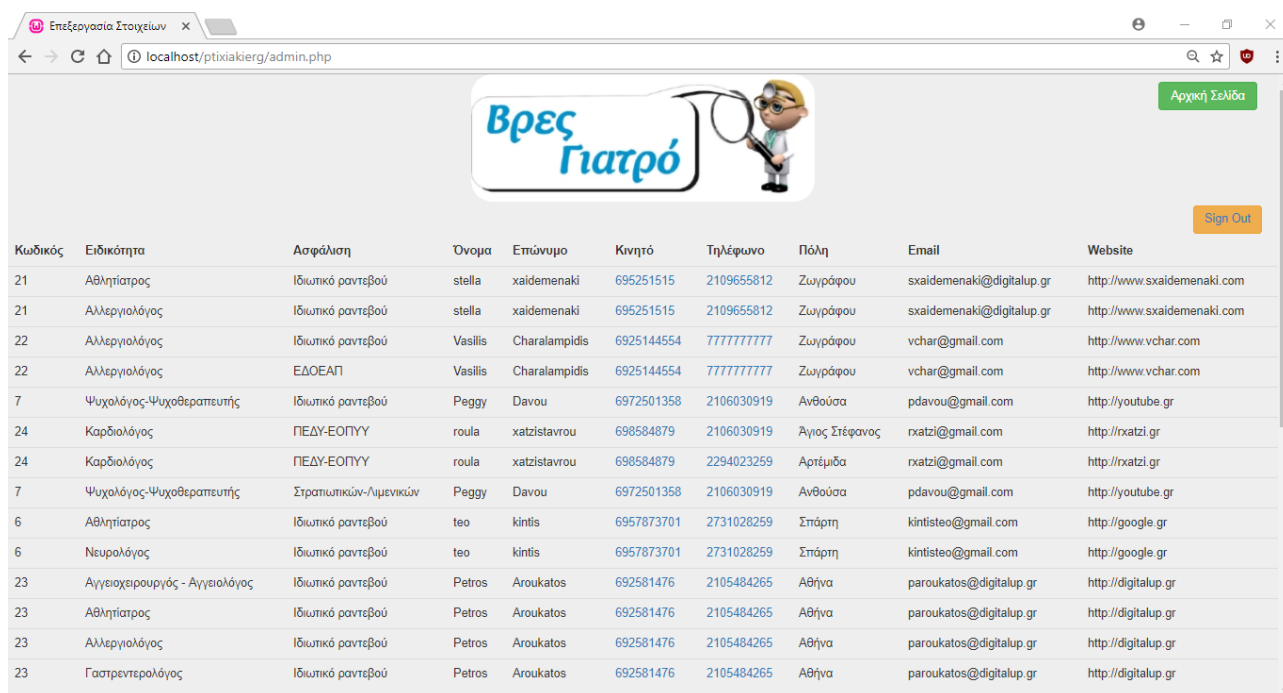
The 'Submit' button remains at the bottom left.

- Αποσύνδεση ιατρού:

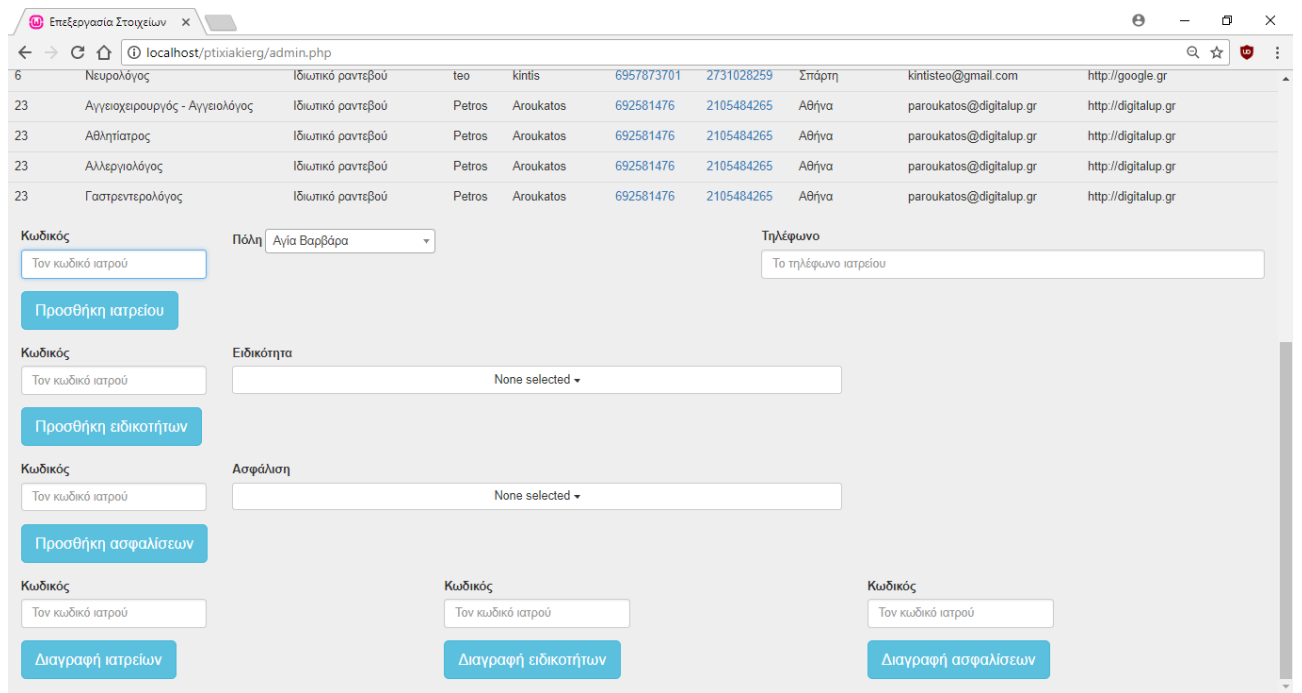
Υπάρχει πάνω δεξιά στην ιστοσελίδα η επιλογή Αποσύνδεση (Logout) όπου ο συνδεδεμένος ιατρός μπορεί να αποσυνδέεται από τον λογαριασμό του. Ύστερα, θα γίνεται ανακατεύθυνση στην αρχική σελίδα.

- Σελίδα του Διαχειριστή

Όταν ο διαχειριστής (admin) συνδεθεί στην πλατφόρμα, δεν θα γίνει ανακατεύθυνση στην ίδια σελίδα με τους ιατρούς, αλλά σε μια άλλη σελίδα που μόνο ο διαχειριστής έχει πρόσβαση. Σε αυτήν την σελίδα μπορεί να προβάλει όλα τα στοιχεία των ιατρών. Επιπρόσθετα, μπορεί να προσθέτει περισσότερα ιατρεία (πόλη και τηλέφωνο), ειδικότητες και ασφαλίσεις στους ιατρούς. Επίσης, μπορεί να διαγράψει στοιχεία όπως είναι οι ειδικότητες, οι ασφαλίσεις και τα ιατρεία.

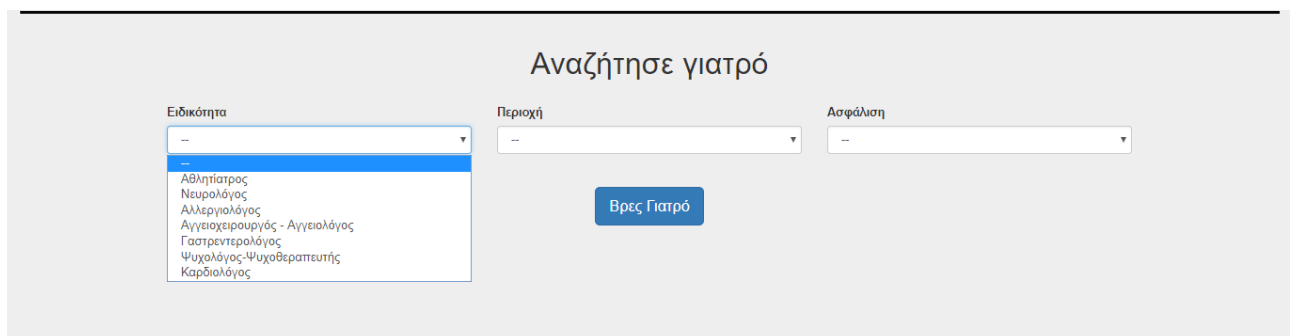


Κωδικός	Ειδικότητα	Ασφάλιση	Όνομα	Επώνυμο	Κινητό	Τηλέφωνο	Πόλη	Email	Website
21	Αθλητίατρος	Ιδιωτικό ραντεβού	stella	xaidemenaki	695251515	2109655812	Ζωγράφου	sxaidemenaki@digitalup.gr	http://www.sxaidemenaki.com
21	Αλλεργιολόγος	Ιδιωτικό ραντεβού	stella	xaidemenaki	695251515	2109655812	Ζωγράφου	sxaidemenaki@digitalup.gr	http://www.sxaidemenaki.com
22	Αλλεργιολόγος	Ιδιωτικό ραντεβού	Vasilis	Charalampidis	6925144554	7777777777	Ζωγράφου	vchar@gmail.com	http://www.vchar.com
22	Αλλεργιολόγος	ΕΔΟΕΑΠ	Vasilis	Charalampidis	6925144554	7777777777	Ζωγράφου	vchar@gmail.com	http://www.vchar.com
7	Ψυχολόγος-Ψυχοθεραπευτής	Ιδιωτικό ραντεβού	Peggy	Davou	6972501358	2106030919	Ανθούσα	pdavou@gmail.com	http://youtube.gr
24	Καρδιολόγος	ΠΕΔΥ-ΕΟΠΥΥ	roula	xatzistavrou	698584879	2106030919	Άγιος Στέφανος	rxatziz@gmail.com	http://rxatziz.gr
24	Καρδιολόγος	ΠΕΔΥ-ΕΟΠΥΥ	roula	xatzistavrou	698584879	2294023259	Αριέμιδα	rxatziz@gmail.com	http://rxatziz.gr
7	Ψυχολόγος-Ψυχοθεραπευτής	Στρατιωτικών-Ύμηνικών	Peggy	Davou	6972501358	2106030919	Ανθούσα	pdavou@gmail.com	http://youtube.gr
6	Αθλητίατρος	Ιδιωτικό ραντεβού	teo	kintis	6957873701	2731028259	Σπάρτη	kintisteo@gmail.com	http://google.gr
6	Νευρολόγος	Ιδιωτικό ραντεβού	teo	kintis	6957873701	2731028259	Σπάρτη	kintisteo@gmail.com	http://google.gr
23	Αγγειοχειρουργός - Αγγειολόγος	Ιδιωτικό ραντεβού	Petros	Aroukatos	692581476	2105484265	Αθήνα	paroukatos@digitalup.gr	http://digitalup.gr
23	Αθλητίατρος	Ιδιωτικό ραντεβού	Petros	Aroukatos	692581476	2105484265	Αθήνα	paroukatos@digitalup.gr	http://digitalup.gr
23	Αλλεργιολόγος	Ιδιωτικό ραντεβού	Petros	Aroukatos	692581476	2105484265	Αθήνα	paroukatos@digitalup.gr	http://digitalup.gr
23	Γαστρεντερολόγος	Ιδιωτικό ραντεβού	Petros	Aroukatos	692581476	2105484265	Αθήνα	paroukatos@digitalup.gr	http://digitalup.gr



- Αναζήτηση με βάση την ειδικότητα:

Στο κέντρο της οθόνης υπάρχει φίλτρο Ειδικότητα, όπου όταν ο χρήστης το επιλέγει θα του εμφανίζεται μια λίστα όλων των ειδικοτήτων των γιατρών που είναι καταχωρημένοι στην βάση και επιλέγει όποια επιθυμεί. Η λίστα περιέχει όλες τις ειδικότητες που είναι καταχωρημένες στη βάση.



- Αναζήτηση με βάση την περιοχή:

Στο κέντρο της οθόνης υπάρχει φίλτρο Περιοχή, όπου όταν ο χρήστης το επιλέγει του εμφανίζεται μια λίστα όλων των περιοχών που είναι καταχωρημένοι στην βάση και θα μπορεί να επιλέγει όποια επιθυμεί. Η λίστα περιέχει όλες τις περιοχές που είναι καταχωρημένες στη βάση.

Αναζήτηση γιατρό

Ειδικότητα:

Περιοχή:

-
- Άγιος Στέφανος
- Ακαδημία
- Πολυτεχνείο
- Ζωγράφου
- Αθήνα
- Ανθούσα

Ασφάλιση:

- Αναζήτηση με βάση την ασφάλιση:

Στο κέντρο της οθόνης υπάρχει φίλτρο Ασφάλιση, όπου όταν ο χρήστης το επιλέγει θα του εμφανίζεται μια λίστα αλφαβητικά όλων των ασφαλίσεων που μπορεί να είναι συμβεβλημένος κάποιος γιατρός όπου μπορεί να είναι ιδιωτικά ταμεία, δημόσια ταμεία και ιδιωτικό ραντεβού. Η λίστα περιέχει όλες τις ασφαλίσεις που είναι καταχωρημένες στη βάση.

Αναζήτηση γιατρό

Ειδικότητα:

Περιοχή:

Ασφάλιση:

-
- ΕΥΔΑΠ-ΕΤΕΑΜ
- ΠΕΔΥ-ΕΟΠΥΥ
- Ιδιωτικό ραντεβού
- ΕΔΟΕΑΠ
- Στρατιωτικών-Λιμενικών
- Τραπεζικών υπαλλήλων


- Αποτελέσματα αναζήτησης:

Όταν ένας χρήστης έχει επιλέξει τα φίλτρα που επιθυμεί, θα πατάει το κουμπί αναζήτηση και θα γίνεται αναζήτηση στην ΒΔ όλων των γιατρών και θα εμφανίζονται στην οθόνη μόνο εκείνοι οι γιατροί οι οποίοι είναι σύμφωνοι με τα χαρακτηριστικά που επιλέχθηκαν κατά την αναζήτηση όσον αφορά την ειδικότητα, την περιοχή και την ασφάλιση. Η λίστα αυτή θα είναι σε πίνακα και για κάθε διαφορετικό στοιχείο του ιατρού θα εμφανίζεται και από μια γραμμή.

Browser tabs: Αναζήτηση Ιατρού, localhost / MySQL / ptixi, Επεξεργασία Στοιχείων

URL: localhost/ptixiakierg/search.php

Αρχική Σελίδα



Αποτελέσματα Αναζήτησης

Ειδικότητα	Ασφάλιση	Πόλη	Τηλέφωνο	Όνομα	Επώνυμο	Κινητό	Email	Website
Καρδιολόγος, Αιματολόγος,	Ιδιωτικό ραντεβού,	Αθήνα,	2102524715,	teo	kintis	6957873701	kintisteo@gmail.com	http://google.gr
Ψυχολόγος-Ψυχοθεραπευτής,	Ιδιωτικό ραντεβού, Στρατιωτικών-Λιμενικών,	Ανθούσα,	2106030919,	Peggy	Davou	6972501358	pdavou@gmail.com	http://youtube.gr
Αιματολόγος, Αλλεργιολόγος,	ΠΕΔΥ-ΕΟΠΥΥ, Ιδιωτικό ραντεβού, Στρατιωτικών-Λιμενικών,	Ζωγράφου,	2105487951,	stella	xaidemenaki	695251515	sxaidemenaki@digitalup.gr	http://www.sxaidemenaki.com

Κεφάλαιο 6

ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

6.1 Γενικά

Σε αυτό το κεφάλαιο θα γίνει ανάλυση του κώδικα που χρησιμοποιήθηκε στα αρχεία php. Το σύστημα μας απαρτίζεται από 8 αρχεία PHP, 1 εικόνα τύπου jpg, 1 αρχείο css και τα αρχεία της bootstrap. Σχεδόν σε κάθε αρχείο php συμπεριλαμβάνεται στην αρχή το functions.php κάνοντάς το include.

```
include_once "./functions.php";
```

Όσα αρχεία php εμφανίζονται στον χρήστη εμπεριέχουν και το header.php.

Ας δούμε αναλυτικότερα κάθε αρχείο.

6.2 Αρχείο functions.php

Το πρώτο αρχείο που δημιουργήθηκε είναι το functions.php όπου έχουμε όλες μαζί τις συναρτήσεις που φτιάξαμε με σκοπό να τις χρησιμοποιούμε σε όλα τα αρχεία που δημιουργήθηκαν ώστε να αποφεύγεται η επανάληψη επομένως δεν εμφανίζεται στον χρήστη.

Η πρώτη συνάρτηση είναι μια συνάρτηση debug οπού βοήθησε να βρίσκουμε τα σφάλματα του κώδικα και είναι η παρακάτω:

```
function debug($msg)
{
    echo "<pre>";
    var_dump($msg);
    echo "</pre>";
}
```

Η δεύτερη συνάρτηση είναι υπεύθυνη για την σύνδεση στην βάση δεδομένων. Είναι εξαιρετικά σημαντική αφού χρησιμοποιήθηκε πάρα πολλές φορές και είναι η παρακάτω:

```
function connectDb()
{
```

```

$servername = "localhost";
$username = "root";
$password = "";
$dbname = "ptixiakierg";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
$conn->set_charset("utf8");
mysqli_select_db($conn, "ptixiakierg");

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
return $conn;
}

```

6.3 Αρχείο mainpage.php

Το δεύτερο αρχείο που δημιουργήθηκε είναι το mainpage.php όπου εδώ είναι η κεντρική σελίδα που εμφανίζεται σε οποιονδήποτε χρήστη μπαίνει στην εφαρμογή.

Στην αρχή όπως και στα περισσότερα αρχεία γίνονται include 2 αρχεία και η σύνδεση με την βάση όπως παρακάτω:

```

<?php
include_once "./header.php";
include_once "./functions.php";
$conn = connectDb();
?>

```


Σε αυτό το αρχείο, αυτό που εμφανίζεται στον χρήστη είναι HTML, Bootstrap και CSS και επομένως είναι responsive. Επίσης, αυτή η σελίδα είναι δυναμική, διότι γίνεται query στην ΒΔ με χρήση της php. Π.χ. για την ειδικότητα έχουμε:

```
<select class="form-control" name="OnomaEid" id="Eidikotites">
  <option value="0">--</option>
  <?php
    $sql = "SELECT DISTINCT OnomaEid FROM eidikotita";
    $result = $conn->query($sql);
    if ($result->num_rows > 0) {
      while ($row = $result->fetch_assoc()){
        echo "<option>" . $row['OnomaEid'] . "</option>";
      }
    }
  ?>
</select>
```

Προεπιλεγμένη τιμή για την αναζήτηση είναι δύο παύλες (--) με τιμή 0 για τις περιπτώσεις όπου ο χρήστης δεν επιθυμεί να φιλτράρει με κάποιο από τα κριτήρια. Π.χ.

```
<option value="0">--</option>
```

6.4 Αρχείο register.php

Σε αυτό το αρχείο γίνεται η αποθήκευση των στοιχείων του ιατρού που ο ίδιος εισήγαγε στην φόρμα (Όνομα, Επώνυμο, Κινητό, Email, Website).

Πιο συγκεκριμένα, γίνεται \$_POST στο κάθε στοιχείο για να πάρουμε τα καταχωρημένα στοιχεία και ύστερα γίνεται το INSERT στον πίνακα iatros.

```
$Onoma = $_POST['Onoma'];
```

```
$Eponimo = $_POST['Eponimo'];
```

```
$Kinito = $_POST['Kinito'];
```

```

$Email = $_POST['Email'];
$Website = $_POST['Website'];
$Username = $_POST['Username'];
$Password = $_POST['Password'];

$conn=connectDb();

$sql = "INSERT into iatros(Onoma, Eponimo, Kinito, Email, Website) VAL-
UES('$Onoma','$Eponimo','$Kinito','$Email','$Website)"; //eisagogi stoixeion ston pinaka
iatros

```

Ύστερα, γίνεται η καταχώρηση του Username και Password στον πίνακα login μαζί με το τελευταίο id του γιατρού κατά τον ακόλουθο τρόπο:

```

if ($conn->query($sql) === TRUE) {
    $lastid = mysqli_insert_id($conn);
    mysqli_query($conn,"INSERT INTO login(Username, Password, DocId) VALUES
('$Username','$Password','$lastid)"); //eisagogi stoixeion ston pinaka login me to last Do-
cid
    header("location: mainpage.php");
}else{
    echo 'Could Not Add Information';
    //echo "Error: " . $sql . "<br>" . $conn->error;
}

```

Για την περίπτωση σφάλματος θα εκτυπωθεί ανάλογο μήνυμα.

6.5 Αρχείο login.php

Σε αυτό το αρχείο, αφού εισάγει ο χρήστης το Username και Password γίνεται λήψη αυτών των στοιχείων

```
$Username = mysqli_real_escape_string($conn, $_POST['Username']); //frontizei na ginei  
real escape sta strings gia na min mporei kapoios na kanei kapoia mysql injection epithesi  
$Password = mysqli_real_escape_string($conn, $_POST['Password']); //frontizei na ginei  
real escape sta strings gia na min mporei kapoios na kanei kapoia mysql injection epithesi
```

Ύστερα, γίνεται query στην βάση και αν ταιριάζουν τα στοιχεία με κάποιο από αυτά που είναι στον πίνακα login, τότε η μεταβλητή count θα επιστρέψει τον αριθμό των αποτελεσμάτων που ταιριάζουν. Έτσι, αν το count επιστρέψει 1 τότε θα καλεστεί η μέθοδος \$_SESSION όπου θα αποθηκευτεί το id του γιατρού και στην συνέχεια γίνεται ανακατεύθυνση στο αρχείο edit.php. Αν ,βέβαια, αυτός που συνδεθεί είναι ο διαχειριστής, θα γίνει ανακατεύθυνση στο σε άλλη σελίδα (την admin.php) όπου είναι η σελίδα του διαχειριστή. Εάν τα στοιχεία που έδωσε ο χρήστης είναι λάθος θα του εμφανιστεί ανάλογο μήνυμα.

6.6 Αρχείο edit.php

Αφού ο χρήστης κάνει login, μεταφέρεται σε αυτό το αρχείο. Εδώ καλείται να εισάγει τα υπόλοιπα στοιχεία όπως είναι: Πόλη, Τηλέφωνο, Ειδικότητες, Ασφαλίσεις. Π.χ. για την πόλη, εμφανίζονται δυναμικά όλες οι πόλεις που είναι καταχωρημένες στον πίνακα πόλεις. Άρα, έχουμε:

```
<label>Πόλη</label>  
<select name="Poli" id="Poleis">  
  <?php  
    $sql = "SELECT * FROM allcities";  
    if($result = mysqli_query($conn, $sql)){  
      if(mysqli_num_rows($result) > 0){  
        while($row = mysqli_fetch_array($result)){  
          echo "<option>" . $row['City'] . "</option>";    }  
        // Free result set  
        mysqli_free_result($result); }  
    }  
</select>
```

```
}
```

```
?>
```

```
</select>
```

Χρησιμοποιείται JQuery όπου οι ειδικότητες και οι ασφαλίσεις είναι τύπου multiselect και οι πόλεις είναι select2. Το multiselect είναι μέθοδος της JQuery όπου έχει ως αποτέλεσμα σε μία φόρμα να μπορεί ο χρήστης να επιλέξει πολλαπλές επιλογές. Στην συγκεκριμένη περίπτωση, αφορά περισσότερες από μία ειδικότητες και ασφαλίσεις. Από την άλλη μεριά, το select2 είναι μέθοδος όπου ο χρήστης μπορεί σε μία φόρμα να επιλέξει κάποιο πεδίο από μια λίστα, αλλά ταυτόχρονα μπορεί να κάνει και αναζήτηση πεδίου. Αναλυτικότερα, ο χρήστης μπορεί να αναζητήσει την πόλη που επιθυμεί.

```
<script type="text/javascript">
```

```
$(document).ready(function() {  
    $('#Eidikotites').multiselect();  
    $('#Asfaliseis').multiselect();  
    $('#Poleis').select2();  
});
```

```
</script>
```

6.7 Αρχείο submit_data.php

Στο σημείο αυτό, ο ιατρός έχοντας κάνει επιτυχώς login, καλείται να εισάγει τα στοιχεία που απαρτίζουν το προφίλ του όπου είναι η Πόλη του ιατρού, το τηλέφωνο, και οι υποστηριζόμενες ασφαλίσεις. Υλοποιώντας το έχουμε:

```
$Poli = $_POST['Poli'];
```

```
$Telefono = $_POST['Telefono'];
```

```
$DocId = $_POST['DoctorId'];
```

```
$OnomaEid = $_POST['OnomaEid'];
```

```
$OnomaAsf = $_POST['OnomaAsf'];
```

```
$conn = connectDb();
```

```
mysqli_query($conn, "INSERT diefthinsi(Poli, Telefono, DocId)  
VALUES ('" . mysqli_real_escape_string($conn, $Poli) . " ,  
" . mysqli_real_escape_string($conn, $Telefono) . " , ". $DocId . "  
");
```

```
foreach ($OnomaEid as $Eid) {
```

```
mysqli_query($conn, "INSERT eidikotita(OnomaEid, DocId)  
VALUES ('" . mysqli_real_escape_string($conn, $Eid) . " , " . $DocId . "  
");
```

```
}
```

```
foreach ($OnomaAsf as $Asf) {
```

```
mysqli_query($conn, "INSERT asfalisi(OnomaAsf, DocId)  
VALUES ('" . mysqli_real_escape_string($conn, $Asf) . " , ". $DocId . "  
");
```

```
}
```

Παράλληλα εκτυπώνεται αντίστοιχο μήνυμα επιτυχίας.

6.8 Αρχείο admin.php

Σε αυτό το αρχείο θα μεταφερθεί μόνο ο Διαχειριστής (admin). Εδώ, μπορεί και βλέπει όλους τους ιατρούς με όλα τα στοιχεία τους που είναι καταχωρημένα στην βάση. Αυτό επιτυγχάνεται χάρις στο παρακάτω query:

```
$query = "  
SELECT DISTINCT iatros.DocId, Onoma, Eponimo, Kinito, Email, Website, Telefono,  
Poli, OnomaEid, OnomaAsf  
FROM ((iatros  
INNER JOIN eidikotita ON iatros.DocId = eidikotita.DocId)  
INNER JOIN diefthinsi ON iatros.DocId = diefthinsi.DocId)
```

```
INNER JOIN asfalisi ON iatros.DocId = asfalisi.DocId)
```

```
";
```

Αν δεν βρεθούν ιατροί εκτυπώνεται ανάλογο μήνυμα, αλλιώς εκτυπώνονται όλα τα στοιχεία μαζί με τον κωδικό (DocId) του κάθε ιατρού όπως φαίνεται παρακάτω:

```
$result = mysqli_query($conn, $query);
$count = mysqli_num_rows($result);
if ($result== FALSE || $count== 0){
    echo "<h3>" . "Δυστυχώς δεν βρέθηκε κάποιος ιατρός που να ταιριάζει στην
αναζήτηση σας" . "</h3>";
}
else {
echo "<table class='table table-hover'>";
echo "<thead>";
echo "<tr>";
echo "<th>Κωδικός</th>";
echo "<th>Ειδικότητα</th>";
echo "<th>Ασφάλιση</th>";
echo "<th>Όνομα</th>";
echo "<th>Επώνυμο</th>";
echo " <th>Κινητό</th>";
echo " <th>Τηλέφωνο</th>";
echo " <th>Πόλη</th>";
echo " <th>Email</th>";
echo " <th>Website</th>";
echo "</tr>";
echo "</thead>";
while ($row = mysqli_fetch_assoc($result)) {
echo "<tr>";
echo "<td>". $row['DocId'] . "</td>";
echo "<td>". $row['OnomaEid'] . "</td>";
echo "<td>". $row['OnomaAsf'] . "</td>";
echo "<td>". $row['Onoma'] . "</td>";
echo "<td>". $row['Eponimo'] . "</td>";
echo "<td>?> <a href=?php echo \"tel:\" . $row['Kinito'] ?>\"> <?php echo $row['Kinito']
.</td>";
    echo "<td>?> <a href=?php echo \"tel:\" . $row['Tilefono'] ?>\"> <?php echo
$row['Tilefono'] . "</td>";
    echo "<td>". $row['Poli'] . "</td>";
    echo "<td>". $row['Email'] . "</td>";
    echo "<td>". $row['Website'] . "</td>";
    echo "</tr>";
}
}
```

```
echo "</table>";  
}
```

```
echo "</div>";
```

Στην συνέχεια, εμφανίζονται η προσθήκη ιατρού, ειδικοτήτων και ασφαλίσεων όπου κάθε ένα κουμπι παραπέμπει σε διαφορετικό αρχείο που θα αναλυθεί στην συνέχεια. Αυτή η διαδικασία υλοποιείται βάσει του συγκεκριμένου κώδικα:

```
<!--prosthiki iatreiou-->  
<form charset=utf-8 action="add_iatreio.php" method="post" >  
  <div class="col-sm-12">  
    <div class="row">  
      <div class="col-sm-2 form-group">  
        <label>Κωδικός</label>  
        <input id="DoctorId" name="DoctorId" type="text" class="form-control" placeholder="Τον κωδικό ιατρού" required="">  
      </div>  
  
      <div class="col-sm-5 form-group">  
        <label>Πόλη</label>  
        <select name="Poli" id="Poleis">  
          <?php  
            $sql = "SELECT * FROM allcities";  
            if($result = mysqli_query($conn, $sql)){  
              if(mysqli_num_rows($result) > 0){  
                while($row = mysqli_fetch_array($result)){  
                  echo "<option>" . $row['City'] . "</option>";  
                }  
                // Free result set  
                mysqli_free_result($result);  
              }  
            }  
          ?>  
        </select>  
      </div>  
  
      <div class="col-sm-5 form-group">  
        <label>Τηλέφωνο</label>  
        <input id="Telefono" name="Telefono" type="text" class="form-control" placeholder="Το τηλέφωνο ιατρού" required="">  
      </div>  
  
    </div>  
    <button type="submit" name="submit" class="btn btn-lg btn-info">Προσθήκη
```

```
ιατρείου</button><br><br>
</div>
</form>
```

```
<!--prosthiki eidikotiton-->
```

```
<form charset=utf-8 action="add_eidikotita.php" method="post" >
```

```
<div class="col-sm-12">
```

```
<div class="row">
```

```
<div class="col-sm-2 form-group">
```

```
<label>Κωδικός</label>
```

```
<input id="DoctorId" name="DoctorId" type="text" class="form-control" place-
holder="Τον κωδικό ιατρού" required="">
```

```
</div>
```

```
<div class="col-sm-6 form-group">
```

```
<label>Ειδικότητα</label>
```

```
<select name="OnomaEid[]" id="Eidikotites" multiple="multiple">
```

```
<?php
```

```
$sql = "SELECT * FROM allresidencies";
```

```
if($result = mysqli_query($conn, $sql)){
```

```
if(mysqli_num_rows($result) > 0){
```

```
while($row = mysqli_fetch_array($result)){
```

```
echo "<option>" . $row['Residency'] . "</option>";
```

```
}
```

```
// Free result set
```

```
mysqli_free_result($result);
```

```
}
```

```
}
```

```
?>
```

```
</select>
```

```
</div>
```

```
</div>
```

```
<button type="submit" name="submit" class="btn btn-lg btn-info">Προσθήκη
```

```
ειδικοτήτων</button><br><br>
```

```
</div>
```

```
</form>
```

```
<!--prosthiki asfaliseon-->
```

```
<form charset=utf-8 action="add_asfalisi.php" method="post" >
```

```
<div class="col-sm-12">
```

```
<div class="row">
```

```
<div class="col-sm-2 form-group">
```

```
<label>Κωδικός</label>
```

```
<input id="DoctorId" name="DoctorId" type="text" class="form-control" place-
holder="Τον κωδικό ιατρού" required="">
```



```

</div>
<div class="col-sm-6 form-group">
  <label>Ασφάλιση</label>
  <select name="OnomaAsf[]" id="Asfaliseis" multiple="multiple">
    <?php
      $sql = "SELECT * FROM alltameia";
      if($result = mysqli_query($conn, $sql)){
        if(mysqli_num_rows($result) > 0){
          while($row = mysqli_fetch_array($result)){
            echo "<option>" . $row['Tameio'] . "</option>";
          }
          // Free result set
          mysqli_free_result($result);
        }
      }
    ?>
  </select>
</div>

</div>
<button type="submit" name="submit" class="btn btn-lg btn-info">Προσθήκη
ασφαλίσεων</button><br><br>
</div>
</form>

```

6.9 Αρχείο add_iatreio.php

Όταν ο διαχειριστής εισάγει τον κωδικό, την πόλη και το τηλέφωνο του ιατρού, προστίθενται στον πίνακα diefthinsi τα συγκεκριμένα στοιχεία με την SQL εντολή INSERT όπως παρακάτω:

```

$Poli = $_POST['Poli'];
$Tilefono = $_POST['Tilefono'];
$DocId = $_POST['DoctorId'];

```

```

$conn = connectDb();

```

```

mysqli_query($conn, "INSERT diefthinsi(Poli, Tilefono, DocId)
  VALUES ('".mysqli_real_escape_string($conn, $Poli)."',
  '".mysqli_real_escape_string($conn, $Tilefono)."', '$DocId.')");

```

```
$conn->close();
```

Και εμφανίζεται αντίστοιχο μήνυμα:

```
<h2 class="text-center">Η εισαγωγή ιατρού ολοκληρώθηκε</h2>
<button type="button" class="btn btn-success center-block"><a href = "ad-
min.php">Admin</a></button>
```

6.10 Αρχείο add_eidikotita.php

Όταν ο διαχειριστής εισάγει τον κωδικό και την ειδικότητα του ιατρού, προστίθενται στον πίνακα eidikotita τα συγκεκριμένα στοιχεία με την SQL εντολή INSERT όπως παρακάτω:

```
$DocId = $_POST['DoctorId'];
$OnomaEid = $_POST['OnomaEid'];
```

```
$conn = connectDb();
```

```
if(isset($OnomaEid)){
    foreach ($OnomaEid as $Eid) {
        mysqli_query($conn, "INSERT eidikotita(OnomaEid, DocId)
            VALUES ('" . mysqli_real_escape_string($conn, $Eid) . "', " . $DocId . ")
        ");
    }
}
```

```
$conn->close();
```

Και εμφανίζεται αντίστοιχο μήνυμα:

```
<h2 class="text-center">Η εισαγωγή ειδικότητας ολοκληρώθηκε</h2>
<button type="button" class="btn btn-success center-block"><a href = "ad-
min.php">Admin</a></button>
```

6.11 Αρχείο add_asfalisi.php

Όταν ο διαχειριστής εισάγει τον κωδικό και την ασφάλιση του ιατρού, προστίθενται στον πίνακα asfalisi τα συγκεκριμένα στοιχεία με την SQL εντολή INSERT όπως παρακάτω:

```
$DocId = $_POST['DoctorId'];
$OnomaAsf = $_POST['OnomaAsf'];
```

```
$conn = connectDb();
```

```

if(isset($OnomaAsf)){
    foreach ($OnomaAsf as $Asf) {
        mysqli_query($conn,"INSERT asfalisi(OnomaAsf, DocId)
            VALUES ('".mysqli_real_escape_string($conn, $Asf)." , ".$DocId.")
            ");
    }
}

$conn->close();

```

Και εμφανίζεται αντίστοιχο μήνυμα:

```

<h2 class="text-center">Η εισαγωγή ασφάλισης ολοκληρώθηκε</h2>
<button type="button" class="btn btn-success center-block"><a href = "ad-
min.php">Admin</a></button>

```

6.12 Αρχείο delete_iatreio.php

Εδώ περιγράφεται η διαγραφή των ιατρείων από τον διαχειριστή μέσα από μια SQL εντολή:

```

$conn = connectDb();
$DocId = $_POST['DoctorId'];

mysqli_query($conn, "DELETE FROM diefthinsi WHERE DocId=$DocId
");

$conn->close();

```

Στην συνέχεια εκτυπώνεται αντίστοιχο μήνυμα:

```

<h2 class="text-center">Η διαγραφή ιατρείου ολοκληρώθηκε</h2>
<button type="button" class="btn btn-success center-block"><a href = "ad-
min.php">Admin</a></button>

```

6.13 Αρχείο delete_eidikotita.php

Εδώ περιγράφεται η διαγραφή των ειδικοτήτων από τον διαχειριστή μέσα από μια SQL εντολή:

```
$conn = connectDb();
$DocId = $_POST['DoctorId'];

mysqli_query($conn, "DELETE FROM eidikotita WHERE DocId=$DocId
");

$conn->close();
```

Στην συνέχεια εκτυπώνεται αντίστοιχο μήνυμα:

```
<h2 class="text-center">Η διαγραφή ειδικοτήτων ολοκληρώθηκε</h2>
<button type="button" class="btn btn-success center-block"><a href = "ad-
min.php">Admin</a></button>
```

6.14 Αρχείο delete_asfalisi.php

Εδώ περιγράφεται η διαγραφή των ασφαλίσεων από τον διαχειριστή μέσα από μια SQL εντολή:

```
$conn = connectDb();
$DocId = $_POST['DoctorId'];

mysqli_query($conn, "DELETE FROM asfalisi WHERE DocId=$DocId
");

$conn->close();
```

Στην συνέχεια εκτυπώνεται αντίστοιχο μήνυμα:

```
<h2 class="text-center">Η διαγραφή ασφαλίσεων ολοκληρώθηκε</h2>
<button type="button" class="btn btn-success center-block"><a href = "ad-
min.php">Admin</a></button>
```

6.15 Αρχείο session.php

Αυτό το αρχείο είναι χρήσιμο μετά την επιτυχή σύνδεση του χρήστη για να παραμείνει συνδεδεμένος.

6.16 Αρχείο logout.php

Εδώ, απλά ο χρήστης πατώντας το κουμπί logout, καλείτε αυτό το αρχείο όπου καταστρέφεται το Session του συγκεκριμένου ιατρού που έχει το συγκεκριμένο id. Στην συνέχεια μεταφέρεται στην αρχική σελίδα

```
if(session_destroy()) {  
    header("Location: mainpage.php");  
}
```

6.17 Αρχείο search.php

Αυτό το αρχείο είναι υπεύθυνο για τα αποτελέσματα της αναζήτησης που θα εμφανιστούν στον χρήστη. Πιο συγκεκριμένα, με την μέθοδο \$_POST συλλέγουμε τα δεδομένα που εισήγαγε ο χρήστης, δηλαδή την ειδικότητα, την πόλη και την ασφάλιση. Στην συνέχεια, ελέγχονται όλες οι περιπτώσεις για τα κριτήρια που φίλτραρε ο χρήστης. Εάν δεν έχει εισάγει κάποιο ή κάποια από τα 3 φίλτρα, τότε λαμβάνονται υπόψιν μόνο αυτά που εισήγαγε.

```
$OnomaEid = mysqli_real_escape_string($conn, $_POST['OnomaEid']);  
$Poli = mysqli_real_escape_string($conn, $_POST['Poli']);  
$OnomaAsf = mysqli_real_escape_string($conn, $_POST['OnomaAsf']);  
  
// initialize where clause  
$whereclause = "";  
  
if ($_POST['OnomaEid'] == '0' && $_POST['Poli'] == '0' && $_POST['OnomaAsf'] == '0'){  
    $whereclause = "";  
}  
  
elseif ($_POST['OnomaEid'] == '0' && $_POST['Poli'] == '0'){  
    $whereclause .= " OnomaAsf='$_POST['OnomaAsf]' ";  
}
```

```

}
elseif ($_POST['OnomaEid'] == '0' && $_POST['OnomaAsf'] == '0'){
$whereclause .= " Poli='$Poli' ";
}
elseif ($_POST['Poli'] == '0' && $_POST['OnomaAsf'] == '0'){
$whereclause .= " OnomaEid='$OnomaEid' ";
}
elseif($_POST['OnomaEid'] == '0') {
    $whereclause .= " Poli='$Poli' AND OnomaAsf='$OnomaAsf' ";
}
elseif ($_POST['Poli'] == '0'){
    $whereclause .= " OnomaEid='$OnomaEid' AND OnomaAsf='$OnomaAsf' ";
}
elseif ($_POST['OnomaAsf'] == '0'){
    $whereclause .= " OnomaEid='$OnomaEid' AND Poli='$Poli' ";
}
else{
    $whereclause .= " OnomaEid='$OnomaEid' AND Poli='$Poli' AND On-
omaAsf='$OnomaAsf' ";
}

```

Στην συνέχεια, φτιάχνεται το query που θα επιλέξει σε όλους τους πίνακες της βάσης, όλους τους ιατρούς που ταιριάζουν με τα κριτήρια της αναζήτησης.

```

$query = "
    SELECT DISTINCT iatros.DocId, Onoma, Eponimo, Kinito, Email, Website, Telefono,
Poli, OnomaEid, OnomaAsf
    FROM ((iatros
    INNER JOIN eidikotita ON iatros.DocId = eidikotita.DocId)
    INNER JOIN diefthinsi ON iatros.DocId = diefthinsi.DocId)

```

```
INNER JOIN asfalisi ON iatros.DocId = asfalisi.DocId)
```

```
WHERE $whereclause";
```

Τα αποτελέσματα για να είναι responsive, είναι σε πίνακα τύπου:

```
<table class='table table-hover'>
```

Εάν ο χρήστης δεν δώσει κάποιο φίλτρο στην αναζήτηση ή θα του εμφανιστεί ανάλογο μήνυμα.

```
$result = mysqli_query($conn, $query);
$count = mysqli_num_rows($result);
if ($result== FALSE || $count== 0){
echo "<h3>" . "Δυστυχώς δεν βρέθηκε κάποιος ιατρός που να ταιριάζει στην αναζήτηση
σας,<br>
  Παρακαλώ δοκιμάστε κάποια άλλη αναζήτηση" . "</h3>";
}
```

Όσο τα αποτελέσματα ταιριάζουν, θα εκτυπώνεται και από μια γραμμή με όλα τα στοιχεία του εκάστοτε ιατρού:

```
while ($hasnext = mysqli_fetch_row($result)) {

// κώδικας που εμφανίζει τις ειδικότητες
$doctorid = $hasnext[0]; // χρειάζομαι το doctorId για να πάρω όλες τις ειδικότητες που
σχετίζονται με γιατρούς
$query2 = "SELECT * FROM eidikotita WHERE eidikotita.DocId = $doctorid"; // επιλέγω
όλες τις εγγραφές από τις ειδικότητες για το συγκεκριμένο γιατρό!
$result2 = mysqli_query($conn, $query2);
echo "<tr>";
echo "<td>"; // φτιάχνω το κελί πριν από τη while για να μη μου βάλει τόσα κελιά όσες και
οι ειδικότητες...
while ($row = mysqli_fetch_assoc($result2)) {
  echo $row['OnomaEid'].", "; // βάζω ένα κόμμα μετά από κάθε ειδικότητα που γράφει.
}
echo "</td>";

$query3 = "SELECT * FROM asfalisi WHERE asfalisi.DocId = $doctorid";
$result3 = mysqli_query($conn, $query3);
echo "<td>";
while ($row = mysqli_fetch_assoc($result3)) {
  echo $row['OnomaAsf'].", "; //ektipono oles tis asfaliseis tou ekastote iatrou
```

```

}
echo "</td>";

$query4 = "SELECT * FROM diefthinsi WHERE diefthinsi.DocId = $doctorid";
$result4 = mysqli_query($conn, $query4);
echo "<td>";
while ($row = mysqli_fetch_assoc($result4)) {
    echo $row['Poli'].", "; //ektipono oles tis poleis tou ekastote iatrou
}
echo "</td>";

$query5 = "SELECT * FROM diefthinsi WHERE diefthinsi.DocId = $doctorid";
$result5 = mysqli_query($conn, $query5);

echo "<td>";
while ($row = mysqli_fetch_assoc($result5)) { ?>
    <a href="<?php echo "tel:" . $row['Tilefono'] ?>"><?php echo $row['Tilefono'].", ";
//ektipono ola ta tilefona tou ekastote iatrou
}
echo "</td>";

$query6 = "SELECT * FROM iatros WHERE iatros.DocId = $doctorid";
$result6 = mysqli_query($conn, $query6);
echo "<td>";
while ($row = mysqli_fetch_assoc($result6)) {
    echo $row['Onoma']; //ektipono to onoma tou ekastote iatrou
}
echo "</td>";

$query7 = "SELECT * FROM iatros WHERE iatros.DocId = $doctorid";
$result7 = mysqli_query($conn, $query7);
echo "<td>";
while ($row = mysqli_fetch_assoc($result7)) {
    echo $row['Eponimo']; //ektipono to eponimo tou ekastote iatrou
}
echo "</td>";

$query8 = "SELECT * FROM iatros WHERE iatros.DocId = $doctorid";
$result8 = mysqli_query($conn, $query8);
while ($row = mysqli_fetch_assoc($result8)) {
    echo "<td>?> <a href="<?php echo "tel:" . $row['Kinito'] ?>"><?php echo
$row['Kinito'] . "</td>"; //ektipono to kinito tou ekastote iatrou
}

$query9 = "SELECT * FROM iatros WHERE iatros.DocId = $doctorid";

```



```
$result9 = mysqli_query($conn, $query9);
echo "<td>";
while ($row = mysqli_fetch_assoc($result9)) {
    echo $row['Email']; //ektipono to email tou ekastote iatrou
}
echo "</td>";

$query10 = "SELECT * FROM iatros WHERE iatros.DocId = $doctorid";
$result10 = mysqli_query($conn, $query10);
echo "<td>";
while ($row = mysqli_fetch_assoc($result10)) {
    echo $row['Website']; //ektipono to website tou ekastote iatrou
}
echo "</td>";

}
```

Κεφάλαιο 7

ΣΥΜΠΕΡΑΣΜΑΤΑ

7.1 Γενικά

Το διαδίκτυο έχει μικρή ιστορία αλλά μεγάλο μέλλον, εν ολίγοις αποτελεί ένα από τα χαρακτηρίστηκα του νέου τρόπου ζωής. Είναι η εξέλιξη της επιστήμης πληροφορικής και αποτελεί ένα από τα μέσα διευκόλυνσης της καθημερινότητας του ανθρώπου εν γένει. Επιπλέον είναι ένα διαρκώς εξελισσόμενο αντικείμενο που βελτιώνεται με ταχύτατους ρυθμούς και χρησιμοποιείται για ποικίλους τρόπους, ως μέσο επικοινωνίας, αγοραπωλησίας ενημέρωσης και επιμόρφωσης. Στη συγκεκριμένη πτυχιακή εργασία προσεγγίσαμε το θέμα του διαδικτύου υπό το πρίσμα της εύρεσης ιατρών που καλύπτουν άμεσα τις ιατρικές ανάγκες μας μέσω μιας πλατφόρμας. Αυτή η πλατφόρμα αποτελεί την απόδειξη ότι το διαδίκτυο εκτός από μέσο ψυχαγωγίας και επικοινωνίας αποτελεί την αμεσότερη και γρηγορότερη λύση για θέματα πρωτεύουσας σημασίας στη ζωή του ανθρώπου όπως είναι η εύρεση του κατάλληλου θεράποντα προκειμένου να εξασφαλιστεί το μέγιστο αγαθό που είναι η υγεία.

7.2 Συμπεράσματα

Πιο συγκεκριμένα να αναφέρουμε ότι με την προσπάθεια κατασκευής βάσης δεδομένων και ιστοσελίδας εξάχθηκαν τα παρακάτω συμπεράσματα.

- Η βήμα προς βήμα κατασκευή βάσης δεδομένων καθώς και ο συχνός επανέλεγχος της οδηγεί σε σωστότερο αποτέλεσμα και μειώνει κατά πολύ την πιθανότητα σφάλματος που θα διαγνωσθεί αργότερα, με αποτέλεσμα να μην μπορεί να εντοπιστεί άμεσα το λάθος και να χρειάζεται περισσότερος χρόνος να διαπιστωθεί το σημείο που χρειάζεται διόρθωση.
- Η Παρθενογένεση δεν υπάρχει ούτε στην τέχνη πόσο μάλλον στην πληροφορική, επομένως η διαρκής επίσκεψη και μελέτη αντίστοιχων πλατφορμών δίνει τα επαρκή ερεθίσματα προκειμένου να φτιάξουμε μια δικού μας χαρακτήρα ιστοσελίδα.
- Η δημιουργία δυναμικής ιστοσελίδας δίνει την δυνατότητα στον κατασκευαστή να συνειδητοποιήσει πως κάνει μια σελίδα για τον άνθρωπο παράλληλα με τον άνθρωπο. Μια σελίδα για και από τον άνθρωπο που διαμορφώνεται και εξελίσσεται βάσει της

ζήτησης, των αναγκών και της υπάρχουσας κατάστασης.

- Ο φοιτητής στα πλαίσια της διπλωματικής του έρχεται σε άμεση επαφή με γλώσσες προγραμματισμού και ουσιαστικά εφαρμόζει τη θεωρητική του γνώση.
- Προσωπικά έμαθα να βρίσκω και να επιλύω τα τυχόν προβλήματα που προέκυπταν στην πράξη σε μια υπαρκτή σελίδα.
- Με την κατασκευή της ιστοσελίδας μια σκέψη προγραμματίστηκε και υλοποιήθηκε.
- Πέρα από τον προγραμματισμό χρειάστηκε η ενασχόληση και με το αισθητικό μέρος της σελίδας. Κάτι που να μπορούσε μελλοντικά να αποτελέσει και θέμα κάποιας άλλης πτυχιακής εργασίας.

7.3 Στόχοι της Εφαρμογής

Οι στόχοι της συγκεκριμένης εφαρμογής ήταν η επαφή του φοιτητή με την ανάπτυξη βάσεων δεδομένων και κατασκευής site για εκπαιδευτικούς σκοπούς, κάτι το οποίο επετεύχθη. Ουσιαστικά η θεωρία έγινε πράξη. Επιπλέον, πιο συγκεκριμένα, σχετικά με την πλατφόρμα ιατρών έγινε προσπάθεια να υπάρχει η δυνατότητα σύνθετης αναζήτησης από το χρήστη. Επιπρόσθετα, ο σκοπός ήταν εξ αρχής να είναι μια εύκολη εφαρμογή που να διευκολύνει το διαχειριστή, τους ιατρούς και τα μέλη στη χρήση του, άλλωστε θα ήταν παράδοξο να ήταν κάτι που θα δυσκόλευε και δε θα ήταν χρηστικό δεδομένου ότι το διαδίκτυο αυτό καθ' εαυτό αναφέρθηκε ότι έχει σκοπό την διευκόλυνση της καθημερινότητας του ατόμου.

Λαμβάνοντας υπόψιν τα προαναφερθέντα δημιουργήθηκε μια φιλική προς τον χρήστη εφαρμογή η οποία είναι εύχρηστη και ασφαλής για τους χρήστες της. Παράλληλα, εξασφαλίζεται η ποιότητα και η αυθεντικότητα των στοιχείων του ιατρού αλλά και της ύπαρξής του, εφόσον ο διαχειριστής έρχεται σε τηλεφωνική επικοινωνία με κάθε ιατρό.

7.4 Μελλοντικοί Στόχοι

Ως μελλοντικό στόχο και προσδοκία για την πορεία της εφαρμογής θα θέταμε την ανάρτησή της στο internet πέραν της δημιουργίας της για εκπαιδευτικούς σκοπούς.

Επιπλέον, είναι σκόπιμη η εγγραφή όλο και περισσότερων ιατρών πανελλαδικά καθώς και η ευρέως διαδεδομένη χρήση της από τους επισκέπτες, οι οποίοι θα μπορούσαν να την καταστήσουν σε μια υψηλής επισκεψιμότητας ιστοσελίδα. Ακόμα, η διηλεκτική βελτίωση της σελίδας και η αναβάθμιση της εμπλουτίζοντας τις λειτουργίες και τις υπηρεσίες που παρέχει είναι ένας στόχος όπως επίσης και η εύρεση χορηγών που θα τη χρηματοδοτούν. Επιπρόσθετα, όσων αφορά το οικονομικό όφελος, οι ιατροί θα μπορούν να πληρώνουν συνδρομή ώστε να γίνει κερδοφόρα η ιστοσελίδα. Ακόμα, θα ήταν χρήσιμο να διαφημιστεί και να γνωστοποιηθεί στο ευρύ κοινό μέσω των social media.

Τέλος, στόχο μπορεί να αποτελέσει να γίνει η δημοφιλέστερη εφαρμογή αναζήτησης ιατρών σε πανελλαδικό επίπεδο.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. W3schools, <https://www.w3schools.com/>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
2. PHP: Hypertext Preprocessor, <http://php.net/>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
3. Wikipedia, <https://el.wikipedia.org/wiki/%CE%A0%CF%8D%CE%BB%CE%B7:%CE%9A%CF%8D%CF%81%CE%B9%CE%B1>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
4. Σημειώσεις για τη γλώσσα HTML (HyperText Markup Language) Διδάσκων Αν. Καθηγητής Ι.Σ.Βενιέρης, http://eclass.sch.gr/modules/document/file.php/EL19109/%CE%95%CE%9C%CE%A0_HTML%20Notes.pdf, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
5. Εισαγωγή στο CSS Από: Μιχαήλια Κομβούτη Βέρου, <http://docplayer.gr/119185-Eisagogi-sto-css-apo-mihailia-komvoyti-veroy.html>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
6. Δουλεύοντας με το Bootstrap, avasil, Σεπτέμβριος 14, 2015, <https://www.toastedweb.gr/blog/150-%CE%B4%CE%BF%CF%85%CE%BB%CE%AD%CF%85%CE%BF%CE%BD%CF%84%CE%B1%CF%82-%CE%BC%CE%B5-%CF%84%CE%BF-bootstrap.html>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
7. Εισαγωγή στις Βάσεις Δεδομένων (DataBases), <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBases.html>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018
8. phpMyAdmin, <https://www.phpmyadmin.net/>, ημερομηνία τελευταίας επίσκεψης 8 Μαΐου 2018

Λοιπές Πτυχιακές

- Υλοποίηση Διαδικτυακού Τύπου Αγγελιών Με Χρήση των PHP & MySQL

ΘΕΟΔΩΡΟΠΟΥΛΟΣ ΔΗΜΗΤΡΙΟΣ Α.Μ.:1484, ΠΕΡΙΣΤΕΡΟΠΟΥΛΟΣ ΘΩΜΑΣ Α.Μ.:1473