

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΝΟΣΗΛΕΥΤΙΚΗΣ

ΔΙΑΠΑΝΕΠΙΣΤΗΜΙΑΚΟ ΔΙΑΤΜΗΜΑΤΙΚΟ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΕΙΔΙΚΕΥΣΗ: ΠΛΗΡΟΦΟΡΙΚΗ ΥΓΕΙΑΣ

**ΣΧΕΔΙΑΣΜΟΣ & ΑΝΑΠΤΥΞΗ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ
ΓΙΑ ΤΗΝ ΥΠΟΣΤΗΡΙΞΗ ΧΕΙΡΟΥΡΓΙΚΟΥ ΤΜΗΜΑΤΟΣ
ΝΟΣΟΚΟΜΕΙΟΥ**

ΜΑΡΙΑ Α. ΠΡΙΑΓΓΕΛΟΥ
ΟΙΚΟΝΟΜΟΛΟΓΟΣ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΘΗΝΑ 2018

**ΣΧΕΔΙΑΣΜΟΣ & ΑΝΑΠΤΥΞΗ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ
ΓΙΑ ΤΗΝ ΥΠΟΣΤΗΡΙΞΗ ΧΕΙΡΟΥΡΓΙΚΟΥ ΤΜΗΜΑΤΟΣ
ΝΟΣΟΚΟΜΕΙΟΥ**

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Επικ. Καθηγητής Α. Πικράκης (Επιβλέπων)

Καθηγητής Ι. Μαντάς

Αναπλ. Καθηγήτρια Α. Τσαλαγίδου

NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS
FACULTY OF NURSING

INTERUNIVERSITY POSTGRADUATE PROGRAM IN HEALTH CARE MANAGEMENT
AND HEALTH CARE INFORMATICS

DESIGN & DEVELOPMENT
OF AN INFORMATION SYSTEM FOR A SURGERY DEPARTMENT
BY MARIA PRIANGELOU

SUMMARY

The aim of this paper is to describe the development of an information system, which will be applied to the surgical department of a hospital, to increase effectiveness. Firstly, a definition of the Information System is given and the current operation of Surgical Departments is described. Also, the importance and the degree of adoption of such a system are mentioned. Furthermore, the tools that were used for the development of a Surgery Information System, are briefly outlined. Lastly, while the Information System is designed as a web application, parts of the code and the interface of its environment are included.

ΠΕΡΙΛΗΨΗ

Η παρούσα εργασία έχει ως σκοπό τη περιγραφή της ανάπτυξης ενός Πληροφοριακού Συστήματος που θα έχει εφαρμογή στο Χειρουργικό Τμήμα ενός Νοσοκομείου, ώστε να διευκολύνεται η λειτουργία του. Στο πρώτο κεφάλαιο περιγράφεται η έννοια του Πληροφοριακού Συστήματος, ο τρόπος λειτουργίας των Χειρουργικών Τμημάτων σήμερα και η σημασία εφαρμογής ενός τέτοιου συστήματος. Στο δεύτερο κεφάλαιο της εργασίας γίνεται μία σύντομη αναφορά στα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη ενός Πληροφοριακού Συστήματος Χειρουργείου ως διαδικτυακή εφαρμογή. Στο τελευταίο κεφάλαιο, περιλαμβάνονται κομμάτια κώδικα της εφαρμογής, αλλά και εικόνες από το περιβάλλον του Συστήματος.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1^ο	7
1. ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΥΓΕΙΑΣ	7
1.1. ΠΛΗΡΟΦΟΡΙΑΚΟ ΣΥΣΤΗΜΑ ΝΟΣΟΚΟΜΕΙΟΥ	7
1.2. ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΧΕΙΡΟΥΡΓΕΙΟΥ	8
1.3. ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΕΦΑΡΜΟΓΗΣ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ	9
1.4. ΕΦΑΡΜΟΓΕΣ ΧΕΙΡΟΥΡΓΕΙΩΝ	10
1.5. ΛΕΙΤΟΥΡΓΙΑ ΧΕΙΡΟΥΡΓΙΚΩΝ ΤΜΗΜΑΤΩΝ ΣΤΗΝ ΕΛΛΑΔΑ	12
1.5.1. ΝΟΣΟΚΟΜΕΙΟ «ΑΓΙΟΣ ΣΑΒΒΑΣ»	12
1.5.2. ΛΑΪΚΟ ΝΟΣΟΚΟΜΕΙΟ	14
1.5.3. ΑΧΙΛΛΟΠΟΥΛΕΙΟ ΝΟΣΟΚΟΜΕΙΟ ΒΟΛΟΥ	16
1.6. ΕΦΑΡΜΟΓΗ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΣΕ ΧΕΙΡΟΥΡΓΙΚΑ ΤΜΗΜΑΤΑ	17
ΚΕΦΑΛΑΙΟ 2^ο	19
2. ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ	19
2.1. ΧΑΜΡΡ	19
2.2. HTML	20
2.3. CSS	20
2.4. PHP	21
2.5. MYSQL	21
2.6. APACHE SERVER	22
2.7. BOOTSTRAP	23
2.8. JAVASCRIPT	24
ΚΕΦΑΛΑΙΟ 3^ο	26
ΔΙΑΔΙΚΤΥΑΚΗ ΕΦΑΡΜΟΓΗ	26
3.0. ΔΙΑΓΡΑΜΜΑ ΠΕΡΙΠΤΩΣΗΣ ΧΡΗΣΗΣ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ	26
3.1. ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	27
3.2. ΣΥΝΔΕΣΗ ΤΗΣ ΒΑΣΗΣ	32
3.3. ΣΥΝΔΕΣΗ ΣΤΗΝ ΕΦΑΡΜΟΓΗ	34
3.4. ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ index.php	36
3.5. ΑΡΧΕΙΟ ΑΣΘΕΝΩΝ	39
3.5.1. ΔΗΜΙΟΥΡΓΙΑ ΝΕΟΥ ΑΣΘΕΝΗ	41
3.5.2. ΠΡΟΒΟΛΗ ΣΤΟΙΧΕΙΩΝ ΑΣΘΕΝΗ	47

3.5.3. ΕΠΕΞΕΡΓΑΣΙΑ ΣΤΟΙΧΕΙΩΝ ΑΣΘΕΝΗ	52
3.5.4. ΔΙΑΓΡΑΦΗ ΣΤΟΙΧΕΙΩΝ ΑΣΘΕΝΗ.....	55
3.5.5. ΑΝΑΖΗΤΗΣΗ ΑΣΘΕΝΗ.....	55
3.6. ΑΡΧΕΙΟ ΙΑΤΡΩΝ	57
3.7. ΑΡΧΕΙΟ ΑΝΑΙΣΘΗΣΙΟΛΟΓΩΝ	58
3.8. ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ	60
3.9. ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ.....	62
3.10. ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ.....	64
3.11. ΥΛΙΚΑ.....	65
ΣΥΜΠΕΡΑΣΜΑΤΑ.....	66
<i>ΒΙΒΛΙΟΓΡΑΦΙΑ</i>	67

ΚΕΦΑΛΑΙΟ 1^ο

1. ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΥΓΕΙΑΣ

Τα Πληροφοριακά Συστήματα Υγείας (ΠΣΥ) μπορούν να οριστούν ως ένα σύνολο από διαδικασίες και υποσυστήματα τα οποία οργανώνονται με σκοπό την δημιουργία πληροφοριών οι οποίες βελτιώνουν τις αποφάσεις της διοίκησης σε όλα τα επίπεδα του συστήματος υγείας για την καλύτερη παροχή υπηρεσιών, αυτοματοποιώντας διαδικασίες και συναλλαγές μεταξύ των διαφορετικών εμπλεκομένων.

Σύμφωνα με τον Παγκόσμιο Οργανισμό Υγείας «Ένα ΠΣΥ παρέχει τα θεμέλια για την λήψη αποφάσεων και έχει τέσσερις κύριες λειτουργίες: την παραγωγή δεδομένων, την επεξεργασία τους, την ανάλυση και σύνθεσή τους και τέλος την επικοινωνία και χρήση τους. Ένα ΠΣΥ συλλέγει δεδομένα από τον τομέα της υγείας και άλλους συναφείς τομείς, τα αναλύει και βεβαιώνει την ποιότητά τους, την αξιοπιστία τους και την επικαιρότητά τους, ενώ τα μετατρέπει σε πληροφορίες με στόχο την λήψη αποφάσεων για την υγεία». (1)

1.1. ΠΛΗΡΟΦΟΡΙΑΚΟ ΣΥΣΤΗΜΑ ΝΟΣΟΚΟΜΕΙΟΥ

Το Πληροφοριακό Σύστημα Νοσοκομείου είναι ένα υπολογιστικό σύστημα στο οποίο άτομα και μηχανές επικοινωνούν και επεξεργάζονται δεδομένα, τα οποία το σύστημα δέχεται και στη συνέχεια τα μεταδίδει, τα αποθηκεύει, τα επεξεργάζεται και παρέχει τις σχετικές πληροφορίες που αφορούν τις ανάγκες του νοσοκομείου σε μια συγκεκριμένη στιγμή για ένα ορισμένο πρόβλημα.

Ένα τέτοιο σύγχρονο σύστημα λειτουργεί με γρήγορους υπολογιστές συνδεδεμένους μεταξύ τους σε ένα βελτιστοποιημένο δίκτυο. Αυτοί οι υπολογιστές είναι προγραμματισμένοι για να συλλέγουν, να ανακτούν και να επεξεργάζονται πληροφορίες για τη φροντίδα του ασθενή και διοικητικές πληροφορίες, ώστε να γίνεται καλύτερη διαχείριση και να λαμβάνονται αποτελεσματικότερες αποφάσεις.

Το Πληροφοριακό Σύστημα Νοσοκομείου περιλαμβάνει πολλές εφαρμογές που απευθύνονται στις ανάγκες διάφορων τμημάτων σε ένα νοσοκομείο, όπως η διαχείριση των δεδομένων σχετικά με τις κλινικές, τα οικονομικά τμήματα, τα εργαστήρια και άλλα τμήματα. Πολλά νοσοκομεία έχουν έως και 200 διαφορετικά συστήματα συνδυασμένα σε ένα ενιαίο Ολοκληρωμένο Πληροφοριακό Σύστημα. Προκειμένου να λειτουργεί αποτελεσματικά ένα τέτοιο σύστημα θα πρέπει να είναι εύκολο στη χρήση, οικονομικά εφικτή η εφαρμογή του, να μπορεί να αναπτυχθεί και να προσαρμοστεί στις ανάγκες των ασθενών και του νοσοκομειακού προσωπικού.

Επιπροσθέτως, ένα Πληροφοριακό Σύστημα Νοσοκομείου είναι πιο αποδοτικό όταν είναι διαδικτυακό και έχουν πρόσβαση σε αυτό μόνο εξουσιοδοτημένοι χρήστες. Με αυτή τη σχεδίαση καταργούνται γεωγραφικά εμπόδια και οι πάροχοι φροντίδας μπορούν να έχουν πρόσβαση στις πληροφορίες όποτε τις χρειάζονται, ενώ παράλληλα επιτρέπει το διαμοιρασμό πληροφοριών μεταξύ διαφορετικών νοσοκομείων. (2) (3)

1.2.ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΧΕΙΡΟΥΡΓΕΙΟΥ

Ένα πληροφοριακό σύστημα χειρουργείου μπορεί να περιλαμβάνει στοιχεία οικονομικού, κλινικού και διαχειριστικού συστήματος. Ορισμένες από τις λειτουργίες ενός πληροφοριακού συστήματος χειρουργείου, είναι ο προγραμματισμός των ασθενών για τις χειρουργικές επεμβάσεις, ο έλεγχος των πρωτοκόλλων των χειρουργών, των επεμβάσεων και των πρακτικών της αναισθησίας, η διανομή προγράμματος επεμβάσεων, η παρακολούθηση της εξέλιξης των επεμβάσεων στο χειρουργείο και η διαχείριση της κοστολόγησης των υλικών.

Ένα τέτοιο σύστημα χρησιμοποιεί μία βάση δεδομένων. Κάθε βάση δεδομένων περιέχει αρχεία, καθένα από τα οποία, αναπαριστά ένα σύνολο ομοιογενών στοιχείων. Σε κάθε αρχείο περιλαμβάνονται επιμέρους πεδία, στα οποία συμπληρώνονται αναλυτικά στοιχεία. Περισσότερα από 200 πεδία, μπορούν να

υπάρχουν για μια επέμβαση, συμπεριλαμβάνοντας στοιχεία για τον χώρο, το χρόνο εισαγωγής και εξόδου από το χειρουργείο και άλλα.

Μία βάση δεδομένων του χειρουργείου μπορεί να αποτελείται εκτός από τη βάση που δίνει πληροφορίες για το χειρουργείο και από άλλες επιμέρους βάσεις που περιέχουν πληροφορίες για τα αποθέματα υλικών και εξοπλισμού, τα πρωτόκολλα χειρουργών, το κόστος ή για το προσωπικό. Για να είναι πλήρης η βάση δεδομένων χειρουργείου, πρέπει να περιέχει όλα τα στοιχεία που αφορούν στον προγραμματισμό, όπως ακυρώσεις, χρόνος καθυστέρησης, συσχετίσεις με συγκεκριμένους χειρουργούς ή ειδικότητες, αλλά και κόστος προμηθειών για κάθε επέμβαση.

Το πληροφοριακό σύστημα Χειρουργείου, καθώς και η βάση δεδομένων που χρησιμοποιείται χαρακτηρίζονται από κάποια στοιχεία:

- Το πληροφοριακό σύστημα του χειρουργείου, πρέπει να καλύπτει πλήρως και αποτελεσματικά, τις πληροφοριακές ανάγκες στο χειρουργείο.
- Η βάση δεδομένων του χειρουργείου, πρέπει να χαρακτηρίζεται από αξιοπιστία και εγκυρότητα.
- Τα δεδομένα, στη βάση δεδομένων του χειρουργείου, επιδέχονται στατιστικής επεξεργασίας και γραφικής απεικόνισης, εφόσον είναι κατά κύριο λόγο ποσοτικά.
- Όλες οι επιμέρους λειτουργικές μονάδες, της βάσης δεδομένων του χειρουργείου, πρέπει να είναι εντοπισμένες, ώστε να επιτρέπουν την ολοκληρωμένη επεξεργασία των δεδομένων. (4)

1.3.ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΕΦΑΡΜΟΓΗΣ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ

Το μεγαλύτερο όφελος που προσφέρει η υιοθέτηση ενός Πληροφοριακού Συστήματος είναι η βέλτιστη διαχείριση του μεγάλου όγκου δεδομένων χωρίς να απαιτούνται τόνοι χαρτιού. Τα έντυπα αρχεία πέρα από τον αυξημένο όγκο που καταλαμβάνουν, είναι πιο εύκολο να καταστραφούν ή να μην είναι άμεσα διαθέσιμα και εύκολα προσβάσιμα όταν απαιτείται. (5)

Η εφαρμογή ενός Πληροφορικού Συστήματος Χειρουργείου προσφέρει τη

δυνατότητα στον ιατρό να έχει άμεση πρόσβαση στις πληροφορίες του ασθενή, στο ιστορικό του και ενδεχομένως σε προηγούμενες ιατρικές εξετάσεις εφ' όσον υπάρχει σύνδεση με τον Ηλεκτρονικό Φάκελο Ασθενή. Κατά αυτό τον τρόπο, ο ιατρός μπορεί να προσφέρει εύκολα φροντίδα εξετάζοντας και εισάγοντας σημειώσεις για τον ασθενή γρηγορότερα και έχοντας ένα επίπεδο ελέγχου σε σχέση με τη πληθώρα των περιστατικών που διαχειρίζεται. Επομένως, ο ιατρός έχει τη δυνατότητα να έχει μία ολική άποψη για τον ασθενή.

Επιπλέον πλεονέκτημα που μπορεί να προσφέρει εκτός από τη φροντίδα του ασθενή είναι η παρακολούθηση και η αξιολόγηση της ποσότητας των υπηρεσιών που προσφέρονται αλλά και των πόρων που είναι διαθέσιμοι. Ελαχιστοποιώντας το χρόνο ανάκτησης δεδομένων και μειώνοντας τα λάθη ή τα κενά που συναντώνται κυρίως σε χειρόγραφα, αυξάνεται η παραγωγικότητα και διασφαλίζεται η αποδοτικότερη λειτουργία του Χειρουργικού Τμήματος. (6) (7)

1.4. ΕΦΑΡΜΟΓΕΣ ΧΕΙΡΟΥΡΓΕΙΩΝ

Εξετάζοντας τη διεθνή βιβλιογραφία γίνεται αντιληπτό ότι έχουν γίνει πολλές προσπάθειες για την εφαρμογή κάποιου είδους οργάνωση των πληροφοριών σε χειρουργικά τμήματα μέσω διάφορων προγραμμάτων και λογισμικών. Παρακάτω περιγράφονται μερικά από αυτά τα εγχειρήματα:

Μία βάση δεδομένων στο Γυναικολογικό Χειρουργικό Μητρώο αναπτύχθηκε το 1990 στην κλινική Mayo. Χρησιμοποιήθηκε περισσότερο για τον εντοπισμό δεδομένων εκ των υστέρων παρά για στοιχεία ποιότητας. Τότε δεν υπήρχε ηλεκτρονικός ιατρικός φάκελος διαθέσιμος και στην πορεία των χρόνων έγιναν πολλές τροποποιήσεις στη βάση, έχοντας ως αποτέλεσμα να εμποδίζονται δύο βασικά στοιχεία: η διαλειτουργικότητα και η συστηματική συλλογή των δεδομένων. Η μελέτη τονίζει την αναγκαιότητα διεύρυνσης των δυνατοτήτων της βάσης μέσω εργαλείων της πληροφορικής, ώστε να διασφαλίζονται τα παραπάνω στοιχεία και να επιτευχθεί η χαρτογράφηση των δεδομένων σύμφωνα με τη ορολογία SNOMED-CT. (8)

Ένα ηλεκτρονικό εργαλείο που επιτρέπει την ηλεκτρονική καταχώριση των πληροφοριών για τον ασθενή κατά τη διάρκεια μίας επέμβασης και αναφέρεται στα στοιχεία της αναισθησίας, είναι το OpChart. Εφαρμόστηκε ως πιλοτική ιστοσελίδα στη κλινική Riverview. Κατά την εισαγωγή του ασθενή στο χειρουργείο, βασικές πληροφορίες όπως η ώρα εισόδου καταχωρούνται σε ένα ηλεκτρονικό έγγραφο. Επιπλέον, περιλαμβάνεται ένα διάγραμμα πραγματικού χρόνου όπου αποτυπώνονται σε πραγματικό χρόνο ζωτικά στοιχεία του ασθενή όπως πίεση, η θερμοκρασία, η επάρκεια οξυγόνου κ.α. Άλλες πληροφορίες όπως τα φάρμακα της αναισθησίας, επιπλοκές ή παρενέργειες και οι εργαστηριακές τιμές καταγράφονται αντίστοιχα. Υπάρχουν ακόμα πεδία, όπου ο χρήστης καταγράφει τυχόν έκτακτα γεγονότα ή καθυστερήσεις. Το διάγραμμα κλείνει μετά από την εισαγωγή των τελικών ζωτικών σημείων του ασθενή στην αίθουσα ανάνηψης. (9)

Μία άλλη μελέτη αναλύει τη προσπάθεια ανάπτυξης ενός πολύ- λειτουργικού υβριδικού συστήματος ηλεκτρονικού ιατρικού φακέλου (HERM) για χρήση στη χειρουργική υπηρεσία. Η ανάπτυξη του βασίστηκε στο ήδη υπάρχον ηλεκτρονικό χειρουργικό μητρώο (ESR) της Υπηρεσίας Τραυμάτων του Pietermaritzburg Metropolitan, σε σύστημα ηλεκτρονικού ιατρικού φακέλου (EMR) και στο σύστημα υποστήριξης κλινικών αποφάσεων (CDSS).

Σε πρώτο στάδιο σκοπός του συστήματος ήταν να καταγράψει τα δεδομένα σε διάφορα στάδια της φροντίδας του ασθενή:

- Στην εισαγωγή του καταγράφονται τα δημογραφικά του στοιχεία, η ώρα και η ημερομηνία προσέλευσης του, ζωτικά στοιχεία, το ιστορικό του και η κλινική εξέταση.
- Στην έξοδο του γίνεται ταξινόμηση σύμφωνα με το ICD-10 των χειρουργικών παθολογικών ευρημάτων και καταγράφονται τα ευρήματα και η κατάσταση του.
- Στην εγχείρηση καταγράφονται οι λεπτομέρειες της, η ημερομηνία και ώρα της εγχείρησης, η διάρκεια της, οι διαδικασίες της και η μέθοδος αναισθησίας.

Τα παραπάνω καταγράφηκαν αρχικά ως ένα χειρόγραφο προσχέδιο. Στο δεύτερο

στάδιο, έγινε η μετατροπή τους σε ηλεκτρονική μορφή και ο σχεδιασμός τους σε μία σχεσιακή βάση δεδομένων. Μετά από τρεις μήνες εφαρμογής του συστήματος HEMR, 1.114 εγγραφές ασθενών καταγράφηκαν στο σύστημα, το επίπεδο προσαρμοστικότητας ήταν της τάξης του 87-100% και το επίπεδο ικανοποίησης των χρηστών πολύ υψηλό. (10)

Τέλος, έχουν αναπτυχθεί προγράμματα για την οργάνωση των προγραμμάτων των χειρουργείων, διευκολύνοντας το προγραμματισμό των επεμβάσεων την επόμενη μέρα. Μελέτες που έλαβαν χώρα σε ένα Ισπανικό Πανεπιστημιακό Νοσοκομείο αναλύουν το τρόπο εφαρμογής τους και αξιολογούν την αποδοτικότητα τους, υπολογίζοντας κατά αυτό τον τρόπο το κόστος από το χρόνο που είναι προγραμματισμένη μία επέμβαση και την ενδεχόμενη καθυστέρηση που μπορεί να έχει από μία σειρά παραμέτρων. (11)

1.5. ΛΕΙΤΟΥΡΓΙΑ ΧΕΙΡΟΥΡΓΙΚΩΝ ΤΜΗΜΑΤΩΝ ΣΤΗΝ ΕΛΛΑΔΑ

Σε αυτή την ενότητα περιγράφεται ο τρόπος λειτουργίας των βασικών διαδικασιών που λαμβάνουν χώρα σε ένα χειρουργικό τμήμα. Αρχικά, αναφέρονται διαδικασίες σύμφωνα με το κανονισμό λειτουργίας των χειρουργείων στο Νοσοκομείο «Άγιος Σάββας». Εν συνεχεία, περιγράφεται η λειτουργία του χειρουργικού τμήματος του Λαϊκού Νοσοκομείου και του Αχιλλοπούλειου Νοσοκομείου Βόλου, σύμφωνα με τις πληροφορίες που λήφθηκαν προφορικά από άτομα που εργάζονται στους συγκεκριμένους χώρους.

1.5.1 ΝΟΣΟΚΟΜΕΙΟ «ΑΓΙΟΣ ΣΑΒΒΑΣ»

Πρακτικό εγχείρησης

Αναγράφεται στο βιβλίο εγχειρήσεων του χειρουργείου με ευθύνη του χειρουργού.

Περιλαμβάνει:

- Τα στοιχεία του ασθενή.
- Τα ονόματα του χειρουργού και των βοηθών.
- Το όνομα του αναισθησιολόγου και το είδος της αναισθησίας.
- Λεπτομερή καταγραφή της εγχείρησης

Το διάγραμμα αναισθησίας (πρακτικό) συμπληρώνεται και υπογράφεται από τον αναισθησιολόγο και τοποθετείται σε ειδικό αρχείο.

Πρόγραμμα εγχειρήσεων: Είναι το έντυπο στο οποίο προγραμματίζεται η εργασία. Το έντυπο αυτό συντάσσεται και αποστέλλεται στο χειρουργείο έγκαιρα την προηγούμενη μέρα για να καταχωρηθεί στο γενικό πρόγραμμα του χειρουργείου. Περιλαμβάνει συνήθως την ώρα και αίθουσα εγχείρησης, το ονοματεπώνυμο και θάλαμο του ασθενή, την προγραμματισμένη εγχείρηση, το χειρουργό, το είδος της αναισθησίας και κάθε άλλη απαραίτητη πληροφορία.

Μητρώο χειρουργείου. Είναι το απαραίτητο έντυπο βιβλίο χειρουργείου, βασικό στοιχείο της διεξαγόμενης ημερήσιας εργασίας. Συντάσσεται από μέλος της χειρουργικής ομάδας και περιέχει λεπτομέρειες για το είδος της εγχείρησης, τη χειρουργική ομάδα, το είδος της αναισθησίας κλπ. Το μητρώο μπορεί να χρησιμοποιηθεί ως βασικό στοιχείο σε περιπτώσεις που χρειάζονται ειδικές πληροφορίες για τη διεξαγωγή εγχειρήσεων, είναι όμως απόρρητο έγγραφο. Το μητρώο χρησιμοποιείται επίσης για τη στατιστική και την ανάλυση της εργασίας, όσον αφορά το χρόνο, τη χρήση των αιθουσών, τον αριθμό και το είδος των εγχειρήσεων, κ.λπ.

Στοιχεία εγχείρησης: Πρόκειται για φύλλο εγχείρησης που αναφέρεται το είδος, η ώρα, η διάρκειά της, τα άτομα που έλαβαν μέρος, την καταμέτρηση γαζών, την τοποθέτηση σωλήνων παροχέτευσης κ.λπ. Τα στοιχεία αυτά καταχωρούνται στο φάκελο του ασθενή και τηρούνται παράλληλα στο μητρώο του χειρουργείου.

Γραμματεία Χειρουργικού Τμήματος: Είναι υπεύθυνη για τη καθημερινή οργάνωση της εργασίας, τα προγράμματα εγχειρήσεων, τη διεκπεραίωση των εντύπων αίτησης και παραλαβής υλικών-εφοδίων, τη τήρηση και τον έλεγχο στοιχείων των μητρώων και τήρηση των βιβλίων υλικού, τη φροντίδα, τη παρακολούθηση προσκομιδής και αποστολής φακέλων ασθενών. (12)

1.5.2. ΛΑΪΚΟ ΝΟΣΟΚΟΜΕΙΟ

Στο Λαϊκό Νοσοκομείο, οι περισσότερες διεργασίες που αφορούν τις διαδικασίες που έπονται και προηγούνται του χειρουργείου, καταγράφονται και οργανώνονται κυρίως χειρόγραφα. Το κομμάτι που γίνεται μηχανογραφημένα αφορά κυρίως τη διαχείριση των υλικών και είναι αρμοδιότητα της γραμματείας του χειρουργείου, Συγκεκριμένα, η γραμματεία εισάγει ηλεκτρονικά σε συγκεκριμένο πρόγραμμα την εγγραφή του ασθενή που θα χειρουργηθεί σε ένα δελτίο πράξης – επέμβασης.

Σε αυτό καταγράφεται ο κωδικός του τμήματος, ημερομηνία εκτέλεσης, αριθμός μητρώου ασθενή, τμήμα και θάλαμος νοσηλείας, κωδικός νόσου-διάγνωσης, σχόλια και ένδειξη εκτέλεσης της επέμβασης.

Ακόμα, συμπληρώνεται μία βεβαίωση χρήσης υλικών με τα στοιχεία του ασθενή, το ασφαλιστικό του ταμείο, αιτιολογία χρήσης των υλικών, ιατρική πράξη, ημερομηνία τοποθέτησης τους, ονομασία υλικού και ποσότητα. Η βεβαίωση εκτυπώνεται και υπογράφεται από τον ιατρό.

Όσον αφορά το πρόγραμμα του χειρουργείου συμπληρώνονται σε excel, εκτυπώνεται και αναρτάται μία μέρα πριν τα προγραμματισμένα χειρουργεία. Περιλαμβάνει τον θάλαμο εισαγωγής του ασθενή, ονοματεπώνυμο ασθενή, πάθηση, επέμβαση, νάρκωση, κωδικό ιατρικής πράξης, ποσότητα αίματος. Το πρόγραμμα ανακοινώνεται κάθε Πέμπτη και αφορά τον εβδομαδιαίο προγραμματισμό χειρουργείων.

Μετά το πέρας της επέμβασης, έξω από το χειρουργείο υπάρχει ένα βιβλίο καταγραφής που συμπληρώνεται χειρόγραφα από τον προϊστάμενο νοσηλευτή.

Το Βιβλίο Καταγραφής Χειρουργικών Επεμβάσεων περιλαμβάνει:

- Α/Α Εγγραφής
- Ημερομηνία
- Ονοματεπώνυμο ασθενή
- Εγχείρηση
- Νάρκωση

- Χειρουργό
- Ώρα εισόδου
- Ώρα εξόδου
- Αδελφές Χειρουργείου
- Παρατηρήσεις (π.χ. ποιοι είναι οι ειδικευόμενοι)
- Αίθουσα Χειρουργείου
- Αν είναι τακτικό ή έκτακτο
- Barcode με στοιχεία ασθενή

Τέλος, ο χειρουργός ιατρός είναι υπεύθυνος για τη συμπλήρωση του πρακτικού χειρουργείου, το οποίο είναι χειρόγραφο και αρχειοθετείται στον φάκελο του ασθενή.

Αναλυτικά, το Πρακτικό Χειρουργείου περιλαμβάνει:

- Ονοματεπώνυμο
- Ημερομηνία Εισαγωγής
- Ημερομηνία Χειρουργείου
- Α.Μ Ασθενή
- Ονοματεπώνυμο Χειρουργού
- Ονοματεπώνυμο Βοηθού
- Ονοματεπώνυμο Αναισθησιολόγου
- Βοηθοί Αναισθησιολόγου

Εργαλειοδοσία:

- Ονοματεπώνυμο Νοσηλεύτρια Κίνησης

Είδος Επέμβασης:

- Περιγραφή Επέμβασης
- Ημερομηνία
- Υπογραφή Γιατρού

Μέσα στο πρακτικό του χειρουργείου περιλαμβάνεται και η καρτέλα αναισθησίας που συμπληρώνεται από τον αναισθησιολόγο. Εκεί καταγράφονται τα φάρμακα που χορηγήθηκαν, πίεση, θερμοκρασία, σφύξεις και οξυγόνο του ασθενή κατά τη διάρκεια της επέμβασης και σχόλια σχετικά με την αναισθησία. Ηλεκτρονικά καταγράφονται μόνο τα εργαστηριακά του ασθενή.

1.5.3. ΑΧΙΛΛΟΠΟΥΛΕΙΟ ΝΟΣΟΚΟΜΕΙΟ ΒΟΛΟΥ

Αντίστοιχες ήταν και οι πληροφορίες που λήφθηκαν για το χειρουργικό τμήμα του Αχιλλοπούλειου Νοσοκομείου. Το πρόγραμμα του χειρουργείου περιλαμβάνει ονοματεπώνυμο ασθενή, ηλικία, πάθηση, επέμβαση, ασφάλεια, μονάδες αίματος που αναμένεται να χρειαστούν και ονοματεπώνυμο χειρουργού.

Όπως και στο λαϊκό νοσοκομείο υπάρχει το βιβλίο καταγραφής των επεμβάσεων που ενημερώνεται χειρόγραφα. Μετά το τέλος της επέμβασης συμπληρώνεται το έντυπο: «Κάρτα Χειρουργείου».

Περιλαμβάνει:

- Ονοματεπώνυμο Ασθενή
- Όνομα Πατρός
- Τόπος Διαμονής
- Ηλικία
- Ταμείο
- Πάθηση-Νόσο
- Είδος επέμβασης
- Αν το χειρουργείο είναι σηπτικό ή άσηπτο
- Αν είναι τακτικό ή έκτακτο
- Αίθουσα Χειρουργείου
- Ώρα εισόδου και εξόδου από την αίθουσα
- Λήψη Χειρουργείου
- Χειρούργο Ιατρό
- Βοηθεί Χειρούργου
- Αναισθησιολόγο
- Νοσηλεύτρια Αναισθησιολογίας
- Εργαλειοδότρια
- Ποσότητα παρασκευάσματος που δόθηκε στο μικροβιολογικό
- Ποσότητα παρασκευάσματος που δόθηκε στο παθολογο-ανατομικό
- Διάφορα υλικά που χρησιμοποιήθηκαν, όπως σωλήνες παροχέτευσης,

- κομπρέσες, γαζόπανα, τρίγωνες, τολυπτά, σύνολο εργαλείων, ράμματα
- Υπογραφή Νοσηλεύτριας Κίνησης
- Υπογραφή Εργαλειοδότριας

Η Κάρτα Χειρουργείου αρχειοθετείται με τη Καρτέλα Αναισθησίας στο φάκελο του ασθενή. Η καρτέλα αναισθησίας περιλαμβάνει πληροφορίες σχετικά με την αναισθησία όπως τεχνική που χρησιμοποιήθηκε, αποτέλεσμα προνάρκωσης, χρόνο νάρκωσης, φάρμακα, αξιολόγηση πόνου, κατάσταση εξόδου, επιπλοκές αλλά και διάφορες τιμές όπως πίεσης και οξυγόνου.

Ο ιατρός συμπληρώνει έντυπη Βεβαίωση Ανάλωσης Υλικού σχετικά με τα υλικά που χρησιμοποιήθηκαν στον ασθενή. Καταγράφει το είδος του υλικού, τον κωδικό υλικού, τη ποσότητα, εταιρεία, τιμή ανά μονάδα και αξία. Η γραμματεία κάνει ηλεκτρονική αίτηση για χορήγηση του υλικού μέσω συγκεκριμένου μηχανογραφημένου προγράμματος προς το Γραφείο Διαχείρισης Υλικού. Στην αίτηση περιλαμβάνεται το ονοματεπώνυμο του ασθενή, ο αριθμός νοσηλείας του, ο κωδικός του υλικού, η περιγραφή του και η ποσότητα χορήγησης.

1.6. ΕΦΑΡΜΟΓΗ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΣΕ ΧΕΙΡΟΥΡΓΙΚΑ ΤΜΗΜΑΤΑ

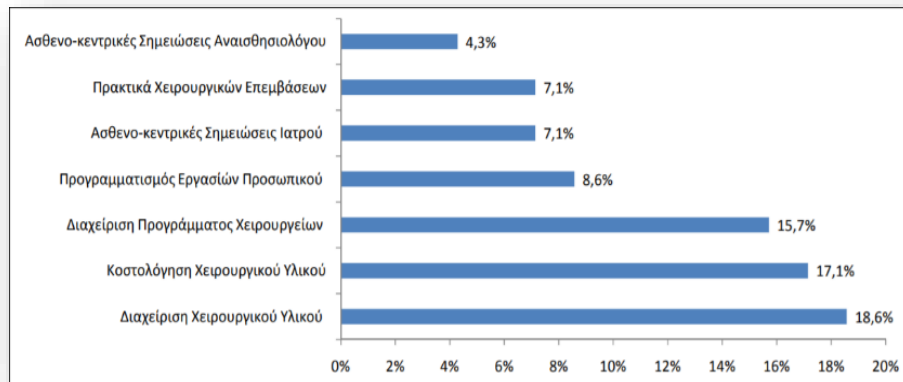
Από τις παραπάνω πληροφορίες είναι εμφανές ότι δεν έχει αναπτυχθεί ιδιαίτερα κάποιου είδους οργάνωσης σε επίπεδο Πληροφοριακού Συστήματος Χειρουργείου στις περιπτώσεις των παραπάνω νοσοκομείων. Ηλεκτρονικά καταγράφεται ένα πολύ μικρό ποσοστό δεδομένων, ενώ η κύρια διαχείριση των λειτουργιών που λαμβάνουν χώρα στο τμήμα και των αντίστοιχων δεδομένων, καταγράφεται και αρχειοθετείται σε χειρόγραφο μορφή.

Αυτή η διαπίστωση ενισχύεται παραθέτοντας τα στοιχεία μίας διδακτορικής διατριβής. (13)Σε αυτή τη μελέτη χρησιμοποιήθηκαν 7 δείκτες μέτρησης αυτοματοποιημένων δραστηριοτήτων στο χειρουργικό τμήμα:

1. Διαχείριση προγράμματος Χειρουργείων
2. Πρόγραμμα Εργασίας Προσωπικού
3. Διαχείριση Χειρουργικού Υλικού

4. Ασθενοκεντρικές Σημειώσεις Αναισθησιολόγου
5. Ασθενοκεντρικές Σημειώσεις Ιατρού
6. Κοστολόγηση Χειρουργικού Υλικού
7. Πρακτικά Χειρουργικών Επεμβάσεων

Τα αποτελέσματα έδειξαν πως από τα 70 νοσηλευτικά ιδρύματα που συμμετείχαν στην έρευνα μόνο τα 11 (15,7%) έχουν υιοθετήσει πληροφοριακό σύστημα για τη διαχείριση των χειρουργικών μονάδων και ως επί το πλείστον η χρήση τους γίνεται από το διοικητικό προσωπικό που διαχειρίζεται τα χειρουργεία και σπανιότερα από το κλινικό προσωπικό. Στον τομέα υπηρεσιών κλινικής φροντίδας όπου εντάσσονται και τα χειρουργεία, το 65% των νοσηλευτικών ιδρυμάτων πραγματοποιεί τις βασικές κλινικές διαδικασίες αποκλειστικά και μόνο υπό έντυπη μορφή.



Εικόνα 1: Γράφημα μηχανοργάνωσης διαδικασιών χειρουργείου

Όπως παρουσιάζεται και στο γράφημα, οι διεργασίες σχετικά με την οργάνωση των χειρουργείων (διαχείριση προγράμματος, διαχείριση χειρουργικού υλικού) αλλά και η κοστολόγηση χειρουργικού υλικού έχουν μηχανογραφηθεί από ποσοστό 8,6 έως 18,6% των νοσοκομείων, ενώ ιδιαίτερα χαμηλό ποσοστό (4,3-7,1%) των αμιγώς κλινικών διεργασιών (ασθενο-κεντρικές σημειώσεις ιατρού και αναισθησιολόγου) που υποστηρίζονται ηλεκτρονικά.

ΚΕΦΑΛΑΙΟ 2^ο

2. ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ

Παρακάτω περιγράφονται τα εργαλεία και οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη ενός Πληροφοριακού Συστήματος Χειρουργείου.

2.1. ΧΑΜΡΡ

Το ΧΑΜΡΡ είναι ένα ελεύθερο λογισμικό ανοικτού κώδικα και περιέχει τον εξυπηρετητή ιστοσελίδων http Apache, την βάση δεδομένων MySQL και ένα διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl. Το ΧΑΜΡΡ είναι ακρωνύμιο και αναφέρεται στα παρακάτω αρχικά:

X (αναφέρεται στο «cross-platform» που σημαίνει λογισμικό ανεξάρτητο πλατφόρμας)

Arache HTTP εξυπηρετητής

MariaDB (παλιότερα mySQL)

PHP

Perl

Τρέχει στα λειτουργικά συστήματα Microsoft Windows, Linux, Solaris, and Mac OS X και χρησιμοποιείται ως πλατφόρμα για την σχεδίαση και ανάπτυξη ιστοσελίδων με τεχνολογίες όπως PHP, JSP και Servlets.

Το ΧΑΜΡΡ μπορεί και χρησιμοποιείται χωρίς σύνδεση στο Internet για τη δοκιμή και ανάπτυξη ιστοσελίδων τοπικά στον υπολογιστή. Όταν το ΧΑΜΡΡ εγκατασταθεί τοπικά στον υπολογιστή διαχειρίζεται τον localhost ως ένα απομακρυσμένο κόμβο, ο οποίος συνδέεται με το πρωτόκολλο μεταφοράς αρχείων FTP. Επίσης, υποστηρίζει την δημιουργία και διαχείριση βάσεων δεδομένων τύπου MySQL και SQLite. (14) (15)

2.2. HTML

Η γλώσσα «HTML (HyperText Markup Language)» είναι η βασική γλώσσα για τη δημιουργία σελίδων στο Internet. Το 1980, ο φυσικός Τιμ Μπέρνερς Λι, ο οποίος εργαζόταν στο CERN, επινόησε το ENQUIRE, ένα σύστημα χρήσης και διαμοιρασμού εγγράφων για τους ερευνητές του CERN, και κατασκεύασε ένα πρωτότυπό του. Αργότερα, το 1989, πρότεινε ένα σύστημα βασισμένο στο διαδίκτυο, το οποίο θα χρησιμοποιούσε υπερκείμενο. Έτσι, έφτιαξε την προδιαγραφή της HTML και έγραψε τον browser και το λογισμικό εξυπηρετητή στα τέλη του 1990. Κύριο χαρακτηριστικό της γλώσσας είναι ο μεγάλος αριθμός από «tags (ετικέτες)». Όπως προκύπτει και το όνομα της ουσιαστικά «μαρκάρουμε» ένα αρχείο κειμένου με tags και έτσι διαμορφώνεται η δομή παρουσίασης του στη σελίδα. Επομένως, όταν ανοίγουμε ένα αρχείο HTML με έναν browser, οι ετικέτες μεταφράζονται στα χαρακτηριστικά που καθορίζουν τη λειτουργικότητα της συγκεκριμένης σελίδας. Η HTML, παρόλο που η αρχική της χρήση περιοριζόταν μόνο στη μορφοποίηση κειμένου, δίνει επιπλέον και τη δυνατότητα δημιουργίας υπερ-συνδέσμων και την εισαγωγή εικόνας και ήχου. (16) (17)

2.3.CSS

Το CSS (Cascading Style Sheets, Διαδοχικά Φύλλα Στυλ) είναι μία γλώσσα που διαμορφώνει το στυλ στις σελίδες HTML. Με το css μπορούμε να αλλάξουμε το χρώμα, τα μεγέθη, τη γραμματοσειρά σε κείμενο, πίνακες και γενικά στα στοιχεία της HTML. Μπορούμε να ορίσουμε ένα γενικότερο στυλ που να εφαρμόζεται σε περισσότερες από μία σελίδες δίνοντας στο site μας ευχρηστία, καθώς μία αλλαγή στο στυλ εφαρμόζεται μόνο μία φορά και όχι σε κάθε σελίδα ξεχωριστά. Το στυλ μπορεί να είναι εσωτερικό (internal) ή εξωτερικό (external). Το εσωτερικό στυλ εφαρμόζεται στο τμήμα head ενός αρχείου html, ενώ το εξωτερικό αποθηκεύεται σε ένα αρχείο της μορφής .css και συνδέεται με την html μέσω <link>. Ο δεύτερος τρόπος είναι προτιμότερος, καθώς ο πηγαίος κώδικας της HTML σελίδας είναι πιο εύκολα διαχειρίσιμος και μικρότερος σε μέγεθος, ενώ παράλληλα το εξωτερικό αρχείο .css αποθηκεύεται στη cache μνήμη του browser, μειώνοντας έτσι τον όγκο των δεδομένων που κατεβάζει ο επισκέπτης από τον server. (18)

2.4. PHP

Η PHP (PHP: Hypertext Preprocessor) είναι γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μέσω ενός συμβατού web server (όπως ο apache), μία σελίδα php περνά από επεξεργασία και παράγεται σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στους επισκέπτες του site σε μορφή κώδικα HTML. Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. *.php, *.php4, *.phtml κ.ά.). Ακόμη κι αν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται κώδικα PHP.

Με τις γλώσσες Perl και C/C++ στις οποίες έχει τις ρίζες της, η PHP έχει εξαιρετική ομοιότητα ως προς τον τρόπο σύνταξης, αλλά και πολλές εντολές της. Η γλώσσα δημιουργήθηκε το 1995 από τον Rasmus Lerdorf. Ο Lerdorf δημιούργησε ένα script με όνομα php.cgi χρησιμοποιώντας τη γλώσσα Perl, προκειμένου να διατηρεί μία λίστα στατικών για άτομα που έβλεπαν online το βιογραφικό του σημείωμα. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page /Form Interpreter. Σήμερα περισσότερα από 16.000.000 web sites, ποσοστό μεγαλύτερο από το 35% των ιστοσελίδων του Διαδικτύου, χρησιμοποιούν scripts γραμμένα με τη γλώσσα PHP, ενώ το υπόλοιπο 65% το μοιράζονται στατικές σελίδες HTML και όλες οι άλλες γλώσσες προγραμματισμού. (19) (20)

2.5. MYSQL

Η MySQL αποτελεί ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (relational database management system - RDBMS) που στο σύνολό της καταμετρά εκατοντάδες εκατομμύρια εγκαταστάσεις. Είναι ανοικτού κώδικα και χρησιμοποιεί την Structured Query Language (SQL), την πιο γνωστή γλώσσα για την προσθήκη, την πρόσβαση και την επεξεργασία δεδομένων σε μία Βάση Δεδομένων. Το πρόγραμμα τρέχει έναν διακομιστή και παρέχει πρόσβαση σε πολλούς χρηστές σε ένα σύνολο βάσεων δεδομένων. Ανήκει και χρηματοδοτείται από τη σουηδική εταιρία MySQL AB, η οποία στις ημέρες μας ανήκει στην Oracle.

Ο πηγαίος κώδικας της MySQL είναι διαθέσιμος μέσω της GNU άδειας καθώς και μέσω άλλου είδους συμφωνιών. Παρέχει αξιοπιστία, ταχύτητα της και ευκολία στη χρήση. Επομένως, αποτελεί τη πιο δημοφιλή βάση δεδομένων για ιστοσελίδες, ενώ χρησιμοποιείται και σε γνωστές υπηρεσίες διαδικτύου όπως το Facebook, το Google, τη Wikipedia, το YouTube και το Twitter. (21)

2.6. APACHE SERVER

Ο Apache Server είναι το πιο διαδεδομένο λογισμικό web server στον κόσμο. Η ανάπτυξη του Apache ξεκίνησε το 1995 από τον Robert McCool. Ο Apache έπαιξε καθοριστικό ρόλο στην αρχική ανάπτυξη του διαδικτύου και έγινε ο κυρίαρχος και πιο δημοφιλής διακομιστής HTTP από το 1996. Το 2009, έγινε το πρώτο λογισμικό web server που εξυπηρετεί περισσότερες από 100 εκατομμύρια ιστοσελίδες.

Ο Apache αναπτύσσεται και συντηρείται από μία ανοιχτή κοινότητα προγραμματιστών υπό την στέγη του Ιδρύματος Λογισμικού Apache. Το λογισμικό είναι διαθέσιμο για όλη την γκάμα των γνωστών λειτουργικών συστημάτων. Από τον Ιούλιο του 2016, ο Apache υπολογίστηκε ότι εξυπηρετεί πάνω από το 50% των ενεργών δικτυακών τόπων και πάνω από 40% των κορυφαίων ιστοσελίδων.

Υποστηρίζει μία ποικιλία χαρακτηριστικών, πολλά υλοποιημένα ως μεταγλωττισμένα modules τα οποία επεκτείνουν τη λειτουργικότητα του πυρήνα. Αυτά μπορεί να κυμαίνονται από μία υποστήριξη server-side γλώσσας προγραμματισμούς έως συστήματα ελέγχου ταυτότητας. Τέτοιες γλώσσες που υποστηρίζονται είναι οι Python, Perl και Tcl. Δημοφιλή modules ταυτοποίησης είναι τα mod_access, mod_auth, mod_digest και mod_auth_digest. Ο Apache διαθέτει δυνατότητα ρύθμισης μηνυμάτων λάθους, ελέγχου ταυτότητας βάσεων δεδομένων και διαπραγμάτευση περιεχομένου. Υποστηρίζεται επίσης από αρκετές γραφικές διεπαφές χρήστη ή αλλιώς GUIs. Υποστηρίζει ακόμα έλεγχο ταυτότητας μέσω κωδικού και αυθεντικοποίηση μέσω ψηφιακού πιστοποιητικού. Επειδή ο πηγαίος κώδικας είναι ανοικτός, ο καθένας μπορεί να προσαρμόσει τον διακομιστή για τις δικές του ανάγκες ενώ υπάρχει και μια μεγάλη δημόσια βιβλιοθήκη με Apache add-ons (πρόσθετα). (22) (23)

2.7. BOOTSTRAP

Το bootstrap είναι ένα ελεύθερο λογισμικό που χρησιμοποιείται για πιο γρήγορο και αποδοτικό σχεδιασμό ιστοσελίδων και διαδικτυακών εφαρμογών. Αναπτύχθηκε από τον Mark Otto και τον Jacob Thornton στο Twitter και κυκλοφόρησε τον Αύγουστο του 2011.

Χρησιμοποιεί HTML, CSS και Javascript. Τα πλεονεκτήματα χρήσης του είναι η προσαρμοστικότητα που δίνει στις σελίδες, η υποστήριξη από όλους τους browsers και η ευκολία που δίνει σε κάποιον που κατέχει βασικές γνώσεις css και html να μορφοποιήσει τη σελίδα του. Από την έκδοση 2.0 υποστηρίζει επίσης ανταποκρίσιμο σχεδιασμό (responsive design), δηλαδή η διάταξη των ιστοσελίδων προσαρμόζεται δυναμικά, λαμβάνοντας υπόψη τα χαρακτηριστικά της συσκευής που χρησιμοποιείται (PC, tablet, κινητό τηλέφωνο).

Ένα πακέτο Bootstrap περιλαμβάνει:

- CSS: Περιέχει ένα ενισχυμένο σύστημα πλέγματος και σύνολο στυλ με βασικούς ορισμούς στυλ για HTML προκειμένου να υπάρχει ομοιομορφία σε πίνακες και κείμενο.
- Επαναχρησιμοποιούμενα στοιχεία: Περιλαμβάνει στοιχεία για dropdown menu, πλοήγηση, προειδοποιήσεις κ.α.
- JavaScript Στοιχεία: Περιέχει JavaScript σε μια μορφή jQuery plugin για δημιουργία παράθυρων διαλόγου, επεξηγήσεων και καρουσέλ.
- Προσαρμοστικότητα: Ο χρήστης μπορεί να προσαρμόσει όλα τα παραπάνω σύμφωνα με τις ανάγκες του.

Υπάρχουν και άλλα προγράμματα που παρέχουν αντίστοιχες δυνατότητες, όμως το Bootstrap παραμένει το πιο δημοφιλές πρόγραμμα στο Github από το 2012.

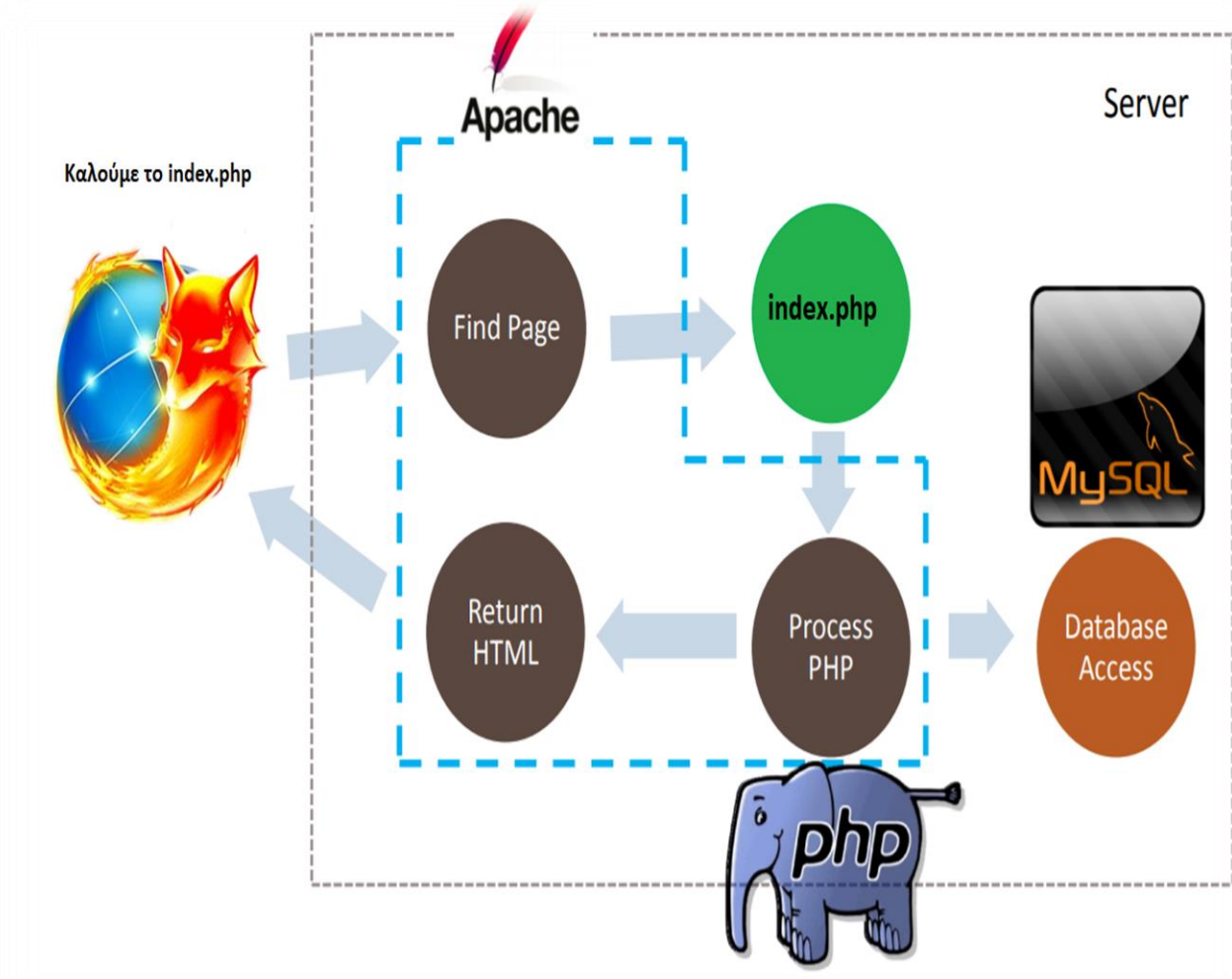
(24) (25)

2.8. JAVASCRIPT

Η JavaScript είναι μία δυναμική αντικειμενοστρεφής γλώσσα προγραμματισμού που χρησιμοποιεί συναρτήσεις πρώτης κατηγορίας και μαζί με την HTML και το CSS αποτελεί τη βασική τεχνολογία για την παραγωγή διαδικτυακού περιεχομένου. Παρόλο που αντιγράφει ονόματα από τη Java δε συσχετίζεται καθόλου μαζί της. Αντίθετα, είναι επηρεασμένη από τη γλώσσα C και οι βασικές της αρχές προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme.

Από την πλευρά του πελάτη, η JavaScript έχει παραδοσιακά εφαρμοστεί ως μια διερμηνευμένη γλώσσα, καθώς αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται. Η πλειοψηφία των δικτυακών τόπων την ενσωματώνουν και όλα τα σύγχρονα προγράμματα περιήγησης στο διαδίκτυο την υποστηρίζουν χωρίς πρόσθετα. Έχει ένα API(Application Programming Interfaces)για εργασία με κείμενο, πίνακες, ημερομηνίες και κανονικές εκφράσεις.

Η JavaScript χρησιμοποιείται και σε περιβάλλοντα που δεν βασίζονται στο διαδίκτυο, όπως PDF έγγραφα καθώς και desktop widgets. Εφαρμόζεται ακόμα στην ανάπτυξη παιχνιδιών, στη δημιουργία desktop και mobile εφαρμογών, καθώς και στον προγραμματισμό server-side δικτύου με run-time περιβάλλοντα όπως το Node.js. (26) (27)



Εικόνα 2: Λειτουργία Συστήματος

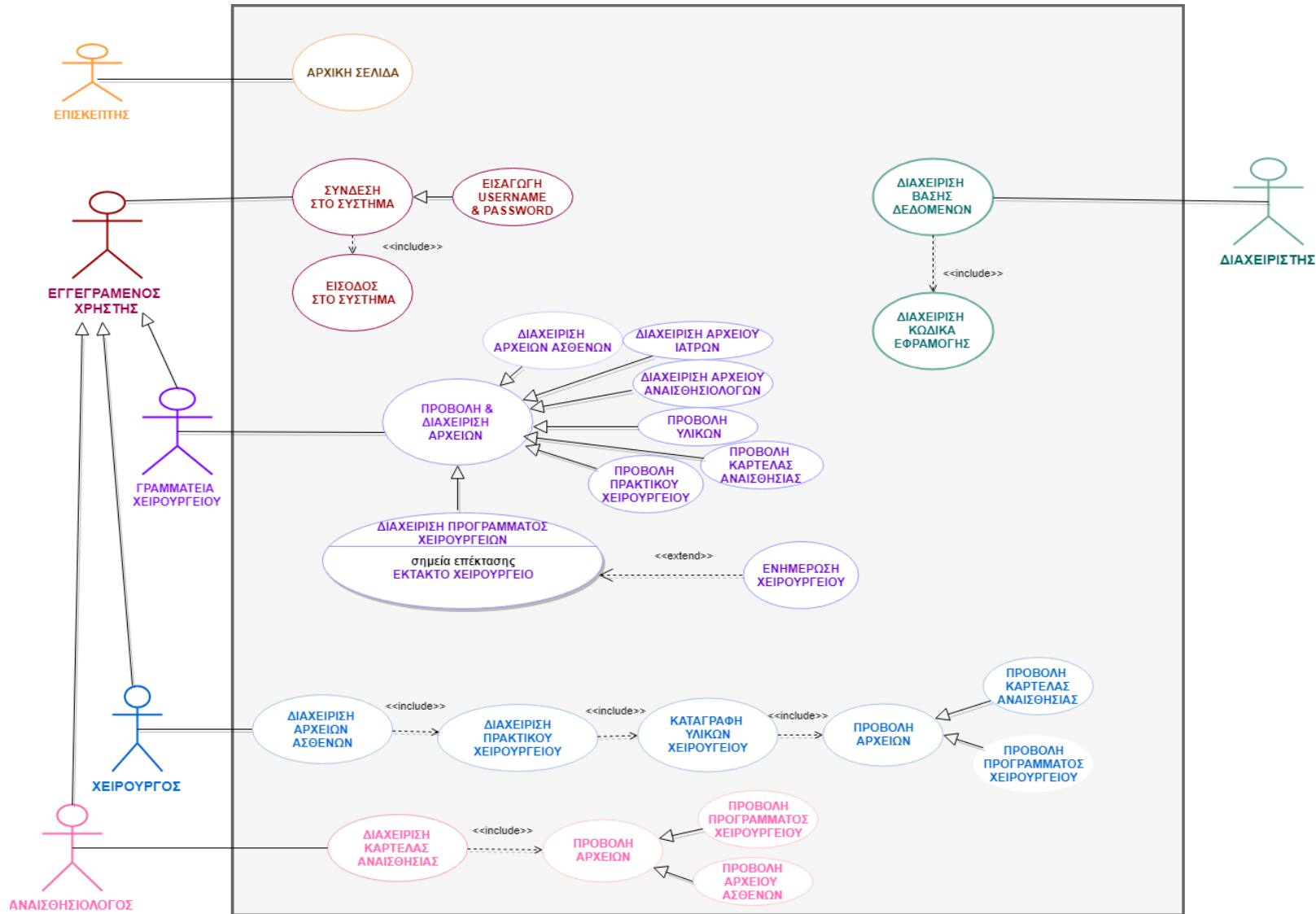
Στην εικόνα φαίνονται πως λειτουργούν οι παραπάνω τεχνολογίες κατά την εκτέλεση της εφαρμογής μας.

ΚΕΦΑΛΑΙΟ 3^ο

ΔΙΑΔΙΚΤΥΑΚΗ ΕΦΑΡΜΟΓΗ

Σε αυτό το κεφάλαιο περιγράφεται η ανάπτυξη ενός Πληροφοριακού Συστήματος Χειρουργείου που θα δίνει τη δυνατότητα στον χρήστη να έχει από οπουδήποτε πρόσβαση μέσω διαδικτύου. Η ανάπτυξη του συστήματος έγινε λαμβάνοντας υπόψιν τις πληροφορίες για το τρόπο λειτουργίας των χειρουργικών τμημάτων και των βασικών αναγκών που πρέπει να καλυφθούν.

3.0. ΔΙΑΓΡΑΜΜΑ ΠΕΡΙΠΤΩΣΗΣ ΧΡΗΣΗΣ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

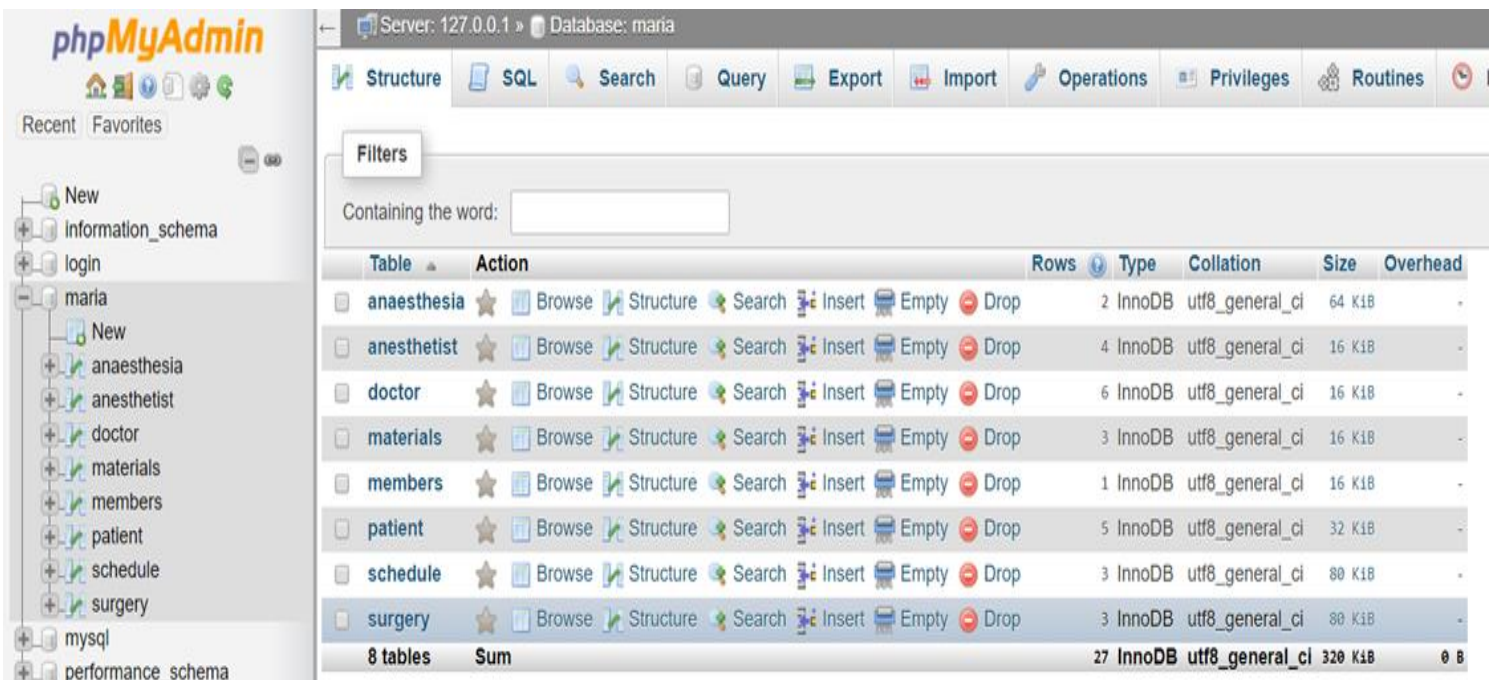


Εικόνα 3: Διάγραμμα Περίπτωσης-Χρήσης(use-case)

Το παραπάνω διάγραμμα περίπτωσης χρήσης (use case) περιγράφει τις τυπικές αλληλεπιδράσεις μεταξύ των χρηστών (χειριστών) και του συστήματος καταδεικνύοντας το τρόπο χρήσης του. Όπως αποτυπώνεται υπάρχουν διαφορετικοί χρήστες στο σύστημα μας και μπορούμε να δώσουμε στον καθένα ένα διαφορετικό ρόλο και διαφορετικές λειτουργίες ανάλογα με τη θέση του στο χειρουργικό τμήμα. Όπως προκύπτει μπορούμε να σχεδιάσουμε το σύστημα μας, ώστε να μην έχουν όλη πρόσβαση σε όλα τα αρχεία ή να έχουν τη δυνατότητα για προβολή ενός αρχείου, αλλά όχι για επεξεργασία ή διαγραφή.

3.1. ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

Το XAMPP εφόσον περιέχει τη MySQL μας δίνει τη δυνατότητα να δημιουργήσουμε μια νέα βάση δεδομένων για να τη χρησιμοποιήσουμε στη σελίδα μας. Το phpmyadmin μας επιτρέπει να διαχειριστούμε έναν διακομιστή MySQL και να δημιουργήσουμε μία βάση δεδομένων μέσω ενός υπερχρήστη. (28) Για τη συγκεκριμένη εφαρμογή δημιουργήθηκε ο superuser maria και μία βάση με το ίδιο όνομα. Η βάση περιέχει τους ακόλουθους πίνακες:




The screenshot shows the phpMyAdmin interface for a MySQL database named 'maria'. The left sidebar shows the database structure with 'maria' expanded to show 8 tables: anaesthesia, anesthetist, doctor, materials, members, patient, schedule, and surgery. The main area displays a table listing these tables with their respective actions and statistics.

Table	Action	Rows	Type	Collation	Size	Overhead
anaesthesia	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8_general_ci	64 K1B	-
anesthetist	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8_general_ci	16 K1B	-
doctor	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8_general_ci	16 K1B	-
materials	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_general_ci	16 K1B	-
members	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8_general_ci	16 K1B	-
patient	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_general_ci	32 K1B	-
schedule	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_general_ci	80 K1B	-
surgery	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_general_ci	80 K1B	-
8 tables	Sum	27	InnoDB	utf8_general_ci	320 K1B	0 B

Εικόνα 4: Η βάση στο phpMyAdmin

Ο πίνακας **patient** αναφέρεται στους ασθενείς που θα χειρουργηθούν, ενώ ο πίνακας **doctor** στους χειρουργούς ιατρούς. Ο πίνακας **anaesthesia** περιλαμβάνει πληροφορίες σχετικά με την αναισθησία που χορηγήθηκε στον ασθενή, ενώ ο πίνακας **anesthetist** αναφέρεται στους αναισθησιολόγους. Ο πίνακας **surgery** αντιστοιχεί στο πρακτικό χειρουργείου, ο πίνακας **schedule** στο πρόγραμμα και ο πίνακας **materials** στα υλικά που χρησιμοποιήθηκαν. Τέλος, ο πίνακας **members** αναφέρεται στους χρήστες που μπορούν να έχουν πρόσβαση στο σύστημα.


Πίνακας 1: Πίνακας Ασθενή

Patient	
id 	int(8)
name	varchar(20)
surname	varchar(200)
ΑΜΚΑ	char(10)
date_of_birth	date
address	varchar(200)
phone_number	char(10)
insurance	varchar(200)
type_of_disease	varchar(500)
medical_history	text
allergies	varchar(500)


Στον παραπάνω πίνακα ως πρωτεύον κλεδί (primary) ορίζεται το id που είναι τύπου ακέραιος(integer) και μεγέθους 8 χαρακτήρων. Το πρωτεύον κλειδί είναι AUTO_INCREMENT, δηλαδή σε κάθε εγγραφή παίρνει από μόνο του την αμέσως επόμενη παραδεκτή τιμή. Σαν μοναδικό κλειδί (unique) που χαρακτηρίζει αποκλειστικά μόνον έναν ασθενή ορίσαμε τον αριθμό ΑΜΚΑ.

Ο ΑΜΚΑ είναι τύπου char σταθερού μήκους 10 χαρακτήρων. Το πεδίο date_of_birth είναι τύπου date, το medical_history τύπου text, ενώ τα υπόλοιπα πεδία είναι τύπου varchar.

Πίνακας 2: Πίνακας Ιατρού


Doctor	
id 	int(15)
name	varchar(20)
surname	varchar(200)
specialization	varchar(500)
address of job	varchar(500)
e-mail	varchar(100)
phone_number	char(10)

Πίνακας 3: Πίνακας Αναισθησιολόγου


Anaesthetist	
id 	int(15)
name	varchar(20)
surname	varchar(200)
address of job	varchar(500)
e-mail	varchar(100)
phone_number	char(10)

Στους παραπάνω πίνακες περιλαμβάνονται τα πεδία για τον ιατρό και τον αναισθησιολόγο. Οι δύο πίνακες έχουν αντίστοιχα πεδία εκτός από το επιπλέον πεδίο της ειδικότητας στον πίνακα «doctor».

Πίνακας 4: Πίνακας Πρακτικού Εγχείρησης

Surgery	
id_surgery 	int(8)
<i>patient_id</i>	int(8)
<i>doctor_id</i>	int(8)
date	date
type	varchar(200)
entry_time	datetime
start_time	datetime
exit_time	datetime
operating_room	int(3)
helpers	varchar(200)
anaesthetist_id	int(15)
helpers_anaestheologist	varchar(200)
ergaleiodosia	varchar(100)
anaesthesia	text
id_anaesthesia	int(8)
description	longtext
results	longtext


Πίνακας 5: Πίνακας Αναισθησίας

Anaesthesia	
id 	int(8)
<i>patient_id</i>	int(8)
<i>anaesthetist_id</i>	int(15)
<i>surgery_id</i>	int(8)
time	time(4)
duration	time(6)
technique	varchar(500)
drugs	varchar(500)
indicator	text
exit	varchar(500)
<i>pain</i>	int(1)
results	longtext


Στον πίνακα surgery πρωτεύον κλειδί είναι το id_surgery, ενώ ξένα κλειδιά είναι το patient_id, το doctor_id, το anaesthetist_id και το id_anaesthesia που είναι τα πρωτεύοντα κλειδιά στους πίνακες patient, doctor, anaesthetist_id και id_anaesthesia αντίστοιχα. Τα πεδία entry_time, start_time και exit_time είναι της μορφής time (ΩΩ:ΛΛ:ΔΔ). Τα πεδία description και results είναι τύπου longtext, δηλαδή πεδίο κειμένου με μέγιστο μήκος 4294967295 χαρακτήρες.

Το πρωτεύον κλειδί στον πίνακα anaesthesia είναι το id, ενώ ξένα κλειδιά είναι το patient_id, το anaesthetist_id και το surgery_id που αναφέρονται στους αντίστοιχους πίνακες.

Πίνακας 6: Πίνακας Προγράμματος

Schedule	
id_schedule 	int(5)
date	datetime
operating_department	varchar(60)
operating_room	varchar(10)
<i>patient_id</i>	int(8)
<i>doctor_id</i>	int(15)
<i>anaesthetist_id</i>	time(6)
age	int(3)
diagnosis	varchar(500)
surgery	varchar(500)
blood	varchar(10)
room	int(4)
indicator	varchar(100)
type	varchar(100)
create_file	timestamp
update_file	timestamp

Πίνακας 7: Πίνακας Υλικών

Materials	
id 	int(8)
<i>surgery_id</i>	int(8)
type	varchar(100)
pieces	int(4)
company	varchar(100)
price	varchar(500)

Στον πίνακα schedule το πρωτεύον κλειδί είναι το id_schedule με ξένα κλειδιά patient_id, doctor_id, anaesthetist_id, ενώ στον πίνακα materials ξένο κλειδί είναι το surgery_id. Αντίστοιχα, πρωτεύον κλειδί στο πίνακα members είναι το memberID.

Πίνακας 8: Πίνακας Μελών

Members	
memberID 	int(11)
username	varchar(255)
password	varchar(255)
email	varchar(255)
active	varchar(255)
resetToken	varchar(255)
resetComplete	varchar(255)

3.2. ΣΥΝΔΕΣΗ ΤΗΣ ΒΑΣΗΣ

Το επόμενο βήμα είναι να συνδέσουμε τη βάση δεδομένων μας με το site μας, ώστε να μπορούμε να αντλούμε και να αποθηκεύουμε τα δεδομένα σε αυτή. Δημιουργούμε ένα αρχείο με ονομασία database.php με τον παρακάτω κώδικα:

1.Κώδικας σύνδεσης με βάση

```
<?php
class Database
{
    private static $dbName = 'maria' ;
    private static $dbHost = 'localhost' ;
    private static $dbUsername = 'maria';
    private static $dbUserPassword = 'mariapriaggelou';

    private static $cont = null;

    public function __construct() {
        die('Init function is not allowed');
    }

    public static function connect()
    {
        if ( null == self::$cont )
        {
            try
            {
                self::$cont = new
                PDO( "mysql:host=".self::$dbHost.";dbname=".self::$dbName,
                self::$dbUsername, self::$dbUserPassword,
                array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8"));
            }
            catch(PDOException $e)
            {
                die($e->getMessage());
            }
        }
        return self::$cont;
    }

    public static function disconnect()
    {
        self::$cont = null;} } ?>
```


Χρησιμοποιούμε το PDO (Php Data Objects), καθώς μπορεί να μας προσφέρει μία ομοιόμορφη μέθοδο πρόσβασης σε πολλαπλές βάσεις. (29) Για να πετύχει η σύνδεση με τη βάση βάζουμε όπου:

- **\$dbName**: Το όνομα της βάσης που χρησιμοποιούμε για να αποθηκεύσουμε τους πίνακες μας.
- **\$dbHost**: Localhost
- **\$dbUsername**: Όνομα χρήστη βάσης δεδομένων.
- **\$dbUserPassword**: Κωδικός χρήστη

Οι τρεις βασικές συναρτήσεις που χρησιμοποιούνται είναι:

1. **__construct()**: Είναι ο κατασκευαστής της κλάσης Database. Αφού είναι στατική κλάση δεν απαγορεύεται η αρχικοποίηση, αλλά για να αποφύγουμε τη κακή χρήση της προσθέτουμε και τη συνάρτηση “die”.
2. **connect**: Είναι η βασική συνάρτηση της κλάσης και δημιουργεί μία σύνδεση. Χρησιμοποιεί «singleton pattern» δηλαδή εξασφαλίζει ότι υπάρχει μόνο μία PDO σύνδεση σε όλη την εφαρμογή. (30)
3. **disconnect**: Αποσύνδεση από τη βάση, θέτοντας τη σύνδεση ως null. (31)

3.3. ΣΥΝΔΕΣΗ ΣΤΗΝ ΕΦΑΡΜΟΓΗ

Στη συνέχεια δημιουργούμε την αρχική σελίδα που θα εμφανίζεται στο χρήστη μόλις εισέρχεται στο σύστημα. Προκειμένου να έχει τη δυνατότητα να συνεχίσει θα πρέπει να συνδεθεί. Η σύνδεση μπορεί να γίνει μόνο από πιστοποιημένους χρήστες, οι οποίοι έχουν εξασφαλίσει τους κωδικούς τους από το αρμόδιο τμήμα.

ΠΛΗΡΟΦΟΡΙΑΚΟ ΣΥΣΤΗΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ



Εισάγετε όνομα & κωδικό χρήστη

Αν ξεχάσατε ή θέλετε να αλλάξετε τον κωδικό πρόσβασης
πατήστε [εδώ](#)

Εικόνα 5: Είσοδος στο σύστημα

Μετά την επιτυχή είσοδο του χρήστη στο σύστημα εμφανίζεται το μήνυμα «Επιτυχής Σύνδεση». Ο χρήστης μπορεί να συνεχίσει στη κεντρική σελίδα.



Εικόνα 5.1: Σύνδεση στο Σύστημα

3.4. ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ index.php

Στη συνέχεια δημιουργούμε το αρχείο index.php που αποτελεί τη κεντρική σελίδα της εφαρμογής μας.

2.Κώδικας της Κεντρικής Σελίδας

```
<!DOCTYPE html>
<html >

<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="id" content="">
  <meta name="author" content="">

  <name>Οργάνωση Χειρουργείου</name>

  <!-- Bootstrap Core CSS -->
  <link href="vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">

  <!-- Theme CSS -->
  <link href="css/freelancer.css" rel="stylesheet">

  <!-- Custom Fonts -->
  <link href="vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css">
  <link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet" type="text/css">
  <link href="https://fonts.googleapis.com/css?family=Lato:400,700,400italic,700italic" rel="stylesheet"
type="text/css">
</head>
<body id="page-top" class="index" >
  <!-- Navigation -->
  <nav id="mainNav" class="navbar navbar-default navbar-fixed-top navbar-custom">
    <div class="container">
      <div class="navbar-header page-scroll">
<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">
  <span class="sr-only">Toggle navigation</span> Menu <i class="fa fa-bars"></i>
</button>
  <a class="navbar-brand" href="#page-top">Οργάνωση Χειρουργείου</a>
  </div>

  <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
    <ul class="nav navbar-nav navbar-right">
      <li class="hidden">
        <a href="#page-top"></a></li>
```

```

<li class="page-scroll">
<a href="#patients"><span class="glyphicon glyphicon-user"></span>&nbsp;&nbsp;&nbsp;<b>ΑΣΘΕΝΕΙΣ</b></a>
</li>
<li class="page-scroll">
<a href="#doctors"><span i class="fa fa-stethoscope" style="font-size:18px"></i></span>&nbsp;&nbsp;&nbsp;<b>ΙΑΤΡΟΙ</b></a>
</li>
<li class="page-scroll">
<a href="#anesthetist"><span<i class="fa fa-user-md" </i></span>&nbsp;&nbsp;&nbsp;<b>ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΙ</b></a>
</li>
<li class="page-scroll">
<a href="#schedules"> <span class="glyphicon glyphicon-file"></span>&nbsp;&nbsp;&nbsp;<b>ΠΡΟΓΡΑΜΜΑ
<b>ΧΕΙΡΟΥΡΓΕΙΟΥ</b></a>
</li>
<li class="page-scroll">
<a href="#practical"><span class="glyphicon glyphicon-list-alt"></span>&nbsp;&nbsp;&nbsp;<b>ΠΡΑΚΤΙΚΟ
<b>ΧΕΙΡΟΥΡΓΕΙΟΥ</b></a>
</li>
<li class="page-scroll">
<a href="#anaesthesia"><span class="glyphicon glyphicon-list-alt"></span>&nbsp;&nbsp;&nbsp;<b>ΚΑΡΤΕΛΑ
<b>ΑΝΑΙΣΘΗΣΙΑΣ</b></a>
</li>
<li class="page-scroll">
<a href="#materials"><span class="glyphicon glyphicon-list-alt"></span>&nbsp;&nbsp;&nbsp;<b>ΥΛΙΚΑ</b></a>
</li>
</ul>
</div>
</div>
</nav>
<header>
<div class="container">
<div class="row">
<div class="col-lg-12">

<div class="intro-text">
<span class="name"><i> <b>Surgery Information System </b></i></span>
<hr class="star-light">
<span class="skills">ΔΙΑΧΕΙΡΙΣΗ ΔΕΔΟΜΕΝΩΝ ΓΙΑ ΧΕΙΡΟΥΡΓΙΚΟ ΤΜΗΜΑ ΝΟΣΟΚΟΜΕΙΟΥ</span>
</div>
</div>
</div>
</div>
</header>

```


Στον παραπάνω κώδικα κάνουμε τη σύνδεση του αρχείου μας με το CSS του Bootstrap και με τα αρχεία. Έπειτα, εισάγουμε τον κώδικα για να δημιουργήσουμε ένα μενού για πλοήγηση στη σελίδα μας, ενώ στο *header* προσθέσαμε και μία φωτογραφία.

Το Κεντρικό Μενού αποτελεί το βασικό μενού πλοήγησης και βρίσκεται στην κορυφή της ιστοσελίδας. Η υλοποίηση μας είναι μια *single page* ιστοσελίδα, πράγμα που σημαίνει ότι κάθε σύνδεσμος (στοιχεία) στο Κεντρικό Μενού οδηγεί σε έναν τομέα (*section*) της αρχικής σελίδας. Το αποτέλεσμα του παραπάνω κώδικα φαίνεται στην παρακάτω εικόνα.

ΟΡΓΑΝΩΣΗ ΧΕΙΡΟΥΡΓΕΙΟΥ

👤 ΑΣΘΕΝΕΙΣ 🩺 ΙΑΤΡΟΙ 🧪 ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΙ 📅 ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ 📄 ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ
📄 ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ 📄 ΥΛΙΚΑ



SURGERY INFORMATION SYSTEM

ΔΙΑΧΕΙΡΙΣΗ ΔΕΔΟΜΕΝΩΝ ΓΙΑ ΧΕΙΡΟΥΡΓΙΚΟ ΤΜΗΜΑ ΝΟΣΟΚΟΜΕΙΟΥ

Εικόνα 6: Αρχική Σελίδα Εφαρμογής

3.5. ΑΡΧΕΙΟ ΑΣΘΕΝΩΝ

3.Κώδικας Δημιουργίας αρχείου ασθενών

```
<!-- Patients Section -->
```

```
<section class="success" id="patients">
```

```
<div class="container">
```

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

```
<div class="col-lg-12 text-center">
```

```
<h2>ΑΡΧΕΙΟ ΑΣΘΕΝΩΝ</h2>
```

```
<hr class="star-light">
```

```
</div>
```

```
</div>
```

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

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

```
<!-- ΔΗΜΙΟΥΡΓΙΑ ΚΟΥΜΠΙΟΥ ΓΙΑ ΝΕΑ ΕΙΣΑΓΩΓΗ-->
```

```
<p> <a href="new patient.php" class="btn btn-success">Νέος Ασθενής</a></p>
```

```
<!-- ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ ΓΙΑ ΔΕΔΟΜΕΝΑ ΑΣΘΕΝΗ-->
```

```
<div class="table-responsive">
```

```
<table class="table table-striped table-bordered">
```

```
<thead>
```

```
<tr>
```

```
<th>ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ</th>
```

```
<th>ΟΝΟΜΑ</th>
```

```
<th>ΕΠΙΘΕΤΟ</th>
```

```
<th>ΑΜΚΑ</th>
```

```
<th>ΗΜ. ΓΕΝΝΗΣΗΣ</th>
```

```
<th colspan="2">ΕΝΕΡΓΕΙΕΣ</th>
```

```
</tr></thead><tbody>
```

```
<?php
```

```
    i. include 'database.php';
```

```
    ii. $pdo = Database::connect();
```

```
    iii. $sql = 'SELECT * FROM patient ORDER BY id DESC';
```

```
    foreach ($pdo->query($sql) as $row) {
```

```
        echo '<tr>';
```

```
        echo '<td>'. $row['id'] . '</td>';
```

```
        echo '<td>'. $row['name'] . '</td>';
```

```
        echo '<td>'. $row['surname'] . '</td>';
```

```
        echo '<td>'. $row['AMKA'] . '</td>';
```

```
        echo '<td>'. $row['date_of_birth'] . '</td>';
```

```
        echo '<td><a class="btn" href="read.php?id='.$row['id'].'">Προβολή</a></td>';
```

```
        echo '<td> <a class="btn btn-
```

```
success"href="update.php?id='.$row['id'].'">Επεξεργασία</a></td>';
```

```
        Database::disconnect();?></tbody></table></div></div></div></div></div>
```

```
</div></section>
```

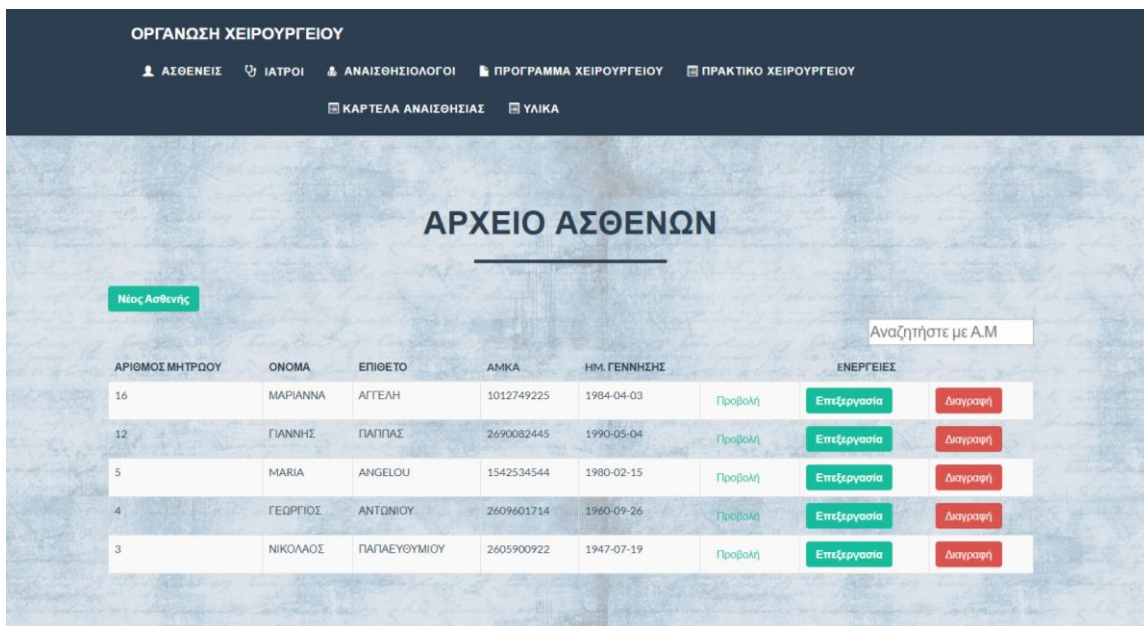
Αρχικά, στον παραπάνω κώδικα συνδέουμε το section που δημιουργήσαμε με την επιλογή «ΑΣΘΕΝΕΙΣ» του κεντρικού μενού. Έπειτα, δημιουργούμε έναν πίνακα που οι επικεφαλίδες του αντιστοιχούν στα πεδία του πίνακα PATIENT της βάσης μας. Έτσι, διαμορφώνουμε έναν πίνακα που εμφανίζει τα εξής στοιχεία ενός ασθενή: Αριθμό Μητρώου(id), Όνομα(name), Επίθετο(surname), ΑΜΚΑ και Ημερομηνία Γέννησης(date_of_birth). Προσθέτουμε και μία στήλη στον πίνακα που ονομάζεται «ΕΝΕΡΓΕΙΕΣ».

Για να γίνει ανάκτηση δεδομένων από τη βάση στο grid που δημιουργήσαμε ακολουθούμε τα εξής τρία βήματα:

- I. Κάνουμε include το αρχείο database.php
- II. Δημιουργούμε μία σύνδεση PDO με τη βάση μας
- III. Επιλέγουμε με την εντολή Select να εμφανιστούν οι εγγραφές από τον πίνακα Patient

Ο βρόχος *foreach* χρησιμοποιείται στο να παίρνουμε τις τιμές των δεδομένων από τον πίνακα. (32)

Επομένως, ο ιατρός ή η γραμματεία του χειρουργικού τμήματος που θέλουν να δουν το αρχείο των ασθενών αρκεί να επιλέξουν να πλοηγηθούν από το μενού.



ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ	ΟΝΟΜΑ	ΕΠΙΘΕΤΟ	ΑΜΚΑ	ΗΜ. ΓΕΝΝΗΣΗΣ	ΕΝΕΡΓΕΙΕΣ
16	ΜΑΡΙΑΝΝΑ	ΛΙΓΓΕΛΗ	1012749225	1984-04-03	Προβολή Επεξεργασία Διαγραφή
12	ΓΙΑΝΝΗΣ	ΠΑΠΠΑΣ	2690082445	1990-05-04	Προβολή Επεξεργασία Διαγραφή
5	MARIA	ANGELOU	1542534544	1980-02-15	Προβολή Επεξεργασία Διαγραφή
4	ΓΕΩΡΓΙΟΣ	ΑΝΤΩΝΙΟΥ	2609601714	1960-09-26	Προβολή Επεξεργασία Διαγραφή
3	ΝΙΚΟΛΑΟΣ	ΠΑΠΑΕΥΘΥΜΙΟΥ	2605900922	1947-07-19	Προβολή Επεξεργασία Διαγραφή

Εικόνα 7: Ενότητα Αρχείου Ασθενών

3.5.1. ΔΗΜΙΟΥΡΓΙΑ ΝΕΟΥ ΑΣΘΕΝΗ

Για να είναι πρακτική η εφαρμογή μας θα πρέπει να επιτρέπει στον χρήστη να κάνει κάποιες ενέργειες. Για να είναι κάτι τέτοιο εφικτό χρησιμοποιούμε το μοντέλο CRUD (Create, Read, Update, Delete). (33) Αυτό το μοντέλο μας επιτρέπει να έχουμε τέσσερις βασικές λειτουργίες:

- i. *Δημιουργία*, δηλαδή να κάνουμε INSERT στη βάση μας
- ii. *Ανάκτηση* δεδομένων από τη βάση με SELECT
- iii. *Τροποποίηση* κάνοντας UPDATE
- iv. *Διαγραφή* δεδομένων με DELETE

Εδώ περιγράφεται η προσθήκη ενός νέου ασθενή και γενικότερα η προσθήκη νέας εγγραφής σε κάθε πίνακα. Για να δώσουμε στον χρήστη αυτή τη δυνατότητα, αρχικά δημιουργήσαμε ένα κουμπί « Νέος ασθενής» στο αρχείο index.php που εμφανίζεται πάνω από τον πίνακα «Αρχείο ασθενών», όπως φαίνεται και στην παραπάνω εικόνα. Στη συνέχεια, δημιουργήσαμε το αρχείο «new patient.php», το οποίο ανοίγει κάθε φορά που κλικάρουμε το κουμπί «Νέος ασθενής» και μας επιτρέπει την εισαγωγή νέας εγγραφής.

4.1.Κώδικας για την εισαγωγή νέου ασθενή

```
<?php
require 'database.php';
    if ( !empty($_POST)) {
        $nameError = null;
        $surnameError = null;
        $AMKAEError = null;
        $date_of_birthError= null;
        $adressErroe= null;
        $phone_numberError= null;
        $insuranceError= null;
        $type_of_diseaseError= null;
        $medical_historyError= null;
```

```

$name = $_POST['name'];
$surname = $_POST['surname'];
$AMKA = $_POST['AMKA'];
$date_of_birth=$_POST['date_of_birth'];
$adress= $_POST['adress'];
$phone_number= $_POST['phone_number'];
$insurance=$_POST['insurance'];
$type_of_disease=$_POST['type_of_disease'];
$medical_history=$_POST['medical_history'];
$valid = true;
if (empty($AMKA)) {
    $AMKAError = 'Υποχρεωτικό Πεδίο;
    $valid = false;
}
// insert data
if ($valid) {
    $pdo = Database::connect();

    $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
    $sql = "INSERT INTO patient
(name,surname,AMKA,date_of_birth,adress,phone_number,insurance,type_of_di
sease,medical_history) values(?, ?, ?, ?, ?, ?, ?, ?, ?)";
    $q = $pdo->prepare($sql);
    $q->execute(array($name,$surname,$AMKA,$date_of_birth,$adress,$phone_nu
mber,$insurance,$type_of_disease,$medical_history ));
    Database::disconnect();
    header("Location:index.php");
}
} ?>

```

Η μεταβλητή \$_POST χρησιμοποιείται για συλλογή δεδομένων από μία φόρμα μέσω της μεθόδου «post» . Η πληροφορία που στέλνεται μέσω της POST μεθόδου είναι αόρατη και δεν έχει όριο μεγέθους για την πληροφορία που αποστέλλεται Αρχικά, ελέγχουμε εάν υπάρχει μία φόρμα που έχει υποβληθεί με τη μεταβλητή \$_POST. Αν υπάρχουν δεδομένα, τσεκάρουμε όλες τις εγγραφές ξεχωριστά για

σιγουρευτούμε ότι δεν είναι άδειες. Αφού γίνει η επαλήθευση των στοιχείων χρησιμοποιώντας το valid, γίνεται εισαγωγή δεδομένων στη βάση χρησιμοποιώντας την κλάση Database. Χρησιμοποιούμε το array (πίνακα) για να αποθηκεύσουμε το σύνολο των δεδομένων μας σε μία μεταβλητή. Τέλος, επαναδηγούμαστε στη σελίδα index.php χρησιμοποιώντας τη συνάρτηση PHP header ()).

Με τον παρακάτω κώδικα δημιουργούμε τις φόρμες μέσα στις οποίες θα εισάγουμε τα δεδομένα για τον νέο ασθενή. Με την εντολή echo εκτυπώνουμε στην οθόνη στα πεδία που δεν έχουν τιμές, το μήνυμα «Υποχρεωτικό Πεδίο».

4.2.Κώδικας για την εισαγωγή νέου ασθενή

```
<!DOCTYPE html>
<html >
<head>
  <script src="js/bootstrap.min.js"></script>
</head>

<body>
  <div class="container">
    <div class="span10 offset1">
      <div class="row">
        <h1>Εισάγετε στοιχεία ασθενή</h1>
      </div>
      <form class="form-horizontal" action="new patient.php" method="post">
        <div class="control-group <?php echo !empty($nameError)?'error:':'';?>">
          <label class="control-label">Όνομα</label>
          <div class="controls">
            <input name="name" type="text" value="<?php
echo !empty($name)?$name:'';?>">
            <?php if (!empty($nameError)): ?>
            <span class="help-inline"><?php echo $nameError;?></span>
            <?php endif; ?></div> </div>

        <div class="control-group <?php echo !empty($surnameError)?'error:':'';?>">
          <label class="control-label">Επίθετο</label>
          <div class="controls">
            <input name="surname" type="text" value="<?php
```

```

echo !empty($surname)?$surname:":?>">
    <?php if (!empty($surnameError)): ?>
        <span class="help-inline"><?php echo $surnameError;?></span>
    <?php endif;?> </div></div>
<div class="control-group <?php echo !empty($AMKAEError)?'error':":?>">
    <label class="control-label">Αμκα</label>
    <div class="controls">
        <input name="AMKA" type="text" value="<?php
echo !empty($AMKA)?$AMKA:":?>">
        <?php if (!empty($AMKAEError)): ?>
            <span class="help-inline"><?php echo $AMKAEError;?></span>
            <?php endif; ?> </div></div>
    <div class="control-group <?php echo !empty($date_of_birthError)?'error':":?>">
        <label class="control-label">Ημ. Γέννησης</label>
        <div class="controls">
            <input name="date_of_birth" type="text" placeholder="YY-MM-DD"
            value="<?php echo !empty($date_of_birth)?$date_of_birth:":?>">
            <?php if (!empty($date_of_birthError)): ?>
                <span class="help-inline"><?php echo $date_of_birthError;?></span>
                <?php endif; ?> </div></div>
    <div class="control-group <?php echo !empty($adressError)?'error':":?>">
        <label class="control-label">Διεύθυνση</label>
        <div class="controls">
            <input name="adress" type="text" value="<?php
echo !empty($adress)?$adress:":?>">
            <?php if (!empty($adressError)): ?>
                <span class="help-inline"><?php echo $adressError;?></span>
                <?php endif; ?> </div></div>
    <div class="control-group <?php
echo !empty($phone_numberError)?'error':":?>">
        <label class="control-label">Τηλέφωνο</label>
        <div class="controls">
            <input name="phone_number" type="text" value="<?php
echo !empty($phone_number)?$phone_number:":?>">
            <?php if (!empty($phone_numberError)): ?>
                <span class="help-inline"><?php echo $phone_numberError;?></span>
                <?php endif; ?> </div></div>
            <div class="control-group <?php
echo !empty($insuranceError)?'error':":?>">
                <label class="control-label">Ασφάλιση</label>

```

```
<div class="controls"><input name="insurance" type="text" value="<?php
echo !empty($insurance)?$insurance:";?>">
<?php if (!empty($insuranceError)): ?>
<span class="help-inline"><?php echo $insuranceError;?></span><?php
endif; ?> </div></div>
```

```
<div class="control-group <?php
echo !empty($type_of_diseaseError)?'error:':";?>">
<label class="control-label">Πρόβλημα Υγείας</label>
<div class="controls">
<input name="type_of_disease" type="text" value="<?php
echo !empty($type_of_disease)?$type_of_disease:";?>">
<?php if (!empty($type_of_diseaseError)): ?>
<span class="help-inline"><?php echo $type_of_diseaseError;?></span>
<?php endif; ?>
</div></div>
```

```
<div class="control-group <?php
echo !empty($medical_historyError)?'error:':";?>">
<label class="control-label">Ιατρικό Ιστορικό</label>
<div class="controls">
<input name="medical_history" type="text"
placeholder="Εισάγετε περιληπτικά το ιστορικό του ασθενή" value="<?php
echo !empty($medical_history)?$medical_history:";?>">
<?php if (!empty($medical_historyError)): ?>
<span class="help-inline"><?php echo $medical_historyError;?></span>
<?php endif; ?>
</div></div>
```

```
<div class="form-actions">
<button type="submit" class="btn btn-success">Δημιουργία</button>
<a class="btn" href="index.php">Επιστροφή</a>
</div>
</form>
</div>
```

```
</div> <!-- /container -->
</body></html>
```

Τέλος προσθέτουμε ένα κουμπί για υποβολή και ένα κουμπί για επιστροφή στην αρχική σελίδα. Το αποτέλεσμα του παραπάνω κώδικα πατώντας την ΕΙΣΑΓΩΓΗ ΝΕΟΥ ΑΣΘΕΝΗ φαίνεται στην εικόνα:

localhost/maria1/new%20patient.php

Εισάγετε στοιχεία ασθενή

Όνομα

Επίθετο

Αμκα

Ημ. Γέννησης
YY-MM-DD

Διεύθυνση

Τηλέφωνο

Ασφάλιση

Πρόβλημα Υγείας

Ιατρικό Ιστορικό
Εισάγετε περιληπτικά το ιστο

Εικόνα 7.1: Εισαγωγή Νέου Ασθενή

3.5.2. ΠΡΟΒΟΛΗ ΣΤΟΙΧΕΙΩΝ ΑΣΘΕΝΗ

Στην αρχική μας σελίδα στο αρχείο των ασθενών εμφανίζονται κάποιες εγγραφές ασθενών με τα στοιχεία Α.Μ, Όνομα, Επίθετο, ΑΜΚΑ και Ημερομηνία Γέννησης. Αν θέλουμε να δούμε όλες τις πληροφορίες ενός ασθενή πατάμε το κουμπί «Προβολή». Δημιουργούμε το αρχείο *read.php*.

5.Κώδικας για την προβολή ασθενή

```
<?php
    require 'database.php';
    $id = null;
    if ( !empty($_GET['id'])) {
        $id = $_REQUEST['id'];
    }

    if ( null==$id ) {
        header("Location: index.php"); }
else {
    $pdo = Database::connect();

    $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
    $sql = "SELECT * FROM patient where id = ?";
    $q = $pdo->prepare($sql);
    $q->execute(array($id));
    $data = $q->fetch(PDO::FETCH_ASSOC);
    Database::disconnect();
}
?>

<!DOCTYPE html>
<html >
<head>
    <meta charset="utf-8">
    <script src="js/bootstrap.min.js"></script>
</head>
<body>
    <div class="container">
```

```

<div class="span10 offset1">
  <div class="row">
    <h2><i><u>Προφίλ Ασθενή</i></u></h2>
  </div>
    <div> 
    </div>
<div class="form-horizontal" >
  <div class="control-group">
<label class="control-label"><h4>Α.Μ. ΑΣΘΕΝΗ</h4></label>
  <div class="controls">
    <label class="checkbox">
      <?php echo $data['id']; ?> </label></div></div>

  <div class="form-horizontal" >
    <div class="control-group">
<label class="control-label"><h4>ΟΝΟΜΑ</h4></label>
      <div class="controls">
        <label class="checkbox">
          <?php echo $data['name']; ?> </label></div></div>

  <div class="control-group">
<label class="control-label"><h4>ΕΠΙΘΕΤΟ</h4></label>
    <div class="controls">
      <label class="checkbox">
        <?php echo $data['surname']; ?> </label></div></div>

  <div class="control-group">
<label class="control-label"><h4>ΑΜΚΑ</h4></label>
    <div class="controls">

```



```

< label class="checkbox">
<?php echo $data['AMKA'];?> </label></div>
  <div class="control-group">
<label class="control-label"><h4>ΗΜ. ΓΕΝΝΗΣΗΣ</h4></label>
  <div class="controls">
    <label class="textbox">
      <?php echo $data['date_of_birth'];?></label></div>

  <div class="control-group">
<label class="control-label"><h4>ΔΙΕΥΘΥΝΣΗ</h4></label>
  <div class="controls">
<label class="checkbox">
<?php echo $data['adress'];?> </label></div>

  <div class="control-group">
<label class="control-label"><h4>ΤΗΛΕΦΩΝΟ</h4></label>
  <div class="controls">
<label class="checkbox">
      <?php echo $data['phone_number'];?> </label></div>

  <div class="control-group">
<label class="control-label"><h4>ΑΣΦΑΛΙΣΗ</h4></label>
  <div class="controls">
<label class="checkbox">
<?php echo $data['insurance'];?> </label></div>

  <div class="control-group">
<label class="control-label"><h4>ΠΡΟΒΛΗΜΑ ΥΓΕΙΑΣ</h4></label>
  <div class="controls">
<label class="checkbox">
      <?php echo $data['type_of_disease'];?> </label></div>

```

```

<div class="control-group">
<label class="control-label"><h4>ΙΣΤΟΡΙΚΟ ΑΣΘΕΝΗ</h4></label>
<div class="controls">
  <label class="checkbox">
    <?php echo $data['medical_history'];?> </label></div></div>

<div class="control-group">
<label class="control-label"><h4>ΑΛΛΕΡΓΙΕΣ</h4></label>
<div class="controls">
<label class="textbox">
<?php echo $data['allergies'];?> </label></div></div>
<a class="btn" href="index.php">Επιστροφή</a></div>
</div></div>
</div> <!-- /container -->

</body>
</html>

```

Η μεταβλητή `$_GET` συγκεντρώνει δεδομένα μετά την υποβολή φορμών. Αρχικά, προσπαθούμε να εντοπίσουμε τη μεταβλητή `$_GET['id']`. Αν δε βρεθεί, ο χρήστης ανακατευθύνεται στο `index.php`. Διαφορετικά διαβάζει τα δεδομένα από τη βάση χρησιμοποιώντας το πεδίο `id` και αποθηκεύει τα δεδομένα σε μία μεταβλητή `$data`. Το κομμάτι της `html` χρησιμεύει για να εμφανίζονται τα δεδομένα αυτής της μεταβλητής.

Προφίλ Ασθενή



A.M. ΑΣΘΕΝΗ

3

ΟΝΟΜΑ

ΝΙΚΟΛΑΟΣ

ΕΠΙΘΕΤΟ

ΠΑΠΑΕΥΘΥΜΙΟΥ

ΑΜΚΑ

2605900922

ΗΜ. ΓΕΝΝΗΣΗΣ

1947-07-19

ΔΙΕΥΘΥΝΣΗ

ΝΙΚΑΙΑΣ 14, ΝΙΚΑΙΑ

ΤΗΛΕΦΩΝΟ

6943730294

ΑΣΦΑΛΙΣΗ

ΙΔΙΩΤΙΚΗ

ΠΡΟΒΛΗΜΑ ΥΓΕΙΑΣ

ΚΑΡΔΙΑΚΑ ΠΡΟΒΛΗΜΑΤΑ

ΙΣΤΟΡΙΚΟ ΑΣΘΕΝΗ

Ο ΑΣΘΕΝΗΣ ΕΧΕΙ ΠΕΡΑΣΕΙ ΕΠΕΙΣΟΔΙΟ ΕΜΦΡΑΓΜΑΤΟΣ.

ΑΛΛΕΡΓΙΕΣ

ΔΕΝ ΕΧΟΥΝ ΑΝΑΦΕΡΘΕΙ

[Επιστροφή](#)

Εικόνα 7.2: Προβολή Στοιχείων Ασθενή

Επομένως, αν θέλουμε να δούμε όλες τις πληροφορίες για τον ασθενή με Α.Μ: 3 και επιλέξουμε προβολή από τις ενέργειες, βλέπουμε πως ο ασθενής ονομάζεται Νικόλαος Παπαευθυμίου, έχει αριθμό ΑΜΚΑ:2605900922 , είναι γεννημένος το 1947, εμφανίζονται τα στοιχεία επικοινωνίας του (διεύθυνση, τηλέφωνο), έχει ιδιωτική ασφάλιση, το πρόβλημα που έχει είναι στην καρδιά του, ενώ μπορούμε να δούμε και ένα σύντομο ιστορικό και τις αλλεργίες του.

3.5.3. ΕΠΕΞΕΡΓΑΣΙΑ ΣΤΟΙΧΕΙΩΝ ΑΣΘΕΝΗ

Η επεξεργασία δίνει τη δυνατότητα στον χρήστη να επεξεργαστεί τις ήδη καταχωρημένες πληροφορίες ενός ασθενή. Παρόμοια με τη δημιουργία και προβολή ασθενή, έχουμε δημιουργήσει ένα αντίστοιχο κουμπί στο αρχείο `index.php`. Δημιουργούμε το αρχείο `update.php`. Αρχικά, τσεκάρουμε αν υπάρχει ήδη μία φόρμα με τη μεταβλητή `$_POST`. Στη συνέχεια, ενημερώνουμε τη βάση μας . Χρησιμοποιώντας τη μεταβλητή `$_GET request`, γίνεται ανάκτηση των ήδη περασμένων πληροφοριών από τη βάση μας.

6.Κώδικας για επεξεργασία ασθενή

```
<?php
require 'database.php';
$id = null;
if ( !empty($_GET['id'])) {
    $id = $_REQUEST['id'];
}
if ( null==$id ) {
    header("Location: index.php");
}

if ( !empty($_POST)) {
    $nameError = null;
    $surnameError = null;
    $AMKAEError = null;
    $date_of_birth= null;
    $adress= null;
    $phone_number= null;
    $insurance= null;
    $type_of_disease= null;
    $medical_history= null;
    $allergies=null;

    $name = $_POST['name'];
    $surname = $_POST['surname'];
    $AMKA = $_POST['AMKA'];
    $date_of_birth=$_POST['date_of_birth'];
    $adress=$_POST['adress'];
    $phone_number=$_POST['phone_number'];
```

```

    $insurance=$_POST['insurance'];
    $type_of_disease=$_POST['type_of_disease'];
    $medical_history=$_POST['medical_history'];
    $allergies=$_POST['allergies'];
$valid = true;
if (empty($AMKA)) {$AMKAError = 'ΥΠΟΧΡΕΩΤΙΚΟ ΠΕΔΙΟ';
    $valid = false; }
// update data
    if ($valid) {
        $pdo = Database::connect();
        $pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
        $sql = "UPDATE patient set name = ?, surname = ?, AMKA = ?,
date_of_birth= ?,adress= ?,
phone_number= ?,insurance= ?,type_of_disease= ?,medical_history= ?,allergies
= ? WHERE id = ?";
        $q = $pdo->prepare($sql);

        $q->execute(array($name,$surname,$AMKA,$date_of_birth,$adress,$phone_nu
mber,$insurance,$type_of_disease,$medical_history,$allergies,$id));
        Database::disconnect();
        header("Location: index.php"); }}
else {
    $pdo = Database::connect();
    $pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    $sql = "SELECT * FROM patient where id = ?";
    $q = $pdo->prepare($sql);
    $q->execute(array($id));
    $data = $q->fetch(PDO::FETCH_ASSOC);
    $name = $data['name'];
    $surname = $data['surname'];
    $AMKA = $data['AMKA'];
    $date_of_birth=$data['date_of_birth'];
    $adress=$data['adress'];
    $phone_number = $data['phone_number'];
    $insurance = $data['insurance'];
    $type_of_disease=$data['type_of_disease'];
    $allergies=$data['allergies']
Database::disconnect(); }?>

```

Στο τέλος του παραπάνω κώδικα προστίθεται και ο κώδικας html για τη δημιουργία φόρμας εισαγωγής των δεδομένων που είναι ακριβώς ίδιος με τον κώδικα html του αρχείου new patient.php.

Για παράδειγμα, πατώντας «Επεξεργασία» στον ασθενή με Α.Μ: 16 , βλέπουμε τις πληροφορίες που είναι περασμένες και μπορούμε να τις ενημερώσουμε και πατώντας καταχώριση να τις τροποποιήσουμε.

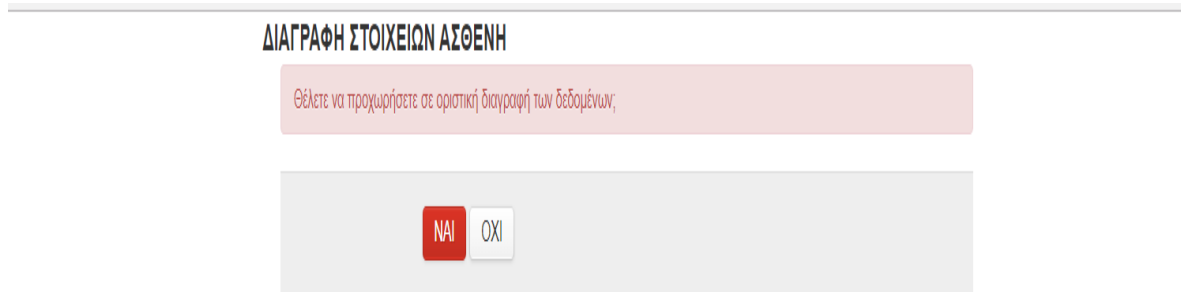
localhost/maria1/update.php?id=16

ΕΠΕΞΕΡΓΑΣΙΑ

Όνομα	MARIANNA
Επίθετο	ΑΓΓΕΛΗ
ΑΜΚΑ	1012749225
Ημ. Γέννησης	1984-04-03
Διεύθυνση	ΠΗΛΙΟΥ 49, ΒΟΛΟΣ
Τηλέφωνο	2421023444
Ασφάλιση	ΙΔΙΩΤΙΚΗ
Ιατρικό Πρόβλημα	Η ΑΣΘΕΝΗΣ ΕΧΕΙ ΠΑΛΙΝ,
Ιστορικό Ασθενή	
Αλλεργίες	Αλλεργία στο φάρμακο Am
ΚΑΤΑΧΩΡΙΣΗ	<u>ΚΛΕΙΣΙΜΟ</u>

Εικόνα 7.3: Επεξεργασία Στοιχείων Ασθενή

3.5.4. ΔΙΑΓΡΑΦΗ ΣΤΟΙΧΕΙΩΝ ΑΣΘΕΝΗ



Εικόνα 7.4: Διαγραφή Ασθενή

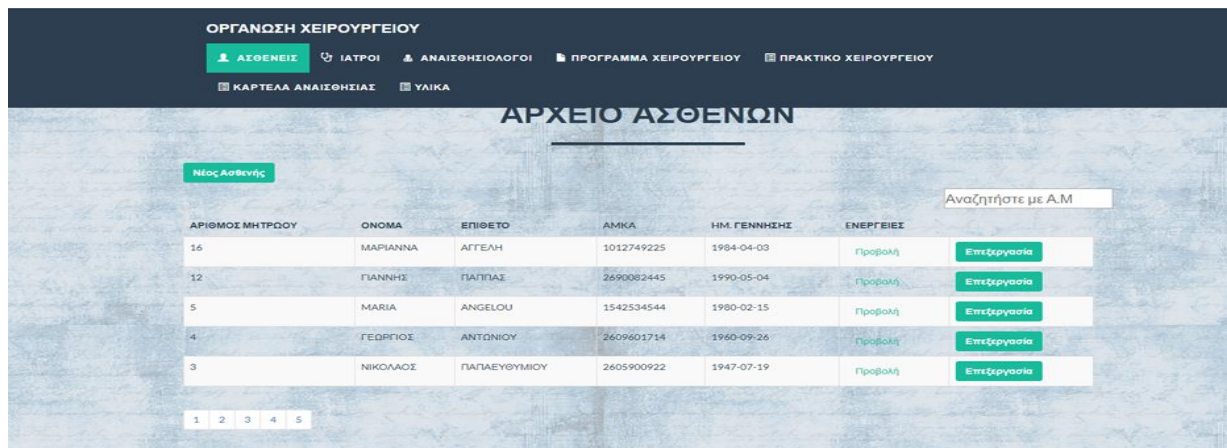
Μία ακόμα δυνατότητα του συστήματος είναι η διαγραφή. Ο χρήστης μπορεί να προχωρήσει σε διαγραφή μίας εγγραφής, αν παραδείγματος χάριν έχει γίνει κάποιος λάθος σε αυτή. Στην εικόνα 6.3 φαίνεται το μήνυμα επιβεβαίωσης της εγγραφής ενός ασθενή. Πατώντας το ναι, πραγματοποιείται η διαγραφή των στοιχείων του ασθενή που επιλέξαμε.

3.5.5. ΑΝΑΖΗΤΗΣΗ ΑΣΘΕΝΗ

Μία άλλη ενέργεια που μπορεί να έχει ο χρήστης είναι η δυνατότητα αναζήτησης ενός ασθενή με βάση τον Αριθμό Μητρώου του. Για να πραγματοποιηθεί αυτή η δυνατότητα εισάγουμε πάνω από τον πίνακα patient του αρχείο index:

```
<input type="text" id="myInput" onkeyup="myFunction()"
placeholder="Αναζητήστε με Α.Μ " title="Type in id">
```

και δημιουργείται το πεδίο της αναζήτησης.



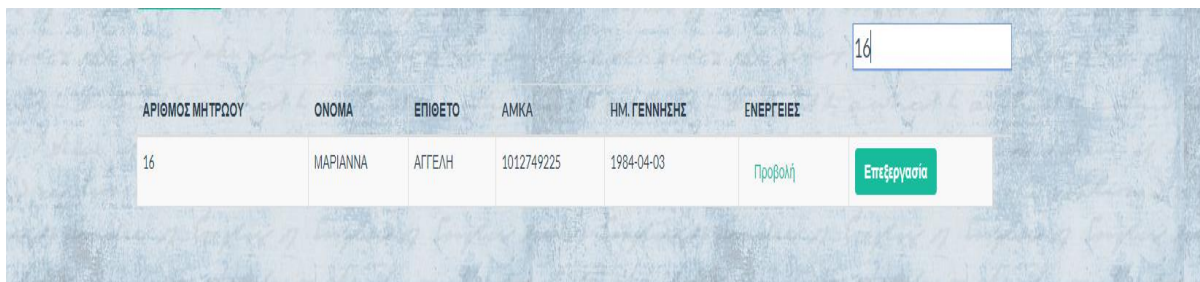
Εικόνα 7.5: Αναζήτηση Ασθενή

Στο τέλος του πίνακα προσθέτουμε τον ακόλουθο κώδικα σε Javascript. (34)

7.Κώδικας για την αναζήτηση ασθενή

```
<script>
function myFunction() {
  var input, filter, table, tr, td, i;
  input = document.getElementById("myInput");
  filter = input.value.toUpperCase();
  table = document.getElementById("myTable");
  tr = table.getElementsByTagName("tr");
  for (i = 0; i < tr.length; i++) {
    td = tr[i].getElementsByTagName("td")[0];
    if (td) {
      if (td.innerHTML.toUpperCase().indexOf(filter) > -1) {
        tr[i].style.display = "";
      } else {
        tr[i].style.display = "none";
      }
    }
  }
}
</script>
```

Παραδείγματος χάριν, αν αναζητήσουμε τον ασθενή με Α.Μ 16, εμφανίζονται μόνο τα στοιχεία του ασθενή με αυτό το μοναδικό αριθμό.



The screenshot shows a search interface with a text input field containing '16'. Below it is a table with the following data:

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ	ΟΝΟΜΑ	ΕΠΙΘΕΤΟ	ΑΜΚΑ	ΗΜ. ΓΕΝΝΗΣΗΣ	ΕΝΕΡΓΕΙΕΣ
16	ΜΑΡΙΑΝΝΑ	ΑΓΓΕΛΗ	1012749225	1984-04-03	Προβολή Επεξεργασία

Εικόνα 7.6: Αποτελέσματα Αναζήτησης

Η δυνατότητα της αναζήτησης μπορεί να προσαρμοστεί και να εφαρμοσθεί για τα υπόλοιπα πεδία, καθώς και για τους υπόλοιπους πίνακες.

3.6. ΑΡΧΕΙΟ ΙΑΤΡΩΝ

Επιλέγοντας από το Κεντρικό μενού την ενότητα «Ιατροί», εμφανίζονται όλοι οι ιατροί που είναι καταχωρημένοι στο σύστημα μας. Εμφανίζεται ο αριθμός μητρώου του ιατρού, ονοματεπώνυμο και ειδικότητα, ενώ αν επιλέξουμε από τις ενέργειες προβολή βλέπουμε τις πλήρης πληροφορίες για τον καθένα.

ΟΡΓΑΝΩΣΗ ΧΕΙΡΟΥΡΓΕΙΟΥ

ΑΣΘΕΝΕΙΣ ΙΑΤΡΟΙ ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΙ ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ

ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ ΥΛΙΚΑ

ΙΑΤΡΙΚΟ ΠΡΟΣΩΠΙΚΟ

Νέος Ιατρός

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ	ΕΠΙΘΕΤΟ	ΕΙΔΙΚΟΤΗΤΑ	ΕΝΕΡΓΕΙΕΣ
27	JOHNSON	ΟΡΘΟΠΕΔΙΚΟΣ	Προβολή Επεξεργασία Διαγραφή
26	ΝΙΚΟΛΟΥΔΗ	ΟΦΘΑΛΜΙΑΤΡΟΣ	Προβολή Επεξεργασία Διαγραφή
25	ΧΑΤΖΕΥΘΥΜΙΟΥ	ΟΡΘΟΠΕΔΙΚΟΣ	Προβολή Επεξεργασία Διαγραφή
12	STONE	ΚΑΡΔΙΟΛΟΓΟΣ	Προβολή Επεξεργασία Διαγραφή
2	ΠΑΝΝΑΚΗ	ΓΥΝΑΙΚΟΛΟΓΟΣ	Προβολή Επεξεργασία Διαγραφή
1	ΜΑΝΟΥ	ΝΕΥΡΟΧΕΙΡΟΥΡΓΟΣ	Προβολή Επεξεργασία Διαγραφή

Εικόνα 8: Προβολή Ενότητας Ιατρών

Έτσι, επιλέγοντας «Προβολή» για τον ιατρό με Α.Μ:12, πληροφορούμαστε πως ονομάζεται Stone, είναι καρδιολόγος αλλά και τα στοιχεία επικοινωνίας του(τηλέφωνο & email) όπως φαίνονται στην παρακάτω εικόνα.



A.M. ΙΑΤΡΟΥ

12

ΟΝΟΜΑ

DAVID

ΕΠΙΘΕΤΟ

STONE

ΕΙΔΙΚΟΤΗΤΑ

ΚΑΡΔΙΟΛΟΓΟΣ

ΔΙΕΥΘΥΝΣΗ

NEW YORK HOSPITAL

EMAIL

stone@outlook.com

ΤΗΛΕΦΩΝΟ

6984512476

[Επιστροφή](#)

Εικόνα 8.1: Επεξεργασία Στοιχείων Ασθενή

3.7. ΑΡΧΕΙΟ ΑΝΑΙΣΘΗΣΙΟΛΟΓΩΝ

Αντίστοιχες πληροφορίες με αυτές των ιατρών είναι οι πληροφορίες που παίρνουμε επιλέγοντας την ενότητα «Αναισθησιολόγοι».

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ	ΕΠΙΘΕΤΟ	ΕΝΕΡΓΕΙΣ
8	ΠΑΠΑΔΟΠΟΥΛΟΣ	Προβολή Επεξεργασία
7	ΧΑΤΖΗΣ	Προβολή Επεξεργασία
6	CURIE	Προβολή Επεξεργασία
5	ΒΑΣΙΛΕΙΟΥ	Προβολή Επεξεργασία

Εικόνα 9: Προβολή Ενότητας Αναισθησιολόγων

Ο αναισθησιολόγος με Α.Μ: 8 ονομάζεται Παπαδόπουλος Πάρης και εργάζεται στο Αναισθησιολογικό Τμήμα του Λαϊκού Νοσοκομείου, ενώ φαίνονται και τα στοιχεία επικοινωνίας του.

ΣΤΟΙΧΕΙΑ ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ



Α.Μ. ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ

8

ΟΝΟΜΑ

ΠΑΡΗΣ

ΕΠΙΘΕΤΟ

ΠΑΠΑΔΟΠΟΥΛΟΣ

ΔΙΕΥΘΥΝΣΗ

ΑΝΑΙΣΘΗΣΙΟΛΟΓΙΚΟ ΤΜΗΜΑ ΛΑΪΚΟΥ ΝΟΣΟΚΟΜΕΙΟΥ

EMAIL

parispap@gmail.com

ΤΗΛΕΦΩΝΟ

6945824141

[Επιστροφή](#)

Εικόνα 9.1: Προβολή Στοιχείων Αναισθησιολόγου

3.8. ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ

Στην ενότητα «Πρόγραμμα Χειρουργείου» εμφανίζεται το πρόγραμμα για όλα τα χειρουργεία. Κάθε εισαγωγή έχει έναν μοναδικό αριθμό εγγραφής, την ημερομηνία και έναρξη της επέμβασης, τον αριθμό της χειρουργικής αίθουσας, Α.Μ. ασθενή, Α.Μ ιατρού, Α.Μ αναισθησιολόγου, είδος επέμβασης, μονάδες αίματος που υπολογίζεται ότι θα χρειαστούν και δωμάτιο νοσηλείας του ασθενή.

ΑΡΙΘΜΟΣ ΕΓΓΡΑΦΗΣ	ΗΜΕΡΟΜΗΝΙΑ & ΩΡΑ	ΧΕΙΡΟΥΡΓΙΚΗ ΑΙΘΟΥΣΑ	Α.Μ. ΑΣΘΕΝΗ	Α.Μ. ΙΑΤΡΟΥ	Α.Μ. ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ	ΕΠΕΜΒΑΣΗ	ΑΙΜΑ	ΔΩΜΑΤΙΟ	ΕΝΕΡΓΕΙΕΣ
1	2017-01-27 09:00:00	A3	3	12	6	BY PASS	3	548	Προβολή ΕΝΗΜΕΡΩΣΗ
2	2017-10-12 08:00:00	A4	12	27	6	ΟΡΘΟΠΕΔΙΚΗ	2	244	Προβολή ΕΝΗΜΕΡΩΣΗ
6	2018-01-30 00:00:00	A4	5	2	5	ΠΑΛΙΝΔΡΟΜΗ ΚΥΗΣΗ	1	362	Προβολή ΕΝΗΜΕΡΩΣΗ

Εικόνα 10: Προβολή Ενότητας Προγράμματος Χειρουργείου

Για παράδειγμα, το πρόγραμμα με αριθμό εγγραφής 1, είναι επέμβαση by pass στη Καρδιολογική Κλινική, στην αίθουσα A3, ο ασθενής έχει Α.Μ:3 είναι 82 ετών, δεν έχει κάποιον δείκτη (HIV ή Ηπατίτιδα), βρίσκεται στο δωμάτιο 548, θα χρειαστεί 3 μονάδες αίματος AB+ και το χειρουργείο του είναι προγραμματισμένο (τακτικό) στις 28/1/2018 09:00 από τον ιατρό ΑΜ:12 και αναισθησιολόγο Α.Μ:6.

ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ



ΚΩΔΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

1

ΗΜΕΡΟΜΗΝΙΑ & ΩΡΑ

2018-01-28 09:00:00

ΤΜΗΜΑ

ΚΑΡΔΙΟΛΟΓΙΚΗ ΚΛΙΝΙΚΗ

ΔΙΘΥΣΙΑ

A3

Α.Μ. ΑΣΘΕΝΗ

3

ΑΜΚΑ

2605900922

Α.Μ. ΙΑΤΡΟΥ

12

Α.Μ. ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ

6

ΗΛΙΚΙΑ

82

ΕΓΧΕΙΡΗΣΗ

BY PASS

ΑΙΜΑ

3 AB+

ΔΩΜΑΤΙΟ

548

ΔΕΙΚΤΕΣ

ΚΑΝΕΝΑΣ

ΕΙΔΟΣ

ΤΑΚΤΙΚΟ

ΔΗΜΙΟΥΡΓΙΑ ΠΡΟΓΡΑΜΜΑΤΟΣ

2018-01-26 10:00:55

ΕΝΗΜΕΡΩΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ

2018-01-28 23:52:17

Επιστροφή

Εικόνα 10.1: Προβολή Προγράμματος Χειρουργείου

Στο τέλος της προβολής της κάθε εγγραφής έχει προστεθεί η ημερομηνία και η ώρα δημιουργίας της, προκειμένου να γνωρίζει ο χρήστης πότε δημιουργήθηκε η εγγραφή. Ακόμα, υπάρχει η δυνατότητα πληροφόρησης για το χρόνο της τελευταίας ενημέρωσης του προγράμματος. Αυτή είναι μία δυνατότητα που μπορεί να επεκταθεί και στα αρχεία των υπόλοιπων κατηγοριών, αλλά και κατά την είσοδο των χρηστών στο σύστημα, ώστε να έχουμε πληροφορίες για τη τελευταία σύνδεση του χρήστη στο σύστημα.

3.9. ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ

Το πρακτικό χειρουργείου συμπληρώνεται από τον χειρουργό ιατρό και περιλαμβάνει τις λεπτομέρειες που αφορούν την επέμβαση.

ΚΩΔΙΚΟΣ ΕΠΕΜΒΑΣΗΣ	Α.Μ ΑΣΘΕΝΗ	Α.Μ ΙΑΤΡΟΥ	ΗΜΕΡΟΜΗΝΙΑ	ΕΙΔΟΣ ΕΠΕΜΒΑΣΗΣ	ΕΝΕΡΓΕΙΕΣ
1	12	25	2017-12-12	ΚΑΤΑΓΜΑ ΚΝΗΜΗΣ	Προβολή Επεξεργασία
5	5	2	2017-12-10	ΠΑΛΙΝΔΟΜΗΚΥΗΣΗ	Προβολή Επεξεργασία
12	3	12	2017-01-27	BY PASS	Προβολή Επεξεργασία

Εικόνα 11: Ενότητα Πρακτικού Χειρουργείου

Η κάθε επέμβαση λαμβάνει ένα τυχαίο αριθμό. Στο πρακτικό φαίνονται οι Α.Μ ασθενή, ιατρού και αναισθησιολόγου. Ο ιατρός καταγράφει τον τύπο και την ημερομηνία της επέμβασης, καθώς και την ώρα εισόδου του ασθενή στην χειρουργική αίθουσα. Η ώρα εισόδου μπορεί να διαφέρει από την ώρα έναρξης της επέμβασης, οπότε καταγράφεται και αυτή. Όταν ολοκληρωθεί η επέμβαση καταγράφεται η ώρα εξόδου του ασθενή. Από το πρακτικό χειρουργείου πληροφορούμαστε για το ποιοι ήταν οι βοηθοί του ιατρού και του αναισθησιολόγου, ποιος νοσηλευτής ήταν στην εργαλειοδοσία, τον τύπο της αναισθησίας και τον Α.Μ της αναισθησίας, ώστε να ανατρέξουμε και να πάρουμε περισσότερες πληροφορίες γι' αυτή. Τέλος, ο ιατρός μπορεί να περιγράψει αναλυτικά τη διαδικασία και τη τεχνική που ακολούθησε κατά την επέμβαση, ενώ μπορεί ακόμα να καταγράψει διάφορα ευρήματα και την έκβαση της.

ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ



ΚΩΔΙΚΟΣ ΕΓΧΕΙΡΙΣΗΣ

12

Α.Μ. ΑΣΘΕΝΗ

3

Α.Μ. ΙΑΤΡΟΥ

12

ΗΜΕΡΟΜΗΝΙΑ

2018-01-28

ΕΙΔΟΣ ΕΠΕΜΒΑΣΗΣ

BY PASS

ΩΡΑ ΕΙΣΟΔΟΥ

09:00:00

ΩΡΑ ΕΞΑΡΞΗΣ

09:15:00

ΩΡΑ ΕΞΟΔΟΥ

12:30:00

ΧΕΙΡΟΥΡΓΙΚΗ ΑΙΘΟΥΣΑ

A3

ΒΟΗΘΟΙ

ΚΑΡΡΑ ΜΑΓΔΑΛΗΝΗ

Α.Μ. ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ

6

ΒΟΗΘΟΙ ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ

ΜΑΝΟΥ ΒΑΣΙΛΗΣ, ΝΤΑΣΗ ΚΑΤΕΡΙΝΑ

ΕΡΓΑΛΕΙΟΛΟΓΙΑ

ΜΑΝΙΚΑ ΓΕΩΡΓΙΑ

ΤΥΠΟΣ ΑΝΑΙΣΘΗΣΙΑΣ

ΓΕΝΙΚΗ

Α.Μ. ΑΝΑΙΣΘΗΣΙΑΣ

3

ΠΕΡΙΓΡΑΦΗ

Έγινε επέμβαση ανοιχτής καρδιάς. Συνδέθηκε μηχανήμα το οποίο συνδέεται με σωλήνες με τα αγγεία του ασθενούς, έτσι ώστε προσωρινά η λειτουργία της καρδιάς και των πνευμόνων να αντικαθίσταται από αυτό

ΑΠΟΤΕΛΕΣΜΑΤΑ/ΕΥΡΗΜΑΤΑ

Η επέμβαση ήταν επιτυχής. Ο ασθενής πρέπει να παρακολουθείται τακτικά κατά την παραμονή του στο νοσοκομείο.

Εικόνα 11.1: Προβολή Πρακτικού Χειρουργείου

3.10. ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ

ΟΡΓΑΝΩΣΗ ΧΕΙΡΟΥΡΓΕΙΟΥ							
ΑΣΘΕΝΕΙΣ	ΙΑΤΡΟΙ	ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΙ	ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ	ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ			
ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ	ΥΛΙΚΑ						
ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ							
Εισάγετε καρτέλα αναισθησίας							
ΑΡΙΘΜΟΣ ΚΑΡΤΕΛΑΣ	Α.Μ ΑΣΘΕΝΗ	Α.Μ ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ	Α.Μ ΕΠΕΜΒΑΣΗΣ	ΩΡΑ ΝΑΡΚΩΣΗΣ		ΕΝΕΡΓΕΙΕΣ	
1	3	6	12	09:15:00	Προβολή	Επέξεργασία	Διαγραφή
2	3	7	5	00:00:00	Προβολή	Επέξεργασία	Διαγραφή

Εικόνα 12: Ενότητα Καρτέλα Αναισθησίας

Από το πρακτικό του χειρουργείου μπορούμε να δούμε τον Α.Μ της αναισθησίας και να ανατρέξουμε στην αντίστοιχη ενότητα. Αντίστοιχα, στην καρτέλα αναισθησίας βλέπουμε τον Α.Μ της επέμβασης (δηλαδή το πρακτικό). Στη καρτέλα αναισθησίας καταγράφεται η ώρα και η διάρκεια της νάρκωσης, η τεχνική που εφάρμοσε ο αναισθησιολόγος και τα φάρμακα που χορήγησε. Στο πεδίο δείκτες καταγράφει αν οι τιμές οξυγόνου, πίεσης, σφίξεων ήταν φυσιολογικές, ενώ στη κατάσταση εξόδου περιγράφει την κατάσταση του ασθενή μετά την αναισθησία δηλαδή το επίπεδο συνείδησης και τη κινητικότητα των κάτω άκρων. Τέλος, σημειώνεται ο βαθμός πόνου μετά την αναισθησία, αλλά και τυχόν επιπλοκές.

ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ



ΑΡΙΘΜΟΣ ΚΑΡΤΕΛΑΣ
1

Α.Μ. ΑΣΘΕΝΗ
3

Α.Μ ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΥ
6

Α.Μ. ΕΠΕΜΒΑΣΗΣ
12

ΩΡΑ ΝΑΡΚΩΣΗΣ
09:15:00

ΔΙΑΡΚΕΙΑ
02:30:00

ΤΕΧΝΙΚΗ
ΤΙΤΑ

ΦΑΡΜΑΚΑ
ΙΣΟΦΛΟΥΡΑΝΙΟ
ΔΕΙΚΤΕΣ

Το οξύγνο και η πίεση του ασθενή ήταν σε φυσιολογικά επίπεδα καθόλη τη διάρκεια της αναισθησίας.

ΚΑΤΑΣΤΑΣΗ ΕΞΟΔΟΥ

Ο ασθενής είχε επανέλθει πλήρως κατά την έξοδο του από τη χειρουργική αίθουσα

ΒΑΘΜΟΣ ΠΟΝΟΥ
2

ΑΠΟΤΕΛΕΣΜΑΤΑ

Δεν υπήρξε κάποια επιπλοκή καθόλη την διάρκεια της αναισθησίας.
Επιστροφή

Εικόνα 12.1: Προβολή Στοιχείων Αναισθησίας

3.11. ΥΛΙΚΑ

Στη τελευταία ενότητα μπορούμε να δούμε τα υλικά που χρησιμοποιήθηκαν για κάθε επέμβαση.

ΟΡΓΑΝΩΣΗ ΧΕΙΡΟΥΡΓΕΙΟΥ

👤 ΛΙΘΩΝΕΙΣ
👨 ΙΑΤΡΟΙ
👨 ΑΝΑΙΣΘΗΣΙΟΛΟΓΟΙ
📅 ΠΡΟΓΡΑΜΜΑ ΧΕΙΡΟΥΡΓΕΙΟΥ
📄 ΠΡΑΚΤΙΚΟ ΧΕΙΡΟΥΡΓΕΙΟΥ

📄 ΚΑΡΤΕΛΑ ΑΝΑΙΣΘΗΣΙΑΣ
📄 ΥΛΙΚΑ

ΥΛΙΚΑ

Εισάγετε υλικά που χρησιμοποιήθηκαν

Α.Μ	ΚΩΔΙΚΟΣ ΕΠΕΜΒΑΣΗΣ	ΕΙΔΟΣ ΥΛΙΚΟΥ	ΕΝΕΡΓΕΙΕΣ		
2	12	PTFE ΣΥΝΘΕΤΙΚΟ ΜΟΣΧΕΥΜΑ	Προβολή	Επέξεργασία	Διαγραφή
3	12	ΣΩΛΗΝΑΣ ΠΑΡΟΧΕΥΤΕΥΣΗΣ	Προβολή	Επέξεργασία	Διαγραφή
4	12	ΡΑΜΜΑΤΑ ΝΑΥΛΟΝ	Προβολή	Επέξεργασία	Διαγραφή

Το παρόν site έγινε για την εκπόνηση της Διπλωματικής Εργασίας στα πλαίσια του Μεταπτυχιακού Προγράμματος:
"Πληροφορική Υγείας".
Τα δεδομένα στους παραπάνω πίνακες δεν είναι πραγματικά.

Copyright © Maria Pringelou

Εικόνα 13: Ενότητα Υλικών

Κάθε καρτέλα υλικού κωδικοποιείται με μοναδικό αριθμό, περιλαμβάνει τον Α.Μ. της επέμβασης στο οποίο χρησιμοποιήθηκε κάθε υλικό, περιγραφή του υλικού, αριθμός τεμαχίων που χρησιμοποιήθηκαν, την εταιρεία προμήθειας του υλικού και τέλος τη τιμή του κάθε υλικού ανά τεμάχιο.

ΥΛΙΚΑ



ΚΩΔΙΚΟΣ ΥΛΙΚΟΥ

2

Α.Μ. ΕΠΕΜΒΑΣΗΣ

12

ΠΕΡΙΓΡΑΦΗ

ΡΤΓΕ ΣΥΝΘΕΤΙΚΟ ΜΟΣΧΕΥΜΑ

ΤΕΜΑΧΙΑ

1

ΕΤΑΙΡΕΙΑ

MEDICALS.AE

ΤΙΜΗ

1000,00

Επιστροφή

Εικόνα 13.1: Προβολή Καρτέλας Υλικών

ΣΥΜΠΕΡΑΣΜΑΤΑ

Καθ' όλη τη διάρκεια σχεδιασμού της συγκεκριμένης εφαρμογής έγινε σαφές πως δεν πρόκειται για μία απλουστευμένη διαδικασία. Για το σχεδιασμό και την εφαρμογή ενός Πληροφοριακού Συστήματος στο συγκεκριμένο χώρο πρέπει να ληφθεί υπόψιν η σωστή διαχείριση ευαίσθητων προσωπικών ιατρικών δεδομένων. Επιπλέον, πρέπει να ικανοποιούνται οι απαιτήσεις των χρηστών και να διευκολύνεται η λειτουργικότητα του συστήματος, ώστε να ελαχιστοποιείται τόσο ο χρόνος ολοκλήρωσης μία διαδικασίας όσο και το κόστος της. Όπως, συμπεραίνουμε ένα τέτοιο σύστημα έχει δυσκολίες εφαρμογής, καθώς πρέπει να σχεδιαστεί, ώστε να είναι πετυχημένο τόσο σε τεχνικό όσο και σε οργανωτικό επίπεδο. Όμως, καθώς η χρήση του αποφέρει σημαντικά οφέλη αξίζει η προσπάθεια υιοθέτησης Πληροφοριακών Συστημάτων στο τομέα της υγείας αρκεί να προσμετρώνται όλοι οι παράμετροι για την επιτυχία του.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. World, Health Organization. *Health Information Systems*. 2008.
2. Hospital Information Systems (HIS). [Ηλεκτρονικό] <http://www.emrconsultant.com/emr-education-center/emr-selection-and-implementation/hospital-information-systems-his/>.
3. Peter L. Reichertz. Health information systems – past, present, future. [Ηλεκτρονικό] 2006. <https://www.sciencedirect.com/science/article/pii/S1386505605001590>.
4. Bourke, Michael K. *Strategy and Architecture of Health Care Information Systems*. 1994.
5. Πληροφοριακά Συστήματα Διοίκησης σε Νοσοκομεία. Κορδάς, Χρήστος. 2012.
6. *An Overview of Surgery Information Systems*. Clarke, Lashan. 2011.
7. *Electronic Data Capture for Registries and Clinical Trials in Orthopaedic Surgery: Open Source versus Commercial Systems*. Shah, Jatin. 2010.
8. MajidRastegar-Mojarad et al. *Need of informatics in designing interoperable clinical registries*. 2017.
9. Sanders, Nathan R. Hoot and David L. *Implementation of OpChart in West*.
10. Laing , G.I. et al. *Development, Implementation, and Evaluation of a Hybrid Electronic Medical Record System Specifically Designed for a Developing World Surgical Service*. 2013.
11. Pulido, Raul et al. *Managing daily surgery schedules in a teaching hospital: a mixed-integer optimization approach*. 2014.
12. Κανονισμός Λειτουργίας Χειρουργείων, ΠΑΟΝΑ«Ο Άγιος Σάββας». [Ηλεκτρονικό] 2002.
http://www.onedaysurgery.net/uploads/9/0/4/3/9043588/kanonismos_liteourgias_xirourgiou_5-1.pdf.
13. Κίτσιου Γ. Σπύρος *Πληροφοριακά Συστήματα Ηλεκτρονικού Επιχειρείν: Μοντέλο Αξιολόγησης Βαθμού Υιοθέτησης και Εξέλιξης των Πληροφοριακών Συστημάτων και των νέων τεχνολογιών Πληροφορικής στο χώρο της Υγείας*. 2010.
14. ΧΑΜΡΡ. [Ηλεκτρονικό]
<https://globalhealthdatamanagement.tghn.org/community/blogs/post/5122/2013/07/xampp/>.
15. Xampp Apache + MariaDB + PHP + Perl. [Ηλεκτρονικό]
<https://www.apachefriends.org/index.html>.
16. HTML 5.2. [Ηλεκτρονικό] <https://www.w3.org/TR/html/introduction.html#background>.
17. HTML tutorial point. [Ηλεκτρονικό]
http://www.tutorialspoint.com/html/html_tutorial.pdf.
18. Μαθήματα CSS. [Ηλεκτρονικό] <http://www.wlearn.gr/index.php/css-84>.
19. PHP. [Ηλεκτρονικό] <http://www.techteam.gr/wiki/PHP>.
20. Php Manual. [Ηλεκτρονικό] <http://php.net/manual/en/index.php>.
21. What is MySQL? [Ηλεκτρονικό] <https://dev.mysql.com/doc/refman/5.5/en/what-is-mysql.html>.
22. About the Apache HTTP Server Project. [Ηλεκτρονικό]
http://httpd.apache.org/ABOUT_APACHE.html.
23. Στατιστικά Στοιχεία Διαδικτύου. [Ηλεκτρονικό] <https://www.udomo.gr/elleniko-statistika-stoicheia-diadiktyou>.
24. Bhaumik, Snig. *Bootstrap Essentials*. 2015.
25. Tutorial, Bootstrap. [Ηλεκτρονικό] <http://wiki.lib.sun.ac.za/images/0/07/Bootstrap-tutorial.pdf>.
26. What is JavaScript? [Ηλεκτρονικό] <https://developer.mozilla.org/en->

US/docs/Learn/JavaScript/First_steps/What_is_JavaScript.
27. What is Javascript? [Ηλεκτρονικό]
http://media.wiley.com/product_data/excerpt/88/07645790/0764579088.pdf.
28. phpMyadmin-Εισαγωγή. [Ηλεκτρονικό]
<https://docs.phpmyadmin.net/el/latest/intro.html>.
29. Why you should be using PHP's PDO for Database Access? . [Ηλεκτρονικό]
<https://code.tutsplus.com/tutorials/why-you-should-be-using-phps-pdo-for-database-access--net-12059>.
30. Singleton Design Pattern. [Ηλεκτρονικό]
https://sourcemaking.com/design_patterns/singleton..
31. PDO. [Ηλεκτρονικό] <http://www.afterhoursprogramming.com/tutorial/PHP/PDO/>.
32. PHP 5 for Loops. [Ηλεκτρονικό]
https://www.w3schools.com/php/php_looping_for.asp.
33. What is Crud? [Ηλεκτρονικό] <https://www.codecademy.com/articles/what-is-crud>.
34. How TO - Filter/Search Table. [Ηλεκτρονικό]
https://www.w3schools.com/howto/howto_js_filter_table.asp.