



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ
ΚΡΗΤΗΣ**

**Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων**

Πτυχιακή εργασία

Κατανεμημένο Ιατρικό Πληροφοριακό Σύστημα



ΔΗΜΗΤΡΑ ΛΟΥΣΑ

Ηράκλειο – Μάιος 2010

Επόπτης Καθηγητής: Παπαδάκης Νικόλαος

Κατανεμημένο Ιατρικό Πληροφοριακό Σύστημα

Η πτυχιακή αυτή εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για την εκπλήρωση των υποχρεώσεων του προγράμματος σπουδών του Τμήματος Εφαρμοσμένης Πληροφορικής και Πολυμέσων του Τ.Ε.Ι. Κρήτης.

Ευχαριστίες

Με τη λήξη αυτού του ακαδημαϊκού κύκλου , θεωρώ χρέος μου να ευχαριστήσω όλους όσους στάθηκαν στο πλευρό μου και με στήριξαν κατά την διάρκεια της εκπόνησης της πτυχιακής εργασίας μου αλλά και καθ'όλη τη διάρκεια των σπουδών μου .

Πρώτα από όλους οφείλω ένα μεγάλο ευχαριστώ στην οικογένεια μου, τους γονείς και την αδερφή μου, για την ψυχολογική, ηθική και οικονομική συμπαράσταση που μου παρείχαν όλα αυτά τα χρόνια .

Θα ήθελα επίσης να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Νίκο Παπαδάκη για την εμπιστοσύνη και την βοήθειά του για την εκπόνηση αυτής της εργασίας .

Τέλος ένα ακόμα ευχαριστώ στα μέλη της εξεταστικής επιτροπής που μου έκαναν την τιμή να αξιολογήσουν την προσπάθειά μου .

Λούσα Δήμητρα
Ηράκλειο , Μάιος 2010

Περίληψη

Η ραγδαία εξέλιξη της τεχνολογίας συνδυασμένη με την ανάγκη για εύκολη πρόσβαση στην πληροφορία οδήγησε στη δημιουργία Πληροφοριακών Συστημάτων. Τα Πληροφοριακά Συστήματα βρίσκουν ολοένα και αυξανόμενες εφαρμογές στο χώρο της υγείας. Μια από αυτές αποτελεί ο Ηλεκτρονικός Φάκελος Ασθενή (Η.Φ.Α), ο οποίος επεκτείνει τον παραδοσιακό ιατρικό φάκελο σε δυνατότητες και λειτουργίες.

Η παρούσα εργασία ασχολείται με το πεδίο αυτό και συγκεκριμένα με τη μελέτη, το σχεδιασμό και την υλοποίηση ενός Η.Φ.Α. Η εφαρμογή που υλοποιήθηκε είναι διαδικτυακή και βασίζεται σε σύγχρονες τεχνολογίες όπως είναι η PHP και η SQL. Περιλαμβάνει στοιχεία του ασθενούς σύμφωνα με την επισκεψιμότητά του και τα επεισόδια που έχουν δημιουργηθεί κατά την διάρκεια αυτών. Περιλαμβάνει επίσης όλα τα δημογραφικά στοιχεία του ασθενούς καθώς και τα ιατρικά στοιχεία που τον αφορούν σύμφωνα με το ατομικό ιστορικό του. Τέλος παρέχει στο ιατρικό προσωπικό την δυνατότητα άντλησης πληροφοριών στατιστικής φύσεως.

Με βάση τα παραπάνω η εφαρμογή που αναπτύχθηκε είναι φιλική προς το χρήστη και πληρεί όλες τις προϋποθέσεις ευχρηστίας και ασφάλειας προσωπικών δεδομένων. Η σημαντικότητα της εφαρμογής έγκειται στο γεγονός ότι τα στοιχεία αποθηκεύονται με ασφάλεια σε μια βάση δεδομένων και είναι ανά πάσα στιγμή στην διάθεση του Ιατρού. Επίσης δόθηκε μεγάλη έμφαση στη δημιουργία ενός εύχρηστου περιβάλλοντος διαχείρισης το οποίο μπορεί να χειριστεί ο χρήστης χωρίς να απαιτούνται ιδιαίτερες γνώσεις πληροφορικής.

Η εργασία αυτή είναι δομημένη σε 10 κεφάλαια. Σκοπός τους είναι να παρουσιάσουν το θεωρητικό υπόβαθρο με ένα τρόπο συνοπτικό όσο αφορά την ιατρική αλλά και την τεχνολογική πλευρά του θέματος. Έτσι, η εργασία αυτή αρχικά περιγράφει τον ρόλο των Πληροφοριακών Συστημάτων στο χώρο της Υγείας καθώς και τον ρόλο και την σημασία εφαρμογής του ηλεκτρονικού φακέλου κυρίως για τα Δημόσια Νοσοκομεία, όπου ο όγκος των πληροφοριών καθώς και ο αριθμός των ασθενών είναι τεράστιος. Επίσης γίνεται αναφορά στα πρότυπα και στις κωδικοποιήσεις που υπάρχουν για τον ηλεκτρονικό ιατρικό φάκελο, καθώς και ποιες τεχνολογίες διαδικτύου υπάρχουν για την υλοποίηση και εφαρμογή του. Τέλος θα αναπτύξουμε την σχεδίαση και την υλοποίηση ενός ΗΙΦ ενός ασθενούς που περιέχει όλα τα δεδομένα που σχετίζονται με αυτόν.

Λέξεις Κλειδιά : Πληροφοριακά Συστήματα Νοσοκομείων, Ηλεκτρονικός Ιατρικός Φάκελος, PHP, MySQL, HTML.

Πίνακας Περιεχομένων

Ευχαριστίες.....	iii
Περίληψη	iv
Πίνακας Περιεχομένων.....	v
Πίνακας Εικόνων	ix
Κεφάλαιο 1 Εισαγωγή	1
1.1 Γενικά.....	1
1.2 Σκοπός.....	2
1.3 Συνοπτική Περιγραφή.....	3
Κεφάλαιο 2 Πληροφοριακά Συστήματα.....	4
2.1 Στοιχεία Θεωρίας Συστημάτων	4
2.1.1 Βασικές έννοιες	4
2.1.2 Τύποι συστημάτων.....	5
2.1.3 Κριτήρια αξιολόγησης συστημάτων	5
2.2 Πληροφορικό Σύστημα.....	5
2.2.1.Ορισμός.....	5
2.2.2 Δραστηριότητες ενός Πληροφοριακού Συστήματος	7
2.2.3 Η επιρροή των πληροφοριακών συστημάτων στους οργανισμούς	8
2.2.4 Προβλήματα στην ανάπτυξη πληροφοριακού συστήματος	8
Κεφάλαιο 3.....	9
Πληροφοριακά Συστήματα Νοσοκομείων.....	9
3.1 Πληροφοριακά Συστήματα Νοσοκομείων.....	9
3.1.1 Αναγκαιότητα εισαγωγής ενός Πληροφοριακού Συστήματος Νοσοκομείου..	9
3.2 Πληροφοριακά Υποσυστήματα	10
3.2.1 Διαχειριστικό Πληροφοριακό Σύστημα Νοσοκομείου (ΔΠΣΝ).....	10
3.2.2 Ιατρικό Πληροφοριακό Σύστημα Νοσοκομείου (ΙΠΣΝ)	11
3.2.3 Πληροφοριακά συστήματα εργαστηρίων (ΠΣΕ)	12
3.2.4 Πληροφορικά Συστήματα Διοίκησης στο Νοσοκομείο(ΠΣΔ).....	12
3.3 Αρχιτεκτονικές υλοποίησης υπολογιστικού συστήματος Νοσοκομείου.....	13
3.3.1 Το κεντρικό μοντέλο	13
3.3.2 Αρθρωτά συστήματα	13
3.3.3 Καταναμημένα συστήματα.....	14
3.4 Χαρακτηριστικά καταναμημένων συστημάτων	15
3.5 Ολοκλήρωση Υποσυστημάτων–Ολοκληρωμένο Πληροφοριακό Σύστημα	16
3.6 Βασικές Προϋποθέσεις Εγκατάστασης και Λειτουργίας Ο.Π.Σ.Ν.....	17
3.6.1. Μακροχρόνιο Στρατηγικό Σχέδιο για την Πληροφορική στον Φορέα.....	17
3.6.2 Εξειδικευμένα Στελέχη- Ισχυρό τμήμα πληροφορικής	17
3.6.3 Βήμα – Βήμα προσέγγιση στην εγκατάσταση συστημάτων	18
3.6.4 Εκπαίδευση του προσωπικού.....	18
3.6.5 Τυποποίηση	19
Κεφάλαιο 4.....	20
Ηλεκτρονικός Ιατρικός Φάκελος Ασθενή.....	20
4.1 Εισαγωγή	20
4.2 Ιατρική Πληροφορία	20
4.3 Ο Ιατρικός Φάκελος.....	21

4.4 Ηλεκτρονικός Ιατρικός Φάκελος	23
4.5 Ιστορική Αναδρομή	23
4.6 Αναλυτικότερη περιγραφή του ΗΙΦ	24
4.7 Η δομή του ΗΙΦ	24
4.7.1 Στοιχειώδης Πληροφορία (ITEM)	24
4.7.2 Επαφή (CONTACT), Επεισόδιο (EPISODE)	25
4.8 Ιατρικές, Διαχειριστικές Πληροφορίες	25
4.9 Φάκελος Πολυμέσων (Multimedia)	25
4.9.1 Ακτινογραφίες	25
4.9.2 Καρδιογραφήματα	26
4.9.3 Άλλα Αντικείμενα (OBJECTS)	27
4.10 Επεξεργασία Δεδομένων	27
4.10.1 Ανάλυση Πλέγματος (Grid Analysis)	27
4.10.2 Ανάλυση Πληθυσμού	27
4.10.3 Ειδικά Εγγραφα	28
4.11 Βάση Φαρμάκων και Κωδικοποίηση	28
4.12 Σύστημα Υποβοήθησης Διάγνωσης	28
4.13 Τηλεϊατρική	29
4.14 Παράγοντες που καθορίζουν την εφαρμογή του ΗΦΥ	29
4.14.1 Παράγοντες του εσωτερικού περιβάλλοντος ή μικρο-περιβάλλον:	29
4.14.2 Παράγοντες του εξωτερικού περιβάλλοντος ή μακρο-περιβάλλον:	30
Κεφάλαιο 5 Πρότυπα και Κωδικοποιήσεις	32
5.1 Η Κωδικοποίηση της Ιατρικής Πληροφορίας	32
5.2 ISO/TC 215	32
5.3 CEN / TC 251	33
5.4 Το Πρότυπο Health Level Seven	33
5.5 Το πρότυπο DICOM	35
5.6 Το πρότυπο PACS	38
5.6.1 Πλεονεκτήματα των PACS	38
5.6.2 Μειονεκτήματα των PACS	38
Κεφάλαιο 6	39
Ασφάλεια Πληροφοριακών Συστημάτων	39
6.1 Εισαγωγή	39
6.2 Η Ασφάλεια Ιατρικών Δεδομένων	39
6.3 Απειλές Ασφάλειας	40
6.4 Μέθοδοι Αντιμέτωπισης Απειλών	41
6.5 Τεχνικές πρόσβασης και ανταλλαγής δεδομένων	41
Κεφάλαιο 7	45
Βάσεις Και Δίκτυα Μεταβίβασης Πληροφοριών	45
7.1 Βάσεις δεδομένων σε Πληροφοριακό Σύστημα	45
7.1.2 Σύστημα Διαχείρισης Βάσεων Δεδομένων	45
7.1.3 Επιπτώσεις από την χρήση Βάσεων Δεδομένων	46
7.1.4 Πλεονεκτήματα των βάσεων δεδομένων	47
7.1.5 Μειονεκτήματα των βάσεων δεδομένων	47
7.2 Κατηγορίες Δικτύων	48
7.3 Τοπολογίες Τοπικών Δικτύων	48
7.3.1 Φυσικές τοπολογίες	49
7.3.1.1 Τοπολογία Αρτηρίας	49
7.3.1.2 Τοπολογία αστέρα	49
7.3.1.3 Τοπολογία δακτυλίου	50

7.3.2 Λογικές τοπολογίες	51
7.3.2.1 Τοπολογία εκπομπής.....	51
7.3.2.2 Τοπολογία μεταφοράς σκυτάλης	52
Κεφάλαιο 8.....	53
Τεχνολογίες και Εργαλεία Που Χρησιμοποιήθηκαν.....	53
8.1 Εισαγωγή	53
8.2 Τι είναι η HTML.....	53
8.2.1 Συγκεντρωτικός πίνακας των Tags της HTML.....	53
8.3 Cascading Style Sheets (CSS)	54
8.3.1 Πλεονεκτήματα CSS	55
8.4 JavaScript.....	55
8.4.1 Χρήσεις της JavaScript.....	56
8.5 Τι είναι η PHP.....	57
8.5.1 Πλεονεκτήματα της PHP	57
8.5.2 Τι μπορεί να κάνει η PHP;	57
8.5.3 Επικοινωνία της PHP με βάσεις δεδομένων	59
8.6 MySQL	61
8.6.1 Πλεονεκτήματα MySQL.....	61
8.6.2 Τρόπος λειτουργίας MySQL Βάσης Δεδομένων.....	62
8.7 Apache Web Server	63
8.8 EasyPHP	63
8.8.1 Εγκατάσταση Easy PHP.....	63
8.9 PhpMyAdmin.....	67
Κεφάλαιο 9 Υλοποίηση Βάσης Δεδομένων.....	68
9.1 Εισαγωγή στην ανάλυση απαιτήσεων	68
9.2 Ανάλυση απαιτήσεων της εφαρμογής	68
9.3 Διαγραμματική Απεικόνιση Βάσης Δεδομένων	69
9.4 Περιγραφή Βάσης της Εφαρμογής	70
9.5 Περιγραφή Πινάκων της Βάσης Δεδομένων	70
9.5.1 Ο πίνακας « patient ».....	70
9.5.2 Ο πίνακας « doctor ».....	71
9.5.3 Ο πίνακας « ergastiriakosdoctor »	72
9.5.4 Ο πίνακας « klinikosdoctor »	73
9.5.5 Ο πίνακας « istoriko ».....	73
9.5.6 Ο πίνακας « exetasi ».....	74
9.5.7 Ο πίνακας « kliniki »	74
9.5.8 Ο πίνακας « nosilia »	75
9.5.9 Ο πίνακας « pragmatopoiisiexet »	75
Κεφάλαιο 10 Παρουσίαση Εφαρμογής	77
10.1 Εισαγωγή	77
10.2 Είσοδος Ιατρού	77
10.3 Αρχική σελίδα.....	78
10.4 Προσθήκη ασθενή.....	79
10.5 Αναζήτηση Ασθενή	80
10.6 Προσθήκη Εξέτασης.....	81
10.7 Στατιστικά Ασθενειών	81
10.8 Στατιστικά Εξετάσεων	83
10.9 Στατιστικά κόστους Εξετάσεων.....	84
10.10 Στοιχεία Γιατρού.....	85
Εικόνα 44: Επεξεργασία στοιχείων Γιατρού	86

Κατανεμημένο Ιατρικό Πληροφοριακό Σύστημα

10.11 Επικοινωνία	86
Παράρτημα Α	88
Παράρτημα Β	91
Βιβλιογραφία	114

Πίνακας Εικόνων

Εικόνα 1: Σχηματική Αναπαράσταση ενός Συστήματος.....	4
Εικόνα 2: Στοιχεία που αποτελούν ένα Πληροφοριακό Σύστημα.....	6
Εικόνα 3: Οι συνιστώσες ενός Πληροφοριακού Συστήματος.....	6
Εικόνα 4: Ακτινογραφία ενσωματωμένη στον Η.Ι.Φ.....	26
Εικόνα 5: ΗΚΓ ενσωματωμένη στον Η.Ι.Φ.....	26
Εικόνα 6: Πρότυπο HL7.....	35
Εικόνα 7: Αρχές λειτουργίας DICOM σε δίκτυο.....	37
Εικόνα 8: Σύστημα κρυπτογράφησης.....	42
Εικόνα 9: Κρυπτογραφικό σύστημα μυστικού κλειδιού.....	43
Εικόνα 10: Κρυπτογραφικό σύστημα δημόσιου κλειδιού.....	43
Εικόνα 11: Σύστημα Βάσης Δεδομένων.....	46
Εικόνα 12: Τοπολογία αρτηρίας.....	49
Εικόνα 13: Τοπολογία αστέρα.....	50
Εικόνα 14: Τοπολογία δακτυλίου.....	51
Εικόνα 15: Εγκατάσταση EasyPHP.....	64
Εικόνα 16: Επιλογή αποδοχής των όρων χρήσης.....	64
Εικόνα 17: Επιλογή διαδρομής για την εγκατάσταση.....	65
Εικόνα 18: Επιλογή Instal.....	65
Εικόνα 19: EasyPHP.....	66
Εικόνα 20: Αρχική σελίδα του EasyPHP.....	66
Εικόνα 21: PhpMyAdmin – Πίνακας της Βάσης.....	70
Εικόνα 22: Πίνακας Patient.....	71
Εικόνα 23: Πίνακας doctor.....	72
Εικόνα 24: Πίνακας ergastiriakosdoctor.....	73
Εικόνα 25: Πίνακας klinikosdoctor.....	73
Εικόνα 26: Πίνακας istoriko.....	74
Εικόνα 27: Πίνακας exetasi.....	74
Εικόνα 28: Πίνακας kliniki.....	75
Εικόνα 29: Πίνακας nosilia.....	75
Εικόνα 30: Πίνακας pragmatopoiisiexet.....	76
Εικόνα 31: Είσοδος γιατρού στην εφαρμογή.....	77
Εικόνα 32: Αρχική σελίδα.....	78
Εικόνα 33: Προσθήκη Ασθενή.....	79
Εικόνα 34: Αναζήτηση Ασθενή.....	80
Εικόνα 35: Επεξεργασία στοιχείων Ασθενή.....	80
Εικόνα 36: Προσθήκη Εξέτασης.....	81
Εικόνα 37: Αναζήτηση περιστατικών ασθενειών.....	82
Εικόνα 38: Ημερολόγιο.....	82
Εικόνα 39: Αποτελέσματα αναζήτησης ασθενειών.....	83
Εικόνα 40: Αναζήτηση συνολικών περιστατικών.....	83
Εικόνα 41: Πίνακας συνολικών ασθενειών.....	84
Εικόνα 42: Αναζήτηση καταστάσεων συνολικού κόστους εξετάσεων.....	84
Εικόνα 43: Πίνακας συνολικού κόστους.....	85
Εικόνα 44: Επεξεργασία στοιχείων Γιατρού.....	86
Εικόνα 45: Φόρμα επικοινωνίας.....	87
Εικόνα 46: Επιβεβαίωση αποστολής.....	87

Κεφάλαιο 1 Εισαγωγή

1.1 Γενικά

Τα τελευταία χρόνια γίνεται ολοένα και πιο έντονη στο χώρο της υγείας η ανάγκη αναβάθμισης της ποιότητας των προσφερόμενων υπηρεσιών. Για το σκοπό αυτό σημαντική βοήθεια έρχονται να προσφέρουν οι τεχνολογίες τηλεπικοινωνιών και πληροφορικής. Έτσι, ολοένα και περισσότερα νοσοκομεία διεθνώς έχουν σε μεγάλο βαθμό υλοποιήσει Ολοκληρωμένα Πληροφοριακά Συστήματα για την κάλυψη των αναγκών διαχείρισης τόσο διαχειριστικών και οικονομικών, όσο και ιατρικών δεδομένων.

Σε ένα Ολοκληρωμένο Πληροφοριακό Σύστημα Νοσοκομείου η πορεία του ασθενούς καταγράφεται και παρακολουθείται πλήρως, από την είσοδό του στο νοσοκομείο μέχρι και μετά την έξοδο του από αυτό. Ο ασθενής αντιμετωπίζεται με ενιαίο τρόπο σε όλη την έκταση του νοσοκομείου και ανεξάρτητα από τις επαφές του με τις διάφορες υπηρεσίες του νοσοκομείου.

Είναι κοινή διαπίστωση ότι ο όγκος των πληροφοριών που σχετίζονται με την φροντίδα του ασθενούς έχει αυξηθεί κατά πολύ τα τελευταία χρόνια, πράγμα που σε μεγάλο βαθμό οφείλεται στην ενσωμάτωση αυξημένου αριθμού εργαστηριακών και παρακλινικών εξετάσεων στους φακέλους των ασθενών. Επιπλέον, τα διαχειριστικά καθήκοντα των γιατρών και των νοσηλευτών γίνονται διαρκώς περισσότερα, καθώς η πολυπλοκότητα των ιδρυμάτων παροχής υπηρεσιών υγείας αυξάνει.

Λόγω της αυξανόμενης ανάγκης για την παροχή καλύτερων υπηρεσιών υγείας με μικρότερο κόστος, οι σύγχρονοι οργανισμοί παροχής υπηρεσιών υγείας έχουν αναγνωρίσει την ανάγκη για αποτελεσματική διαχείριση ιατρικών πληροφοριών και έπαψαν να αρκούνται στη χρήση της πληροφορικής μόνο για τα λογιστήρια και τις διοικητικές τους ανάγκες. Έτσι εισήγαγαν την έννοια του Ηλεκτρονικού Φακέλου Ασθενή και την τήρηση των ιατρικών στοιχείων.

Τα λογισμικά Ηλεκτρονικού Ιατρικού Φακέλου ασθενών (Η.Φ.Α.), αποτελούν συστήματα διαχείρισης ιατρικών φακέλων που βασίζονται σε ηλεκτρονικούς υπολογιστές. Ως εκ τούτου, η αποθήκευση και ανάκληση των δεδομένων γίνεται γρήγορα και με ασφάλεια. Επιπλέον, καθίσταται δυνατή η επεξεργασία των δεδομένων και η άμεση μεταφορά τους με ηλεκτρονικά μέσα σε οποιαδήποτε απόσταση. Χάρη στους ηλεκτρονικούς υπολογιστές, το σύστημα καταγραφής των δεδομένων που σχετίζονται με τους ασθενείς γίνεται τώρα πιο αποτελεσματικό και εμπλουτίζεται εκμεταλλευόμενο τις δυνατότητες της νέας τεχνολογίας.

1.2 Σκοπός

Στόχος της παρούσας πτυχιακής εργασίας ήταν η ανάλυση, η σχεδίαση και η ανάπτυξη μιας διαδικτυακής εφαρμογής ηλεκτρονικού ιατρικού φακέλου ασθενούς για την αποθήκευση και επεξεργασία ιατρικών δεδομένων. Η ιστοσελίδα είναι δυναμική και για την υλοποίησή της χρησιμοποιήθηκε η γλώσσα προγραμματισμού PHP.

Η PHP είναι μια γλώσσα προγραμματισμού με αντικειμενοστραφή χαρακτήρα η οποία είναι ιδανική για την ανάπτυξη εφαρμογών για το περιβάλλον του διαδικτύου. Πέρα από την PHP χρησιμοποιήθηκαν και άλλες τεχνολογίες για την ανάπτυξη της εφαρμογής όπως η Γλώσσα Σήμανσης Υπερκειμένου HTML, τα Cascading Stylesheets (CSS) και η JavaScript.

Η εφαρμογή δίνει την δυνατότητα εκτέλεσης των παρακάτω :

- Προσθήκη δημογραφικών στοιχείων ασθενή .
- Επεξεργασία στοιχείων ασθενή .
- Διαγραφή στοιχείων ασθενή.
- Αναζήτηση συγκεκριμένου ασθενή .
- Εμφάνιση Ιστορικού ασθενή .
- Εισαγωγή στοιχείων εξέτασης ασθενή.
- Συγκεντρωτικές καταστάσεις ασθενειών .
- Συγκεντρωτικές καταστάσεις εξετάσεων .
- Συγκεντρωτικές καταστάσεις κόστους .
- Επεξεργασία στοιχείων γιατρού .
- Επικοινωνία γιατρού με τον διαχειριστή.

Πέρα από την υλοποίηση της εφαρμογής με την χρήση των προαναφερόμενων τεχνολογιών, καθορίστηκαν οι απαιτήσεις δεδομένων για τη σχεδίαση και δημιουργία μιας βάσης δεδομένων σε εξυπηρετητή MySQL, η οποία θα εξυπηρετήσει στην αποθήκευση της σχετικής πληροφορίας . Τέλος χρησιμοποιήθηκε ένας εξυπηρετητής ιστού (Web Server) Apache, για τη συνεργασία των υπηρεσιών με τη βάση δεδομένων πάνω από το διαδίκτυο .

1.3 Συνοπτική Περιγραφή

Προχωρώντας σε μία σύντομη επισκόπηση του παρόντος εγγράφου :

Στο **2^ο κεφάλαιο** γίνεται η εισαγωγή στα Πληροφοριακά Συστήματα περιγράφοντας τις συνιστώσες και τις δραστηριότητες ενός τέτοιου συστήματος, καθώς επίσης και την επιρροή και τα τυχόν προβλήματα που υπάρχουν.

Στο **3^ο κεφάλαιο** παρουσιάζεται η αναγκαιότητα εισαγωγής Πληροφοριακού Συστήματος στα Νοσοκομεία. Περιγράφονται οι λειτουργίες, τα υποσυστήματα που το απαρτίζουν, οι αρχιτεκτονικές υλοποιήσεις, καθώς και οι προϋποθέσεις εγκατάστασης – λειτουργίας, αποτιμώντας την επίδραση του στις λειτουργίες του οργανισμού.

Στο **4^ο κεφάλαιο** προσδιορίζεται η έννοια και τα κύρια στοιχεία του ηλεκτρονικού ιατρικού φακέλου. Περιγράφεται η δομή του και αναφέρονται οι παράγοντες που καθορίζουν την εφαρμογή του ΗΦΥ.

Στο **5^ο κεφάλαιο** δίνουμε περιληπτικά μια εικόνα για τα πρότυπα και τις κωδικοποιήσεις που υπάρχουν για διάφορους οργανισμούς , τόσο εθνικούς όσο και παγκόσμιους .(ISO/TC215, HL7,DICOM,PACS)

Στο **6^ο κεφάλαιο** γίνεται μια εισαγωγή στην ασφάλεια των Ιατρικών δεδομένων και στις μεθόδους αντιμετώπισης απειλών, καθώς επίσης αναλύονται οι τεχνικές πρόσβασης και ανταλλαγής δεδομένων.

Στο **7^ο κεφάλαιο** περιγράφεται το απαραίτητο τεχνολογικό περιβάλλον ανάπτυξης των Πληροφοριακών Συστημάτων Νοσοκομείων με επίκεντρο τις Βάσεις Δεδομένων και τα Δίκτυα Υπολογιστών .

Στο **8^ο κεφάλαιο** αναπτύσσεται η τεχνολογία που χρησιμοποιήθηκε για την υλοποίηση της εφαρμογής .

Στο **9^ο κεφάλαιο** περιγράφεται η βάση δεδομένων και αναλύονται όλοι οι πίνακες της βάσης που χρησιμοποιούνται για τη λειτουργία του συστήματος. Επίσης, αναλύονται τα επιμέρους στοιχεία των πινάκων και δίνεται μία μικρή περιγραφή για τη λειτουργία τους.

Στο **10^ο κεφάλαιο** παρέχεται ένα εγχειρίδιο χρήσης της εφαρμογής που υλοποιήθηκε όπου με τη χρήση εικόνων γίνεται προσπάθεια να κατανοηθεί ο τρόπος λειτουργίας της και να εκμεταλλευτούν όλες οι δυνατότητες που παρέχονται από το σύστημα.

Τέλος, στα **παραρτήματα Α και Β** παρέχονται ο κώδικας sql δημιουργίας των πινάκων της βάσης δεδομένων του συστήματος καθώς και ο κώδικας html και php των αρχείων της εφαρμογής.

Κεφάλαιο 2 Πληροφοριακά Συστήματα

2.1 Στοιχεία Θεωρίας Συστημάτων

2.1.1 Βασικές έννοιες

Σύστημα (system) είναι ένα σύνολο από οντότητες (άνθρωποι, μηχανές, διαδικασίες κλπ) που συνεργάζονται για την επίτευξη ενός στόχου. Ο στόχος αυτός είναι ο λόγος ύπαρξης του συστήματος. Κάθε σύστημα έχει εισόδους, εξόδους και επεξεργασίες και περιβάλλεται από ένα περιβάλλον από το οποίο διαχωρίζεται από ένα όριο.



Εικόνα 1 : Σχηματική Αναπαράσταση ενός Συστήματος

Είσοδος, ή εισροές (input) είναι τα στοιχεία εκείνα τα οποία εισέρχονται στο σύστημα.

Επεξεργασίες (process) είναι τα απαραίτητα στοιχεία για το μετασχηματισμό των εισόδων σε εξόδους.

Έξοδος, ή εκροές (output) είναι τα προϊόντα που παράγει το σύστημα.

Τα στοιχεία ενός συστήματος διαχωρίζονται από το **Περιβάλλον** (environment) που αποτελείται από οντότητες που δεν ανήκουν στο σύστημα, δηλαδή δεν είναι εισοδοί, έξοδοι ή επεξεργασίες του, αλλά παίζουν σημαντικό ρόλο στην απόδοση του συστήματος.

Η **ανάδραση** ή ανατροφοδότηση (feedback) είναι πληροφορία που αφορά την απόδοση του συστήματος. Η συνεχής παρακολούθηση και αξιολόγηση της ανάδρασης για να προσδιοριστεί εάν το σύστημα βαίνει προς ολοκλήρωση των στόχων του αποτελεί μέρος του **ελέγχου** (control). Όταν διαπιστώνεται ότι το σύστημα δεν ικανοποιεί επαρκώς τους στόχους πρέπει να βρεθεί τρόπος ώστε να επηρεαστεί η συμπεριφορά του. Αυτό επιτυγχάνεται με την τροποποίηση των εισόδων ή και των διαδικασιών του συστήματος.

2.1.2 Τύποι συστημάτων

Ανοικτό (open system) λέγεται το σύστημα το οποίο αλληλεπιδρά με το περιβάλλον του δεχόμενο εισόδους από αυτό και αποδίδοντας σε αυτό τις παραγόμενες εξόδους, ενώ κλειστό (closed system) λέγεται το σύστημα το οποίο δεν έχει καμία αλληλεπίδραση με το περιβάλλον του. Στον πραγματικό κόσμο βέβαια δεν υπάρχει κλειστό σύστημα. Η έννοια του κλειστού συστήματος είναι μία επινόηση για την διευκόλυνση της μελέτης των συστημάτων. Τα συστήματα τα οποία έχουν την δυνατότητα να αλλάζουν ώστε να επιβιώσουν ονομάζονται προσαρμοσμένα.

2.1.3 Κριτήρια αξιολόγησης συστημάτων

Τα συστήματα αξιολογούνται βάσει δύο κριτηρίων: της παραγωγικότητας και της αποτελεσματικότητας. Η **παραγωγικότητα** ή **αποδοτικότητα** (efficiency) είναι οι εισοδοί που χρησιμοποιήθηκαν για την επίτευξη των στόχων .

Αποτελεσματικότητα (effectiveness) από την άλλη μεριά, είναι ο βαθμός επίτευξης των σωστών στόχων.

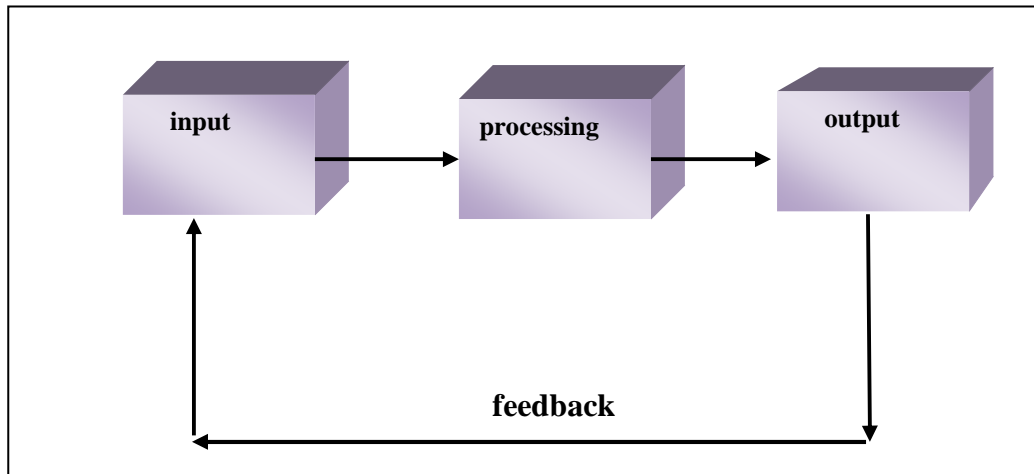
Η Γενική Θεωρία Συστημάτων είναι ένα επιστημονικό πεδίο που ασχολείται με την ανάλυση, τον σχεδιασμό και την βελτίωση των συστημάτων και συνδυάζει πολλούς άλλους τομείς επιστημών σε μία καθολική μελέτη των συστημάτων.

Θα πρέπει να τονιστεί ότι κάθε σύστημα είναι ένα εννοιολογικό πλαίσιο που ορίζεται από έναν άνθρωπο (παρατηρητή) - δεν υπάρχει από μόνο του. Ο παρατηρητής θεωρεί κάτι ως σύστημα για κάποιο λόγο και ο λόγος αυτός καθορίζει τον ορισμό του συστήματος. Επιπλέον, δύο παρατηρητές που μελετούν μία κατάσταση για τον ίδιο λόγο, μπορεί να ορίσουν το ίδιο σύστημα διαφορετικά. Αυτό γίνεται γιατί κάθε άτομο επηρεάζεται από τις γνώσεις του, την κοσμοθεωρία του, τις προτιμήσεις του κλπ .

2.2 Πληροφοριακό Σύστημα

2.2.1.Ορισμός

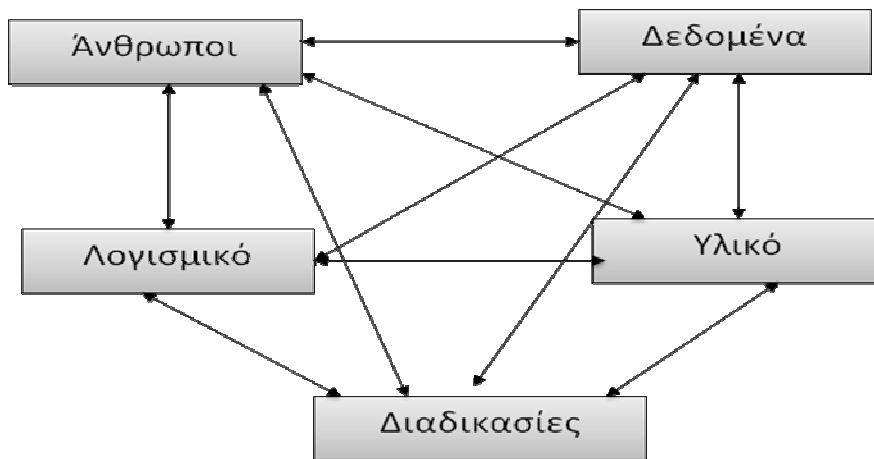
Πληροφοριακό Σύστημα (Π.Σ) (information system) είναι ένα σύνολο από αλληλεπιδρούσες συνιστώσες που δουλεύουν μαζί για την συλλογή , επεξεργασία , αποθήκευση και διανομή της πληροφορίας με τελικό στόχο την δημιουργία πληροφοριών που είναι αναγκαίες ή / και χρήσιμες στον Οργανισμό /Επιχείρηση για να επιτελέσει τον σκοπό του/της.Απλουστεύοντας , θα μπορούσαμε να πούμε ότι Πληροφοριακό Σύστημα είναι το σύστημα εκείνο που παίρνει σαν είσοδο δεδομένα τα οποία επεξεργάζεται και τα αποδίδει στην έξοδο ως πληροφορίες .Διαγραμματικά τα παραπάνω μπορούμε να τα αποδώσουμε με το παρακάτω σχήμα :



Εικόνα 2 : Στοιχεία που αποτελούν ένα Πληροφοριακό Σύστημα

Κάθε επιχείρηση/ οργανισμός έχει ένα πληροφοριακό σύστημα που, ειδικότερα, αποτελείται από τις εξής κυρίως συνιστώσες:

- ✓ Άνθρωποι
- ✓ Διαδικασίες
- ✓ Λογισμικό
- ✓ Δεδομένα
- ✓ Υλικό



Εικόνα 3 : Οι συνιστώσες ενός Πληροφοριακού συστήματος

α) Άνθρωποι

Οι άνθρωποι ενός Πληροφοριακού Συστήματος θα μπορούσαν να ταξινομηθούν σε δυο κατηγορίες.

- στους χρήστες (users) και
- στους χειριστές (operators) του συστήματος

Στην πρώτη κατηγορία ανήκουν οι κυρίως χρήστες , οι προϊστάμενοί τους και ο ιδιοκτήτης του συστήματος , ενώ στην δεύτερη κατηγορία ανήκουν οι χειριστές των Η/Υ ,δηλαδή όσοι εισάγουν στοιχεία και όσοι συντηρούν το υλικό και/ή το λογισμικό .

β) Διαδικασίες

Διαδικασία είναι μια σειρά από οδηγίες , οι οποίες καθορίζουν τον τρόπο με τον οποίο θα ενεργήσουν οι άνθρωποι σε συγκεκριμένες περιστάσεις και απευθύνονται στους ανθρώπους που συμμετέχουν στο σύστημα . Επιγραμματικά μπορούμε να πούμε ότι μια διαδικασία :

- Υποστηρίζει ανθρώπινες δραστηριότητες
- Εξασφαλίζει τη πληροφορία που έχει ο συγκεκριμένος άνθρωπος τη συγκεκριμένη χρονική στιγμή
- Δίνει τον τρόπο μετασχηματισμού της πληροφορίας

γ) Λογισμικό

Το λογισμικό ενός Πληροφοριακού Συστήματος μπορούμε να το διακρίνουμε στις παρακάτω κατηγορίες:

- **Το λογισμικό του συστήματος** (System Software) όπως τα Λειτουργικά συστήματα (π.χ. διάφορα είδη των Windows, UNIX) και οι Μεταγλωττιστές (π.χ. COBOL, CLIPPER).
- **Το λογισμικό εφαρμογών** (Application Software) το οποίο συνήθως αφορά συγκεκριμένο εργασιακό χώρο ή εργασιακούς χώρους (π.χ. έλεγχος αποθεμάτων, έκδοση μισθοδοσίας, παρακολούθηση προμηθειών και παρακολούθηση ασθενών).
- **Το λογισμικό για την αύξηση της παραγωγικότητας** (Productivity Software) όπως, επεξεργαστές πινάκων (π.χ. EXCEL), επεξεργαστές κειμένου (WORD), εργαλεία παρουσίασης (π.χ. POWERPOINT) και συστήματα διαχείρισης βάσεων δεδομένων (π.χ. ORACLE).

δ) Δεδομένα

Ο όρος δεδομένα σημαίνει μια παράσταση γεγονότων, εννοιών ή εντολών κατά τέτοιο τρόπο που να είναι σε μορφή κατάλληλη για επικοινωνία , ερμηνεία ή επεξεργασία από άνθρωπο ή από αυτοματοποιημένο μηχάνημα (π.χ. Η/Υ).

ε) Υλικό

Η συνιστώσα αυτή είναι όλος ο εξοπλισμός των υπολογιστών του πληροφοριακού συστήματος. Σε αυτόν συμπεριλαμβάνονται οι περιφερειακές συσκευές καθώς και ο πιθανός δικτυακός εξοπλισμός .

2.2.2 Δραστηριότητες ενός Πληροφοριακού Συστήματος

1. Συλλογή δεδομένων

Τα δεδομένα συλλέγονται από διάφορες πηγές:

- από εσωτερικές πηγές (internal sources)

- από εξωτερικές πηγές (external sources)
- από το περιβάλλον

2. Αποθήκευση δεδομένων

- Με την αποθήκευση τα δεδομένα φυλάσσονται με έναν οργανωμένο τρόπο για μελλοντική χρήση.

3. Επεξεργασία δεδομένων

- Η επεξεργασία των δεδομένων περιλαμβάνει υπολογισμούς, συγκρίσεις, ταξινομήσεις και κατηγοριοποιήσεις.

4. Διάδοση πληροφοριών

Ο στόχος ενός Π.Σ είναι η διάδοση πληροφοριών. Η πληροφορία μπορεί να διαδοθεί σε διάφορες μορφές (μηνύματα, φόρμες, αναφορές, λίστες, γραφήματα, κλπ).

2.2.3 Η επιρροή των πληροφοριακών συστημάτων στους οργανισμούς

Τα πληροφοριακά συστήματα ασκούν επίδραση στην οργάνωση και τη λειτουργία των οργανισμών. Πρώτον, από οικονομικής πλευράς λόγω της σταδιακής αντικατάστασης χειρογραφικών εργασιών με αυτοματοποιημένες αλλά και τη φθηνότερη απόκτηση και διαχείριση δεδομένων και πληροφοριών.

Δεύτερον, από πλευράς αλλαγής του τρόπου λήψης αποφάσεων μέσα στον οργανισμό με τη βοήθεια πληροφοριακού συστήματος. Οι σύγχρονες τάσεις στην ανάπτυξη πληροφοριακού συστήματος υπό την πίεση των τεχνολογικών εξελίξεων στους διάφορους τομείς και ιδίως η εμφάνιση του Διαδικτύου, του παγκόσμιου ιστού, ήδη επηρεάζει τον τρόπο με τον οποίο οργανισμοί και επιχειρήσεις δρουν μέσα στην αγορά και τη διοίκηση και τις οδηγούν σε ορισμένες αναθεωρήσεις και προσαρμογές.

2.2.4 Προβλήματα στην ανάπτυξη πληροφοριακού συστήματος

Είναι κοινή διαπίστωση ότι η ανάπτυξη ενός αυτοματοποιημένου πληροφοριακού συστήματος, ανεξάρτητα από το μέγεθος και τη συμπλοκότητά του είναι ένα έργο δύσκολο. Τα συνήθη προβλήματα που παρουσιάζονται είναι τα εξής :

- Οι χρήστες κάνουν παράπονα ότι αναπτύσσονται συστήματα που δε βοηθούν γιατί είναι πολύπλοκα και παράγουν πληροφορίες άχρηστες.
- Είναι δύσκολος ο καθορισμός των πραγματικών απαιτήσεων του χρήστη.
- Παρουσιάζει δυσκολίες ο προσδιορισμός ενός συστήματος που να ικανοποιεί τις απαιτήσεις των χρηστών.
- Η ανάπτυξη του λογισμικού θέλει μεγάλη προσοχή γιατί δύσκολα γίνονται αλλαγές στο software κι αν γίνονται, κοστίζουν.

Κεφάλαιο 3

Πληροφοριακά Συστήματα Νοσοκομείων

3.1 Πληροφοριακά Συστήματα Νοσοκομείων

Πληροφοριακό Σύστημα Νοσοκομείου (ΠΣΝ) είναι εκείνο το υπολογιστικό σύστημα ,το οποίο φροντίζει για την συνύπαρξη και την επικοινωνία της εξωτερικής και της εσωτερικής ροής των πληροφοριών σε ένα Νοσοκομείο ,καθώς και για τον κοινό τρόπο λειτουργίας στις εφαρμογές μέσα στο Νοσοκομείο .

3.1.1 Αναγκαιότητα εισαγωγής ενός Πληροφοριακού Συστήματος Νοσοκομείου.

Οι λόγοι που καθιστούν αναγκαία στις μέρες μας, την εισαγωγή πληροφοριακού συστήματος στα σύγχρονα Νοσοκομεία, απορρέουν από τη γενικότερη ανάγκη βελτίωσης τόσο του τρόπου λειτουργίας τους, όσο και των παρεχόμενων υπηρεσιών υγείας. Οι βασικοί επιμέρους στόχοι που θα πρέπει να ικανοποιηθούν για το σκοπό αυτό είναι :

Η γενικότερη αναβάθμιση των υπηρεσιών του Νοσοκομείου (βελτίωση της ποιότητας περίθαλψης και εξυπηρέτησης των ασθενών). Ο στόχος αυτός μπορεί να επιτευχθεί με:

- Την εισαγωγή και τη διαχείριση ηλεκτρονικού φακέλου ασθενούς, που θα συγκεντρώνει και θα παρουσιάζει κατάλληλα όλα τα στοιχεία που αφορούν στους κρίσιμους παράγοντες περίθαλψης, την πορεία της πάθησης κλπ.
- Το συσχετισμό των παραπάνω στοιχείων σύμφωνα με τους κανόνες της ιατρικής επιστήμης, ώστε να εξυπηρετούνται οι ιατροί στη λήψη αποφάσεων σχετικών με την προτεινόμενη αγωγή.
- Την παροχή δυνατότητας πρόσβασης σε παλαιότερα στοιχεία περίθαλψης (στο ίδιο ή /και σε άλλο νοσηλευτικό ίδρυμα), ώστε να είναι δυνατή η άμεση αναδρομή στο ιστορικό του ασθενούς.
- Τη μείωση της γραφειοκρατίας.
- Τη βελτίωση της πληροφόρησης των συναλλασσομένων και της ταχύτητας εξυπηρέτησής τους.
- Την ελαχιστοποίηση των λαθών.

Ο περιορισμός των χειρόγραφων διαδικασιών και η βελτίωση του εργασιακού περιβάλλοντος. Ο στόχος αυτός μπορεί να επιτευχθεί με:

- Την αυτοματοποίηση των διαδικασιών.
- Την διασύνδεση και την ολοκλήρωση των επί μέρους συστημάτων σε ένα πλήρες σύστημα.
- Την αναβάθμιση του εσωτερικού εργασιακού περιβάλλοντος.
- Την εξασφάλιση αποτελεσματικότητας στη διεκπεραίωση καθημερινών εργασιών.
- Τη διαχείριση και αξιοποίηση του ανθρωπίνου δυναμικού.
- Την αξιοποίηση σύγχρονων τεχνολογιών πληροφορικής.

Η ελαχιστοποίηση του κόστους παροχής περίθαλψης. Ο στόχος αυτός μπορεί να επιτευχθεί με:

- Την ορθολογική διαχείριση των πόρων του Νοσηλευτικού ιδρύματος (έλεγχος ανάλωσης υλικού, προγραμματισμός διαδικασιών, αυτοματοποίηση ελέγχων, κ.λ.π.).
- Την αποφυγή άσκοπων ιατρικών πράξεων (π.χ. αποφυγή επανάληψης εξετάσεων).

Η παροχή ικανών και αξιόπιστων πληροφοριών στη διοίκηση του Νοσοκομείου. Η πληροφόρηση αυτή μπορεί να περιλαμβάνει τόσο διαχειριστικά, όσο και επιστημονικά στοιχεία. Χαρακτηριστικά αναφέρονται:

- Η πληρότητα θαλάμων, ο μέσος χρόνος νοσηλείας κ.λ.π.
- Η παρακολούθηση των ποσοτικών και οικονομικών δεικτών τόσο ανά κατηγορία, όσο και ανά κέντρο κόστους.
- Το κόστος νοσηλείας ανά διάγνωση ή ομάδα διαγνώσεων.
- Τα ποσοστά αποθεραπείας ανά διάγνωση ή ομάδα διαγνώσεων.

Η δημιουργία ενός εύελικτου εργαλείου υποστήριξης στη λήψη αποφάσεων για τον καθορισμό και τον έλεγχο των διαφορετικών πολιτικών οργάνωσης της παροχής υγείας, κοστολόγησης και τιμολόγησης των υπηρεσιών της.

3.2 Πληροφοριακά Υποσυστήματα

Τα πληροφοριακά υποσυστήματα που συγκροτούν ένα Ολοκληρωμένο Πληροφοριακό Σύστημα Νοσοκομείου (ΟΠΣΝ) διακρίνονται στα ακόλουθα υποσυστήματα : το διαχειριστικό / οικονομικό , το ιατρικό , το εργαστηριακό και της διοίκησης .

3.2.1 Διαχειριστικό Πληροφοριακό Σύστημα Νοσοκομείου (ΔΠΣΝ)

Το ΔΠΣΝ καλύπτει τις λειτουργίες και παρακολουθεί τις διαδικασίες της διαχειριστικής και της οικονομικής οργάνωσης του νοσοκομείου .

Οι εφαρμογές διαχειριστικού χαρακτήρα συνήθως περιλαμβάνουν :

- Διαχείριση Ασθενών.
- Νοσηλευομένων (Γραφείο Κίνησης)
- Εξωτερικών Ασθενών (Γραμματεία Εξωτερικών Ιατρείων)
- Επείγοντων περιστατικών (Τμήμα Επείγοντων Περιστατικών)
- Διαχείριση προσωπικού
- Διαχείριση υλικών
- Διαχείριση προμηθειών
- Διαχείριση εγκαταστάσεων
- Τιμολόγηση παρεχόμενων Υπηρεσιών(νοσηλείας, ιατρικών πράξεων, εργαστηριακών εξετάσεων, χρήσης υλικών και φαρμάκων)

Οι εφαρμογές οικονομικού χαρακτήρα συνήθως περιλαμβάνουν :

- Γενική Λογιστική
- Αναλυτική Λογιστική
- Ταμειακό Προγραμματισμό
- Προϋπολογισμό
- Λογιστήριο ασθενών
- Εκκαθάριση ασφαλιστικών ταμείων
- Διαχείριση παραμέτρων νοσηλίων
- Εισπράξεις / Πληρωμές
- Διαχείριση παγίων
- Μισθοδοσία προσωπικού

3.2.2 Ιατρικό Πληροφοριακό Σύστημα Νοσοκομείου (ΙΠΣΝ)

Το ΙΠΣΝ καλύπτει τις ανάγκες διεκπεραίωσης των εργασιών που επιτελούνται στα κλινικά τμήματα του νοσοκομείου . Οι εφαρμογές του ΙΠΣΝ μπορούν να διακριθούν περαιτέρω σε :

- Εφαρμογές παροχής ιατρικής φροντίδας (ΙΦ)
- Εφαρμογές παροχής νοσηλευτικής φροντίδας (ΝΦ)

Οι εφαρμογές παροχής ιατρικής φροντίδας , υποστηρίζουν το κλινικό τμήμα στην υλοποίηση της καθαρά ιατρικής φροντίδας που παρέχεται στον ασθενή κατά την διάρκεια της νοσηλείας του και περιλαμβάνουν :

- Διαχείριση ασθενή (εισαγωγή, έξοδος, μετακίνηση ασθενή).
- Διαχείριση ιστορικού.
- Παρακολούθηση και καταγραφή υγείας (συμπτώματα ασθενή, κλινικά σημεία, διαγνώσεις, πορεία νόσου κλπ).
- Διαχείριση ιατρικών εντολών και παρουσίαση αποτελεσμάτων.

Ανάλογα με την ιατρική εξειδίκευση του κλινικού τμήματος (Καρδιολογικό, Χειρουργικό, Νεφρολογικό, Ογκολογικό, κ.λ.π.) υπάρχουν πρόσθετες απαιτήσεις πληροφοριακής υποστήριξης, οι οποίες ενσωματώνονται στις λειτουργίες του υποσυστήματος ιατρικής φροντίδας.

Παράλληλα, στις παραπάνω εφαρμογές λογισμικού εντάσσεται και ένα σύνολο από απαραίτητες υποστηρικτικές εφαρμογές, όπως:

- Προγραμματισμός ιατρικού και νοσηλευτικού προσωπικού
- Διαχείριση τακτικών εξωτερικών ιατρείων
- Νοσοκομειακό φαρμακείο
- Διαχείριση τακτικών εξωτερικών ιατρείων
- Προγραμματισμός χειρουργείων
- Διαιτολογικό

Οι εφαρμογές νοσηλευτικής φροντίδας υποστηρίζουν το νοσηλευτικό προσωπικό στη διαχείριση του νοσηλευτικού τους έργου και περιλαμβάνουν τις παρακάτω ενότητες :

- Σχεδιασμός Νοσηλευτικής Φροντίδας
- Νοσηλευτική παρακολούθηση

- Νοσηλευτικές ενέργειες και πράξεις
- Φαρμακολογική παρακολούθηση ασθενή

3.2.3 Πληροφοριακά συστήματα εργαστηρίων (ΠΣΕ)

Τα εργαστηριακά πληροφοριακά συστήματα (Laboratory Information Systems-LIS) είναι λογισμικό εγκατεστημένο σε ηλεκτρονικό υπολογιστή, ο οποίος είναι συνδεδεμένος με τον κατάλληλο ιατρικό εξοπλισμό. Είναι υπεύθυνα για την αποθήκευση κλινικών δεδομένων, την επαλήθευση της ακρίβειας των εξετάσεων, τη βαθμονόμηση των οργάνων, τη δημιουργία ή ενημέρωση αρχείων ασθενών, τη συλλογή πληροφοριών από ένα πλήθος συσκευών όπως συσκευές ανάλυσης αίματος.

Οι ιατρικές συσκευές που πραγματοποιούν τις διάφορες μετρήσεις ονομάζονται εργαστηριακοί αναλυτές και διαθέτουν μικροεπεξεργαστές, που ελέγχουν και συντονίζουν τη σωστή λειτουργία των συσκευών. Ο χρήστης μπορεί να μεταφέρει την ίδια στιγμή ηλεκτρονικά στο εργαστηριακό πληροφοριακό σύστημα τις μετρήσεις από τις συσκευές. Οι χρησιμοποιούμενοι εργαστηριακοί αναλυτές διασυνδέονται στο όλο σύστημα μέσω ειδικών διατάξεων, που συνδέονται σε Η/Υ και το σύστημα, έτσι, αποτελεί ενιαίο κορμό παραγωγής.

Ένα ιδανικό ολοκληρωμένο πληροφοριακό σύστημα εργαστηρίων για να είναι καταξιωμένο στον ιατρικό χώρο χρειάζεται να είναι προσαρμοσμένο στις ανάγκες και τις ιδιαιτερότητες των εργαστηρίων κάθε νοσοκομείου ή διαγνωστικού κέντρου. Γενικά χαρακτηριστικά ενός πληροφοριακού εργαστηριακού συστήματος είναι:

- Μονόδρομη και αμφίδρομη επικοινωνία με πληθώρα αυτόματων αναλυτών
- Παραγγελία εργαστηριακών εξετάσεων σε πραγματικό χρόνο
- Έγκριση και ανάγνωση αποτελεσμάτων σε πραγματικό χρόνο
- Δυνατότητα σύνδεσης αποτελεσμάτων και διαγνώσεων
- Διαχείριση ποιότητας ιατρικών συσκευών
- Παρακολούθηση αναλώσιμων
- Στατιστική ανάλυση

Σ' ένα τέτοιο σύστημα το λογισμικό είναι δομημένο με τη συλλογιστική πολλών χρηστών, που ο καθένας έχει διαφορετικές αρμοδιότητες και προσβάσεις στις διακινούμενες πληροφορίες. Διαθέτει πλήρη παραμετροποίηση επιτρέποντας το διαχωρισμό του συνόλου των εργαστηρίων σε επί μέρους τμήματα, τον καθορισμό του προσωπικού του τμήματος όπως και τις εξετάσεις που πραγματοποιεί το κάθε τμήμα. Διαχειρίζεται το ιστορικό των εξετάσεων όλων των ασθενών παρακολουθώντας τις εξετάσεις ανά ασθενή, τμήμα εργαστηρίου, κλινική, ασφαλιστικό φορέα και προαιρετικά μπορεί να εκτελεί τιμολογήσεις και να παρακολουθεί όλα τα σχετικά οικονομικά στοιχεία.

3.2.4 Πληροφορικά Συστήματα Διοίκησης στο Νοσοκομείο(ΠΣΔ)

Η λήψη των αποφάσεων της διοίκησης του νοσοκομειακού οργανισμού, πρέπει να βασίζεται στη διαθεσιμότητα, στην ανάλυση και στην επεξεργασία δεδομένων και πληροφοριών που είναι δυνατόν να συγκεντρωθούν από τα επιμέρους τμήματα του οργανισμού. Το Πληροφοριακό Σύστημα Διοίκησης (Management

Information System, MIS) παρέχει την δυνατότητα της συγκέντρωσης στοιχείων από όλες τις δραστηριότητες του οργανισμού, ώστε μέσα από την κατάλληλη σύνθεσή τους να προκύψουν οι δείκτες εκείνοι που θα αξιολογήσουν τις δραστηριότητες αυτές και θα βοηθήσουν το διοικητικό μηχανισμό στη λήψη αποφάσεων. Το Π.Σ.Δ στο Νοσοκομείο αντλεί πληροφορίες από όλα τα πληροφοριακά υποσυστήματα του Νοσοκομείου και τις παρουσιάζει με κατανοητό και επεξεργάσιμο τρόπο στη διοίκηση. Τα δεδομένα, τα οποία και χρειάζεται κατά ελάχιστο ένα Π.Σ.Δ.Ν είναι ενδεικτικά τα εξής :

- Κοστολογικά δεδομένα.
- Δεδομένα προσωπικού και μισθολογικά δεδομένα.
- Ιατρικές πράξεις, στις οποίες υποβάλλονται οι ασθενείς.
- Διαγνώσεις.

3.3 Αρχιτεκτονικές υλοποίησης υπολογιστικού συστήματος Νοσοκομείου

Τα υπολογιστικά νοσοκομειακά συστήματα που έχουν αναπτυχθεί μέχρι σήμερα μπορούν να καταταχθούν σε τρεις εναλλακτικές αρχιτεκτονικές υλοποίησης : το κεντρικό, αρθρωτό και κατανεμημένο μοντέλο.

3.3.1 Το κεντρικό μοντέλο

Το κεντρικό μοντέλο (central model), χαρακτηρίζεται από μια αρχιτεκτονική βασισμένη σε έναν κεντρικό υπολογιστή (συνήθως mainframe), με τα τερματικά και τα άλλα περιφερειακά διατεταγμένα σε συνδεσμολογία αστέρα. Ο κεντρικός υπολογιστής επιτελεί όλη την απαιτούμενη διαχείριση πληροφορίας, συχνά με ενιαίο λογισμικό, χωρίς να υπάρχουν σημαντικές διαφοροποιήσεις μεταξύ διαφορετικών νοσοκομειακών τμημάτων.

Το κύριο πλεονέκτημα των κεντρικών συστημάτων είναι ο μεγάλος βαθμός ολοκλήρωσης του συστήματος, καθώς η συμβατότητα της πληροφορίας των διαφόρων τμημάτων είναι η μέγιστη δυνατή. Είναι όμως ακριβώς αυτό το πλεονέκτημα, το οποίο δημιουργεί ένα σημαντικό αρνητικό σημείο: τα κεντρικά συστήματα δεν διαθέτουν ευελιξία προσαρμογής στις ιδιαίτερες απαιτήσεις συγκεκριμένων νοσοκομειακών τμημάτων.

Συνεπακόλουθα, δεν διαθέτουν σημαντική δυνατότητα προσαρμογής σε νέες απαιτήσεις. Η αγορά και η συντήρηση ενός τόσο μεγάλου υπολογιστικού συστήματος αποτελεί επένδυση συχνά δυσβάσταχτη για αρκετά ιδρύματα, η δε εγκατάσταση του δημιουργεί αναστάτωση σε πολλά τμήματα του Νοσοκομείου ταυτόχρονα.

3.3.2 Αρθρωτά συστήματα

Στις αρχές της δεκαετίας του '80 λόγω της μείωσης του κόστους του υλικού και της προόδου που παρατηρήθηκε στον τομέα του λογισμικού, δημιουργήθηκαν τα αρθρωτά συστήματα.

Με τη χρήση αυτής της αρχιτεκτονικής κάθε τμήμα του Νοσοκομείου εξυπηρετείται από διαφορετικό υπολογιστικό σύστημα, στο οποίο και λειτουργεί εξειδικευμένη εφαρμογή λογισμικού σε συνεργασία με ένα κεντρικό σύστημα που εξυπηρετεί τις ανάγκες του πυρήνα (A.D.T). Η συμβατότητα μεταξύ των συνιστωσών πληροφοριακών υποσυστημάτων επιτυγχάνεται με χρήση καθορισμένων πλαισίων στον τρόπο ανταλλαγής των πληροφοριών. Με τη χρήση αυτής της αρχιτεκτονικής, υπολογιστικά συστήματα εγκαταστάθηκαν στην αρχή στις διοικητικές υπηρεσίες και στον τομέα των εργαστηρίων.

Στα πλεονεκτήματα της αρχιτεκτονικής αυτής συγκαταλέγονται: η αυξημένη προσαρμοστικότητα του συστήματος στις ανάγκες των επιμέρους νοσοκομειακών τμημάτων και η δυνατότητα προμήθειας υλικού και λογισμικού από διαφορετικούς οίκους. Η δαπάνη της επένδυσης, σε αντίθεση με την κεντρική προσέγγιση, μπορεί να γίνει προοδευτικά, συμβάλλοντας σε ένα πιο προσιτό σύστημα. Η ανταπόκριση στους χρήστες του συστήματος του είναι πιο άμεση, ενώ η εγκατάσταση και η αναβάθμιση του δεν δημιουργεί λειτουργικά προβλήματα στο Νοσοκομείο.

3.3.3 Κατανεμημένα συστήματα

Τα κατανεμημένα συστήματα (distributed systems) αναπτύχθηκαν τη δεκαετία του '80, ως αποτέλεσμα του μεγάλου βαθμού ανάπτυξης των δικτύων υπολογιστών και των κοινών πρωτοκόλλων επικοινωνιών και συστημάτων διαχείρισης βάσεων δεδομένων. Αποτελούνται από μια συλλογή ανεξάρτητων υπολογιστικών συστημάτων, καθένα από τα οποία εξυπηρετεί ένα διαφορετικό τμήμα του Νοσοκομείου. Τα κύρια συστήματα περιλαμβάνουν τον εξυπηρετητή, ο οποίος παρέχει λειτουργίες A.D.T, τους εξυπηρετητές των πτερύγων του Νοσοκομείου, καθώς και εξυπηρετητές ακτινολογικού τμήματος, κλινικών εργαστηρίων, οικονομικών υπηρεσιών κτλ,

Η κατανεμημένη προσέγγιση δεν απαιτεί κεντρικό υπολογιστή. Αντίθετα, όλοι οι εξυπηρετητές είναι συνδεδεμένοι σε ένα τοπικό δίκτυο υψηλού εύρους ζώνης (local area network). Τα τερματικά των χρηστών συνήθως είναι σταθμοί εργασίας με δυνατότητες επίδειξης πολυμέσων, αφού καλούνται να διαχειριστούν αντικείμενα προερχόμενα από διάφορους εξυπηρετητές (π.χ στο ίδιο σταθμό εργασίας να εμφανίζονται απεικονίσεις από το R.I.S, L.I.S, αλλά και στοιχεία από τον Ιατρικό Φάκελο).

Με τη χρήση ενός κατανεμημένου συστήματος επιτυγχάνεται η κάλυψη των ιδιαίτερων αναγκών των επιμέρους νοσοκομειακών τμημάτων. Η αρχιτεκτονική του συστήματος είναι "ανοικτή", επιτρέποντας τη σύνδεση στο τοπικό δίκτυο υπολογιστικών συστημάτων από διαφορετικές κατασκευάστριες εταιρείες, ανεξαρτήτως λειτουργικού συστήματος και λοιπού λογισμικού.

Καθίσταται φανερό ότι το κύριο πλεονέκτημα της αρχιτεκτονικής αυτής προέρχεται από τη σχεδόν απόλυτη ελευθερία επιλογής υλικού, λειτουργικού συστήματος και λογισμικού, η οποία επιτρέπει συχνές και οικονομικές αναβαθμίσεις κατά τμήμα του Νοσοκομείου. Η ταχύτητα ανταπόκρισης του συστήματος είναι αυξημένη.

Βασικό μειονέκτημα της προσέγγισης αυτής αποτελεί η δυσκολία ολοκλήρωσης των επιμέρους συστημάτων σε ένα λειτουργικό υπολογιστικό σύστημα, η οποία προϋποθέτει ανταλλαγή συμβατής πληροφορίας. Προϋπόθεση της ομαλής λειτουργίας ενός κατανεμημένου συστήματος αποτελεί η τήρηση μιας

προσυμφωνημένης κωδικοποίησης της πληροφορίας από τα ανεξάρτητα υπολογιστικά συστήματα.

3.4 Χαρακτηριστικά κατανεμημένων συστημάτων

Τα κύρια χαρακτηριστικά των κατανεμημένων συστημάτων είναι ο διαμοιρασμός πόρων (resource sharing) , η ευρύτητα (openness) , η κλιμάκωση (scalability) , ο συγχρονισμός (concurrency) , η διαφάνεια (transparency) και η ανοχή στα λάθη (fault tolerance) .

Αναλυτικά :

- **Διαμοιρασμός πόρων :** Οι πόροι που παρέχονται από έναν υπολογιστή , ο οποίος είναι μέλος ενός κατανεμημένου συστήματος , μπορούν να διαμοιραστούν και να αξιοποιηθούν από όλους τους υπόλοιπους υπολογιστές του συστήματος . Ο διαμοιρασμός των πόρων επιτυγχάνεται με τη χρήση του απαραίτητου δικτυακού εξοπλισμού . Ο αποτελεσματικός διαμοιρασμός των πόρων επιτυγχάνεται με τη χρήση λογισμικού το οποίο παρέχει τη δυνατότητα αξιοποίησης των πόρων ενός υπολογιστή από όλους τους υπόλοιπους . Το λογισμικό αυτό ονομάζεται διαχειριστής πόρων .
- **Ευρύτητα :** Η ευρύτητα ενός κατανεμημένου συστήματος είναι η ιδιότητα που επιτρέπει την επέκταση των δυνατοτήτων και των πόρων που παρέχονται χωρίς να δημιουργούνται προβλήματα στις παρεχόμενες υπηρεσίες ή να εμφανίζεται πλεονασμός των παρεχόμενων πόρων . Η ευρύτητα ενός συστήματος μπορεί να διαχωριστεί σε δυο τομείς : την ευρύτητα του υλικού και την ευρύτητα του λογισμικού , δηλαδή το κατά πόσο το σύστημα μπορεί να αξιοποιήσει υλικό και λογισμικό από διάφορους άλλους κατασκευαστές .
- **Κλιμάκωση :** Η κλιμάκωση είναι το χαρακτηριστικό που περιγράφει το κατά πόσο το λογισμικό του συστήματος και των εφαρμογών αλλάζει με την αύξηση της κλίμακας του συστήματος . Η κλιμάκωση είναι ιδιαίτερα σημαντική , καθώς το μέγεθος και οι ανάγκες ενός πληροφοριακού συστήματος αυξάνονται συνεχώς . Η αύξηση των αναγκών απαιτεί την προσθήκη επιπλέον υλικού και λογισμικού . Ένα πληροφοριακό σύστημα πρέπει να είναι ικανό να διαχειριστεί την αύξηση αυτή και να αξιοποιήσει αποτελεσματικά τις νέες δυνατότητες .
- **Συγχρονισμός :** Ο συγχρονισμός περιγράφει την ιδιότητα επεξεργασίας πολλών υπολογισμών την ίδια χρονική στιγμή . Όταν υπάρχει ένα σύνολο από διεργασίες σε ένα σύστημα με έναν επεξεργαστή , είναι δυνατή η ταυτόχρονη εκτέλεση όλων των διεργασιών με ενδιάμεσα διαστήματα εναλλαγής της χρήσης του επεξεργαστή . Αντίθετα , ένα κατανεμημένο σύστημα περιλαμβάνει πολλούς υπολογιστές καθένας εκ των οποίων περιέχει έναν ή περισσότερους επεξεργαστές . Η ύπαρξη πολλών επεξεργαστών επιτρέπει στο σύστημα να εκτελέσει ταυτόχρονα πολλαπλούς υπολογισμούς εξυπηρετώντας τις ανάγκες πολλών χρηστών με μεγάλη ταχύτητα .
- **Ανοχή στα λάθη :** Η ανοχή στα λάθη περιγράφει την ικανότητα ενός κατανεμημένου συστήματος να παρέχει μηχανισμούς αντιμετώπισης των σφαλμάτων που εμφανίζονται κατά τη λειτουργία του συστήματος . Ένα σύστημα με καλή ανοχή στα λάθη χαρακτηρίζεται από μεγάλο βαθμό διαθεσιμότητας . Η διαθεσιμότητα ενός συστήματος είναι ενδεικτική του

χρόνου , που είναι διαθέσιμο στους χρήστες . Η ανοχή στα λάθη μπορεί να επιτευχθεί με τον πλεονασμό του υλικού και με μηχανισμούς ανάκαμψης του λογισμικού . Στην πρώτη περίπτωση , παρέχονται μηχανισμοί άμεσης αποκατάστασης δυσλειτουργιών του υλικού . Ο στόχος αυτός επιτυγχάνεται με το διπλασιασμό του υλικού που μπορεί να εκτελέσει ακριβώς τις ίδιες διεργασίες . Στη δεύτερη περίπτωση παρέχονται μηχανισμοί ελέγχου των σφαλμάτων λογισμικού και αποκατάστασής τους όταν ανιχνευθούν . Η αποκατάσταση σφαλμάτων λογισμικού μπορεί να επιτευχθεί με διάφορους τρόπους , όπως με μηχανισμούς επαναφοράς του συστήματος και των εφαρμογών σε προηγούμενη κατάσταση .

- **Διαφάνεια :** Η διαφάνεια περιγράφει την ιδιότητα του συστήματος να γίνεται αντιληπτό από το χρήστη ως μια ενότητα και όχι ως μια ομάδα από διαφορετικά υποσυστήματα . Υπάρχουν οχτώ διαφορετικές μορφές διαφάνειας :
 - Διαφάνεια πρόσβασης : Περιγράφει την ιδιότητα ενός συστήματος να παρέχει πρόσβαση με ομοιόμορφο τρόπο σε πληροφορίες που είναι αποθηκευμένες είτε τοπικά είτε σε απομακρυσμένα σημεία .
 - Διαφάνεια περιοχής : Περιγράφει την ιδιότητα του συστήματος να παρέχει πρόσβαση στους χρήστες σε αντικείμενα πληροφοριών , αποκρύπτοντας τη θέση τους .
 - Διαφάνεια συγχρονισμού : Επιτρέπει σε πολλές διεργασίες να εκτελούνται ταυτόχρονα χρησιμοποιώντας διαμοιραζόμενους πόρους χωρίς , ωστόσο , να αλληλεπιδρά η μια με την άλλη .
 - Διαφάνεια αντιγράφων : Επιτρέπει την ύπαρξη πολλαπλών στιγμιοτύπων – αντιγράφων των πληροφοριακών αντικειμένων , χωρίς να είναι αντιληπτή η ύπαρξη των αντιγράφων από τις εφαρμογές του συστήματος .
 - Διαφάνεια αποτυχίας : Επιτρέπει την απόκρυψη λαθών επιτρέποντας στους χρήστες και στις εφαρμογές να συνεχίζουν απρόσκοπα τη λειτουργία τους παρά την ύπαρξη κάποιας αστοχίας υλικού ή λογισμικού .
 - Διαφάνεια μετανάστευσης : Επιτρέπει την μετακίνηση πληροφοριακών αντικειμένων μέσα στο σύστημα χωρίς να επηρεάζονται οι ενέργειες των χρηστών ή των εφαρμογών .
 - Διαφάνεια επίδοσης : Επιτρέπει στο σύστημα να ρυθμίζεται με στόχο την βελτίωση της απόδοσής του σε περίπτωση αύξησης των απαιτήσεων .
 - Διαφάνεια κλιμάκωσης : Επιτρέπει στο σύστημα να επεκτείνει την κλίμακά του χωρίς να υπάρχει αλλαγή στη δομή του ή στους αλγόριθμους των εφαρμογών .

3.5 Ολοκλήρωση Υποσυστημάτων–Ολοκληρωμένο Πληροφοριακό Σύστημα

Όλα τα πληροφοριακά υποσυστήματα που αναλύθηκαν παραπάνω πρέπει να διασυνδέονται μεταξύ τους με τρόπο διαφανή στο χρήστη, ώστε να αποτελούν ένα **Ολοκληρωμένο Πληροφοριακό Σύστημα Νοσοκομείου (Ο.Π.Σ.Ν)**. Το ενδιαφέρον

δεν πρέπει να εστιάζεται τόσο στο διαχωρισμό και την ονοματολογία των υποσυστημάτων, όσο στην πληρότητα όλων των υποστηρικτικών εφαρμογών για τη καλύτερη δυνατή λειτουργία του Νοσοκομείου.

Έτσι ένα Ο.Π.Σ.Ν θα έχει τις εξής ιδιότητες:

- Όλες οι εφαρμογές των υποσυστημάτων θα έχουν κοινή μεθοδολογία ανάπτυξης και τεκμηρίωσης.
- Τα δεδομένα θα εισάγονται μία φορά και θα διατίθενται οποτεδήποτε ζητούνται από τους εξουσιοδοτημένους χρήστες, μέσω των επιμέρους εφαρμογών.
- Τα δεδομένα θα είναι διαθέσιμα από όλους τους σταθμούς εργασίας του Νοσοκομείου
- Δεν υπάρχουν μεμονωμένες εφαρμογές για την κάλυψη συγκεκριμένων αναγκών χωρίς διασύνδεση μεταξύ τους.
- Το περιβάλλον προσπέλασης του χρήστη είναι παρόμοιο σε όλες τις εφαρμογές.
- Δεν υπάρχουν διπλά αντίγραφα εφαρμογών που εξυπηρετούν τον ίδιο σκοπό.

3.6 Βασικές Προϋποθέσεις Εγκατάστασης και Λειτουργίας Ο.Π.Σ.Ν

Ο σχεδιασμός, η ανάπτυξη και υλοποίηση πληροφοριακών συστημάτων για τα Νοσοκομεία είναι εξαιρετικά πολύπλοκο έργο, λόγω των ιδιομορφιών και της φύσης των συστημάτων.

Για να εγκατασταθεί και να λειτουργήσει ένα Π.Σ.Ν θα πρέπει να ικανοποιηθούν μια σειρά από προϋποθέσεις :

3.6.1. Μακροχρόνιο Στρατηγικό Σχέδιο για την Πληροφορική στον Φορέα.

Η εισαγωγή Ο.Π.Σ στον χώρο των Νοσοκομείων αποτελεί μακρά και επίπονη διαδικασία για την επιτυχία της οποίας απαιτείται σωστός προγραμματισμός και συνεχής αφιέρωση. Για το λόγο αυτό, η εκπόνηση ενός ολοκληρωμένου επιχειρησιακού σχεδίου για την πληροφορική, με συγκεκριμένο χρονοδιάγραμμα ενεργειών και στόχων αποτελεί βασική προϋπόθεση επιτυχής εισαγωγής και εφαρμογής ενός Π.Σ. Έτσι, η εφαρμογή της πληροφορικής δεν θα αντιμετωπίζεται αποσπασματικά, απλά για να καλύψει κάποιες πρόσκαιρες ανάγκες, αλλά με τρόπο στρατηγικό και επιπλέον ως βασικό μέσο για τη βελτίωση του νοσοκομειακού φορέα.

3.6.2 Εξειδικευμένα Στελέχη- Ισχυρό τμήμα πληροφορικής

Στελέχη ικανά να διαμορφώσουν λεπτομερείς απαιτήσεις από το σύστημα (ερευνητές, ειδικοί κοστολόγοι, κλπ) που θα εργάζονται στο περιβάλλον του Νοσοκομείου ή σε κεντρικό επίπεδο (Περιφερειακά Συστήματα Υγείας –Υπουργείο). Οι άνθρωποι αυτοί είναι απαραίτητοι, ώστε να ορίσουν μεθόδους και δείκτες μέτρησης και αξιολόγησης της αποδοτικότητας του Οργανισμού (Νοσοκομείου), και να προτείνουν τρόπους βελτίωσης ή ανασχεδιασμού των περιοχών δυσλειτουργίας. Η

λεπτομερής αποτύπωση και ποσοτική καταγραφή των μεγεθών είναι απαραίτητο στοιχείο για την επιτυχή προώθηση λύσεων.

Η κατάρτιση ενός μακροχρόνιου στρατηγικού σχεδίου για την ανάπτυξη της πληροφορική στο Νοσοκομείο, αλλά κυρίως η υλοποίηση του σχεδίου αυτού, απαιτεί την παρουσία ειδικών στα πληροφοριακά συστήματα υγείας, αλλά και σε όλες τις σχετικές ειδικότητες (ασφάλεια συστημάτων, δίκτυα υπολογιστών, ποιότητα, κ.λ.π.). Για το λόγο αυτό το τμήμα πληροφορικής του Νοσοκομείου πρέπει να είναι πλήρως στελεχωμένο.

3.6.3 Βήμα – Βήμα προσέγγιση στην εγκατάσταση συστημάτων

Η εισαγωγή συστημάτων πληροφορικής σε κάθε εργασιακό χώρο, επιφέρει μεταβολές στον τρόπο διεκπεραίωσης των καθημερινών εργασιών των χρηστών. Η αλλαγή αυτή δεν γίνεται πάντα αποδεκτή με ευκολία, ιδιαίτερα από τους μεγαλύτερους σε ηλικία εργαζομένους, οι οποίοι κατά τεκμήριο είναι και οι λιγότερο εξοικειωμένοι στα υπολογιστικά συστήματα.

Υπό την έννοια αυτή, ένα φιλόδοξο σχέδιο εισαγωγής ενός πλήρους και ολοκληρωμένου πληροφοριακού συστήματος είναι μάλλον απίθανο να στεφθεί από επιτυχία. Αντίθετα, μια προσέγγιση βήμα-βήμα, με πιο λογικούς και ρεαλιστικούς στόχους και με υποσυστήματα που θα ολοκληρωθούν σε προδιαγεγραμμένο χρονικό διάστημα, είναι πολύ πιο αποτελεσματική, δεδομένου ότι το προσωπικό θα έχει αρκετό χρόνο στη διάθεσή του να προσαρμοστεί στις αλλαγές και να τις αφομοιώσει.

Τα πρώτα μάλιστα βήματα πρέπει να εστιάζονται στη βελτίωση των υπάρχοντων διαδικασιών προς όφελος των χρηστών, μειώνοντας ή διευκολύνοντας το έργο τους. Μόνο αφού φανούν τα θετικά αποτελέσματα της πρώτης προσέγγισης, είναι δυνατόν να επιχειρούνται ενέργειες αναδιοργάνωσης και επανασχεδιασμού των διαδικασιών, ώστε να βελτιωθεί η λειτουργία του εγκατεστημένου συστήματος.

3.6.4 Εκπαίδευση του προσωπικού

Η αποδοχή του εγκατεστημένου συστήματος από τους χρήστες παίζει ιδιαίτερα σημαντικό ρόλο στην αποτελεσματική λειτουργία ενός Ο.Π.Σ.Ν. Για να εξασφαλιστεί αυτή απαιτείται η εκπόνηση και η εκτέλεση ενός εμπειριστατωμένου προγράμματος εκπαίδευσης. Το πρόγραμμα αυτό πρέπει να λαμβάνει υπόψη του το επίπεδο εκπαίδευσης των διαφόρων κατηγοριών των εργαζομένων και τις ιδιαίτερες ανάγκες τους (ιατρικό, νοσηλευτικό, διοικητικό προσωπικό).

Για την υλοποίηση του προγράμματος μπορούν να χρησιμοποιηθούν διάφοροι μέθοδοι, όπως η κλασική μέθοδος του αμφιθεάτρου-παρουσιάσεις, η εκπαίδευση κατά τη διάρκεια της εργασίας (on the job training), αλλά και πιο σύγχρονες μέθοδοι, όπως προγράμματα πολυμέσων που μπορεί να χρησιμοποιηθούν οποτεδήποτε από τους χρήστες.

Η ύπαρξη ενός ``γραφείου βοήθειας`` (help desk) για το πληροφοριακό σύστημα, μπορεί να βοηθήσει σημαντικά στην αντιμετώπιση των προβλημάτων χρήσης ενός Π.Σ.Ν ενισχύοντας σημαντικά το βαθμό αποδοχής από το προσωπικό.

3.6.5 Τυποποίηση

Στον χώρο της υγείας και ειδικότερα στον χώρο των Π.Σ.Ν, υπάρχει ιδιαίτερη ανάγκη για τυποποίηση. Αρκετοί φορείς υγείας (ιδιαίτερα στις αναπτυγμένες χώρες), έχουν μηχανογραφήσει τις κυριότερες λειτουργίες τους. Η εικόνα όμως που παρουσιάζεται είναι αυτή των απομονωμένων πληροφοριακών υποσυστημάτων που λειτουργούν αυτόνομα, χωρίς επικοινωνιακή σύνδεση μεταξύ τους.

Βασική αιτία για την έλλειψη επικοινωνίας και ολοκλήρωσης μεταξύ των διαφόρων πληροφοριακών υποσυστημάτων είναι η έλλειψη προτύπων.

Για την επιτυχή εφαρμογή ενός Ο.Π.Σ.Ν απαιτείται η υιοθέτηση προτύπων ανάπτυξης και επικοινωνίας. Το θέμα της τυποποίησης είναι κάτι το οποίο θα πρέπει να αντιμετωπιστεί συνολικά σε εθνικό επίπεδο από ένα φορέα, ο οποίος θα είναι αρμόδιος για την υιοθέτηση, εξέλιξη και επιβολή προτύπων

Κεφάλαιο 4 Ηλεκτρονικός Ιατρικός Φάκελος Ασθενή

4.1 Εισαγωγή

Η ηλεκτρονική υγεία είναι η εφαρμογή τεχνολογιών πληροφοριών και επικοινωνιών στον τομέα της υγείας . Η ηλεκτρονική υγεία έχει ως στόχο τη συγκέντρωση, ανάλυση και αποθήκευση κλινικών δεδομένων σε όλες τις μορφές καθώς και την ανταλλαγή αυτών των δεδομένων ανάμεσα στις μονάδες παροχής υγείας, τους ασφαλιστικούς φορείς και τις υγειονομικές αρχές. Ένα από τα συστατικά που αποτελούν την ηλεκτρονική υγεία είναι ο ΗΙΦ ασθενούς, ορισμός του οποίου θα δοθεί στη συνέχεια του κεφαλαίου. Ο ΗΙΦ είναι ένα εργαλείο που έχει ήδη εφαρμοστεί διεθνώς ενώ στη χώρα μας δεν έχει αναπτυχθεί παρά την επιτακτική ανάγκη ανάπτυξης και εφαρμογής του. Παράδειγμα των πλεονεκτημάτων που παρουσιάζει η εφαρμογή του έχουμε κατά την διακομιδή ασθενών από το ένα νοσοκομείο στο άλλο ή κατά την εισαγωγή των ασθενών από το ένα τμήμα στο άλλο. Στις περιπτώσεις αυτές με την χρήση του επιβάλλεται η πλήρης, διαφανής και αποτελεσματική ροή της ιατρικής πληροφορίας, ενώ παράλληλα τα δεδομένα απαιτούν διατήρηση και συντήρηση από τους γιατρούς και το νοσηλευτικό προσωπικό με στόχο τη βελτίωση της υγείας. Επίσης, η πρόσβαση στον ηλεκτρονικό φάκελο ασθενή υποστηρίζει το σχεδιασμό κλινικών επεμβάσεων και γενικότερα διευκολύνει την έρευνα και τη διαχείριση της δημόσιας υγείας, εφόσον ακολουθούνται τα διεθνή πρότυπα τόσο για τις κωδικοποιήσεις όσο και για τις απαιτούμενες διασυνδέσεις.

4.2 Ιατρική Πληροφορία

Γενικά κάθε πληροφορία ως οργανωμένο στοιχείο ή γνώση που παρέχει μια βάση για τη λήψη αποφάσεων, καθορίζεται κυρίως δια μέσου των λειτουργιών της. Όταν ένας γιατρός αποφασίζει για τον ασθενή, θα πρέπει πρώτα να επεξεργασθεί το ιστορικό του ασθενή, όπως ευρήματα από εργαστηριακές και απεικονιστικές εξετάσεις, υποθέσεις και προηγούμενες νοσηλείες. Η ιατρική πληροφορία είναι ιδιαίτερα πολύπλοκη στα νοσοκομεία που παρέχουν υπηρεσίες υγείας σε εκατοντάδες ασθενείς ημερησίως, λόγω του ότι ο όγκος των δεδομένων και των πληροφοριών που δημιουργούνται είναι τεράστιος .

Διάφορες έρευνες έχουν δείξει πως αρκετά ιατρικά σφάλματα οφείλονται στην αδυναμία του νοσοκομειακού συστήματος να παρέχει στους γιατρούς όλες τις απαραίτητες πληροφορίες που απαιτούνται για τη λήψη σωστών αποφάσεων . Τα ιατρικά σφάλματα έχουν όχι μόνο μοιραία αποτελέσματα για την υγεία των ασθενών αλλά και συνεπάγονται σημαντική οικονομική επιβάρυνση τόσο για τους ασθενείς όσο και για το σύστημα υγείας . Επίσης πολλές φορές έχει γίνει χρήση μη τεκμηριωμένων παρεμβάσεων υγείας για συγκεκριμένα ιατρικά προβλήματα, λόγω ελλιπούς ιατρικής πληροφόρησης .

Η καλή διαχείριση της ιατρικής πληροφορίας μπορεί να βελτιώσει την ποιότητα, την αποτελεσματικότητα και την αποδοτικότητα των υπηρεσιών υγείας, και το σπουδαιότερο μπορεί να επιτρέψει στο ιατρικό και νοσηλευτικό προσωπικό να

δαπανήσει τον απαιτούμενο χρόνο με τους ασθενείς. Όμως, η σωστή ιατρική πληροφόρηση είναι ένα δύσκολο και πολύπλοκο θέμα διότι απαιτεί πρόσβαση των νοσοκομειακών γιατρών στο κλινικό ιστορικό με χρήση κοινών αρχείων, συμμετοχή στη λήψη αποφάσεων και μεγαλύτερη αυτοδιαχείριση. Στον αντίποδα, το πεδίο που αφορά την παροχή ιατρικής περίθαλψης δεν μπορεί να βασίζεται στην υποκειμενική αντίληψη και ατομική εμπειρία του εκάστοτε κλινικού γιατρού, αφού η εμπειρία αυτή εμπεριέχει στοιχεία μεροληψίας.

Επιπλέον η διακίνηση της ιατρικής πληροφορίας είναι μια πολυπαραγοντική διαδικασία όχι μόνο γιατί σε αυτήν εμπίπτουν θέματα εμπιστευτικότητας των στοιχείων των ασθενών, αλλά και γιατί σχετίζεται με την έλλειψη κατάλληλων οργανωτικών υποδομών στα νοσοκομεία που να εξασφαλίζουν τη χρήση προτύπων και κατευθυντήριων οδηγιών κλινικής πρακτικής .

Σημαντικό επίσης πρόβλημα αποτελεί το ότι αρκετές φορές ο ασθενής είναι ο μοναδικός που γνωρίζει ποιοι ιατροί συμμετείχαν στην πρόοδο της θεραπείας του αλλά και πού βρίσκονται σχετικές πληροφορίες για αυτή. Έτσι ο ασθενής αποκτά ένα προβάδισμα γνώσεων απέναντι του ιατρού σχετικά με την κατάσταση της υγείας του, το τρόπο ζωής και το ιστορικό του, για τα οποία πολλές φορές δεν δίνει πληροφορίες από φόβο, ντροπή ή άγνοια. Συνεπώς ο ρόλος του ασθενή ως πηγή ιατρικής πληροφόρησης αποτελεί μόνο μια προσεγγιστική και μη συστηματοποιημένη πληροφορία. Επιπλέον όταν στο νοσοκομειακό περιβάλλον λαμβάνεται μια απόφαση για την δημιουργία ενός νέου τμήματος ή για μια νέα κλινική υπηρεσία ή για μια νέα σύμβαση ιατρικών ειδών, θα πρέπει να ομαδοποιούνται τα δεδομένα κατά περίπτωση και να συνδυάζονται με τα στοιχεία που απαιτούνται για την επάνδρωση, τις εγκαταστάσεις, και άλλους υλικούς πόρους.

Για τους παραπάνω λόγους υπάρχει η ανάγκη να δημιουργηθεί μια πληροφοριακή δομή που να παρέχει τη μεγαλύτερη δυνατή πληροφορία για τα παραπάνω, ενώ μεγαλύτερη αξιοπιστία στην αξιολόγηση των κλινικών δεδομένων επιτυγχάνεται με την συστηματική καταγραφή και ανάλυση της παραγόμενης ιατρικής πληροφορίας. Το σημαντικότερο στοιχείο που αποτελεί και τη βάση της διαχείρισης της ιατρικής πληροφορίας στο χώρο και στο χρόνο είναι ο Ιατρικός Φάκελος.

4.3 Ο Ιατρικός Φάκελος

Σύμφωνα με την Ευρωπαϊκή Επιτροπή Προτυποποίησης, Ιατρικός Φάκελος είναι η *αποθήκη όλων των πληροφοριών που αφορούν στο ιατρικό ιστορικό του ασθενούς, έτσι ώστε να αποτελεί τη βάση της διάγνωσης και της θεραπευτικής αντιμετώπισης του ασθενούς αλλά και τη βάση επιδημιολογικών ερευνών*. Επιπλέον παρέχει πληροφορίες διοικητικής, οικονομικής και στατιστικής φύσεως καθώς και ποιοτικού ελέγχου .

Συνεπώς ο Ιατρικός Φάκελος είναι η συστηματοποιημένη συλλογή του ιστορικού και της κατάστασης υγείας ενός ασθενούς. Συνήθως δημιουργείται, διατηρείται και συντηρείται από έναν ιατρό ή μια Μονάδα Υγείας ή άλλον επαγγελματία φροντίδας υγείας.

Σύμφωνα με τη διεθνή βιβλιογραφία, οι ιατρικοί φάκελοι ταξινομούνται σε σχέση με τα ακόλουθα στοιχεία:

1. *Το περιεχόμενο:* Φάκελος ενδο-νοσοκομειακών ασθενών, Φάκελος εξω-νοσοκομειακών ασθενών, Φάκελος Φροντίδας Υγείας.
2. *Τη δομή:* Φάκελος προσανατολισμένος στο πρόβλημα, Φάκελος προσανατολισμένος στο χρόνο, Φάκελος προσανατολισμένος στην εργασία, Φάκελος προσανατολισμένος στην αντιμετώπιση του ασθενή.
3. *Το σκοπό:* Νοσηλευτικός φάκελος, Ακτινολογικός φάκελος, Φαρμακευτικός φάκελος.
4. *Το μέσο που χρησιμοποιείται για την καταγραφή:* Χειρόγραφος φάκελος, Ηλεκτρονικός φάκελος, Φάκελος Πολυμέσων, Φάκελος ασθενή σε μικροφίλμ.

Ανεξάρτητα από τη μορφή που έχει, κάθε ιατρικός φάκελος θα πρέπει να περιέχει όλα τα δεδομένα και τη πληροφορία που σχετίζεται με την κατάσταση υγείας του ασθενή. Η πληροφορία αυτή αφορά στο ιστορικό, στη κλινική εξέταση, στη διάγνωση, στα αποτελέσματα εργαστηριακών και διαγνωστικών ή παρακλινικών εξετάσεων, στις απεικονιστικές εξετάσεις κ.α

Με άλλα λόγια ένας τέτοιος φάκελος χρειάζεται να συνδυάζει μια πλειάδα από διαφορετικού τύπου πληροφορίες. Αυτές οι πληροφορίες είναι:

- Δημογραφικά στοιχεία
- Ιατρικό ιστορικό – Παράγοντες κινδύνου (risk factors)
- Κλινικά δεδομένα φυσικής εξέτασης – διαγνώσεις και σημεία
- Νοσηλείες – Εγχειρήσεις
- Ιατροφαρμακευτική περίθαλψη
- Εργαστηριακές εξετάσεις (ανάλυση αίματος ,ούρων, κλπ)
- Καταγραφές βιοδυναμικών (ηλεκτροκαρδιογράφημα, ηλεκτρομυογράφημα, κλπ.)
- Ιατρικές πράξεις
- Παραπεμπτικά - Γνωματεύσεις
- Διαγνωστικές εξετάσεις και ιατρικές εικόνες (Ακτινογραφίες, μαγνητικές τομογραφίες, αξονικές τομογραφίες, κλπ)
- Διαχειριστικά – οικονομικά στοιχεία ιατρικών πράξεων και νοσηλειών
- Πιθανά αρχεία παλιών ιατρικών φακέλων

Οι αντίστοιχες εξετάσεις συνοδεύουν τον φάκελο του ασθενούς συνήθως υπό την μορφή με την οποία δημιουργούνται στα αντίστοιχα εργαστήρια. Τέτοιες μορφές είναι για παράδειγμα προτυπωμένα έντυπα για τις αιματολογικές, τις μικροβιολογικές και τις βιοχημικές εξετάσεις, ακτινογραφικά φιλμ, έντυπα ηλεκτροκαρδιογραφημάτων και συνοδευτικά χειρόγραφα δυσανάγνωστα φύλλα ιστορικών, τα οποία χαρακτηρίζονται από σύνθετες, αποδιοργανωμένες σημειώσεις και περιγραφές ελεύθερων κειμένων, με συνώνυμα ή συντμήσεις κ.λπ.

Όλα αυτά αντίκεινται στην αυστηρή πληροφοριακή οργάνωση που θα ήταν επεξεργάσιμη από ένα πληροφοριακό σύστημα. Επίσης, παράπλευρα αποτελέσματα της αναπαράστασης αυτής είναι i) η παραγωγή ενός μεγάλου όγκου ιατρικού φακέλου, ii) η μεγάλη πιθανότητα απώλειας δεδομένων, iii) η μεγάλη δυσκολία ανάκτησης πληροφορίας και iv) η ασύγχρονη συσχέτιση του ιστορικού με τις εξετάσεις.

4.4 Ηλεκτρονικός Ιατρικός Φάκελος

Μέχρι σήμερα δεν υπάρχει ένας σαφής ορισμός ή μια ξεκάθαρη άποψη για τα συστήματα Ηλεκτρονικών Ιατρικών Φακέλων (ΗΙΦ) ή Electronic Medical Record/EMR. Στα συστήματα υγείας διαφόρων κρατών δεν υπάρχει ομοφωνία ως προς την έννοια του EMR, με συνέπεια αυτός να αποδίδεται με διαφορετικές θεωρήσεις: Άλλοτε αποδίδεται ως αντίγραφο του χειρόγραφου φακέλου με διαδικασίες αυτόματης ανάγνωσης, άλλοτε ως αυτοματοποιημένος εργαστηριακός φάκελος (Laboratory Medical Record/LMR) και άλλοτε ως Ηλεκτρονικός φάκελος Υγείας (ΗΦΥ – Electronic Health Record/EHR). Η έλλειψη ενός διεθνώς αποδεκτού ορισμού προκαλεί την έλλειψη συμφωνίας τόσο για το τι είναι ο ΗΙΦ όσο και για την αντίστοιχη ηλεκτρονική υποδομή του Εθνικού Συστήματος Υγείας.

Ωστόσο μια προσέγγιση του όρου Ηλεκτρονικός Ιατρικός Φάκελος δίνεται από το Ινστιτούτο Ιατρικής των ΗΠΑ. Σύμφωνα με αυτό ο ΗΙΦ είναι το σύστημα που είναι σχεδιασμένο έτσι ώστε να υποστηρίζει την απόλυτη διαθεσιμότητα και ακρίβεια ιατρικών ή άλλων πληροφοριών, με στόχο την παροχή ιατρικής περίθαλψης. Ως συνέπεια του ανωτέρω ορισμού, ο EMR χαρακτηρίζεται από τις παρακάτω ιδιότητες:

- Ατομικότητα, εφόσον παρέχει με κάθε λεπτομέρεια στοιχεία που αφορούν τη περιγραφή της κατάστασης υγείας των ασθενών,
- Συνέπεια, εφόσον λόγω των πληροφοριών που παρέχει μπορεί να οδηγήσει με συνέπεια σε λήψη κλινικών αποφάσεων,
- Εξουσιοδότηση, εφόσον έχει τη δυνατότητα να ενεργοποιήσει τη κοινωνική διάσταση της ασθένειας, αποτελώντας σε πολλές περιπτώσεις νομικό έγγραφο, το οποίο καθορίζει ακόμα και τη πολιτική ζωή.

Ο EHR μπορεί να είναι είτε κλασσικός και να περιέχει τη στοιχειώδη κλινική πληροφορία, είτε μοντέρνος, και να περιέχει επιπλέον την καταναμημένη στα επιμέρους υποσυστήματα πληροφορία για τις ιατρικές απεικονίσεις, να παράγει μηνύματα και να διασυνδέεται με άλλες μονάδες υγείας.

Θεωρητικά ο ιδανικός EHR παρέχει τη δυνατότητα σχεδιασμού ιατρικών συμπερασμάτων από τα δεδομένα του, με τη χρήση αλγόριθμων εξόρυξης δεδομένων και με τη ταυτόχρονη μετάφραση κλινικών δεδομένων, διαμέσου της επεξεργασίας της φυσικής γλώσσας κειμένων. Ωστόσο ο κλασσικός EHR θα πρέπει να περιέχει τουλάχιστον και ανά κάθε χρονική στιγμή την επαφή γιατρού- ασθενή, τα δημογραφικά στοιχεία και το ιστορικό του ασθενή, καθώς και τις διαγνώσεις, συνοδευμένες από σαφείς λεπτομέρειες των εκάστοτε νοσηλειών, όπως συνταγογραφία και τιμές εργαστηριακών εξετάσεων.

4.5 Ιστορική Αναδρομή

Η ιδέα του Ηλεκτρονικού φακέλου ξεκίνησε το 1969 από τον Dr. William Edward Hammond II ως το μέρος όπου αποθηκεύονται για πάντα όλες οι πληροφορίες για έναν ασθενή, προσφέροντας του έτσι τις καλύτερες υπηρεσίες, παρέχοντας δηλαδή τη δυνατότητα της γνώσης κάθε λεπτομέρειας του ιστορικού του ασθενή (εξετάσεις, διαγνώσεις, φάρμακα κτλ) και συνεπώς τη συνολική αντίληψη των προβλημάτων υγείας. Το μέρος αυτό είναι οι ηλεκτρονικοί υπολογιστές αντί των χάρτινων χειρόγραφων φακέλων, μέσω των οποίων επιτυγχάνεται η συλλογή και η χρονική

παρουσίαση των δεδομένων της κατάστασης υγείας των ασθενών ανά πάσα χρονική στιγμή.

Η υλοποίηση του Ιατρικού φακέλου (TMR) πραγματοποιήθηκε με την κατασκευή μιας διασύνδεσης ανάμεσα σε ένα σκάνερ και έναν προσωπικό υπολογιστή (τύπου PDP 12), με ένα πρόγραμμα σε γλώσσα assembly που εκτύπωνε το ιατρικό ιστορικό άμεσα από τον ασθενή στο Health Department at Duke University. Από το 1973 το κλείσιμο ραντεβού και οι πληρωμές των εξωτερικών ασθενών λειτουργούσαν βάσει του πρώτου Ηλεκτρονικού Ιατρικού φακέλου (CPR). Αργότερα ομάδα από πέντε γιατρούς και φοιτητές κατασκεύασε το GEMISCH, δηλαδή μια command line γλώσσα που έτρεχε στα λειτουργικά συστήματα εκείνης της εποχής (RSX and VMS Operating Systems), βάσει του οποίου ειδικές εφαρμογές αντικαταστάθηκαν από γενικότερες εφαρμογές. Έτσι δημιουργήθηκε ένα λεξικό από μετα-δεδομένα, παράγοντας τον TMR που εφαρμόστηκε σε ένα καρκινικό νοσοκομείο 60 κρεβατιών .

4.6 Αναλυτικότερη περιγραφή του ΗΙΦ

Η περιγραφή που ακολουθεί βασίζεται στην αρχιτεκτονική Ηλεκτρονικού Ιατρικού Φακέλου που έχει προτείνει το Ευρωπαϊκό Ερευνητικό πρόγραμμα Good European Health Record.

Ορισμός του Ιατρικού Φακέλου (κείμενο CEN/TC251/WG1/N8 της Ευρωπαϊκής Επιτροπής Προτυποποίησης): “ Ο Ιατρικός Φάκελος είναι η αποθήκη όλων των πληροφοριών που αφορούν στο ιατρικό ιστορικό του ασθενούς. Αποτελεί επομένως την βάση της διάγνωσης και της θεραπευτικής αντιμετώπισης του ασθενούς αλλά και τη βάση επιδημιολογικών ερευνών. Επιπλέον, παρέχει πληροφορίες διοικητικής, οικονομικής και στατιστικής φύσεως, καθώς και ποιοτικού ελέγχου.

4.7 Η δομή του ΗΙΦ

Ο φάκελος είναι ένα πρόγραμμα διαχείρισης βάσεων δεδομένων. Εφόσον ο φάκελος του ασθενούς περιέχει δεδομένα διαφόρων μορφών, αυτά πρέπει να καταχωρηθούν στον ΗΙΦ κατά τέτοιο τρόπο, ώστε να βρίσκονται σε απόλυτη συσχέτιση μεταξύ τους, προκειμένου να διατηρηθούν οι πληροφορίες που εμπεριέχονται σε αυτή τη συσχέτιση

4.7.1 Στοιχειώδης Πληροφορία (ITEM)

Η βάση του φακέλου ενός ασθενή είναι στα πλαίσια της λογικής αυτής, η “στοιχειώδης πληροφορία (item)”. Στοιχειώδης πληροφορία είναι η ελάχιστη ιατρική πληροφορία που έχει νόημα από τη στιγμή που προσδιορίζεται. Έτσι η πληροφορία “ήπαρ” προσδιορίζει το αντίστοιχο όργανο, η ιδιότητά του “διογκωμένο” αναφέρεται αντίστοιχα σε μια παθολογική κατάσταση του ύπατος. Η πληροφορία “διογκωμένο” δεν προσδιορίζει κάτι συγκεκριμένο (διογκωμένο μπορεί να είναι οτιδήποτε), δεν μπορεί να είναι συνεπώς μία στοιχειώδης πληροφορία.

4.7.2 Επαφή (CONTACT), Επεισόδιο (EPISODE)

Οι στοιχειώδεις πληροφορίες που απαιτούνται προκειμένου να περιγραφεί μία συγκεκριμένη κατάσταση του ασθενούς οργανώνονται σε μια “Επαφή (contact)”. Η επαφή αποτελεί μία ενότητα δεδομένων που περιγράφουν μια επίσκεψη του ασθενούς στον ιατρό.

Ένα σύνολο από τέτοιες επαφές που αναφέρονται στο ίδιο πρόβλημα του ασθενούς, ονομάζεται “Επεισόδιο (episode)”. Περιγράφει τη χρονική εξέλιξη της υγείας του ασθενούς. Είναι σαφές ότι ένας τέτοιος φάκελος επιτρέπει την παρακολούθηση της εξέλιξης της υγείας του.

4.8 Ιατρικές , Διαχειριστικές Πληροφορίες

Το σύνολο των επαφών ενός φακέλου, μαζί με τις βασικές (αμετάβλητες) παραμέτρους του ασθενούς (ατομικό αναμνηστικό, κληρονομικό ιστορικό, ομάδα αίματος κ.τ.λ.) αποτελεί το ιατρικό τμήμα του φακέλου (ιατρικές πληροφορίες).

Εκτός του ιατρικού συμπληρώνεται και το διαχειριστικό τμήμα του φακέλου, αφού είναι εκείνο το τμήμα που περιέχει πληροφορίες όπως το όνομα, το επώνυμο του ασθενούς, ασφαλιστικές πληροφορίες κ.ο.κ.

Η οργάνωση αυτή του ιατρικού φακέλου επιτρέπει την αποθήκευση και τη μεταφορά των πληροφοριών με τον καλύτερο τρόπο. Η επιλογή των κατάλληλων στοιχειωδών πληροφοριών για την περιγραφή μιας κατάστασης, η απόδοση των κατάλληλων χαρακτηριστικών σε αυτές, η οργάνωσή τους σε επαφές και επεισόδια δίνουν την δυνατότητα της ακριβούς αναπαράστασης της γνώσης που αφορά στον ασθενή και ακολούθως την αποθήκευσή της. Εφόσον οι κανόνες που διέπουν την οργάνωση των στοιχειωδών πληροφοριών είναι γνωστοί, πληροφορίες που έχουν αναπαρασταθεί και αποθηκευθεί με βάση τους κανόνες αυτούς μπορεί στο μέλλον να ανακληθούν, να αναγνωσθούν και να επεξεργασθούν, χωρίς να αλλοιωθούν οι αρχικές πληροφορίες.

Τα προγράμματα ΗΙΦ χρησιμοποιούν ακριβώς αυτή την δομή για να αποθηκεύσουν τα δεδομένα. Η ιδιαιτερότητα αυτή δεν μας επιτρέπει να κάνουμε χρήση οποιουδήποτε προγράμματος διαχείρισης βάσεων δεδομένων για την τήρηση των φακέλων των ασθενών. Τα συνήθη προγράμματα προσανατολίζονται στην άριστη διαχείριση των δεδομένων που φυλάσσουν και όχι στην οργάνωσή τους με βάση τις ανάγκες της ιατρικής πρακτικής.

4.9 Φάκελος Πολυμέσων (Multimedia)

4.9.1 Ακτινογραφίες

4.9.3 Άλλα Αντικείμενα (OBJECTS)

Η ενσωμάτωση άλλων αντικειμένων στο φάκελο γίνεται με τον τρόπο ακριβώς των ακτινογραφιών και των ΗΚΓ. Ως παράδειγμα μπορούμε να αναφέρουμε την ενσωμάτωση:

- Ηχητικών σημάτων (ηχοκαρδιογράφημα)
- Ακολουθιών video ενδοσκοπίας
- GEHR γραφημάτων (clinical drawings)

Η νέα τεχνολογία επιτρέπει την επεξεργασία τόσο του κειμένου του φακέλου, όσο και των αντικειμένων που είναι ενσωματωμένα σε αυτόν. Οι ακτινογραφίες μπορούν να αναστραφούν, να περιστραφούν, να μεγεθυνθούν, να μεταβληθεί η αντίθεσή τους (contrast) κ.ο.κ. Ακόμη είναι δυνατό να διαγνωστεί το ΗΚΓ από τον υπολογιστή, προκειμένου να βοηθηθεί ο γιατρός στο έργο του.

4.10 Επεξεργασία Δεδομένων

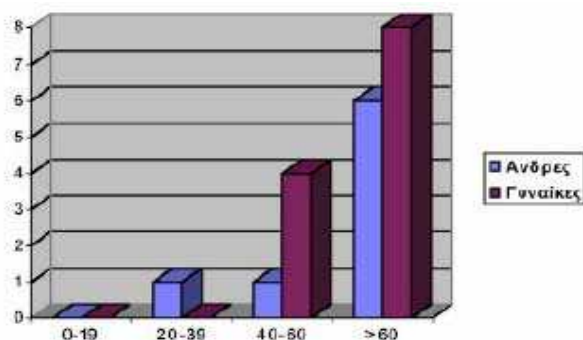
4.10.1 Ανάλυση Πλέγματος (Grid Analysis)

Πρόκειται για συνοπτική παρουσίαση των δεδομένων όλων των επαφών ενός ασθενούς και περιέχει τη δυνατότητα συγκρίσεων. Έτσι έχουμε τη δυνατότητα της παρακολούθησης της εξέλιξης των παραμέτρων στο χρόνο. Στις αναλύσεις αυτές περιλαμβάνονται η Μικρο-ανάλυση και η Ανάλυση Προβλήματος. Η Μικρο-ανάλυση παρουσιάζει την χρονική εξέλιξη μιας παραμέτρου του ασθενούς κατά τις διάφορες επαφές με το γιατρό. Η Ανάλυση Προβλήματος παρουσιάζει την χρονική εξέλιξη της παραμέτρου «πρόβλημα» κατά τις διάφορες επαφές με το γιατρό .

4.10.2 Ανάλυση Πληθυσμού

Πρόκειται για ένα πρώτο επίπεδο στατιστικής επεξεργασίας. Δίνεται η δυνατότητα ανάλυσης των φακέλων των ασθενών προκειμένου να επιτευχθεί, για παράδειγμα η έγκυρη ειδοποίησή τους για τη διενέργεια εμβολιασμού, δοκιμασιών Παπανικολάου, μαστογραφιών.

Στην παρακάτω εικόνα φαίνεται το αποτέλεσμα της στατιστικής επεξεργασίας των φακέλων της κοινότητας Βρουτσειού Αμοργού. Ο Ιατρικός Φάκελος μπορεί να εξάγει τα δεδομένα σε διάφορους τύπους (format) πράγμα που καθιστά δυνατό την επεξεργασία τους.



Γράφημα 1 - Αριθμός ανδρών και γυναικών κατά ηλικία, που πάσχουν από υπερχοληστερολαιμία/ υπερτριγλυκεριδαιμία

4.10.3 Ειδικά Έγγραφα

Ο Φάκελος συνδέεται με έγγραφα που μπορούν να έχουν χρησιμότητα στη καθημερινή ζωή. Τα έγγραφα είναι γενικά πρότυπα (templates) και παίρνουν συγκεκριμένη μορφή ανάλογα με τον ιδιαίτερο κάθε φορά εξεταζόμενο ασθενή. Συμπληρώνονται αυτόματα με τα συγκεκριμένα στοιχεία αυτού. Παραδείγματα τέτοιων εγγραφών είναι τα ακόλουθα:

1. Παραπεμπτικό εξετάσεων.
2. Διακομιστήριο.
3. Πρόσκληση για διενέργεια τέστ δοκιμασίας Παπανικολάου.
4. Πρόσκληση για διενέργεια εμβολιασμών.
5. Πρόσκληση για διενέργεια μαστογραφίας.

4.11 Βάση Φαρμάκων και Κωδικοποίηση

Κάθε φάρμακο καταχωρείται στη Βάση Φαρμάκων με συγκεκριμένο τρόπο (format) ώστε να είναι δυνατή η επεξεργασία στοιχείων που αφορούν τη χορήγησή τους. Η Βάση αυτή θα εμπλουτίζεται συνεχώς κατά τη χρήση των ΗΙΦ.

4.12 Σύστημα Υποβοήθησης Διάγνωσης

Πρόκειται για έξυπνα προγράμματα, τα οποία “διαβάζουν” και επεξεργάζονται τα δεδομένα ενός ή περισσότερων ιατρικών φακέλων, προκειμένου να υποβοηθήσουν τον ιατρό στο έργο του. Η υποβοήθηση αυτή συνίσταται, συνήθως, στην παροχή οδηγιών: επιστούν την προσοχή σε σημαντικές παραμέτρους του ασθενούς, εντοπίζουν αλληλεπιδράσεις φαρμάκων, συνιστούν καλύτερη θεραπεία.

4.13 Τηλεϊατρική

Ο ΗΙΦ ενός ασθενούς μπορεί να μεταφερθεί μέσω απλών τηλεφωνικών γραμμών, δορυφορικών ζευξέων, γραμμών ISDN ή οποιουδήποτε άλλου επικοινωνιακού μέσου σε οποιαδήποτε απόσταση. Μπορεί έτσι ο ιατρός να ενημερώσει σχετικά με κάποιον ασθενή, οποιονδήποτε συναδέρφό του διαθέτει συμβατό λογισμικό ΗΙΦ. Ως εκ τούτου, μπορεί να ζητήσει την συμβουλή του συναδέρφου του όσον αφορά στο συγκεκριμένο περιστατικό. Προκειμένου να καταστεί δυνατό όλα τα λογισμικά ΗΙΦ να είναι συμβατά μεταξύ τους, έτσι ώστε όλοι οι ιατροί να μπορούν να ανταλλάξουν τους φακέλους των ασθενών τους, αναπτύσσονται πρότυπες αρχιτεκτονικές Ηλεκτρονικών Ιατρικών Φακέλων σε επίπεδο Ευρωπαϊκό και παγκόσμιο.

4.14 Παράγοντες που καθορίζουν την εφαρμογή του ΗΦΥ

Υπάρχουν πολλοί παράγοντες που καθορίζουν την εφαρμογή, αποδοχή και χρήση του ΗΦΥ στα Ελληνικά Δημόσια Νοσοκομεία. Αυτοί γενικά ομαδοποιούνται σε δύο κύριες κατηγορίες :

4.14.1 Παράγοντες του εσωτερικού περιβάλλοντος ή μικρο-περιβάλλον:

- **Η νοοτροπία του ιατρικού προσωπικού:** Οι γιατροί κατά το μεγαλύτερο μέρος είναι αρνητικοί στη χρήση της ΤΠ&Ε, θεωρούν ότι η ηλεκτρονική καταχώρηση δεν αποτελεί ιατρικό έργο αλλά πάρεργο και δεν αποδέχονται την αλλαγή των κανόνων της λειτουργίας και των διαδικασιών που απαιτούνται κατά την εφαρμογή της ηλεκτρονικής διαχείρισης. Άλλωστε, οι γιατροί κατέχουν απόλυτη εξουσία πάνω στην ασθένεια και μονοπωλούν την θεραπεία της, κυριαρχώντας έτσι στο νοσοκομειακό χώρο, θέτοντας οι ίδιοι τους κανόνες λειτουργίας.
- **Ο κοινωνικός παράγοντας:** Οι επαγγελματίες υγείας αντιδρούν στην ηλεκτρονική επεξεργασία προσωπικών δεδομένων, φοβούμενοι το προσωπικό απόρρητο και κυρίως το ενδεχόμενο του στιγματισμού ή ακόμα και της απαξίωσης.
- **Η κατάρτιση του ιατρικού, νοσηλευτικού, τεχνικού και διοικητικού προσωπικού στη χρήση υπολογιστών:** Τα συστήματα επαγγελματικής εκπαίδευσης είναι μάλλον απαξιωμένα, ενώ τις περισσότερες φορές κυριαρχεί η απογοήτευση, η πτώση των ηθικών αξιών και η απουσία αφοσίωσης και επαγγελματισμού. Επιπλέον η εργασία με την εφαρμογή της ΤΠ&Ε, γίνεται περισσότερο διατμημένη, ενώ η ανταπόκριση σε κάθε νεωτερισμό είναι υπόθεση των νεότερων, και η αποδοτικότητα φθίνει με την ηλικία.
- **Τα παντός είδους συμφέροντα** που αναπτύσσονται και επιζούν σε ποικίλες καταστάσεις δημόσιας αδιαφάνειας, έλλειψης συγκεκριμένων δομών, διαδικασιών και λειτουργιών, οι οποίες συντηρούνται με διάφορους τρόπους, μεταξύ των οποίων είναι και εκείνη της αντίδρασης σε κάθε νέα ριζοσπαστική πρόταση αλλαγής ή βελτίωσης της υπάρχουσας κατάστασης. Άλλωστε «...το ιατρικό επάγγελμα θεωρείται ως μια από τις πάμπολλες ομάδες συμφερόντων

που δρουν μέσα στην κοινωνία, αφού οι ηθικοί κώδικες και οι νόμοι που ρυθμίζουν την άσκηση της ιατρικής είναι επίσης μηχανισμοί προστατευτικού του επαγγέλματος, από τις παρεμβάσεις του κοινού και από τον ανταγωνισμό...» .

- **Η πολυδιάσπαση τμημάτων σε συνδυασμό με την έξω-χωρική καταχώρηση:** Κατά τη διάρκεια της επίσκεψης στους θαλάμους, οι γιατροί κρατούν χειρόγραφες σημειώσεις για την κατάσταση των ασθενών, τις οποίες μεταγενέστερα θα πρέπει να μεταφέρουν στο ηλεκτρονικό ιατρικό υποσύστημα. Ωστόσο, η ενέργεια αυτή απαιτεί τη σπατάλη πολύτιμου χρόνου, οποίος θα μπορούσε να αποδοθεί σε ασθενείς.

Το πρόβλημα αυτό θα μπορούσε να επιλυθεί με τη εφαρμογή ασύρματου τοπικού δικτύου (Wireless Local Area Network/WLAN) με pen-based ή rocketable συσκευές. Με τον τρόπο αυτό είναι εφικτή η καταχώριση του ιστορικού από τους γιατρούς σε πραγματικό χρόνο, ακόμα και αν αυτοί βρίσκονται εν κινήσει. Υπάρχουν διάφορα προβλήματα σχετικά με την εφαρμογή της τεχνολογίας των WLANs.

Τέτοια είναι: ι) υστέρηση στον τομέα της ασφάλειας –είναι αρκετά ευάλωτα σε επιθέσεις και σε παρεμβολές , ιι) υπερχείλιση καναλιών στις ασύρματες συχνότητες (channel flood), ιιι) το μπλοκάρισμα συχνοτήτων (signal jamming), ιιιι) η καταγραφή δεδομένων που κινούνται στο δίκτυο (sniffing) κ.ά ν) η σχετικά χαμηλή ταχύτητα ανταλλαγής δεδομένων και το υψηλό κόστος του εξοπλισμού. Όμως η τεχνολογία τους εξελίσσεται συνεχώς και τα προβλήματα αυτά ξεπερνιούνται επιτυχώς.

4.14.2 Παράγοντες του εξωτερικού περιβάλλοντος ή μακρο-περιβάλλον:

- **Η έλλειψη της ταυτοποίησης ασθενών** με ενιαίους αριθμούς μητρώων, έτσι ώστε να αποφεύγονται οι πολύ-εγγραφές σε εθνικό επίπεδο, ως πρώτο βήμα δόμησης ηλεκτρονικού ιατρικού φακέλου.
- **Η μη αποδοχή της χρήσης των διεθνώς αποδεκτών κλινικών κωδικοποιήσεων και προτύπων** για την συστηματική καταγραφή των ιατρικών δεδομένων.. Στο σημείο αυτό θα πρέπει να τονιστεί η αμφισβήτηση και διαφωνία μέρους του ιατρικού σώματος σχετικά με την ελληνική μετάφραση της διεθνούς κωδικοποίησης ICD-9/10.
- **Η απουσία στενής επικοινωνίας και συνεργασίας** ανάμεσα σε νοσοκομεία ακόμα της ίδιας υγειονομικής περιφέρειας, με αποτέλεσμα να χάνεται πολύτιμος χρόνος για σχεδιασμό δομών που ήδη έχουν εφαρμοσθεί επιτυχώς σε άλλα νοσοκομεία.
- **Η έλλειψη ιατρικών και εργαστηριακών πληροφοριακών συστημάτων** στα δημόσια νοσοκομεία, αφού στη χώρα, η πληροφοριακή υποδομή των δημόσιων νοσοκομείων σε αντίθεση με τα ιδιωτικά, έχει επικεντρωθεί κυρίως στη διαχείριση λογιστικών και όχι ιατρικών εφαρμογών.
- **Η έλλειψη ποιότητας και ανταγωνισμού** των δημόσιων νοσοκομείων έναντι του ιδιωτικού τομέα, οδηγεί στην αποτυχία εφαρμογής της ΤΠ&Ε, αφού δεν είναι δυνατό να σχεδιάζονται δομές και ενέργειες, χωρίς την παροχή οποιασδήποτε μορφής κινήτρων.

- **Η τεχνολογική πρόοδος**, η οποία λόγω της ταχύτητας με την οποία εξελίσσεται, επιβάλλει δομικά και λειτουργικά σχήματα ευέλικτα και προσαρμόσιμα σε αυτήν, έτσι ώστε να μην αποτυγχάνει μία προσπάθεια πριν ακόμα εφαρμοστεί και δοκιμαστεί.
- **Η έλλειψη σαφών και συγκεκριμένων στόχων** οι οποίοι πρέπει να προσδιοριστούν κατά την έναρξη μιας εφαρμογής διαχείρισης της ιατρικής πληροφορίας, σε σχέση με τους οποίους μπορεί να αξιολογηθεί η πραγματική απόδοση της.
- **Η ελλιπής χρηματοδότηση των δημόσιων νοσοκομείων** για την εφαρμογή της ΤΠ&Ε, αφού για παράδειγμα, από τα 129 δημόσια νοσοκομεία, μόνο τα 25 επιλεγμένα από το Υπουργείο Υγείας, Πρόνοιας και Κοινωνικών Ασφαλίσεων, χρηματοδοτήθηκαν για διαχειριστικά πληροφοριακά συστήματα ετερογενούς προέλευσης, ενώ τα υπόλοιπα, προμηθεύονται τα διαχειριστικά τους συστήματα με ιδίους πόρους.
- **Τα σχετικά ελλιπή δημοσιευμένα στοιχεία** για την μελέτη αξιολόγησης των αυτοματοποιημένων πληροφοριακών συστημάτων στην υγειονομική περίθαλψη, αφού ίσως η αξιολόγηση των επενδύσεων ιατρικής πληροφορικής είναι πολυσύνθετη και προβληματική.

Κεφάλαιο 5

Πρότυπα και Κωδικοποιήσεις

5.1 Η Κωδικοποίηση της Ιατρικής Πληροφορίας

Πολύ μεγάλη σημασία έχει ο βαθμός τελειότητας και ακρίβειας της κωδικοποίησης της ιατρικής πληροφορίας, διότι ακόμα και η παραμικρή διαφοροποίηση μπορεί να αντανακλά σε πραγματικές διαφορές ποιότητας. Μέτριας ποιότητας κωδικοποιήσεις μπορεί να μην ανταποκρίνονται στην εγκυρότητα, γιατί περιορίζουν την ικανότητα ορθών εκτιμήσεων από τα διαχειριστικά δεδομένα. Το πόσο έγκυρη είναι μια κωδικοποίηση δεν επιδέχεται μια σαφή απάντηση της απόλυτης κατάφασης ή απόρριψης.

Ο τρόπος κωδικοποίησης δεν θα πρέπει απλά να κάνει τα δεδομένα χρήσιμα για περιγραφικούς σκοπούς, αλλά θα πρέπει να διερευνάται σε μεγαλύτερο βάθος με στόχο την αξιοποίηση κλινικών και οικονομικών πληροφοριών. Επίσης είναι πολύ πιθανό, ότι οι νοσοκομειακοί γιατροί με την ίδια ειδικότητα μπορεί να χρησιμοποιούν τις ίδιες εκφράσεις για διαφορετικές έννοιες.

Απαιτείται περαιτέρω έρευνα, η οποία θα καθορίσει τις περιοχές στις οποίες οι περισσότεροι γιατροί συμφωνούν σχετικά με την έννοια των όρων για τις διαγνώσεις, καθώς και μια κοινή γλώσσα ιατρικής ορολογίας, τόσο σε επίπεδο κωδικοποίησης όσο και σε επίπεδο ονοματολογίας, έτσι ώστε να αποδίδεται *αξιοπιστία* και *ποιότητα* στην παραγόμενη ιατρική πληροφορία. Η αξιοπιστία παράγεται με την εξασφάλιση της σταθερότητας, της ικανότητας αναπαραγωγής και ακρίβειας, ενώ η ποιότητα με τη συνέπεια των δεδομένων, δηλαδή σωστή απόδοση των όρων έτσι ώστε να επιτρέπεται η ανάκτηση των δεδομένων με έναν συνεπή τρόπο.

5.2 ISO/TC 215

Ο οργανισμός ISO έχει ιδρύσει την Τεχνική Επιτροπή 215 (TC215) με στόχο την προτυποποίηση στον τομέα της ιατρικής πληροφορικής. Τα πρότυπα (standards) κατά ISO/TC 215 είναι η παγκόσμια κορυφή για τον ΗΙΦ, όπως και για άλλα standards που αφορούν στην Ιατρική Πληροφορική.

Κάποιοι οργανισμοί έκαναν χρήση ήδη συγκεκριμένων προτυποποιήσεων διεθνών οργανισμών, όπως είναι το ISO. Μερικοί από τους οργανισμούς που το έκαναν αυτό είναι οι DICOM, IEEE, CEN, HL7. Είναι γνωστό άλλωστε πως οι τρεις τελευταίοι οργανισμοί έχουν ειδική συμφωνία με τον ISO που εξουσιοδοτούν τις υπάρχουσες προτυποποιήσεις προκειμένου να γίνουν πρότυπα κατά ISO.

Ο οργανισμός ISO/TC 215 έχει έξι ομάδες εργασίας (working groups) οι οποίες είναι οι παρακάτω:

WG1: Ιατρικοί φάκελοι και συντονισμός των μοντέλων. Η επιδίωξη είναι ένα πρότυπο ιατρικού φακέλου, όπου η κατάλληλη πληροφορία θα είναι διαθέσιμη όταν και όπου απαιτείται η υποστήριξη αποφάσεων.

WG2: Μετάδοση πληροφορίας και επικοινωνία

WG3: Αναπαράσταση ιατρικών ήχων

WG4: Ασφάλεια

WG5: Ιατρικές κάρτες

WG6: Ηλεκτρονικό φαρμακείο

5.3 CEN / TC 251

Η Ευρωπαϊκή Επιτροπή Τυποποίησης (Committee European Normalization – The European Standards Organization – CEN) έχει δημοσιεύσει ένα PreStandard για την αρχιτεκτονική ηλεκτρονικού ιατρικού φακέλου με την ονομασία ENV 13606, το οποίο δημοσιεύθηκε το 2000.

Σκοπός του CEN είναι να παράγει μια ακριβή, άκαμπτη και μεγάλη σε διάρκεια αρχιτεκτονική η οποία να παριστάνει τον Ηλεκτρονικό Ιατρικό Φάκελο. Στόχος είναι να υποστηρίξει την διαλειτουργικότητα των συστημάτων καθώς και τις συνιστώσες οι οποίες χρειάζονται για να αλληλεπιδρούν οι υπηρεσίες του Ηλεκτρονικού Ιατρικού Φακέλου ως

- Διακριτά συστήματα
- Να έχει πρόσβαση, να μεταφέρει, να προσθέτει καθώς και να μορφοποιεί διάφορες εισόδους νέων ιατρικών φακέλων
- Να κάνει χρήση ηλεκτρονικών μηνυμάτων ή καταναεμημένων αντικειμένων
- Να διατηρεί το αρχικό κλινικό δεδομένο που προηγείται απο τον σχεδιαστή του

5.4 Το Πρότυπο Health Level Seven

Ο οργανισμός Health Level Seven Inc. (HL7) σχηματίστηκε το 1987 στις Η.Π.Α. με σκοπό την ανάπτυξη προτύπων σχετικά με την ηλεκτρονική ανταλλαγή δεδομένων και την αυτόματη ανταλλαγή πληροφορίας μεταξύ των διαφορετικών πληροφοριακών συστημάτων στην υγειονομική περίθαλψη.

Το HL7 είναι το πλέον ευρέως χρησιμοποιημένο πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων σε κλινικό περιβάλλον. Χρησιμοποιείται σε όλες τις ηπείρους. Εάν περιοριστεί κανείς στην Ευρώπη θα δει ότι χρησιμοποιείται σχεδόν σε κάθε χώρα ως πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων ανάμεσα στα διάφορα υποσυστήματα.

Σχεδόν όλα τα ευφυή διαγνωστικά μηχανήματα (ιατροτεχνολογικός εξοπλισμός) μπορούν να "μιλήσουν" HL7 και σχεδόν όλα τα ιατρικά πληροφοριακά συστήματα υψηλού επιπέδου είναι σε θέση να στείλουν και να λάβουν τα κατάλληλα HL7 μηνύματα, χρησιμοποιώντας τους κανόνες ανταλλαγής μηνυμάτων του HL7 (του πρωτοκόλλου).

Επίσης το HL7 είναι ξεκάθαρα το πιο ώριμο πρότυπο ανταλλαγής πληροφοριών μέσω μηνυμάτων. Η έρευνα από την ακαδημαϊκή κοινότητα και την βιομηχανία και τις εταιρίες συμβούλων οδήγησε σ' αυτό το πρότυπο, την κυριότητα του οποίου την κατέχει ο μη κερδοσκοπικός οργανισμός Health Level Seven Inc. Ο οποίος έχει τοπικά υποκαταστήματα σε όλες σχεδόν τις χώρες της Ευρώπης, στις Ηνωμένες

Πολιτείες της Αμερικής, στην Αυστραλία / Νέα Ζηλανδία, την Ασία και στη ζώνη του Ειρηνικού.

Το πρότυπο HL7 έχει αναγνωριστεί από πολλά εθνικά ιδρύματα προτυποποίησης, όπως ο ANSI (USA) και ο DIN (Γερμανία). Επίσης, το HL7 χρησιμοποιείται καθημερινά σε εκατοντάδες νοσοκομεία σε όλο τον κόσμο, συνδέοντας μια μεγάλη ποικιλία εφαρμογών και συστημάτων.

Η αποστολή του οργανισμού "HL7 Inc." είναι η δημιουργία αξιόπιστων προτύπων ανταλλαγής, διαχείρισης και ολοκλήρωσης δεδομένων που αφορούν την κλινική φροντίδα του ασθενή, και την διαχείριση, οργάνωση και αξιολόγηση υπηρεσιών ιατρικής περίθαλψης.

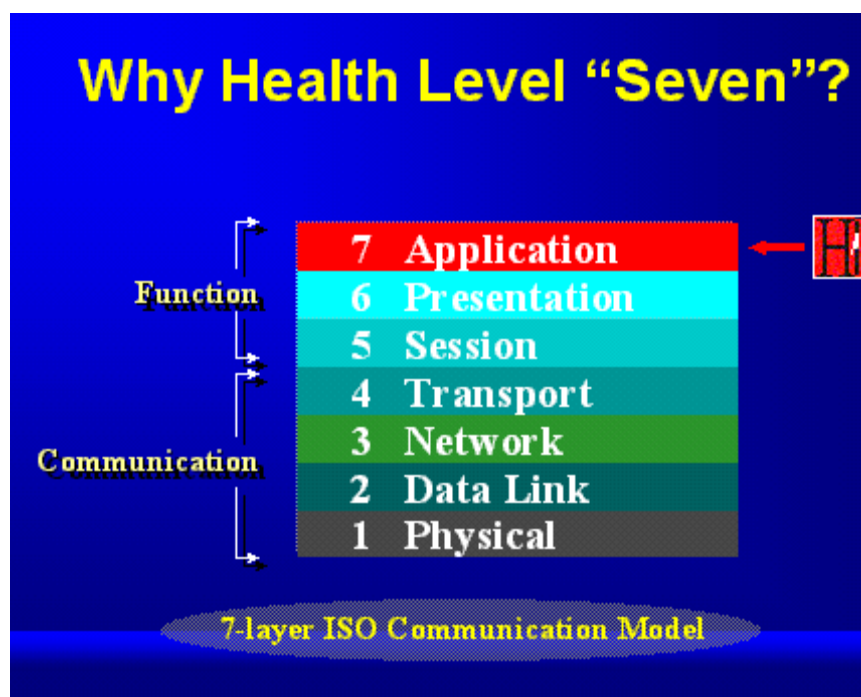
Ο οργανισμός ενθαρρύνει τη δημιουργία ευέλικτων προτύπων, οδηγιών, μεθοδολογιών, πρωτοκόλλων και άλλων συναφών υπηρεσιών και προϊόντων, προκειμένου να καταστεί εφικτή η διαλειτουργικότητα πληροφοριακών συστημάτων στην Υγεία - Πρόνοια και η ανταλλαγή στοιχείων του ηλεκτρονικού φακέλου ασθενή.

Ο οργανισμός "HL7 Inc." δημιουργήθηκε προκειμένου να λειτουργεί ως αξιόπιστο μέσο επικοινωνίας μεταξύ των ενδιαφερομένων φορέων στον τομέα της ιατρικής περίθαλψης, γεγονός που αποτυπώνεται στην ποικιλία που παρουσιάζουν τα μέλη του όπως εταιρίες ιατρικής πληροφορικής, ιδιωτικοί και δημόσιοι φορείς υγείας - πρόνοιας, ειδικοί σύμβουλοι, εμπειρογνώμονες, εταιρίες ολοκλήρωσης πληροφοριακών συστημάτων (system integrators), ασφαλιστικοί φορείς, εταιρίες ιατροτεχνολογικού εξοπλισμού, φορείς παροχής υπηρεσιών υγείας - πρόνοιας, κλπ.

Αναγνωρίζοντας λοιπόν την ανάγκη υποστήριξης των τοπικών ομάδων που δραστηριοποιούνται στην προώθηση των προτύπων, ο "HL7 Inc." στηρίζει τις προσπάθειες αυτές με την δημιουργία τοπικών παραρτημάτων (HL7 affiliates). Μέχρι σήμερα έχουν ήδη ιδρυθεί 23 τέτοια παραρτήματα (Ηνωμένο Βασίλειο, Καναδάς, Αυστραλία, Νέα Ζηλανδία, Νότιος Αφρική, Γερμανία, Ολλανδία, Φιλανδία, Ινδία, Ιαπωνία, Αργεντινή, Κίνα, Κορέα, Τσεχία, Λιθουανία, Ελβετία, Βραζιλία, Κροατία, Μεξικό, Ιταλία, Δανία και Ταϊβάν).

Τα τοπικά παραρτήματα είναι ανεξάρτητοι οργανισμοί διεθνούς χαρακτήρα που στοχεύουν στην ανάπτυξη, υποστήριξη, αποδοχή και χρήση των προτύπων HL7 σε παγκόσμια κλίμακα με την μεταφορά αυτών στην αντίστοιχη γλώσσα του παραρτήματος.

Το HL7 προτυποποιεί τα πρωτόκολλα και τις δομές για την ανταλλαγή μηνυμάτων ιατρικού ενδιαφέροντος στο επίπεδο της εφαρμογής, στο επίπεδο του μοντέλου OSI, δηλαδή είναι ανεξάρτητο από συγκεκριμένες πλατφόρμες και τεχνολογίες.



Εικόνα 6: Πρότυπο HL7

Οι εκδόσεις HL7 2.x παρά την ευρύτερη αποδοχή και τις υλοποιήσεις, παρουσιάζουν αρκετά μειονεκτήματα, και συγκεκριμένα:

- Δεν υπάρχει ένα λογικό μοντέλο αναφοράς της πληροφορίας που ανταλλάσσεται στα μηνύματα, ούτε τρόπος αναπαράστασης της σχέσης μεταξύ των δεδομένων.
- Χρησιμοποιεί πολύ ειδική σύνταξη στα μηνύματα, καθιστώντας δύσκολη την εκμάθηση και την υλοποίηση του προτύπου
- Έχει πολλά προαιρετικά χαρακτηριστικά, κάτι που παρέχει ευελιξία και συνεισφέρει αποφασιστικά στη διάδοσή του, αλλά και που καθιστά σχεδόν αδύνατο τον έλεγχο της συμμόρφωσης προς το πρότυπο των διαφόρων υλοποιήσεων. Έτσι απαιτείται μεγάλη προσπάθεια για να εξασφαλισθεί ότι οι δύο εφαρμογές που θα «μιλήσουν μεταξύ τους, χρησιμοποιούν τα ίδια χαρακτηριστικά».

Η έκδοση HL7 Version 3, αντιμετωπίζει ουσιστικά τα παραπάνω θέματα

5.5 Το πρότυπο DICOM

Η λέξη **DICOM** είναι συντομογραφία του **Digital Imaging and Communication in Medicine** και αναφέρεται στο πρότυπο των ACR – NEMA (American College of Radiology - National Electrical Manufacturers Association) και αναπτύχθηκε με σκοπό την κάλυψη της ανάγκης διασύνδεσης διαφόρων ιατρικών απεικονιστικών μηχανημάτων αρχικά με σύνδεση ανά δύο συσκευές και στη συνέχεια σε δίκτυο.

Το πρότυπο DICOM δομήθηκε σαν κείμενο με πολλά τμήματα (multi-part document) με βάση την οδηγία ISO/IEC Directive 1989 part 3: Drafting and presentation of International Standards. Η σχεδίαση του DICOM σε πολλά τμήματα (parts)

εξυπηρετεί την εύκολη επέκταση του προτύπου σε καινούργιες απαιτήσεις, σε όλες τις μορφές τηςιατρικής απεικόνισης.

Το πρότυπο του DICOM δίνει επίσης ένα μέσο στους χρήστες των μηχανημάτων ιατρικής απεικόνισης να μπορούν να αποφανθούν αν δύο συσκευές που ισχυρίζονται ότι συμμορφώνονται με το πρότυπο μπορούν να ανταλλάξουν μεταξύ του μια ελάχιστη πληροφορία με νόημα.

Μεταγενέστερες προσθήκες στο DICOM έχουν συμπεριλάβει τη δημιουργία αρχείων σε μεταφερόμενα μέσα (όπως οπτικοί δίσκοι και μαγνητικές ταινίες), και καινούργιες δομές δεδομένων όπως π.χ. για την αγγειογραφία και το χειρισμό της εκτύπωσης των εικόνων σε φιλμ και χαρτί.

Το DICOM ψηφίστηκε και εγκρίθηκε το 1992 στο ετήσιο συνέδριο του RSNA (Radiology Society of North America) και συγκεκριμένα το part 1 (Introduction and Overview) και το part 8 (Network Communication Support for MessageExchange).

Από τότε το πρότυπο DICOM βρίσκεται σε συνεχή εξέλιξη και αυτή τη στιγμή βρίσκεται στην έκδοση DICOM Version 3.0 με τις τελευταίες αλλαγές να έχουν γίνει το 1999 και σε συνεργασία πλέον με τους οργανισμούς τυποποίησης CEN TC251 της Ευρώπης και του JIRA στην Ιαπωνία με την εποπτεία από άλλους οργανισμούς όπως IEEE, HL7 και ANSI.

Ένα απλό μοντέλο κατανεμημένης διεργασίας μπορεί να εξηγήσει τους μηχανισμούς και την ορολογία του προτύπου DICOM.

Μια κατανεμημένη διεργασία έχει τουλάχιστον δύο επιμέρους διεργασίες που μοιράζονται πληροφορίες και η κάθε μια στηρίζεται σε δεδομένα που θα πάρει από την άλλη. Ένας αριθμός κατανεμημένων διεργασιών που δρουν από κοινού συνήθως ορίζουν μια υπηρεσία (service). Στις περισσότερες κατανεμημένες διεργασίες, οι διαδικασίες που αφορούν την εφαρμογή είναι ανεξάρτητες από τις διαδικασίες που αφορούν την επικοινωνία μεταξύ των διεργασιών.

Πριν από τη κοινή δράση κατανεμημένων διεργασιών πρέπει να καθοριστούν οι ρόλοι κάθε πλευράς. Αρχικά πρέπει να καθοριστεί ο ρόλος του client και ο ρόλος του server. Η πλευρά που χρησιμοποιεί τις δυνατότητες της άλλης πλευράς έχει συνήθως το ρόλο του client ενώ η άλλη το ρόλο του server. Οι απαιτήσεις που έχει η μία πλευρά από την άλλη καθορίζουν τη *σχέση (relation)* που τις διέπει. Η σχέση καθορίζει κάτω από ποιες συνθήκες θα αρχικοποιηθεί η διεργασία. Στις περισσότερες περιπτώσεις το έναυσμα το δίνει ο client αλλά μερικές φορές την αρχικοποίηση την κάνει ο server. Εκτός από τους ρόλους πρέπει να συμφωνηθεί ποιες πληροφορίες (information) θα ανταλλάγουν. Σε αυτό το σημείο μας ενδιαφέρει η ανταλλαγή πληροφοριών σημασιολογικά και όχι από την άποψη του τρόπου που παρουσιάζεται και κωδικοποιείται (syntax).

Η πληροφορία καθορίζεται από το *περιβάλλον (context)* της υπηρεσίας (service) και από την διεργασία που υλοποιείται. Κάθε ξεχωριστή διεργασία έχει μια επιλεκτική όψη της πληροφορίας, αλλά αυτή η όψη πρέπει να συμβαδίζει με όλο το περιβάλλον της διεργασίας.

Η λειτουργία (operation) καθορίζει την επεξεργασία που θα υποστεί η ανταλλαγμένη πληροφορία στην άλλη πλευρά π.χ. αποθήκευση ή επιστροφή κάποιου αποτελέσματος. Με όλα τα προηγούμενα θέματα ασχολείται το *application domain* της κατανεμημένης διεργασίας. Ο συνδυασμός των context, relation, operation και

information είναι ο ακρογωνιαίος λίθος κάθε κατανεμημένης διεργασίας και πρέπει να καθοριστούν πριν πραγματοποιηθεί κάθε επιτυχημένη υλοποίηση αυτής της διεργασίας.

Ο ακριβής τρόπος ανταλλαγής των δεδομένων δεν είναι θέμα του application domain αλλά υπηρεσιών χαμηλότερων επιπέδων (π.χ. TCP/IP) που κανονίζονται από το *exchange domain* της διεργασίας. Ο client και ο server πρέπει να είναι σε θέση να χρησιμοποιούν τις υπηρεσίες χαμηλότερων επιπέδων. Οι υπηρεσίες των χαμηλότερων επιπέδων χειρίζονται την ανταλλαγή της πληροφορίας και γενικά δεν είναι εμφανείς από το application domain (είτε του client είτε του server).

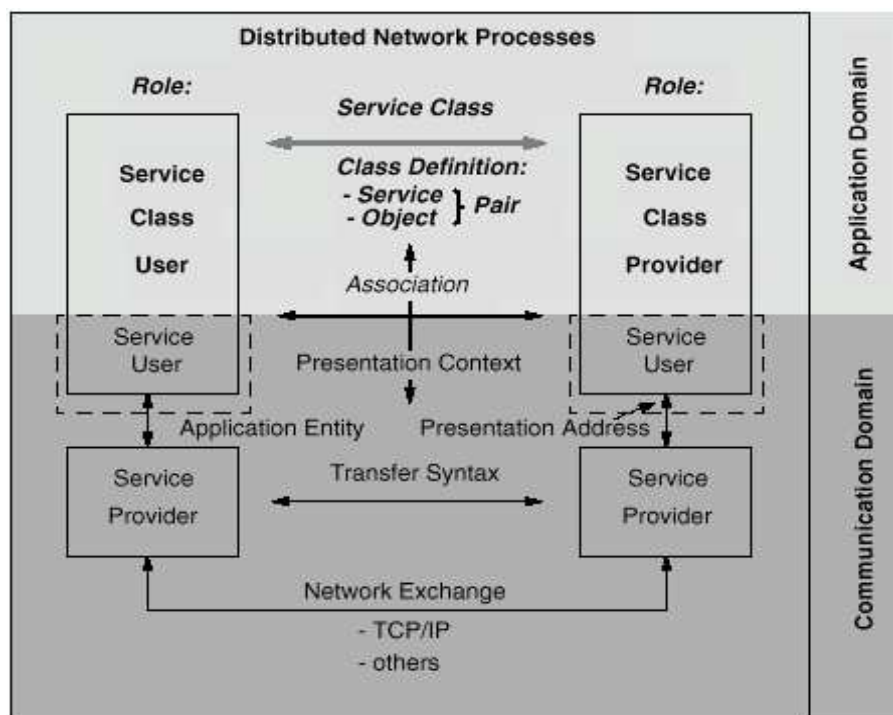
Το μέλος εκείνο που ζητά κάποια υπηρεσία είναι ο *service user* ενώ το άλλο μέλος είναι ο *service provider*. Οι δύο πλευρές μπορεί να έχουν εντελώς διαφορετική υλοποίηση αλλά αυτό που έχει σημασία είναι να έχουν κοινή γνώση για τον τρόπο ανταλλαγής των δεδομένων (protocol) και να έχουν το ίδιο λογικό interface μεταξύ του service user και service provider.

Ακόμη πρέπει και οι δύο πλευρές να γνωρίζουν τον τρόπο που η πληροφορία αντιπροσωπεύεται σε bits και bytes. Ο service provider πρέπει να προσδιορίζει το format που η πληροφορία μεταφέρεται και να την μετατρέψει στην *αναπαράσταση* (*representation*) εκείνη, που είναι κατανοητή από το application domain.

Μετά την ανταλλαγή της πληροφορίας οι δύο πλευρές πρέπει να έχουν την ίδια πληροφορία ανεξαρτήτως των πόσων μετατροπών αυτή υποβλήθηκε και του τρόπου μεταφοράς της πληροφορίας.

Η φυσική ανταλλαγή (*physical exchange*) μεταξύ των service providers κάθε πλευράς μπορεί να γίνει είτε μέσω του δικτύου είτε μέσω των μέσων (π.χ. οπτικών δίσκων ή μαγνητικών ταινιών).

Στην παρακάτω φαίνονται οι αρχές λειτουργίας DICOM σ' ένα δίκτυο.



Εικόνα 7 : Αρχές λειτουργίας DICOM σε δίκτυο

5.6 Το πρότυπο PACS

Πρόκειται για τα αρχικά του *Picture Archiving and Communication Systems*. Σημαντικό κομμάτι για τον ηλεκτρονικό ιατρικό φάκελο παίζει και η διαδικασία μεταφοράς και επεξεργασίας της εικόνας. Το PACS είναι ένα πρότυπο με τέτοια χαρακτηριστικά, με στόχο να αρχειοθετεί, διαχειρίζεται, διανέμει και αποθηκεύει ιατρικές εικόνες και δεδομένα, με τρόπο ώστε η πρόσβαση σε αυτά μέσα σε κατάλληλα διαμορφωμένο δίκτυο, από εξουσιοδοτημένα τερματικά, να είναι βατή.

Οι εικόνες με τη βοήθεια του PACS μπορούν να προβληθούν σε οθόνες όπως, σωλήνες καθοδικών ακτινών, ή σε flat panel. Είναι επίσης εφικτό να καταγραφούν με video ή laser camera, σε φωτογραφικό φιλμ που μετά τη χημική επεξεργασία μπορεί να προβληθούν σε κατάλληλες οθόνες. Τέλος, όπου χρειαστεί, η ψηφιακή εικόνα μετατρέπεται σε αναλογική με ένα DAC (Digital to Analog Converter).

5.6.1 Πλεονεκτήματα των PACS

- Άμεση πρόσβαση σε λήθος εικόνων από πολλούς χρήστες
- Δυνατότητα επεξεργασίας της εικόνας και σύγκρισής της με παλαιότερες
- Μείωση της απώλειας εξετάσεων –φιλμ
- Μείωση χώρου αποθήκευσης των εξετάσεων-αναφορών
- Δυνατότητα διάγνωσης με τη βοήθεια λογισμικού

5.6.2 Μειονεκτήματα των PACS

- Συχνή ανανέωση εξοπλισμού
- Ανάγκη συνεχούς παρουσίας τεχνικού προσωπικού υποστήριξης του συστήματος
- Χαμηλότερο δυναμικό εύρος σε σχέση με το φιλμ
- Χειρότερη διακριτική ικανότητα στην οθόνη σε σχέση με το φιλμ
- Χρόνος προσαρμογής των ακτινολόγων στη νέα τεχνολογία
- Ασφάλεια και αξιοπιστία (πιθανή πρόσβαση μη εξουσιοδοτημένων ατόμων)

Κεφάλαιο 6

Ασφάλεια Πληροφοριακών Συστημάτων

6.1 Εισαγωγή

Το πρόβλημα της ασφάλειας των πληροφοριών και της προστασίας των προσωπικών δεδομένων είναι ιδιαίτερα σημαντικό στα σύγχρονα Πληροφοριακά Συστήματα και πρωτίστης προτεραιότη-τας στον τομέα της υγείας .Τα ιατρικά αρχεία ενός ασθενούς αποτελούν ιδιαίτερα ευαίσθητα προσωπικά δεδομένα. Αυτό σημαίνει ότι όποιος τα αναλύει ή έχει πρόσβαση σε αυτά, πρέπει να είναι άτομο το οποίο δε θα τα χρησιμοποιήσει προς ίδιον όφελος.

Η ασφάλεια των ιατρικών δεδομένων είναι ένα σημαντικότερο θέμα για το οποίο,ωστόσο, η τεχνολογία έχει δώσει ουσιαστικές λύσεις, οι οποίες μάλιστα μπορεί να θεωρηθούν αποτελεσματικότερες από αυτές που μέχρι σήμερα εφαρμόζονται για την τήρηση και φύλαξη των ιατρικών φακέλων των ασθενών.

Στον ηλεκτρονικό ιατρικό φάκελο πρέπει να δίνεται ιδιαίτερη έμφαση στην προστασία των προσωπικών δεδομένων τα οποία θα αρχειοθετούνται. Λόγω της ευαισθησίας των προσωπικών στοιχείων είναι επιτακτική η ανάγκη να πληρούνται όλες εκείνες οι προϋποθέσεις ασφάλειας που θα εξασφαλίζουν το αδιάβλητων δεδομένων.

6.2 Η Ασφάλεια Ιατρικών Δεδομένων

Η έννοια της ασφάλειας των Πληροφοριακών Συστημάτων συνδέεται στενά με τρεις βασικές έν-νοιες :

- Την Εμπιστευτικότητα(Confidentiality)
- Την Ακεραιότητα(Integrity), και
- Την Διαθεσιμότητα(Availability)

Η αρχή της **εμπιστευτικότητας** αφορά την προστασία των δεδομένων (στοιχεία ασθενή,περιεχόμενο ιατρικών εγγράφων) ενάντια σε μη εξουσιοδοτημένη πρόσβαση ή γνωστοποίησή τους. Συνεπώς, μόνο εξουσιοδοτημένοι χρήστες (άτομα ή κατηγορίες προσωπικού) μπορούν να προσπε-λάσουν συγκεκριμένες πληροφορίες . **Ακεραιότητα** είναι η προστασία των δεδομένων πό μη εξουσιοδοτημένη τροποποίηση ή αντικα-τάστασή τους .

Οι πληροφορίες είναι **διαθέσιμες** (24ώρες το 24ωρο) σε εξουσιοδοτημένους χρήστες. Για παρά- δειγμα , κατά την αντιμετώπιση ενός κρίσιμου περιστατικού , το περιεχόμενο του ηλεκτρονικού φακέλου του αθενή πρέπει να είναι άμεσα διαθέσιμο στο γιατρό που χειρίζεται το περιστατικό. Αλλιώς , ενδέχεται να αγνοηθούν πληροφορίες μείζονος σημασίας και να προκληθούν ιατρικά λάθη .

Το πρόβλημα της ασφάλειας των ιατρικών πληροφοριακών συστημάτων μπορεί να αναλυθεί σε τέσσερις βασικές συνιστώσες :

1. Την φυσική ασφάλεια (physical security)

Αντικείμενό της είναι η προστασία του ίδιου του υπολογιστή και του σχετικού εξοπλισμού από φυσικές καταστροφές όπως : κλοπή , φωτιά , πλημμύρες , βανδαλισμούς αλλά και από εμπλοκή ανθρώπων που δεν έχουν καμία δικαιοδοσία στη χρήση τους .

2. Την ασφάλεια του λειτουργικού συστήματος

Εκτός από την ασφάλεια του υλικού , σημαντική είναι και η προστασία του λειτουργικού συστήματος που διαχειρίζεται τους πόρους του υπολογιστή , αλλά και των ειδικών προγραμμάτων εφαρμογών που διαχειρίζονται τα δεδομένα . Η δυσλειτουργία του λειτουργικού συστήματος μπορεί να προκαλέσει την απώλεια των δεδομένων ή τον πλήρη απο-συντονισμό των λειτουργιών του συστήματος , με άμεση συνέπεια την απώλεια της λειτουργικότητάς του .

3. Την ασφάλεια των βάσεων δεδομένων(database security)

Αντικείμενό της είναι η ικανότητα του συστήματος να εφαρμόσει μια προκαθορισμένη πολιτική προστασίας των πληροφοριών που αφορά την δυνατότητα προσπέλασης ,την διαθεσιμότητα και την δυνατότητα τροποποίησης ή διαγραφής των πληροφοριών της βά-σης δεδομένων .

4. Την ασφάλεια των Δικτύων επικοινωνιών του συστήματος (network security)

Ο κύριος στόχος των δικτύων είναι η διασφάλιση της μετάδοσης των δεδομένων που διακινούνται δια μέσου του δικτύου , ώστε να καταλήγουν στον προορισμό τους χωρίς προσθήκες ,αφαιρέσεις ή αλλαγές .

6.3 Απειλές Ασφάλειας

Με τον όρο απειλή κατά της ασφάλειας ενός Υπολογιστικού Συστήματος ορίζεται η πιθανή εκμετάλλευση μιας ευπάθειας του συστήματος με δυνητικό κίνδυνο την μη εξουσιοδοτημένη πρόσβαση, την αποκάλυψη πληροφοριών, την χρήση, την κλοπή ή την καταστροφή των πόρων του συστήματος.

Όπως αναφέρεται στο βιβλίο 'Ασφάλεια Πληροφοριών ' της Ελληνικής Εταιρείας Επιστημόνων Ηλεκτρονικών Υπολογιστών και Πληροφορικής οι πιο γνωστές απειλές είναι οι ακόλουθες :

- Μη εξουσιοδοτημένη χρήση κατά την οποία επιχειρείται προσπέλαση στα δεδομένα ή τις προσφερόμενες υπηρεσίες του δικτύου από μη εξουσιοδοτημένους χρήστες (hackers) οι οποίοι προσπαθούν να αποκτήσουν πρόσβαση στο δίκτυο με αθέμιτα μέσα .
- Μη ενεργή παρακολούθηση κατά την οποία απειλείται η εμπιστευτικότητα των ανταλλασσόμενων μηνυμάτων από μη ενεργούς παρεμβολείς της διεξαγόμενης επικοινωνίας ,δηλαδή από άτομα που υποκλέπτουν πληροφορίες χωρίς να αλλάζουν το περιεχόμενό τους .
- Ενεργή παρακολούθηση κατά την οποία επιχειρείται τροποποίηση των ανταλλασσόμενων μηνυμάτων στο δίκτυο . Αν και ο ενεργός παρεμβολέας μπορεί να εντοπισθεί ευκολότερα από ένα μη ενεργό , θεωρείται πιο επικίνδυνος γιατί μπορεί να προκαλέσει μεγαλύτερη ζημιά στο δίκτυο , να εισάγει τα δικά του δεδομένα και να κατευθύνει τα μηνύματα σύμφωνα με τις επιθυμίες του .

- Καταλογισμός ευθύνης όπου ένας εξουσιοδοτημένος χρήστης μπορεί να αρνηθεί την αποστολή ή την παραλαβή ενός συγκεκριμένου μηνύματος ή ακόμα και να κατασκευάσει ένα μη έγκυρο μήνυμα .
- Άρνηση εξυπηρέτησης κατά την οποία το δίκτυο δεν ανταποκρίνεται στο απαιτούμενο επίπεδο εξυπηρέτησης και λειτουργικότητας . Η απειλή αυτή μπορεί να προκαλέσει απώλεια μηνυμάτων ή καθυστερήσεις , ενώ είναι πιθανό να συμβαίνουν μικρές ή μεγάλες διακοπές της λειτουργίας του .
- Ανάλυση επικοινωνίας κατά την οποία παρακολουθείται η μετάδοση των μηνυμάτων όχι απαραίτητα για την αποκάλυψη του περιεχομένου τους , αλλά για τον εντοπισμό της προέλευσης και της αποστολής τους .
- Οι Ιοί μπορούν να καταστρέψουν τα δεδομένα , να παραποιήσουν την ακεραιότητά τους ή να προκαλέσουν μικρότερα προβλήματα .

6.4 Μέθοδοι Αντιμετώπισης Απειλών

Κάθε οργανισμός υγείας που χρησιμοποιεί πληροφοριακά συστήματα στη διεκπεραίωση των διαδικασιών του , πρέπει να υιοθετεί μια αυστηρή πολιτική ασφαλείας για να προστατεύει τόσο τα συστήματά του όσο και τα δεδομένα που διαχειρίζεται . Καταρχήν είναι απαραίτητο να δοθεί προτεραιότητα σε θέματα ασφαλείας από την ίδια την διοίκηση και να λαμβάνονται άμεσες και γενναίες αποφάσεις .Με αυτό τον τρόπο θα ‘περάσει’ και στους ίδιους τους εργαζόμενους το μήνυμα της ασφαλούς διαχείρισης των πληροφοριών, της προστασίας των συστημάτων, της αναγνώρισης των κινδύνων, ευαισθητοποιώντας τους σε θέματα που προκύπτουν από την ανάπτυξη και συντήρηση ανασφαλών πληροφοριακών συστημάτων .Αυτά τα βήματα αποτελούν τη μη τεχνική παρέμβαση σε θέματα ασφαλείας .

6.5 Τεχνικές πρόσβασης και ανταλλαγής δεδομένων

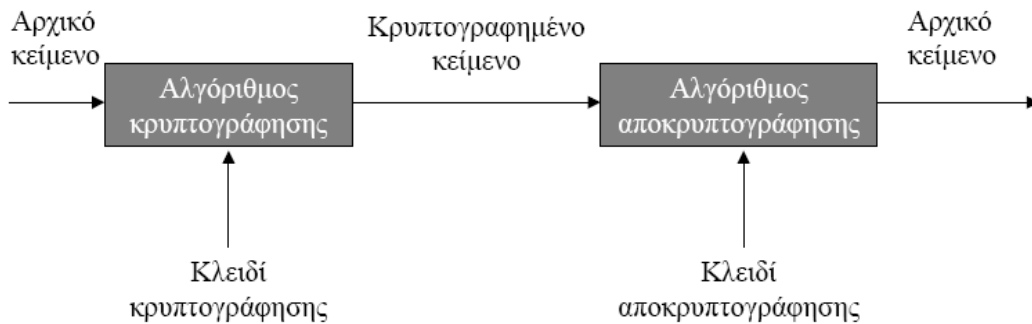
Οι τεχνικές πρόσβασης και ανταλλαγής δεδομένων περιλαμβάνουν διάφορες μεθόδους διασφάλισης της εξουσιοδοτημένης πρόσβασης των χρηστών σε αυτά .

Η **πιστοποίηση** (authentication) είναι η πιο γνωστή μέθοδος και αναφέρεται στις διαδικασίες αναγνώρισης και επιβεβαίωσης της ταυτότητας ενός ατόμου . Η μέθοδος αυτή μπορεί να υλοποιηθεί με τη χρήση κωδικών (password) , με τη χρήση ενός ηλεκτρονικού μέσου ή ακόμα και μέσω βιομετρικών μεθόδων ταυτοποίησης προσώπων , όπως για παράδειγμα με την αναγνώριση των δακτυλικών αποτυπωμάτων , της φωνής και της ίριδας του ματιού του ατόμου . Ακολούθως ,δίνεται η εξουσιοδότηση για περαιτέρω πρόσβαση στα δεδομένα .

Μια άλλη πολύ διαδεδομένη τεχνική είναι η αφάλεια που παρέχεται από την **Έμπιστη Τρίτη Οντότητα** . Όπως δηλώνεται και από το όνομα , η Έμπιστη Τρίτη Οντότητα αναμειγνύεται στη διαδικασία της συναλλαγής μεταξύ δυο οντοτήτων ,όπως για παράδειγμα κατά την ανταλλαγή μηνυμάτων μεταξύ των πληροφοριακών συστημάτων δυο νοσοκομείων. Πρόκειται για τον έλεγχο της διαδικασίας επικοινωνίας ώστε , αφενός ο αποστολέας δεδομένων να μην μπορεί να αρνηθεί τη δημιουργία και αποστολή του μηνύματος , αφετέρου ο παραλήπτης να μη μπορεί να αρνηθεί την παραλαβή ενός μηνύματος .

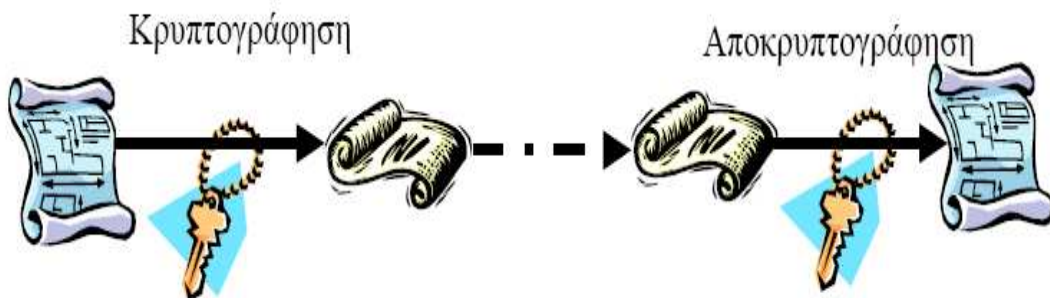
Πολύ σημαντικός είναι και ο ρόλος των **συστημάτων κρυπτογράφησης** . Σε ένα σύστημα κρυπτογράφησης ο κρυπτογράφος μετασχηματίζει το αρχικό κείμενο σε κρυπτογράφημα , δηλαδή ένα κωδικοποιημένο κείμενο , χρησιμοποιώντας ένα κλειδί και ο αποκρυπτογράφος εκτελεί την αντίστροφη διαδικασία με το ίδιο ή κάποιο άλλο κλειδί . Ουσιαστικά , συμμετέχουν πάλι δυο οντότητες μόνο που δεν εμποτεύονται από κάποια άλλη , όπως στην περίπτωση της Έμπιστης Τρίτης Οντότητας . Σήμερα υπάρχουν δυο κατηγορίες κρυπτογραφικών συστημάτων για την προστασία των πληροφοριών :

- Συστήματα μυστικού κλειδιού
- Συστήματα δημόσιου κλειδιού



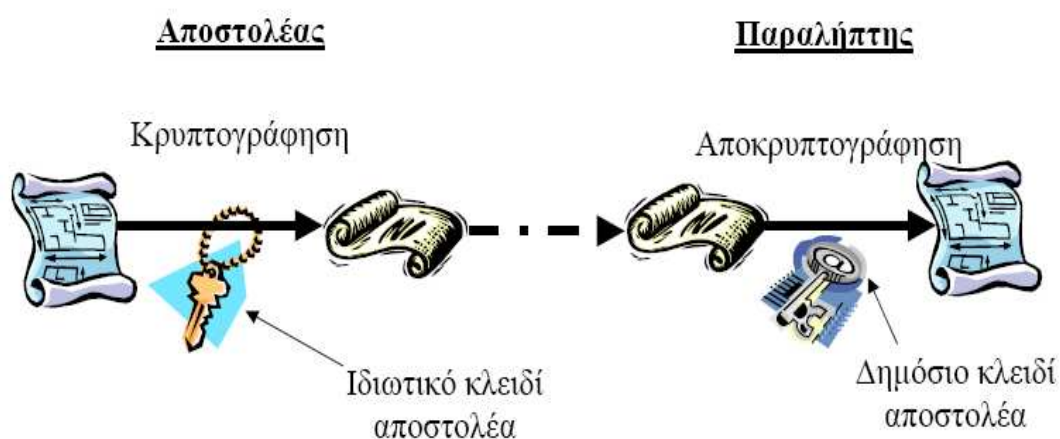
Εικόνα 8 : Σύστημα Κρυπτογράφησης

Στα **συστήματα μυστικού κλειδιού** χρησιμοποιείται το ίδιο κλειδί για την κρυπτογράφηση και αποκρυπτογράφηση του μηνύματος . Η διαδικασία περιγράφεται συνοπτικά ως εξής : Ο κρυπτογράφος παράγει το κρυπτογράφημα με το μυστικό κλειδί και ένα αλγόριθμο και το αποστέλλει μέσω ενός διαύλου επικοινωνίας στον παραλήπτη. Παράλληλα , μέσω ενός άλλου ασφαλούς διαύλου ο παραλήπτης ο παραλήπτης λαμβάνει από τον αποστολέα και το μυστικό κλειδί . Ακολούθως , αποκρυπτογραφείται το κρυπτογράφημα με το μυστικό κλειδί και τον αντίστοιχο αλγόριθμο , άποτε και προκύπτει το αρχικό μήνυμα. Στην περίπτωση δυο νοσοκομείων που ανταλλάσσουν πληροφορίες , το μήνυμα αυτό μπορεί να είναι μια απλή ιατρική γνωμάτευση , κάποια οικονομικά στοιχεία ή άλλα διοικητικά έγγραφα .



Εικόνα 9 : Κρυπτογραφικό σύστημα μυστικού κλειδιού

Τα συστήματα δημόσιου κλειδιού χρησιμοποιούν δυο διαφορετικά κλειδιά , ένα δημόσιο για την κρυπτογράφηση και ένα μυστικό για την αποκρυπτογράφηση . Παρόλο που τα κλειδιά αυτά σχετίζονται , η γνώση του δημόσιου κλειδιού δεν καθιστά εφικτό τον υπολογισμό του μυστικού κλειδιού . Το δημόσιο κλειδί γίνεται ευρέως διαθέσιμο , οπότε οι υπόλοιποι χρήστες μπορούν να κρυπτογραφούν μηνύματα με αυτό και να είναι σίγουροι ότι μόνο ο κάτοχος του μυστικού κλειδιού μπορεί να διαβάσει αυτά τα μηνύματα . Ο τρόπος λειτουργίας ενός τέτοιου συστήματος φαίνεται στο παρακάτω σχήμα :



Εικόνα 10: Κρυπτογραφικό σύστημα δημόσιου κλειδιού

Μια βασική εφαρμογή κρυπτογράφησης είναι και η ψηφιακή υπογραφή . Η ψηφιακή υπογραφή βοηθά τον παραλήπτη να πιστοποιήσει την αφετηρία ενός μηνύματος , την ακεραιότητα του περιεχομένου του και το ότι ο αποστολέας δεν θα διαψεύσει την αποστολή του μηνύματος . Ο αποστολέας από την πλευρά του διασφαλίζει τη μη άρνηση της παραλαβής του μηνύματος από τον παραλήπτη . Ένα ασφαλές σύστημα ψηφιακών υπογραφών αποτελείται από δυο μέρη :

- Υπογραφή ενός κειμένου από τον αποστολέα με ορθό τρόπο
- Επαλήθευση της ψηφιακής υπογραφής από τον παραλήπτη

Εκτος από τις τεχνικές που προαναφέρθηκαν , δοκιμάζονται διαρκώς νέες και πιο πολύπλοκες μέθοδοι προστασίας των δεδομένων της υγείας .Οι προαναφερθείσες έχουν δοκιμαστεί με επιτυχία και εξακολουθούν να βρίσκουν πολλές εφαρμογές .

Τέλος , στην περίπτωση των συστημάτων βάσεων δεδομένων , το σύστημα που διαχειρίζεται τη βάση διαθέτει πολλούς μηχανισμούς ασφάλειας .Καταρχήν ,οι συγκεκριμένοι μηχανισμοί καθορίζουν πολλαπλά επίπεδα πρόσβασης , οπότε κάθε χρήστης έχει άποψη των δεδομένων που τον ενδιαφέρουν, ενώ του αποκρύπτονται εκείνα που δεν πρέπει να δει .

Εκτός από την εμπιστευτικότητα των δεδομένων , οι μηχανισμοί αυτοί προστατεύουν και την ακεραιότητά τους. Έτσι , για συγκεκριμένα πεδία δεν επιτρέπεται η εισαγωγή μη πραγματικών ή ανακριβών τιμών . Επιπλέον , δεν επιτρέπουν την κατάργηση

εγγραφών που σχετίζονται με άλλες .Ομοίως , το σύστημα διαχείρισης της βάσης δεδομένων δεν επιτρέπει την εισαγωγή μιας οντότητας χωρίς να δοθεί σε αυτή κάποιο αναγνωριστικό ταυτοποίησης που να τη χαρακτηρίζει μοναδικά.

6.6 Μέθοδοι πρόληψης και αντιμετώπισης δυσάρεστων καταστάσεων

Πέραν των τεχνικών που καθορίζουν την πρόσβαση των χρηστών στα δεδομένα ,υπάρχουν κάποιες μέθοδοι που προλαμβάνουν την έλευση δυσάρεστων καταστάσεων .Έτσι , για να προληφθεί κάποια καταστροφή από φυσικά αίτια , πρέπει να υπάρχει σαφές σχέδιο των εγκαταστάσεων όπου τοποθετείται το υπολογιστικό υλικό . Στο σχέδιο αυτό καθορίζονται διάφοροι παράγοντες ,όπως :

Η τοποθέτηση του υλικού σε χώρους που ασφαρίζονται με κλειδαριά ασφαλείας ή με μηχανήματα ανάγνωσης μαγνητικών καρτών , για να επιτραπεί η είσοδος .

Ο εξοπλισμός του χώρου με συστήματα πυρόσβεσης .

Η εγκατάσταση συστημάτων κλιματισμού ώστε να ελέγχεται η θερμοκρασία και η υγρασία .

Η προστασία του υλικού σε περίπτωση πλημμύρας .

Η δημιουργία αντιγράφων ασφαλείας είναι πολύ σημαντική γιατί , ακόμα και σε περίπτωση οργανωμένου σχεδίου πρόληψης , είναι πιθανόν να υπάρξουν απώλειες .

Κεφάλαιο 7

Βάσεις Και Δίκτυα Μεταβίβασης Πληροφοριών

7.1 Βάσεις δεδομένων σε Πληροφοριακό Σύστημα.

Για την αποτελεσματική διοίκηση και λειτουργία των σύγχρονων Νοσοκομείων απαιτούνται πληροφορίες που παράγονται από τη συσχέτιση και επεξεργασία δεδομένων, τα οποία συλλέγονται κατά την εκτέλεση των δραστηριοτήτων των διαφόρων λειτουργικών τους μονάδων. Για την ικανοποίηση τέτοιων πληροφοριακών απαιτήσεων, ο σχεδιασμός του Π.Σ πρέπει να εστιάζεται προς τα συνολικά δεδομένα του οργανισμού και όχι προς τα επιμέρους δεδομένα κάθε λειτουργικής του μονάδας. Η τεχνολογία των βάσεων δεδομένων (data bases) παρέχει το κατάλληλο υπόβαθρο για την ανάπτυξη και υλοποίηση τέτοιων Π.Σ.

Οι βάσεις δεδομένων και η τεχνολογία τους εξασκούν σημαντική επίδραση στην αυξανόμενη χρήση των υπολογιστών. Είναι εύλογο να ειπωθεί ότι οι βάσεις δεδομένων θα διαδραματίσουν κρίσιμο ρόλο σε όλες τις περιοχές όπου χρησιμοποιούνται υπολογιστές, όπως στις επιχειρήσεις, στη μηχανική, στην ιατρική, στα νομικά, στην εκπαίδευση και στη βιβλιοθηκονομία, για να αναφέρουμε μόνο μερικές από αυτές. Η φράση βάση δεδομένων βρίσκεται σε τόσο κοινή χρήση που πρέπει να αρχίσουμε με μια προσπάθεια του ορισμού της.

Βάση δεδομένων (database) είναι μια ολοκληρωμένη και δομημένη συλλογή από σχετιζόμενα δεδομένα.

Ο παραπάνω ορισμός δεν είναι μοναδικός, που μπορεί να δοθεί, αλλά περιγράφει με πολύ λιτό τρόπο την έννοια των βάσεων δεδομένων.

Με τον όρο **δεδομένα** εννοούμε γνωστά γεγονότα που μπορούν να καταγραφούν και που έχουν κάποια λογική σχέση μεταξύ τους. Για παράδειγμα ένα ευρετήριο με τα ονόματα, τους αριθμούς τηλεφώνων και τις διευθύνσεις των ανθρώπων που γνωρίζετε είναι μια συλλογή από λογικά σχετιζόμενα μεταξύ τους δεδομένα και, επομένως, είναι μια βάση δεδομένων. Ο πιο πάνω ορισμός μιας βάσης δεδομένων είναι αρκετά γενικός.

Με άλλα λόγια μια **βάση δεδομένων** είναι μια συλλογή από δεδομένα προκαθορισμένης μορφής και των σχέσεων μεταξύ τους, και με οργανωτική δομή προσανατολισμένη στην ευέλικτη εισαγωγή, ενημέρωση, παρουσίαση και ανάκτηση των δεδομένων της.

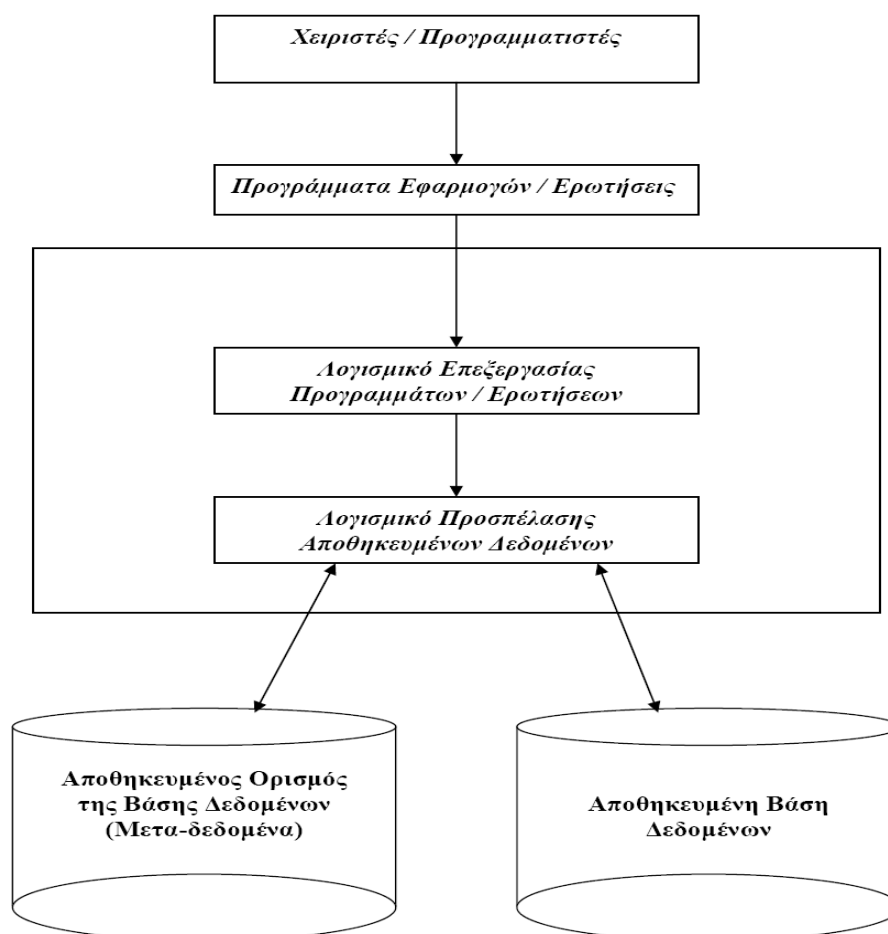
7.1.2 Σύστημα Διαχείρισης Βάσεων Δεδομένων

Οι βάσεις δεδομένων σε υπολογιστή δημιουργούνται και συντηρούνται είτε από μια ομάδα προγραμμάτων εφαρμογών που έχουν γραφεί ειδικά για το σκοπό αυτόν, είτε από ένα **σύστημα διαχείρισης βάσεων δεδομένων**.

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων (**ΣΔΒΔ**)(**database management system -DBMS**) είναι μια συλλογή από προγράμματα που επιτρέπουν στους χρήστες

να δημιουργήσουν και να συντηρήσουν τη βάση. Επομένως, το ΣΔΒΔ είναι ένα γενικής χρήσης σύστημα λογισμικού που διευκολύνει τις διαδικασίες ορισμού, κατασκευής και χειρισμού βάσεων δεδομένων για διάφορες εφαρμογές. Το λογισμικό αυτό λειτουργεί ένα επίπεδο ψηλότερα από το λειτουργικό σύστημα του Η/Υ και αποκρύπτει τις λεπτομέρειες του δεύτερου από τον χρήστη του DBMS.

ΣΥΣΤΗΜΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ



Εικόνα 11: Σύστημα Βάσης Δεδομένων

7.1.3 Επιπτώσεις από την χρήση Βάσεων Δεδομένων

Η ανάπτυξη ενός πληροφοριακού συστήματος σε περιβάλλον βάσεων δεδομένων παρέχει σημαντικά πλεονεκτήματα στο Νοσοκομείο, σε ότι αφορά τη διαχείριση των λειτουργικών τήρηση κοινών προτύπων για τα δεδομένα και την ικανοποίηση των σφαιρικών πληροφοριακών απαιτήσεων του οργανισμού.

Τα πλεονεκτήματα προκύπτουν από τη δημιουργία μιας ολοκληρωμένης θεώρησης των δεδομένων του Νοσοκομείου και από τη δυνατότητα άσκησης κεντρικού ελέγχου σ' αυτά, με στόχο την τήρηση κοινών προτύπων για τα δεδομένα και την ικανοποίηση των σφαιρικών πληροφοριακών απαιτήσεων του οργανισμού.

Τα μειονεκτήματα προκύπτουν από τις ιδιαιτερότητες των βάσεων δεδομένων, σε σχέση με τα παραδοσιακά συστήματα αρχείων και αφορούν λειτουργικά και οικονομικά θέματα.

Στη συνέχεια περιγράφονται τα κύρια πλεονεκτήματα και μειονεκτήματα των βάσεων δεδομένων.

7.1.4 Πλεονεκτήματα των βάσεων δεδομένων

Τα βασικά πλεονεκτήματα που προκύπτουν από τη χρήση των βάσεων δεδομένων είναι:

α) Μείωση των περιττών δεδομένων. Περιττά δεδομένα (redundant data) ονομάζονται εκείνα που όταν διαγραφούν από τη βάση δεδομένων δεν μειώνεται το πληροφοριακό της περιεχόμενο. Γενικά, τα περιττά δεδομένα μπορεί να είναι είτε επαναλήψεις δεδομένων που αποθηκεύονται σε διαφορετικά αρχεία, είτε αποθηκευμένα δεδομένα που μπορεί να εξαχθούν από άλλα. Για παράδειγμα, αν τα προγράμματα εφαρμογών του Ιατρικού Φακέλου και του L.I.S ενός οργανισμού χρησιμοποιούν δύο διαφορετικά αρχεία, με κοινά δεδομένα για τους νοσηλευόμενους στο Νοσοκομείο, τα δεδομένα που επαναλαμβάνονται είναι περιττά.

β) Ανεξαρτησία των δεδομένων. Ανεξαρτησία των δεδομένων (data independence) σημαίνει ότι τα προγράμματα εφαρμογών δεν εξαρτώνται από τον τρόπο οργάνωσης και προσπέλασης των αρχείων δεδομένων που χρησιμοποιούν. Έτσι, οι εφαρμογές δεν επηρεάζονται από τυχόν αλλαγές στη λογική ή στη φυσική δόμηση των δεδομένων της βάσης δεδομένων, αλλά και σε περίπτωση πρόσθεσης νέων πεδίων στις εγγραφές της βάσης οι τροποποιήσεις περιορίζονται μόνο στα προγράμματα που χρησιμοποιούν αυτά τα νέα πεδία. Για παράδειγμα αλλαγή στα πεδία του Γραφείου Κίνησης δεν θα επηρεάσει τα στοιχεία ασθενή που χρησιμοποιούνται από τον Ιατρικό Φάκελο.

γ) Κοινοχρησία των δεδομένων. Τα δεδομένα που έχουν αποθηκευτεί στη βάση δεδομένων μπορούν να χρησιμοποιηθούν από πολλές εφαρμογές, γεγονός που συμβάλει στην αποτελεσματική λειτουργία του πληροφοριακού συστήματος του Νοσοκομείου. Παράδειγμα τα στοιχεία που καταχωρούνται στο Γραφείο Προσωπικού είναι δυνατό να χρησιμοποιηθούν από την εφαρμογή της Μισθοδοσίας.

δ) Ασφάλεια των Δεδομένων. Ο κεντρικός έλεγχος στη βάση δεδομένων μπορεί να διασφαλίσει την πρόσβαση ευαίσθητων δεδομένων (πχ στοιχεία του Ιατρικού Φακέλου) από ορισμένες εφαρμογές και από χρήστες μόνο που διαθέτουν σχετική δικαιοδοσία.

ε) Ακεραιότητα των δεδομένων. Η διατήρηση της ακεραιότητας σημαίνει ότι τα δεδομένα που καταχωρούνται στη βάση είναι ακριβή. Μηχανισμοί του D.B.M.S εξασφαλίζουν παραβίαση ακεραιότητας από λάθη πληκτρολόγησης, λάθη προγραμμάτων εφαρμογών και από μη ολοκληρωμένες επεξεργασίες δεδομένων ως αποτέλεσμα βλαβών των υπολογιστικών συστημάτων.

7.1.5 Μειονεκτήματα των βάσεων δεδομένων.

α) Απαιτήσεις σε εξειδικευμένο προσωπικό. Για την αποτελεσματική ανάπτυξη του πληροφοριακού συστήματος βασισμένου σε περιβάλλον βάσης δεδομένων απαιτείται προσωπικό που να είναι κατάλληλα καταρτισμένο στην αξιοποίηση των δυνατοτήτων του D.B.M.S, αλλά και στο σχεδιασμό βάσεων δεδομένων. Η αδυναμία στελέχωσης των Τμημάτων Πληροφορικής των Νοσοκομείων με προσωπικό αντίστοιχων προσόντων οδηγεί τους οργανισμούς σε τρίτους φορείς για την εξυπηρέτηση των αναγκών του με υψηλό κόστος.

β) Κόστος του D.B.M.S. Το κόστος προμήθειας ενός D.B.M.S (συμπεριλαμβανόμενων και των δαπανών υποστήριξης, εκπαίδευσης και συντήρησης

) εξαρτάται από τον αριθμό των αδειών χρήσης (license) και το υπολογιστικό σύστημα για το οποίο και προορίζεται.

γ) Προβλήματα στην Ακεραιότητα και Ασφάλεια δεδομένων.

Σε περίπτωση αδυναμιών του D.B.M.S στην υποστήριξη διαδικασιών ασφάλειας και ακεραιότητας δεδομένων, το Νοσοκομείο είναι δυνατό να αντιμετωπίσει σημαντικά λειτουργικά προβλήματα.. Επίσης, το στάδιο σχεδιασμού της βάσης δεδομένων είναι το πλέον σημαντικό βήμα για την διασφάλιση της ακεραιότητας και ασφάλειας των δεδομένων.

7.2 Κατηγορίες Δικτύων

Ένα **δίκτυο** είναι ένας συνδυασμός υλικού, λογισμικού και καλωδίωσης ο οποίος επιτρέπει την επικοινωνία πολλαπλών υπολογιστικών συσκευών.

Υπάρχουν τρεις κατηγορίες δικτύων ανάλογα με τη γεωγραφική έκταση που εκτείνονται:

- τοπικά δίκτυα (Local Area Networks - LANs)
- δίκτυα μητροπολιτικής περιοχής (Metropolitan Area Networks - MANs)
- δίκτυα ευρείας περιοχής (Wide Area Networks - WANs)
- Ένα **τοπικό δίκτυο** είναι ένα ιδιόκτητο δίκτυο το οποίο περιορίζεται σε μία συγκεκριμένη γεωγραφική περιοχή (συνήθως ένα γραφείο, ένα κτίριο ή συγκρότημα κτιρίων). Η δημιουργία ενός τοπικού δικτύου επιτρέπει την κοινή χρήση αρχείων και εκτυπωτών από πολλούς χρήστες, τη μεταφορά δεδομένων (π.χ. σε νοσοκομείο μεταφορά εικόνων από το ένα τμήμα στο άλλο) κ.λπ. Μερικές πολύ γνωστές τεχνολογίες τοπικών δικτύων είναι: Ethernet ,Token Ring , FDDI .

Ένα **δίκτυο ευρείας περιοχής** καλύπτει μία μεγάλη γεωγραφική περιοχή (ακόμα μία ή περισσότερες ηπείρους). Το πιο γνωστό παράδειγμα δικτύου ευρείας περιοχής είναι το Διαδίκτυο. Ένα τέτοιο δίκτυο επιτρέπει στους χρήστες να επικοινωνήσουν σε πραγματικό χρόνο με άλλους χρήστες, παροχή υπηρεσιών εξ αποστάσεως (τηλεϊατρική).

Ένα δίκτυο μητροπολιτικής περιοχής συνδέει δύο ή περισσότερα τοπικά δίκτυα τα οποία βρίσκονται σε μία συγκεκριμένη γεωγραφική περιοχή, συνήθως μία πόλη. Συνήθως ένας πάροχος τηλεπικοινωνιακών υπηρεσιών συνδέει δύο ή περισσότερα τοπικά δίκτυα χρησιμοποιώντας ιδιωτικές γραμμές επικοινωνίας (γραμμές χαλκού ή οπτικές ίνες). Ένα δίκτυο μητροπολιτικής περιοχής μπορεί επίσης να δημιουργηθεί χρησιμοποιώντας ασύρματη τεχνολογία ή τεχνολογία οπτικής (για παράδειγμα, οπτική ελεύθερου χώρου (Free Space Optics - FSO)).

7.3 Τοπολογίες Τοπικών Δικτύων

Τοπολογία (topology) ενός τοπικού δικτύου αναφέρεται στον τρόπο με τον οποίο συνδέονται οι υπολογιστές μεταξύ τους. Με άλλα λόγια, μία τοπολογία δικτύου περιγράφει την κατάστρωση (layout) των συσκευών και των καλωδίων καθώς επίσης και τις διαδρομές που χρησιμοποιούνται κατά τη μετάδοση των δεδομένων. Η τοπολογία επηρεάζει σημαντικά τη λειτουργία ενός δικτύου.

Τα δίκτυα μπορούν να έχουν μια **φυσική (physical)** και μία **λογική (logical)** τοπολογία. Η φυσική τοπολογία αναφέρεται στον τρόπο τοποθέτησης στο φυσικό χώρο και της διασύνδεσης με καλώδια των διαφόρων συσκευών. Η λογική τοπολογία καθορίζει τον τρόπο με τον οποίο οι υπολογιστές χρησιμοποιούν το μέσο μετάδοσης για την αποστολή δεδομένων.

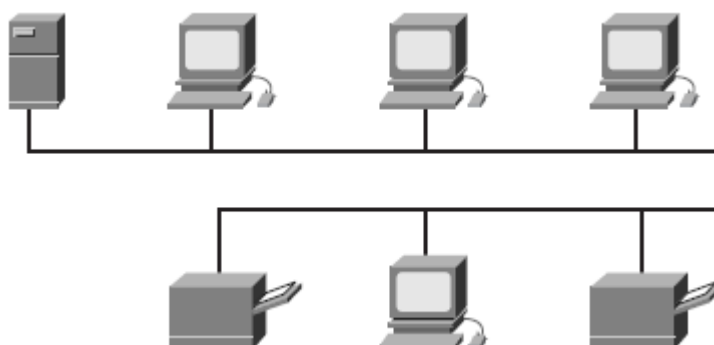
7.3.1 Φυσικές τοπολογίες

Τα πιο γνωστά είδη φυσικών τοπολογιών τοπικού δικτύου είναι:

- τοπολογία αρτηρίας
- τοπολογία αστέρα
- τοπολογία δακτυλίου

7.3.1.1 Τοπολογία Αρτηρίας

Στην **τοπολογία αρτηρίας** (bus topology) όλοι οι υπολογιστές συνδέονται σε σειρά σε ένα κοινό μέσο μετάδοσης (συνήθως είναι ένα ομοαξονικό καλώδιο για ενσύρματα δίκτυα) κατάλληλα τερματισμένο στα δύο άκρα του για να μην υπάρχει ανάκλαση του μεταδιδόμενου σήματος.

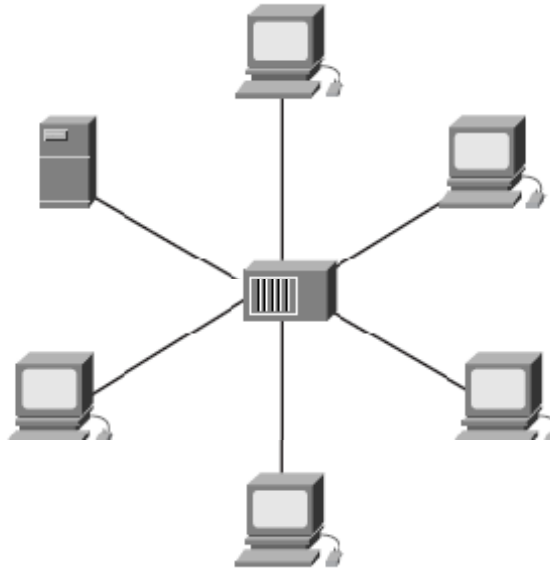


Εικόνα 12: Τοπολογία Αρτηρίας

Η επικοινωνία μεταξύ δύο υπολογιστών γίνεται ως ακολούθως: ο υπολογιστής που στέλνει τα δεδομένα "εξετάζει" αν το μέσο μετάδοσης (π.χ. ομοαξονικό καλώδιο) είναι ελεύθερο ή όχι. Αν ο το μέσο μετάδοσης είναι ελεύθερο, τότε αρχίζει τη μετάδοση, διαφορετικά περιμένει να ελευθερωθεί το μέσο μετάδοσης. Επειδή όλοι οι υπολογιστές είναι συνδεδεμένοι στο κοινό μέσο μετάδοσης, λαμβάνουν τα μεταδιδόμενα δεδομένα. Ο υπολογιστής για τον οποίο προορίζονται τα δεδομένα τα αντιγράφει, ενώ όλοι οι άλλοι υπολογιστές τα αγνοούν.

7.3.1.2 Τοπολογία αστέρα

Στην **τοπολογία αστέρα** (star topology), υπάρχει μία κεντρική δικτυακή συσκευή (και οι υπόλοιποι υπολογιστές συνδέονται μόνο με αυτή. Η επικοινωνία μεταξύ δύο υπολογιστών γίνεται μέσω της κεντρικής δικτυακής συσκευής. Είναι η πιο ευρέως χρησιμοποιούμενη τοπολογία σήμερα.



Εικόνα 13: Τοπολογία αστέρα

Η τοπολογία αυτή έχει τα εξής πλεονεκτήματα:

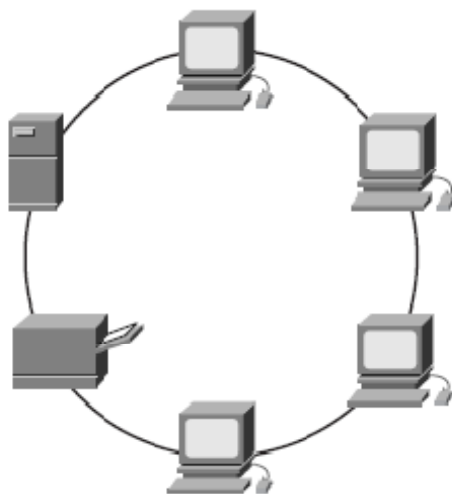
- Καλύτερη απόδοση: Σε κάθε επικοινωνία εμπλέκονται το πολύ τρεις συσκευές, δηλαδή οι δύο υπολογιστές που επικοινωνούν και η κεντρική δικτυακή συσκευή, σε αντίθεση με άλλες τοπολογίες όπου εμπλέκονται περισσότερες συσκευές.
- Απομόνωση υπολογιστών: Κάθε υπολογιστής επικοινωνεί απευθείας μόνο με την κεντρική συσκευή. Συνεπώς, αν τεθεί εκτός λειτουργίας ο υπολογιστής δεν επηρεάζεται λειτουργία του υπόλοιπου δικτύου.
- Συγκεντρωτισμός: Αφού όλη η κίνηση γίνεται μέσω της κεντρικής συσκευής είναι εύκολη η παρακολούθηση της κίνησης του δικτύου και η ανίχνευση ανεπιθύμητης συμπεριφοράς (π.χ. παραβίαση δικτύου).
- Απλότητα: Η τοπολογία αυτή είναι εύκολα κατανοητή και υλοποιήσιμη.

Το βασικό μειονέκτημα της τοπολογίας αυτής είναι η εξάρτηση της λειτουργίας του δικτύου από την κεντρική δικτυακή συσκευή, αφού μια βλάβη αυτής καθιστά ανενεργό ολόκληρο το δίκτυο. Επιπλέον, η απόδοση και η επεκτασιμότητα του δικτύου εξαρτώνται από τις δυνατότητες της κεντρικής συσκευής. Συγκεκριμένα, το μέγεθος του δικτύου περιορίζεται από το πλήθος των συνδέσεων που μπορεί να υποστηρίξει η κεντρική συσκευή και η απόδοση για ολόκληρο το δίκτυο καθορίζεται από τη διεκπεραιωτική ικανότητά της.

7.3.1.3 Τοπολογία δακτυλίου

Στην **τοπολογία δακτυλίου** (ring topology), κάθε υπολογιστής συνδέονται με άλλους δύο για να σχηματίσουν ένα δακτύλιο . Τα προς μετάδοση δεδομένα κινούνται στο δακτύλιο σε μία κατεύθυνση (π.χ δεξιόστροφα) . Όταν ένας υπολογιστής λάβει τα δεδομένα εξετάζει αν προορίζονται για εκείνον. Αν ναι, τότε τα αντιγράφει και τα προωθεί στο γειτονικό υπολογιστή. Αν όχι, απλώς τα προωθεί.

Όταν τα δεδομένα φτάσουν στον υπολογιστή από τον οποίο στάλθηκαν, αποσύρονται από το δίκτυο.



Εικόνα 14 : Τοπολογία Δακτυλίου

Η τοπολογία δακτυλίου έχει τα ακόλουθα πλεονεκτήματα:

- Καλύτερη απόδοση από την τοπολογία αστέρα όταν το δίκτυο είναι φορτωμένο.
- Μπορούν να δημιουργηθούν μεγαλύτερα δίκτυα.
- Δεν απαιτείται κάποια κεντρική συσκευή για την επικοινωνία μεταξύ δύο υπολογιστών.

Είναι προφανές ότι η τοπολογία δακτυλίου έχει το σοβαρό μειονέκτημα ότι βλάβη σε έναν υπολογιστή καθιστά προβληματική τη λειτουργία ολόκληρου του δικτύου. Για να ξεπεραστεί το πρόβλημα αυτό έχουν αναπτυχθεί διάφορες παραλλαγές. Για παράδειγμα, στα δίκτυα **FDDI** (Fiber Distributed Data Interface) τα δεδομένα στέλνονται και προς τις δύο κατευθύνσεις (δεξιόστροφα και αριστερόστροφα). Στα **δίκτυα 802.5**, επίσης γνωστά ως IBM δίκτυα δακτυλίου με σκυτάλη, χρησιμοποιούνται ειδικές δικτυακές συσκευές (MAU – Multistation Access Unit) για να συνδέσουν τους υπολογιστές σε τοπολογία αστέρα, διατηρώντας όμως τη λογική της μετάδοσης δεδομένων με χρήση σκυτάλης. Οι συσκευές αυτές περιέχουν ειδικές διατάξεις για να απομονώνουν από το δίκτυο υπολογιστές που δε λειτουργούν.

7.3.2 Λογικές τοπολογίες

Η λογική τοπολογία ενός δικτύου καθορίζει τον τρόπο με τον οποίο επικοινωνούν οι συσκευές στο μέσο μετάδοσης. Οι δύο πιο κοινί τύποι λογικής τοπολογίας είναι:

- τοπολογία εκπομπής
- τοπολογία μεταφοράς σκυτάλης

7.3.2.1 Τοπολογία εκπομπής

Στην τοπολογία εκπομπής (broadcast topology) κάθε συσκευή στέλνει τα δεδομένα σε μία συγκεκριμένη κάρτα διασύνδεσης (NIC), μία διεύθυνση πολλαπλής εκπομπής

(multicast address) ή σε μία διεύθυνση εκπομπής (broadcast address) on the network medium. Δεν υπάρχει συγκεκριμένος κανόνας που να καθορίζει την προτεραιότητα μετάδοσης δεδομένων, δηλαδή οι συσκευές ανταγωνίζονται για προσπέλαση στο μέσο μετάδοσης. Όποια συσκευή προλάβει, μεταδίδει πρώτα τα δεδομένα της.

7.3.2.2 Τοπολογία μεταφοράς σκυτάλης

Στην **τοπολογία μεταφοράς σκυτάλης (token passing)**, για την επικοινωνία μεταξύ δύο συσκευών περιλαμβάνει τη χρήση **σκυτάλης (token)**. Συγκεκριμένα, στο δίκτυο κυκλοφορεί συνεχώς μία σκυτάλη, δηλαδή μία συγκεκριμένη ακολουθία από bits, η οποία μεταφέρεται από συσκευή σε συσκευή. Μία συσκευή μπορεί να μεταδώσει μόνο όταν έχει στη διάθεσή της τη σκυτάλη. Στην περίπτωση αυτή, η συσκευή τροποποιεί τη σκυτάλη και εισαγάγει τα δεδομένα και τη διεύθυνση του υπολογιστή προορισμού. Τα δεδομένα μεταδίδονται μέχρι να φτάσουν στον υπολογιστή για τον οποία προορίζονται, ο οποίος τα αντιγράφει και επαναφέρει τη σκυτάλη στην αρχική της μορφή. Όταν η σκυτάλη φτάσει στον αρχικό υπολογιστή, αυτός αφού διαπιστώσει ότι η σκυτάλη είναι στην αρχική της μορφή αποσύρει τα δεδομένα από το δίκτυο και απελευθερώνει τη σκυτάλη.

Κεφάλαιο 8

Τεχνολογίες και Εργαλεία Που Χρησιμοποιήθηκαν

8.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναφέρουμε κάποια στοιχεία και εργαλεία που χρησιμοποιήθηκαν στην ανάπτυξη της ιστοσελίδας . Συγκεκριμένα , θα δούμε την Γλώσσα Σήμανσης Υπερκειμένου HTML , τα Cascading Style Sheets , την γλώσσα PHP και SQL και το πακέτο λογισμικού EasyPHP .

8.2 Τι είναι η HTML

Η **HTML** είναι το ακρωνύμιο των λέξεων HyperText Markup Language, δηλαδή Γλώσσα Χαρακτηρισμού Υπερ-Κειμένου και βασίζεται στη γλώσσα **SGML**, Standard Generalized Markup Language, που είναι ένα πολύ μεγαλύτερο σύστημα επεξεργασίας εγγράφων.

Τα αρχεία της HTML είναι απλά αρχεία κειμένου , τα οποία χρησιμοποιούν ετικέτες για την περιγραφή της δομής και της παρουσίασης μιας ιστοσελίδας , η οποία μπορεί να περιέχει κείμενο , εικόνα , φόρμες , συνδέσεις κ.α.

Για να δημιουργήσουμε ένα αρχείο HTML αρκεί ένας απλός συντάκτης κειμένου .

Η HTML ορίζει ένα σύνολο κοινών στυλ για τις Web σελίδες, όπως τίτλοι (titles), επικεφαλίδες (headings), παράγραφοι (paragraphs), λίστες (lists) και πίνακες (tables). Ορίζει επίσης στυλ χαρακτήρων, όπως η έντονη γραφή (boldface) και οι ενότητες κώδικα.

Κάθε στοιχείο έχει ένα όνομα και περιέχεται μέσα στα σύμβολα <>, που αποκαλούνται **tags** (ετικέτες). Όταν γράφουμε μια Web σελίδα με την HTML, στην ουσία δίνουμε τίτλους στα διάφορα στοιχεία της σελίδας μ' αυτά τα tags.

Οι φυλλομετρητές, μαζί με τη δυνατότητά τους να ανακτούν σελίδες από το Web, λειτουργούν επίσης και σαν μορφοποιητές για την HTML. Όταν διαβάζουμε μια σελίδα γραμμένη με την HTML σ' έναν φυλλομετρητή, ο φυλλομετρητής διαβάζει (διερμηνεύει) τα tags της HTML και μορφοποιεί το κείμενο και τις εικόνες στην οθόνη.

Διαφορετικοί φυλλομετρητές, οι οποίοι τρέχουν σε διαφορετικούς υπολογιστές, μπορεί να αντιστοιχίζουν διαφορετικά στυλ σε κάθε στοιχείο μιας σελίδας. Αυτό σημαίνει ότι οι σελίδες που δημιουργούμε με την HTML μπορεί να δείχνουν εντελώς διαφορετικές από σύστημα σε σύστημα και από φυλλομετρητή σε φυλλομετρητή. Δηλαδή, οι πραγματικές πληροφορίες και οι σύνδεσμοι που περιέχουν οι σελίδες μας θα είναι πάντα εκεί, αλλά η εμφάνιση των σελίδων στην οθόνη θα είναι διαφορετική.

8.2.1 Συγκεντρωτικός πίνακας των Tags της HTML

Η παρακάτω λίστα περιέχει τα πιο συνηθισμένα HTML tags.

tag	Περιγραφή
<html>	Το αρχικό tag για κάθε σελίδα HTML
<head>	Η επικεφαλίδα της σελίδας
<meta>	Στοιχεία για τις μηχανές αναζήτησης
<title>	Ο τίτλος της σελίδα που θα εμφανιστεί στο Browser.
<body>	Το κύριο σώμα της σελίδας
	Μορφοποίηση γραμματοσειράς
 	Αλλαγή γραμμής (δεν έχει tag τέλους)
<h?>	Επικεφαλίδα (? από 1 έως 6)

<pre>	Εισάγουμε το κείμενο στις θέσεις που θέλουμε
	Δημιουργία έντονων χαρακτήρων (bold)
<u>	Υπογράμμιση χαρακτήρων(underline)
<tt>	Χαρακτήρες γραφομηχανής (Typewriter)
<i>	Δημιουργία πλάγιων χαρακτήρων (italic)
<strike>	Διακριτή διαγραφή (strike)
<a>	Σύνδεσμος (link)
	Εισαγωγή εικόνας
<small>	Μικρά γράμματα
<big>	Μεγάλα γράμματα
<p>	Αλλαγή παραγράφου
<hr>	Οριζόντια γραμμή
<sub>	Δείκτης
<sup>	Εκθέτης
<!--	Σχόλιο (δεν εμφανίζονται στον Browser)

8.3 Cascading Style Sheets (CSS)

Τα Διαδοχικά Φύλλα Στυλ (CSS, Cascading Style Sheets) αποτελούν ένα πολύ καλό εργαλείο για να μπορούμε να αλλάζουμε την εμφάνιση και τη διάταξη (layout) των ιστοσελίδων μας. Μπορούν να μας γλυτώσουν από πολύ χρόνο και κόπο και μας

δίνουν τη δυνατότητα να σχεδιάζουμε τις ιστοσελίδες μας με μια εντελώς καινούργια φιλοσοφία. Η κατανόηση των CSS απαιτεί να υπάρχει κάποια βασική εμπειρία με την HTML.

Για να δουλέψουμε με τα CSS δεν χρειάζεται να χρησιμοποιήσουμε κάποιο πρόγραμμα δημιουργίας ιστοσελίδων, όπως είναι τα γνωστά FrontPage, DreamWeaver ή και το Word, καθώς θα σταθούν εμπόδιο στην κατανόηση των CSS. Αυτό που χρειαζόμαστε είναι ένας απλός επεξεργαστής κειμένου, όπως είναι το Σημειωματάριο (Notepad) των Windows.

Μπορούμε να χρησιμοποιήσουμε όποιον φυλλομετρητή (browser) επιθυμούμε για να βλέπουμε πώς θα εμφανίζονται οι ιστοσελίδες που θα δημιουργούμε. Συνεπώς αυτό που χρειαζόμαστε είναι ένας φυλλομετρητής (browser) και ένας απλός επεξεργαστής κειμένου (text editor).

8.3.1 Πλεονεκτήματα CSS

- Πολύ μεγαλύτερη ευελιξία. Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασσική 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 μας.

8.4 JavaScript

Η γλώσσα προγραμματισμού JavaScript αναπτύχθηκε από την εταιρεία Netscape, σε συνεργασία με την Sun Microsystems και η πρώτη της έκδοση δημοσιεύτηκε το 1995. Ακολούθησε η αντίστοιχη γλώσσα της Microsoft η οποία ονομάστηκε Jscript και η επόμενη έκδοση της JavaScript που είχε το όνομα ECMAScript που αργότερα όμως καθιερώθηκε με το όνομα που είναι γνωστό μέχρι σήμερα.

Η JavaScript είναι μία διερμηνευμένη (interpreted) γλώσσα προγραμματισμού με ιδιότητες αντικειμενοστραφούς γλώσσας προγραμματισμού, χωρίς όμως να μπορεί να χαρακτηριστεί ως πλήρης αντικειμενοστραφής.

Η γλώσσα αυτή, κτίστηκε ουσιαστικά πάνω στο πρότυπο των γλωσσών C, C++ και Java. Από την άλλη όμως έχει μία πολύ σημαντική διαφορά στο ότι διαχειρίζεται τους

τύπους δεδομένων πιο χαλαρά (loosely typed) σε σχέση με τη σφικτή διαχείριση τύπων δεδομένων (strongly typed) που γίνεται στις προαναφερόμενες γλώσσες.

Στην JavaScript οι μεταβλητές δεν είναι απαραίτητο να έχουν ένα συγκεκριμένο τύπο ή ακόμη είναι δυνατόν να αλλάζουν τύπο κατά τη διάρκεια της ζωής τους. Επίσης, δεν πρέπει να συγχέεται η JavaScript με την Java της Sun Microsystems. Η μία δεν έχει καμία σχέση με την άλλη. Η χρήση του ονόματος JavaScript έγινε για λόγους προώθησης της γλώσσας σε μία εποχή που η εξάπλωση της Java ήταν πολύ μεγάλη.

8.4.1 Χρήσεις της JavaScript

Η γλώσσα JavaScript χρησιμοποιείται κυρίως για την εξυπηρέτηση των παρακάτω σκοπών:

- Λιγότερος φόρτος των server: Ο έλεγχος και η επικύρωση των δεδομένων που εισάγονται από τους χρήστες γίνεται από τη μεριά του browser κι έτσι δεδομένα τα οποία δεν είναι σε κατάλληλη μορφή δεν αποστέλλονται στον server. Αυτό όμως δεν σημαίνει ότι ο έλεγχος δεν πρέπει να γίνεται και στη μεριά των εξυπηρετητών καθώς κάποιος χρήστης μπορεί να μην έχει ενσωματωμένη την JavaScript στον browser του ή υπάρχει πιθανότητα να την έχει απενεργοποιήσει.
- Άμεση αλληλεπίδραση με τους χρήστες: Με την χρήση της JavaScript για τον έλεγχο των δεδομένων μειώνονται οι χρόνοι αναμονής του χρηστών αφού αυτοί δεν χρειάζεται να περιμένουν μεγάλα χρονικά διαστήματα επαναφόρτωσης της σελίδας σε περίπτωση που έχουν ξεχάσει να εισάγουν κάποιο δεδομένο ή έχουν εισάγει κάτι λάθος.
- Αυτόματη διόρθωση λαθών: Ένα παράδειγμα που μπορεί να κάνει περισσότερο κατανοητό το πώς μπορεί να χρησιμοποιηθεί η JavaScript με αυτόν τον τρόπο είναι αυτό της ημερομηνίας. Πολλά συστήματα βάσεων δεδομένων αποθηκεύουν δεδομένα ημερομηνιών σε μορφή dd-mm-yyyy. Αν κάποιος χρήστης εισάγει κάποια ημερομηνία σε μορφή dd/mm/yyyy τότε κάτι τέτοιο θα μπορούσε να ανιχνευτεί αυτόματα από τον browser και να μετατραπεί στην σωστή μορφή πριν τα δεδομένα αποσταλούν στον server.
- Αυξημένη χρηστικότητα: Αυτό επιτυγχάνεται επιτρέποντας στον χρήστη την αλλαγή και αλληλεπίδραση με το γραφικό περιβάλλον χωρίς την επαναφόρτωση της σελίδας. Ένα τέτοιο παράδειγμα είναι τα πτυσσόμενα μενού.
- Αυξημένη δυνατότητα αλληλεπίδρασης: Ένα τέτοιο παράδειγμα όπου κάτι τέτοιο επιτυγχάνεται είναι τα μενού τα οποία αλληλεπιδρούν όταν ο χρήστης περάσει το mouse πάνω από αυτά – η λειτουργία hover – κάτι το οποίο έχει ως αποτέλεσμα να δημιουργηθεί μία σειρά από γεγονότα τα οποία έχουν προγραμματιστεί να λειτουργούν με έναν συγκεκριμένο τρόπο.

8.5 Τι είναι η PHP



Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.

Η PHP είναι μια γλώσσα script από την πλευρά του διακομιστή, σχεδιασμένη ειδικά για το Web. Μέσα σε μια HTML σελίδα μπορείτε να ενσωματώσετε PHP κώδικα, που θα εκτελείται κάθε φορά που θα επισκέπτεστε τη σελίδα. Ο PHP κώδικας μεταφράζεται στο Web διακομιστή και δημιουργεί HTML ή άλλη έξοδο που θα δει ο επισκέπτης.

Η PHP δημιουργήθηκε το 1994 και ήταν αρχικά η δουλειά ενός ατόμου, του **Rasmus Lerdorf**. Υιοθετήθηκε και από άλλα ταλαντούχα άτομα και έχει περάσει από τρεις βασικές εκδόσεις. Τον Ιανουάριο του 2001 ήταν σε χρήση σχεδόν σε πέντε εκατομμύρια τομείς παγκόσμια και αυτός ο αριθμός μεγαλώνει γρήγορα.

Η PHP είναι ένα προϊόν ανοιχτού κώδικα. Θα έχετε πρόσβαση στον κώδικα προέλευσης. Μπορείτε να τον χρησιμοποιήσετε, να τον αλλάξετε και να τον αναδιανεμίσετε, χωρίς χρέωση.

Η PHP αρχικά σήμαινε *Personal Home Page* (προσωπική αρχική σελίδα), αλλά άλλαξε σύμφωνα με την σύμβαση GNU και τώρα σημαίνει *PHP Hypertext Preprocessor* (προεπεξεργαστής κειμένου PHP). Η τρέχουσα βασική έκδοση της PHP είναι η 4. Αυτή η έκδοση έχει μερικές μεγάλες βελτιώσεις στη γλώσσα.

8.5.1 Πλεονεκτήματα της PHP

Κάποιοι από τους βασικούς ανταγωνιστές της PHP είναι ο Perl, Microsoft Active Server Pages (ASP), Java Server Pages (JSP) και Allaire Cold Fusion.

Σε σύγκριση με αυτά τα προϊόντα, η PHP έχει πολλά πλεονεκτήματα όπως :

- Υψηλή απόδοση
- Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων
- Ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες Web διαδικασίες
- Χαμηλό κόστος
- Ευκολία μάθησης και χρήσης
- Μεταφερσιμότητα
- Διαθεσιμότητα του κώδικα προέλευσης.

8.5.2 Τι μπορεί να κάνει η PHP;

Οτιδήποτε. Η PHP επικεντρώνεται κυρίως στο server-side scripting, έτσι μπορείτε να κάνετε οτιδήποτε ένα άλλο CGI πρόγραμμα μπορεί να κάνει, όπως να μαζέψει δεδομένα, να παράγει δυναμικό περιεχόμενο σελίδων, ή να στείλει και να πάρει cookies. Αλλά η PHP μπορεί να κάνει πολύ περισσότερα.

Υπάρχουν τρεις κύριοι τομείς που χρησιμοποιείται ένα PHP script.

- **Server-side scripting.** Αυτό είναι το πιο παραδοσιακό και το κύριο πεδίο για την PHP. Χρειάζεστε τρία πράγματα για να δουλέψει αυτό. Τον PHP μεταγλωττιστή (parser) (CGI ή server module), ένα webserver (εξηγηρητή σελίδων) και ένα web browser ("φυλλομετρητή"). Πρέπει να τρέξετε τον webserver, με μια συνδεδεμένη εγκατάσταση της PHP. Μπορείτε να προσπελάσετε τα αποτελέσματα του PHP προγράμματος με ένα web browser, βλέποντας την σελίδα PHP μέσα από τον server.
- **Command line scripting.** Μπορείτε να φτιάξετε ένα PHP script για να το τρέχετε χωρίς server ή browser. Χρειάζεστε μόνο τον PHP μεταγλωττιστή για να την χρησιμοποιήσετε με αυτό τον τρόπο. Αυτός ο τύπος είναι ιδανικός για script που εκτελούνται συχνά με τη χρήση της cron (σε *nix ή Linux) ή με τον Task Scheduler (στα Windows). Αυτά τα script μπορούν επίσης να χρησιμοποιηθούν για απλές εργασίες επεξεργασίας κειμένου.
- **Εγγραφή client-side GUI εφαρμογών (Γραφικά περιβάλλοντα χρηστών).** Η PHP ίσως να μην είναι η πιο καλή γλώσσα για να γράψει κανείς παραθυριακές εφαρμογές, αλλά αν ξέρετε PHP πολύ καλά και θέλετε να χρησιμοποιήσετε κάποια προχωρημένα χαρακτηριστικά της PHP στις client-side εφαρμογές σας, μπορείτε επίσης να χρησιμοποιήσετε το PHP-GTK για αυτού του είδους τα προγράμματα. Έχετε επίσης τη δυνατότητα να γράφετε cross-platform εφαρμογές με αυτό τον τρόπο. Το PHP-GTK είναι μια επέκταση της PHP και δεν συμπεριλαμβάνεται στην κύρια διανομή.

Η PHP μπορεί να χρησιμοποιηθεί σε όλα τα κύρια λειτουργικά συστήματα, συμπεριλαμβανομένου του Linux, πολλών εκδοχών του Unix (HP-UX, Solaris και OpenBSD), Microsoft Windows, Mac OS X, RISC OS και πιθανώς σε άλλα. Η PHP υποστηρίζει επίσης τους Apache, Microsoft Internet Information Server, Personal Web Server, Netscape και iPlanet servers, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd, και πολλούς άλλους webserver. Για την πλειοψηφία των server η PHP έχει ένα module, για τους υπόλοιπους η PHP μπορεί να λειτουργήσει ως ένας CGI επεξεργαστής.

Έτσι με την PHP έχετε την ελευθερία επιλογής ενός λειτουργικού συστήματος και ενός web server. Επιπλέον, έχετε επίσης την ελευθερία να χρησιμοποιήσετε συναρτησιακό (procedural) ή αντικειμενοστρεφή (object oriented) προγραμματισμό ή μια ανάμειξη τους. Αν και η παρούσα έκδοση δεν υποστηρίζει όλα τα πρότυπα χαρακτηριστικά, μεγάλες βιβλιοθήκες κώδικα και μεγάλες εφαρμογές (συμπεριλαμβανομένης και της βιβλιοθήκης PEAR) είναι γραμμένες μόνο με αντικειμενοστρεφή κώδικα.

Με την PHP δεν είστε περιορισμένοι να εξάγετε HTML. Οι δυνατότητες της PHP συμπεριλαμβάνουν την εξαγωγή εικόνων, αρχείων PDF, ακόμη και ταινίες Flash (χρησιμοποιώντας τα libswf και Ming) παράγονται αμέσως. Μπορείτε επίσης να εξάγετε εύκολα οποιοδήποτε κείμενο όπως XHTML και οποιοδήποτε άλλο XML αρχείο. Η PHP μπορεί να δημιουργεί αυτόματα αυτά τα αρχεία και να τα αποθηκεύει

στο σύστημα αρχείων, αντί να τα εκτυπώνει, αποτελώντας έτσι μια server-side cache για το δυναμικό σας περιεχόμενο.

Ένα από τα πιο δυνατά και σημαντικά χαρακτηριστικά της PHP είναι η υποστήριξη που έχει για ένα μεγάλο σύνολο βάσεων δεδομένων. Η συγγραφή μιας σελίδας που υποστηρίζει βάσεις δεδομένων είναι εξαιρετικά απλή. Οι εξής βάσεις δεδομένων υποστηρίζονται μέχρι στιγμής:

Adabas D	Ingres	Oracle (OCI7 and OCI8)
dBase	InterBase	Ovrimos
Empress	FrontBase	PostgreSQL
FilePro (read-only)	mSQL	Solid
Hyperwave	Direct MS-SQL	Sybase
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

Έχουμε επίσης μια αφαιρετική επέκταση DBX βάσεων δεδομένων (DBX database abstraction extension) που σας επιτρέπει διάφανα να χρησιμοποιείτε οποιαδήποτε βάση δεδομένων υποστηρίζεται από αυτή την επέκταση. Επιπλέον η PHP υποστηρίζει το ODBC, το Open Database Connection standard (Ανοιχτό πρότυπο Σύνδεσης Βάσεων δεδομένων) έτσι μπορείτε να συνδεθείτε σε οποιαδήποτε βάση δεδομένων που υποστηρίζει αυτό το παγκόσμιο πρότυπο.

Η PHP έχει επίσης υποστήριξη για επικοινωνία με άλλες υπηρεσίες χρησιμοποιώντας πρωτόκολλα όπως LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (στα Windows) και αμέτρητα άλλα. Μπορείτε επίσης να ανοίξετε raw network sockets και να αλληλεπιδράσετε με οποιοδήποτε άλλο πρωτόκολλο. Η PHP έχει ακόμη υποστήριξη για την περίπλοκη ανταλλαγή δεδομένων WDDX μεταξύ σχεδόν όλων των Web programming γλωσσών. Μιλώντας για δια-επικοινωνία, η PHP υποστηρίζει instantiation αντικειμένων Java και τα χρησιμοποιεί διάφανα σαν αντικείμενα PHP. Μπορείτε επίσης να χρησιμοποιήσετε την CORBA επέκταση μας για να προσπελάσετε remote (απομακρυσμένα) αντικείμενα.

8.5.3 Επικοινωνία της PHP με βάσεις δεδομένων

Η PHP θεωρείται η καλύτερη επιλογή για την δημιουργία δυναμικών εφαρμογών που επικοινωνούν με βάσεις δεδομένων, ένας βασικός λόγος για να την επιλέξουμε είναι ότι διαθέτει ενσωματωμένες συναρτήσεις που επικοινωνούν με μεγάλο αριθμό εμπορικών συστημάτων βάσεων δεδομένων.

Το παρακάτω παράδειγμα παρουσιάζει τις συναρτήσεις της PHP, που χρησιμοποιούνται για τη σύνδεση με μια βάση δεδομένων στη MySQL, τη δημιουργία ερωτήσεων και την ανάκτηση των αποτελεσμάτων για περαιτέρω επεξεργασία τους από την εφαρμογή.

```
<?php
```

```
//Δημιουργία σύνδεσης προς τον MySQL Server
$connection = mysql_connect("localhost","username","passwd");
//Εμφάνιση κωδικού σφάλματος σε περίπτωση αποτυχημένης σύνδεσης
if (!$connection) {
die('Could not connect: ' . mysql_error());
}
//Επιλογή μιας βάσης δεδομένων
mysql_select_db("mydb", $connection);
// Δημιουργία μιας επερώτησης (query) προς τη βάση
$result = mysql_query ("SELECT * FROM mytable", $connection);
// Επιστροφή των αποτελεσμάτων του query
while ($row = mysql_fetch_array($result, MYSQL_NUM))
{
// επεξεργασία των αποτελεσμάτων
}
//Κλείσιμο της σύνδεσης με τον Server
mysql_close ($connection);
?>
```

- **mysql_connect()** χρησιμοποιείται για τη σύνδεση με τον MySQL Server και δέχεται τις εξής παραμέτρους: το hostname ή την IP διεύθυνση του server, το όνομα του χρήστη που έχει πρόσβαση στον server και τον κωδικό ασφαλείας του. Η τιμή που επιστρέφει η συνάρτηση χρησιμοποιείται σαν παράμετρος στις μετέπειτα συναρτήσεις που καλούνται προς τον MySQL Server.
- **mysql_error()** επιστρέφει το μήνυμα λάθους σε περίπτωση αποτυχίας της προηγούμενης ενέργειας που έγινε προς τον MySQL Server.
- **mysql_select_db()** διαλέγει μια συγκεκριμένη βάση δεδομένων η οποία έχει ήδη δημιουργηθεί στον MySQL Server. Οι παράμετροι που εμπεριέχονται είναι το όνομα της βάσης και η τιμή που επιστράφηκε από την **mysql_connect()**.
- **mysql_query()** χρησιμοποιείται για την αποστολή μιας επερώτησης (query) στη βάση δεδομένων που ορίζει η δεύτερη παράμετρος (\$connection). Η πρώτη παράμετρος είναι το query που θέλουμε να εκτελεστεί.
- **mysql_fetch_array()** επιτρέπει την ανάκτηση των εγγραφών (rows) που επέστρεψε προηγούμενο query προς τη βάση.
- **mysql_close()** χρησιμοποιείται για να κλείσει την σύνδεση που ορίζει η παράμετρος \$connection, αν και αυτό δεν είναι απαραίτητο αφού όλες οι συνδέσεις προς την βάση δεδομένων κλείνουν αυτόματα όταν τελειώνει το script.

8.6 MySQL



Η MySQL είναι ένα πολύ γρήγορο και δυνατό , σύστημα διαχείρισης βάσεων δεδομένων. Μια βάση δεδομένων επιτρέπει την αποθήκευση , την αναζήτηση , την ταξινόμηση και επίσης δίνει τη δυνατότητα να ανακαλέσει ο προγραμματιστής τα δεδομένα αποτελεσματικά . Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα σας , για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα , για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση.

Συνεπώς η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Χρησιμοποιεί την SQL (Structured Query Language) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια . Η MySQL είναι διαθέσιμη από το 1996 αλλά η ιστορία της ξεκινά από το 1979 .

Ένα από τα μεγαλύτερα πλεονεκτήματα της MySQL είναι η δυνατότητα που δίνει για σύνδεση σε πολλές διαφορετικές βάσεις δεδομένων. Οι βάσεις δεδομένων που υποστηρίζονται περιλαμβάνουν τις: Adabas D, InterBase, PostgreSQL, dBase, FrontBase, SQLite, Empress, mSQL, Solid, FilePro (read-only), Direct MS-SQL, Sybase, Hyperwave, MySQL, Velocis, IBM, ODBC.

8.6.1 Πλεονεκτήματα MySQL

Μερικοί από τους κύριους ανταγωνιστές της MySQL είναι οι PostgreSQL , Microsoft SQL και Oracle. Η MySQL έχει πολλά πλεονεκτήματα , όπως χαμηλό κόστος , εύκολη διαμόρφωση και μάθηση και ο κώδικας προέλευσης είναι διαθέσιμος.

- **Απόδοση** : Η MySQL είναι χωρίς αμφιβολία γρήγορη. Μπορείτε να δείτε την σελίδα δοκιμών <http://web.mysql.com/benchmark.html> . Πολλές από αυτές τις δοκιμές δείχνουν ότι η MySQL είναι αρκετά πιο γρήγορη από τον ανταγωνισμό.
- **Χαμηλό κόστος** : Η MySQL είναι διαθέσιμη δωρεάν , με άδεια ανοικτού κώδικα (Open Source) ή με χαμηλό κόστος , αν πάρετε εμπορική άδεια, αν απαιτείται από την εφαρμογή σας.
- **Ευκολία Χρήσης** : Οι περισσότερες μοντέρνες βάσεις δεδομένων χρησιμοποιούν SQL. Αν έχετε χρησιμοποιήσει ένα άλλο σύστημα διαχείρισης βάσεων δεδομένων δεν θα έχετε πρόβλημα να προσαρμοστείτε σε αυτό.

- **Μεταφερσιμότητα :** Η MySQL μπορεί να χρησιμοποιηθεί σε πολλά διαφορετικά συστήματα Unix όπως επίσης και στα Microsoft Windows .
- **Κώδικας Προέλευσης :** Όπως και με την PHP , μπορείτε να πάρετε και να τροποποιήσετε τον κώδικα προέλευσης της MySQL.
- **Νέα έκδοση :** Η νέα έκδοση MySQL 5 έχει έρθει με νέες εντυπωσιακές λειτουργίες. Είναι πλέον ικανή να υποστηρίξει πολύ μεγάλα projects με υψηλή αξιοπιστία.

8.6.2 Τρόπος λειτουργίας MySQL Βάσης Δεδομένων

Ο τρόπος λειτουργίας της MySQL είναι ίδιος με αυτόν που ακολουθούν όλες οι Web βάσεις δεδομένων. Τα βήματα λειτουργίας των αρχιτεκτονικών των Web Βάσεων Δεδομένων μπορούν να συνοψιστούν στα παρακάτω:

- Ο web browser ενός χρήστη κάνει μια HTTP αίτηση για μια συγκεκριμένη ιστοσελίδα. Για παράδειγμα, μπορεί να κάνει μια αναζήτηση για όλους τους χρήστες οι οποίοι είναι εγγεγραμμένοι σε ιστοσελίδα. Έστω ότι η σελίδα των αποτελεσμάτων αναζήτησης ονομάζεται results.php
- Ο web διακομιστής λαμβάνει την αίτηση για τη σελίδα results.php, ανακαλεί το αρχείο και το περνά στην μηχανή PHP για επεξεργασία.
- Η μηχανή PHP αρχίζει την ανάλυση του script. Μέσα στο script, υπάρχει μια εντολή που συνδέει την βάση δεδομένων και εκτελεί ένα ερώτημα (την αναζήτηση των χρηστών). Η PHP ανοίγει μια σύνδεση με τον MySQL διακομιστή και στέλνει το κατάλληλο ερώτημα.
- Ο MySQL διακομιστής λαμβάνει το ερώτημα της βάσης δεδομένων και το επεξεργάζεται και στέλνει τα αποτελέσματα (μια λίστα χρηστών), ξανά στη μηχανή PHP.
- Η μηχανή PHP σταματά την εκτέλεση του script, που συνήθως περιλαμβάνει την μορφοποίηση των αποτελεσμάτων του ερωτήματος σε HTML. Επιστρέφει μετά την τελική HTML σελίδα στο web διακομιστή.
- Ο web διακομιστής περνά την HTML σελίδα ξανά στο browser, όπου ο χρήστης μπορεί να δει τη λίστα των σπουδαστών που ζήτησε.

8.7 Apache Web Server



Ο Apache Web Server είναι ένας πολύ δημοφιλής διακομιστής διαδικτύου που διανέμεται ελεύθερα στο διαδίκτυο. Αναπτύχθηκε και συντηρείται από μια ομάδα εθελοντών που ήθελαν να υλοποιήσουν έναν εύρωστο κώδικα για διακομιστή δικτύου, που να είναι εμπορικός και να έχει πολλά χαρακτηριστικά.

Ο Apache όπως έχει αποδειχτεί είναι ο πιο γρήγορος, σταθερός, ασφαλής και υποστηρίζει τα περισσότερα χαρακτηριστικά από οποιονδήποτε άλλο διακομιστή δικτύου. Ο Apache είναι εγκατεστημένος στο 80% των διακομιστών παγκοσμίως (πάνω από 6 εκατομμύρια διακομιστές). Πάνω του είναι εγκατεστημένα εκατομμύρια sites που δέχονται εκατομμύρια hits καθημερινά χωρίς να παρουσιάζεται κανένα απολύτως πρόβλημα.

Σήμερα ο Apache θεωρείται από τους πιο σταθερούς διακομιστές δικτύου που κυκλοφορούν και θα πρέπει να τονίσουμε ότι αρκετοί εμπορικοί διακομιστές διαδικτύου, όπως ο HTTP Server της IBM, χρησιμοποιούν τον πυρήνα του Apache.

8.8 EasyPHP

Για την ανάπτυξη και διανομή διαδικτυακών εφαρμογών είναι απαραίτητο κάποιο εργαλείο, το οποίο να υποστηρίζει την εκτέλεση δυναμικών ιστοσελίδων. Δηλαδή, να διαθέτει έναν web server που να μπορεί να εκτελεί server side scripts καθώς και να υποστηρίζει τη χρήση βάσεων δεδομένων.

Η εφαρμογή EasyPHP 1.8, η οποία χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής, αποτελείται από τα παρακάτω πακέτα λογισμικού:

1. Το Διαδικτυακό Εξυπηρετητή (web server) Apache 1.3.33

Στον web server αποθηκεύονται όλα τα αρχεία (HTML, PHP) μιας ιστοσελίδας και είναι υπεύθυνος για την αλληλεπίδραση με τα προγράμματα περιήγησης των επισκεπτών.

2. Τη γλώσσα σεναρίων PHP 4.3.10

Έτσι ώστε να μπορεί να εκτελεί σενάρια στον εξυπηρετητή και να αποστέλλει τα αποτελέσματα στον επισκέπτη.

3. Το εργαλείο για την διαχείριση βάσεων δεδομένων phpMyAdmin 2.6.1

Με αυτή την εφαρμογή, δίνεται η δυνατότητα διαχείρισης των διάφορων βάσεων δεδομένων, με εύκολο τρόπο μέσω web browser και χωρίς την πληκτρολόγηση εντολών.

4. Τον εξυπηρετητή βάσεων δεδομένων MySQL 4.1.9

8.8.1 Εγκατάσταση Easy PHP

Τρέχουμε το αρχείο EasyPHP1.8 και ακολουθούμε τα βήματα εγκατάστασης όπως φαίνονται στα screenshots που ακολουθούν.



Εικόνα 15 : Εγκατάσταση EasyPHP

Στη συνέχεια αποδεχόμαστε τους όρους χρήσης και πατάμε “next”.



Εικόνα 16 : Επιλογή αποδοχής των όρων χρήσης

Επιλέγουμε τη διαδρομή για την εγκατάσταση. Καλύτερα είναι να αφήσουμε αυτό που μας προτείνει .



Εικόνα 17 : Επιλογή διαδρομής για την εγκατάσταση

Στη συνέχεια πατάμε “install” ώστε να ξεκινήσει η εγκατάσταση



Εικόνα 18 : Επιλογή Install

Εφόσον έχουμε ολοκληρώσει την εγκατάσταση ανοίγουμε τον EasyPHP



Εικόνα 19 : EasyPHP

Στη συνέχεια ανοίγουμε την αρχική σελίδα του EasyPHP είτε πληκτρολογώντας τη διεύθυνση : <http://127.0.0.1/home/> είτε κάνοντας δεξι κλικ->Administration πάνω στο server μας.



Εικόνα 20 : Αρχική σελίδα του EasyPHP

8.9 PhpMyAdmin



Το PhpMyAdmin είναι ένα ελεύθερο λογισμικό ανοιχτού κώδικα με το οποίο ο χρήστης /προγραμματιστής έχει την δυνατότητα να διαχειρίζεται την MySQL στο δίκτυο. Μπορεί να χειρίζεται πλήρως βάσεις δεδομένων, πίνακες, πεδία πινάκων αλλά και ολόκληρο τον MySQL Server. Υποστηρίζει 47 γλώσσες μεταξύ των οποίων και τα Ελληνικά .

Οι δυνατότητες του PhpMyAdmin είναι οι εξής:

- Να δημιουργεί και να διαγράφει βάσεις δεδομένων .
- Να δημιουργεί, τροποποιεί, διαγράφει, αντιγράφει και μετονομάζει πίνακες σε μια υπάρχουσα βάση .
- Να κάνει την απαραίτητη συντήρηση της βάσης .
- Να προσθέτει, διαγράφει και τροποποιεί τα πεδία των πινάκων .
- Να εκτελεί Sql ερωτήματα προς την βάση .
- Να δημιουργεί αυτόματα πολύπλοκα ερωτήματα χρησιμοποιώντας το QBE (Query-by-example) ,που είναι σε στυλ συμπλήρωσης φορμών .
- Να κάνει εύρεση δεδομένων ,γενικά στη βάση ή ειδικότερα σε κάποια υποδιαίρεσή της (πίνακα ή πεδίο).
- Να διαχειρίζεται κλειδιά των διαφόρων πεδίων των πινάκων της βάσης .
- Να φορτώνει αρχεία κειμένου σε πίνακες της βάσης .
- Να δημιουργεί και να διαβάζει αναφορά οποιασδήποτε τυχόν απόρριψης των πινάκων της βάσης .
- Να εισάγει και να εξάγει δεδομένα σε μορφή CVS, Latex και XML .
- Να διαχειρίζεται πολλαπλούς διακομιστές .
- Διαχειρίζεται τους χρήστες MySQL και τα δικαιώματά τους .
- Να ελέγχει την αναφορική ακεραιότητα των δεδομένων των πινάκων της βάσης .
- Να δημιουργεί γραφικά το σχεδιασμό της βάσης δεδομένων σε μορφή PDF .

Κεφάλαιο 9

Υλοποίηση Βάσης Δεδομένων

9.1 Εισαγωγή στην ανάλυση απαιτήσεων

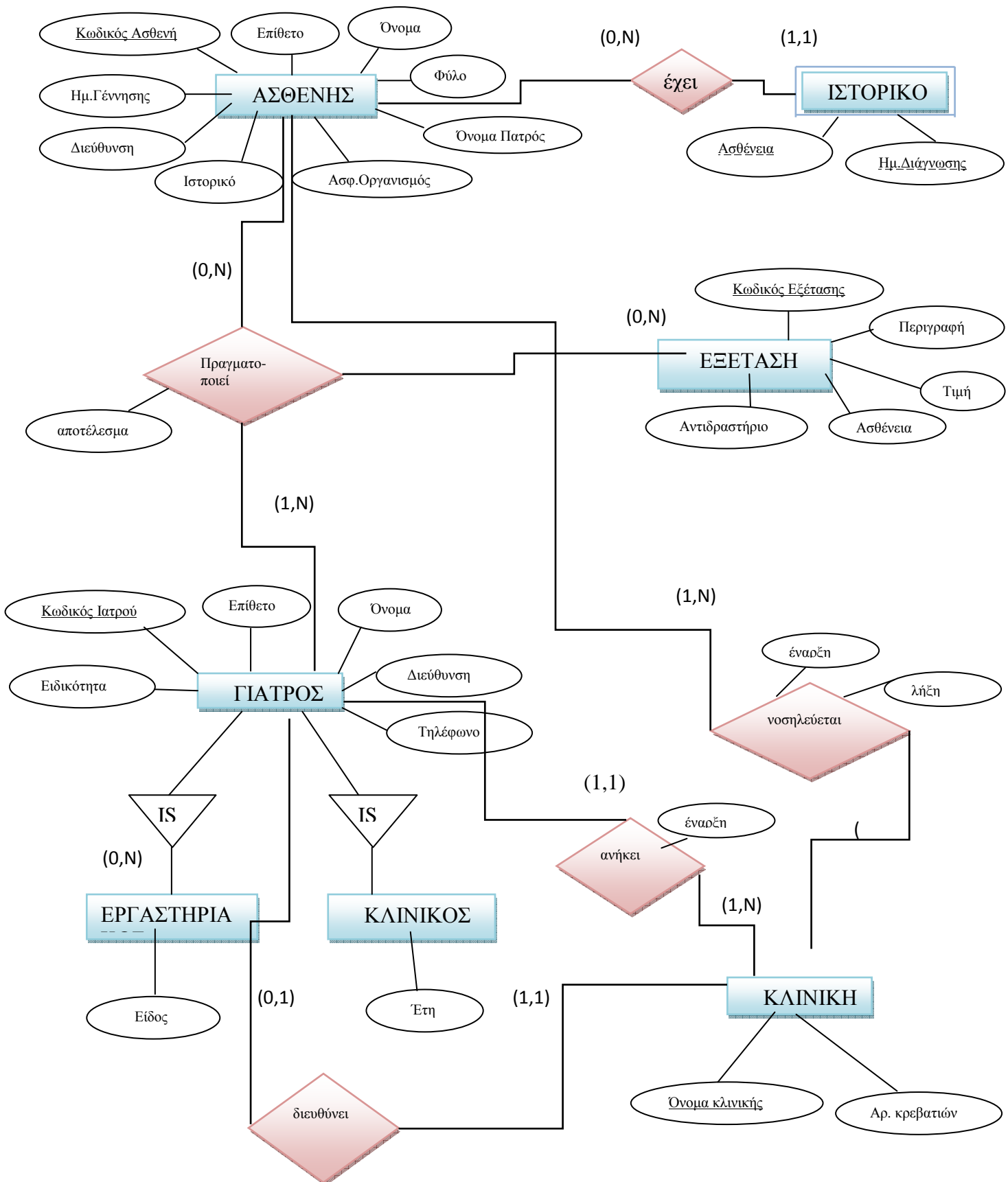
Η ανάλυση απαιτήσεων είναι μια διαδικασία κατάρτισης μιας λίστας, όπου αναφέρονται οι προδιαγραφές που πρέπει να πληρεί η εφαρμογή που πρόκειται να δημιουργηθεί. Οι προδιαγραφές που προσδιορίζονται μπορεί να είναι τεχνολογικές, επιχειρηματικές, λειτουργικές, να σχετίζονται με τη μορφή, το κόστος, τη διάρκεια ή ακόμη και το χρόνο απόσβεσης.

Η λίστα που καταρτίζεται χρησιμεύει τόσο σε αυτούς που θα αναπτύξουν την εφαρμογή όσο και σ' εκείνους που θα τη χρησιμοποιήσουν. Η ανάλυση απαιτήσεων είναι μία συνεργατική διαδικασία όπου διαφορετικά άτομα με διαφορετικές αφετηρίες συναντιούνται, αλληλεπιδρούν, διαφωνούν και συμφωνούν γύρω από το ίδιο αντικείμενο: το έργο.

9.2 Ανάλυση απαιτήσεων της εφαρμογής

- Προσθήκη Ασθενή
Ο χρήστης έχει την δυνατότητα εισαγωγής των δημογραφικών στοιχείων του ασθενή , τα οποία αποθηκεύονται στη Βάση Δεδομένων .
- Αναζήτηση Ασθενή
Ο χρήστης έχει την δυνατότητα μέσω της αναζήτησης την εμφάνιση των προσωπικών στοιχείων ενός συγκεκριμένου ασθενή , καθώς επίσης έχει την δυνατότητα επξεργασίας και διαγραφής των στοιχείων αυτών .
- Εμφάνιση Ιστορικού
Ο χρήστης έχει την δυνατότητα εμφάνισης του ιστορικού ενός ασθενή .
- Προσθήκη Εξέτασης
Ο χρήστης μπορεί να εισάγει στην Βάση Δεδομένων τα στοιχεία της εξέτασης που αφορούν ένα ασθενή .
- Στατιστικά Ασθενειών
Ο χρήστης μπορεί να αναζητήσει πόσα περιστατικά μια συγκεκριμένης ασθένειας έλαβαν χώρα σε ένα συγκεκριμένο χρονικό διάστημα.
- Στατιστικά Κόστους Εξετάσεων
Ο χρήστης μπορεί να ενημερωθεί για το συνολικό κόστος των εξετάσεων που έγιναν σε ένα ορισμένο χρονικό διάστημα.
- Στατιστικά Εξετάσεων
Ο χρήστης μπορεί να ενημερωθεί για το πόσες εξετάσεις καθώς και ποιο είναι το κόστος κάθε είδους εξέτασης για κάποιο συγκεκριμένο χρονικό διάστημα.
- Επικοινωνία με Administrator
Σε περίπτωση εμφάνισης κάποιου προβλήματος ο χρήστης μπορεί να έρθει σε επικοινωνία με τον διαχειριστή του συστήματος για επίλυση κάποιου προβλήματος ώστε να εξασφαλιστεί η ομαλή λειτουργία της εφαρμογής.

9.3 Διαγραμματική Απεικόνιση Βάσης Δεδομένων



9.4 Περιγραφή Βάσης της Εφαρμογής

Για τις ανάγκες της διαδικτυακής μας εφαρμογής δημιουργήθηκε μια Βάση Δεδομένων , η οποία αποθηκεύει τις απαραίτητες πληροφορίες Έτσι μας δίνεται η δυνατότητα να έχουμε μια ιστοσελίδα με περιεχόμενο που ανανεώνεται δυναμικά, κάθε φορά που τροποποιούνται τα στοιχεία της Βάσης . Με αυτό τον τρόπο γίνεται πιο εύκολη η διαχείριση και η προβολή του περιεχομένου της εφαρμογής.

Η Βάση Δεδομένων που δημιουργήθηκε ονομάζεται medico και αποτελείται από 9 πίνακες.(Εικόνα: 21). Οι πίνακες αυτοί που προέκυψαν από την ανάλυση απαιτήσεων του συστήματος παρουσιάζονται συνοπτικά παρακάτω .

Για τις ανάγκες της διαδικτυακής μας εφαρμογής δημιουργήθηκε μια Βάση Δεδομένων , η οποία αποθηκεύει τις απαραίτητες πληροφορίες Έτσι μας δίνεται η δυνατότητα να έχουμε μια ιστοσελίδα με περιεχόμενο που ανανεώνεται δυναμικά, κάθε φορά που τροποποιούνται τα στοιχεία της Βάσης . Με αυτό τον τρόπο γίνεται πιο εύκολη η διαχείριση και η προβολή του περιεχομένου της εφαρμογής.

Η Βάση Δεδομένων που δημιουργήθηκε ονομάζεται medico και αποτελείται από 9 πίνακες.(Εικόνα: 21). Οι πίνακες αυτοί που προέκυψαν από την ανάλυση απαιτήσεων του συστήματος παρουσιάζονται συνοπτικά παρακάτω .

Διακομιστής: localhost ▶ Βάση: medico

Δομή SQL Εξαγωγή Αναζήτηση Επερώτημα κατά παράδειγμα Λειτουργίες Διαγραφή

Πίνακας	Ενέργεια	Εγγραφές	Τύπος	Collation	Μέγεθος	Επιβάρυνση
doctor		5	MyISAM	latin1_swedish_ci	3,3 KB	-
ergastiriakosdoctor		0	MyISAM	latin1_swedish_ci	1,0 KB	-
exetasi		19	MyISAM	latin1_swedish_ci	2,7 KB	-
istoriko		4	MyISAM	latin1_swedish_ci	2,1 KB	-
kliniki		0	MyISAM	latin1_swedish_ci	1,0 KB	-
klinikosdoctor		0	MyISAM	latin1_swedish_ci	1,0 KB	-
nosilia		0	MyISAM	latin1_swedish_ci	1,0 KB	-
patient		9	MyISAM	latin1_swedish_ci	2,7 KB	-
pragmatopoiisixet		0	MyISAM	latin1_swedish_ci	1,0 KB	-
9 Πίνακας/Πίνακες	Σύνολο	37	--	latin1_swedish_ci	15,8 KB	0 Bytes

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εμφάνιση για εκτύπωση Λεξικό δεδομένων

Δημιουργία νέου πίνακα στη βάση medico:

Όνομα:

Πεδία: Εκτέλεση























































Εικόνα 21 : PhpMyAdmin –Πίνακες της Βάσης

9.5 Περιγραφή Πινάκων της Βάσης Δεδομένων

9.5.1 Ο πίνακας « patient »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των δημογραφικών στοιχείων των ασθενών .Τα πεδία αυτού του πίνακα είναι :

Πίνακας : Patient	Περιγραφή
patientid (varchar)	Μοναδικό αναγνωριστικό του ασθενή. Είναι το κύριο κλειδί του πίνακα
lname (varchar)	Επίθετο ασθενή
fname (varchar)	Όνομα ασθενή
patronimo (varchar)	Όνομα πατρός ασθενή
address (varchar)	Διεύθυνση ασθενή
bdate (date)	Ημερομηνία γέννησης ασθενή
sex (varchar)	Φύλο ασθενή
insurance (varchar)	Ασφαλιστικός Οργανισμός ασθενή
History (text)	Ιστορικό ασθενή

	Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	lname	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	fname	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	patronimo	varchar(10)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	id	varchar(7)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	address	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	bdate	date			Όχι	0000-00-00		     
<input type="checkbox"/>	sex	varchar(5)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	insurance	varchar(10)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	history	text	latin1_swedish_ci		Όχι			     























































Εικόνα 22 : Πίνακας patient

9.5.2 Ο πίνακας « doctor »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των στοιχείων των γιατρών .Τα πεδία αυτού του πίνακα είναι :

Πίνακας : Doctor	Περιγραφή
doctortid (varchar)	Μοναδικό αναγνωριστικό του γιατρού. Είναι το κύριο κλειδί του πίνακα .

Iname (varchar)	Επίθετο γιατρού
fname (varchar)	Όνομα γιατρού
eidikotita (varchar)	Ειδικότητα γιατρού
address (varchar)	Διεύθυνση γιατρού
phone (int)	Τηλέφωνο γιατρού
idkliniki (varchar)	Κωδικός κλινικής
idexetasis (varchar)	Κωδικός εξέτασης
username (varchar)	Όνομα χρηστη
Password (varchar)	Κωδικός πρόσβασης













Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> Lname	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/> Fname	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/> doctorid	varchar(10)	latin1_swedish_ci		Όχι	0		     
<input type="checkbox"/> idkliniki	varchar(7)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/> idexetasis	varchar(7)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/> idikotita	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/> phone	int(10)			Όχι	0		     
<input type="checkbox"/> address	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/> username	varchar(20)	latin1_swedish_ci		Όχι			     

Εικόνα 23 : Πίνακας doctor

9.5.3 Ο πίνακας « *ergastiriakosdoctor* »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των στοιχείων των εργαστηριακών γιατρών .Τα πεδία αυτού του πίνακα είναι :

Πίνακας : Ergastiriakosdoctor	Περιγραφή
doctortid (varchar)	Μοναδικό αναγνωριστικό του γιατρού. Είναι το κύριο κλειδί του πίνακα .
eidos (varchar)	Το είδος , δηλαδή αν ασχολούνται με ιογενής ή κληρονομικές ασθένειες.













	Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	doctorid	varchar(8)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	eidoid	varchar(15)	latin1_swedish_ci		Όχι			     

Εικόνα 24: Πίνακας ergastiriakosdoctor

9.5.4 Ο πίνακας « *klinikosdoctor* »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των στοιχείων των κλινικών γιατρών .Τα πεδία αυτού του πίνακα είναι :

Πίνακας :Klinikosdoctor	Περιγραφή
doctortid (varchar)	Μοναδικό αναγνωριστικό του γιατρού.
years (varchar)	Τα έτη που έχουν εργαστεί μέσα σε νοσοκομειακούς χώρους



















	Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	doctorid	varchar(7)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	years	int(5)			Όχι	0		     

Εικόνα 25: Πίνακας klinikosdoctor

9.5.5 Ο πίνακας « *istoriko* »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση του ιστορικού ενός ασθενή . Τα πεδία αυτού του πίνακα είναι :

Πίνακας :Istoriko	Περιγραφή
patientid (varchar)	Μοναδικό αναγνωριστικό του ασθενή. Είναι το κύριο κλειδί του πίνακα .
asthenia (varchar)	Περιγραφή της ασθένεια του ασθενή.
Datediagnosis (date)	Ημερομηνία διάγνωσης της ασθένειας .































	Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	patientid	varchar(10)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	asthenia	varchar(15)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	datediagnvsis	bigint(20)			Όχι	0		     

Εικόνα 26: Πίνακας istoriko

9.5.6 Ο πίνακας « *exetasi* »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των δεδομένων μιας εξέτασης .
Τα πεδία αυτού του πίνακα είναι :

Πίνακας :Exetasi	Περιγραφή
<u>Idexetasis</u> (varchar)	Μοναδικό αναγνωριστικό της εξέτασης . Είναι το κύριο κλειδί του πίνακα .
perigrافي (varchar)	Περιγραφή του είδους της εξέτασης .
asthenia (varchar)	Ασθένεια
adidrastirio (varchar)	το αντιδραστήριο τα οποίο χρησιμοποιεί .
timiexetasis (varchar)	Η τιμή της εξέτασης .

	Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	idexetasis	varchar(10)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	description	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	asthenia	varchar(20)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	timiexetasis	varchar(10)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	antidrastirio	varchar(10)	latin1_swedish_ci		Όχι			     






















Εικόνα 27: Πίνακας exetasi

9.5.7 Ο πίνακας « *kliniki* »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των δεδομένων της κλινικής .
Τα πεδία αυτού του πίνακα είναι :

Πίνακας : Kliniki	Περιγραφή
<u>idkliniki</u> (varchar)	Μοναδικό αναγνωριστικό της κλινικής .

	Είναι το κύριο κλειδί του πίνακα
idmanager (varchar)	Ο κωδικός του διευθυντή της κλινικής
nbed (int)	Ο αριθμός των κρεβατιών της κλινικής





























Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> idkliniki	varchar(20)	latin1_swedish_ci		Όχι			      
<input type="checkbox"/> idmanager	varchar(7)	latin1_swedish_ci		Όχι			      
<input type="checkbox"/> nbed	int(11)			Όχι	0		      

Εικόνα 28: Πίνακας kliniki

9.5.8 Ο πίνακας « nosilia »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των δεδομένων της νοσηλείας ενός ασθενή στο νοσοκομείο . Τα πεδία αυτού του πίνακα είναι :

Πίνακας : Nosilia	Περιγραφή
patientid (varchar)	Μοναδικό αναγνωριστικό του ασθενή. Είναι το κύριο κλειδί του πίνακα .
idkliniki (varchar)	Μοναδικό αναγνωριστικό της κλινικής . Είναι το κύριο κλειδί του πίνακα .
startdate (date)	Ημερομηνία έναρξης της νοσηλείας .
enddate (date)	Ημερομηνία λήξης της νοσηλείας .



















Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> patientid	varchar(7)	latin1_swedish_ci		Όχι			      
<input type="checkbox"/> idkliniki	varchar(7)	latin1_swedish_ci		Όχι			      
<input type="checkbox"/> startdate	date			Όχι	0000-00-00		      
<input type="checkbox"/> enddate	date			Όχι	0000-00-00		      

Εικόνα 29: Πίνακας nosilia

9.5.9 Ο πίνακας « pragmatopoiisiexet »

Ο πίνακας αυτός χρησιμοποιείται για την αποθήκευση των δεδομένων των αποτελεσμάτων μιας εξέτασης . Τα πεδία αυτού του πίνακα είναι :

Πίνακας : pragmatopoiisiexet	Περιγραφή
<u>patientid</u> (varchar)	Μοναδικό αναγνωριστικό του ασθενή. Είναι το κύριο κλειδί του πίνακα
<u>Idexetasis</u> (varchar)	Μοναδικό αναγνωριστικό της εξέτασης . Είναι το κύριο κλειδί του πίνακα .
apotelesma (varchar)	Αποτέλεσμα .

	Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	patientid	varchar(7)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	idexetasis	varchar(7)	latin1_swedish_ci		Όχι			     
<input type="checkbox"/>	apotelesma	varchar(20)	latin1_swedish_ci		Όχι			     

Εικόνα 30: Πίνακας pragmatopoiisiexet

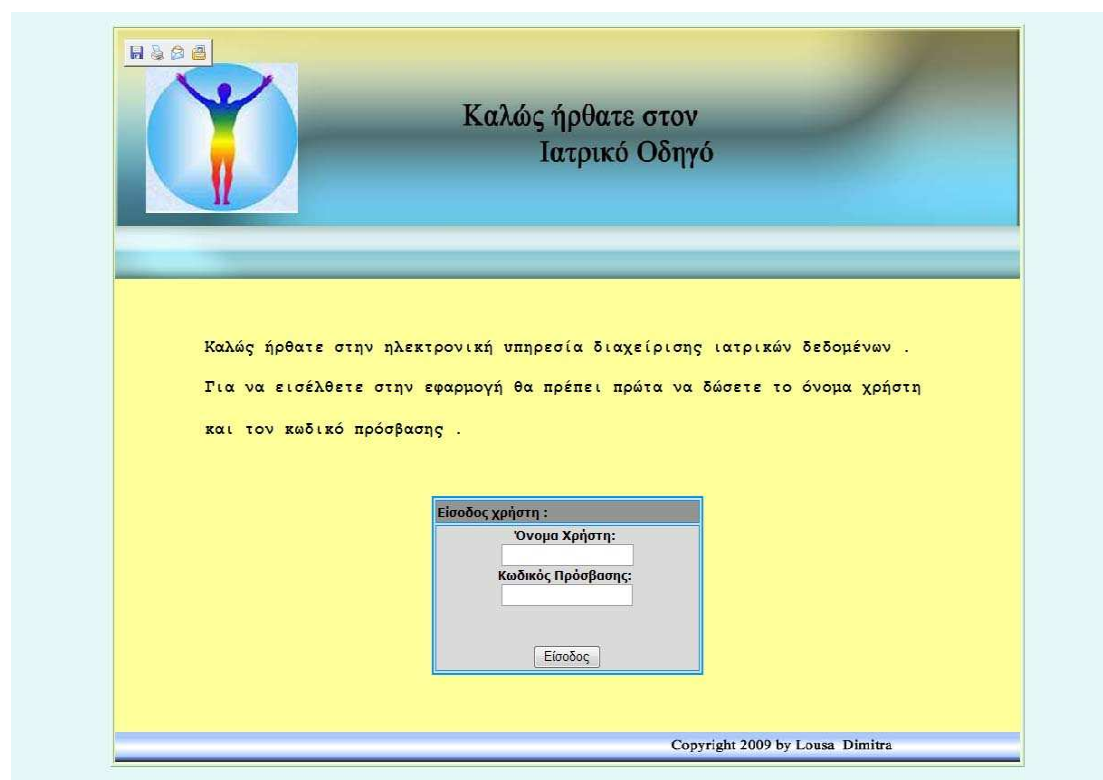
Κεφάλαιο 10 Παρουσίαση Εφαρμογής

10.1 Εισαγωγή

Στο κεφάλαιο αυτό γίνει μία αναλυτική παρουσίαση της εφαρμογής που υλοποιήθηκε από τη σκοπιά του χρήστη του συστήματος, ενώ παράλληλα με κάθε βήμα θα αναλύονται οι εσωτερικές λειτουργίες του συστήματος και θα αιτιολογούνται οι επιλογές που έχουν γίνει.

10.2 Είσοδος Ιατρού

Όπως φαίνεται στην παρακάτω εικόνα , κατά την είσοδο του Ιατρού στην εφαρμογή η πρώτη σελίδα που εμφανίζεται είναι , η σελίδα εισόδου στην οποία ο Ιατρός θα πρέπει να εισάγει τα στοιχεία του σωστά για να μπορέσει να μπει στο σύστημα. Τα στοιχεία που καταχωρούνται στην φόρμα αυτή αναζητούνται στην Βάση Δεδομένων και συγκεκριμένα στον πίνακα doctor. Αν τα στοιχεία που έχει εισάγει ο χρήστης υπάρχουν στον πίνακα “doctor” , του επιτρέπει να μπει στην κεντρική σελίδα του site αλλιώς , τον παραπέμπει να ξαναπροσπαθήσει έως ότου δώσει τα σωστά στοιχεία .



Εικόνα 31: Είσοδος Γιατρού στην εφαρμογή

Ο έλεγχος που γίνεται, έτσι ώστε να διαπιστωθεί αν τα στοιχεία που έχει εισάγει ο χρήστης στην φόρμα είναι τα σωστά , πραγματοποιείται με το παρακάτω κώδικα php:

```

$con=mysql_connect($dbServer,$dbUser,$dbPass) or exit("Cannot connect
to database!");
$username=$_POST['username'];
$password=$_POST['password'];
$sql="SELECT * FROM doctor WHERE username='$username' AND
password='$password'";
$res=mysql_db_query($dbDatabase , $sql);
if (($res!=null) && (mysql_num_rows($res)>0))
{
$_SESSION['username'] = $_POST['username'];
$_SESSION['password'] = $_POST['password'];
echo '<META HTTP-EQUIV="Refresh"
CONTENT="0;URL=http://127.0.0.1/second.html">';
}
else
{
echo '<META HTTP-EQUIV="Refresh"
CONTENT="0;URL=http://127.0.0.1/login.html">';
}
?>

```

Με τις εντολές `$_SESSION['username'] = $_POST['username']` και `$_SESSION['password'] = $_POST['password']` γίνεται αποθήκευση του ονόματος του χρήστη και του κωδικού πρόσβασης σε μεταβλητές `$_SESSION` έτσι ώστε να ανακτηθούν και να τροποποιηθούν από οποιαδήποτε άλλη σελίδα της εφαρμογής

10.3 Αρχική σελίδα

Ύστερα από την επιτυχή είσοδο του γιατρού η πρώτη σελίδα που εμφανίζεται είναι αυτή που φαίνεται παρακάτω .



Εικόνα 32 : Αρχική σελίδα

Όπως φαίνεται στην εικόνα 32 η σελίδα χωρίζεται σε τρία βασικά μέρη:

- 1) Την κεφαλίδα, η οποία περιλαμβάνει το λογότυπο του ιατρικού οδηγού .
- 2) Την πλευρική γραμμή μενού που έχει σαν στόχο την εμφάνιση των ενεργειών που μπορεί να πραγματοποιήσει ο χρήστης μέσω της εφαρμογής , και εμφανίζεται σε όλες τις σελίδες για ευκολότερη πρόσβαση του χρήστη στις διάφορες υπηρεσίες .
- 3) Και τον υποδοχέα στο κέντρο της σελίδας που θα περιλαμβάνει κάθε φορά τη βασική πληροφορία .

10.4 Προσθήκη ασθενή

Η πρώτη συνάντηση που γίνεται μεταξύ του γιατρού και του συγκεκριμένου ασθενούς , απαιτεί από την μεριά του γιατρού την καταγραφή κάποιων πληροφοριών σχετικών με τον ασθενή , όπως ονοματεπώνυμο, ημ/νία γέννησης, φύλο κ.λ.π.Έτσι, όταν ο χρήστης επιλέξει από το μενού την επιλογή « Προσθήκη Ασθενή » εμφανίζεται η παρακάτω σελίδα :

Καλώς ήρθατε στον
Ιατρικό Οδηγό

Προσθήκη Ασθενή
Αναζήτηση Ασθενή
Εμφάνιση Ιστορικού
Εισαγωγή Εξέτασης
Στατιστικά Ασθενειών
Στατιστικά Εξετάσεων
Στατιστικά Κόστους
Στοιχεία Ιατρού
Επικοινωνία
Έξοδος

Στοιχεία Ασθενή

Επίθετο :
Όνομα :
Όνομα πατρός :
Αρ. ταυτότητας :
Διεύθυνση :
Ημ. γέννησης :
Φύλο : Ανδρας Γυναίκα
Ασφαλιστικός Οργανισμός : Δημόσιο
Ιστορικό :
Εισαγωγή

Copyright 2009 by Lousa Dimitra

Εικόνα 33: Αρχική σελίδα

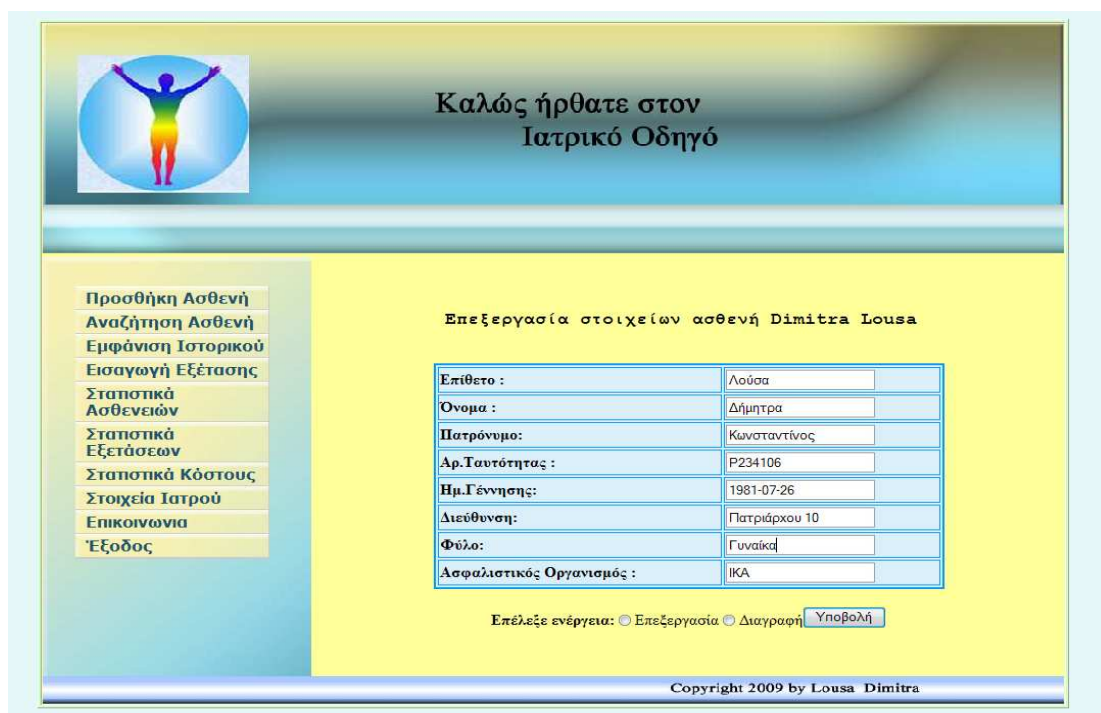
Στη φόρμα που εμφανίζεται ο χρήστης πρέπει να εισάγει τα απαραίτητα στοιχεία του ασθενή. Αφού συμπληρώσει τις απαραίτητες πληροφορίες πατώντας το κουμπί « Εισαγωγή » γίνεται η καταχώρηση των δεδομένων στην Βάση Δεδομένων και εμφανίζεται ένα μήνυμα που μας ενημερώνει ότι η εγγραφή πραγματοποιήθηκε με επιτυχία .

10.5 Αναζήτηση Ασθενή

Σε αυτό το σημείο ο Ιατρός μπορεί να αναζητήσει οποιοδήποτε ασθενή γράφοντας στην φόρμα που εμφανίζεται το αναγνωριστικό κωδικό του κάθε ασθενή. Στην εφαρμογή χρησιμοποιείται ο αριθμός Δελτίου Ταυτότητας σαν id των ασθενών , ο οποίος είναι μοναδικός και εύκολος στην απομνημόνευση του .



Εικόνα 34: Αναζήτηση Ασθενή



Εικόνα 35 : Επεξεργασία στοιχείων ασθενή

Μετά την εισαγωγή του αναγνωριστικού κωδικού του ασθενή ,ο Ιατρός έχει την δυνατότητα προβολής των στοιχείων αυτών, τα οποία μπορεί στην συνέχεια αν το επιθυμεί να προβεί σε επεξεργασία η διαγραφή των δεδομένων που είναι αποθηκευμένα στη βάση .

Η εικόνα που βλέπει ο Ιατρός μετά την αναζήτηση στην οθόνη του είναι η παραπάνω (Εικόνα 35).

10.6 Προσθήκη Εξέτασης

Σε αυτό το σημείο ο Ιατρός καταχωρεί τα στοιχεία της διάγνωσης του μετά την εξέταση του ασθενή , όπως είναι η ασθένεια ,η ημερομηνία διάγνωσης ,η τιμή της εξέτασης κλπ.

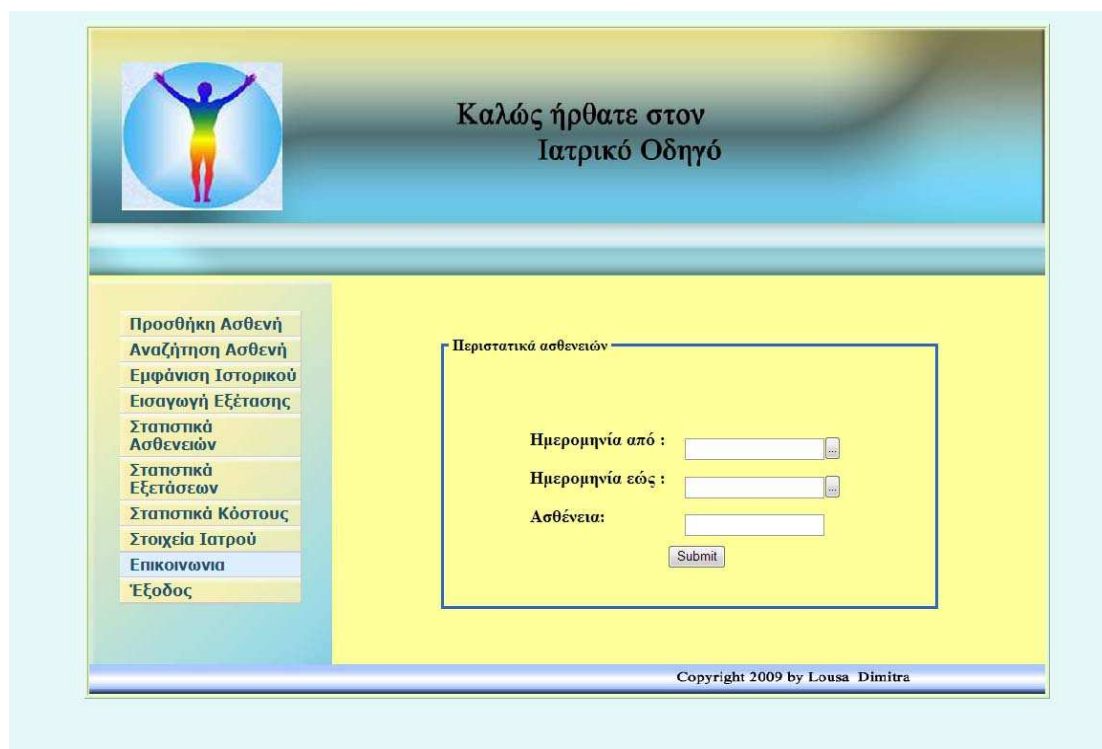


The screenshot shows a web application interface. At the top, there is a header with a logo of a human figure and the text "Καλώς ήρθατε στον Ιατρικό Οδηγό". Below the header is a navigation menu with the following items: Προσθήκη Ασθενή, Αναζήτηση Ασθενή, Εμφάνιση Ιστορικού, Εισαγωγή Εξέτασης, Στατιστικά Ασθενειών, Στατιστικά Εξετάσεων, Στατιστικά Κόστους, Στοιχεία Ιατρού, Επικοινωνία, Έξοδος. The main content area is titled "Προσθήκη Εξέτασης" and contains a form with the following fields: Κωδικός Εξέτασης, Ασθένεια, Τιμή Εξέτασης, Αντιδραστήριο, Ημ.Επίσκεψης, and Περιγραφή. There is an "Εισαγωγή" button at the bottom of the form. The footer of the page reads "Copyright 2009 by Lousa Dimitra".

Εικόνα 36 : Προσθήκη Εξέτασης

10.7 Στατιστικά Ασθενειών

Πολλές φορές στην Ιατρική για διάφορες μελέτες οι Ιατροί χρειάζονται κάποιες συγκεντρωτικές καταστάσεις . Στην παρακάτω σελίδα ο Ιατρός μπορεί να ορίσει κάποιο χρονικό διάστημα για την εξαγωγή αποτελεσμάτων μιας συγκεκριμένης ασθένειας .



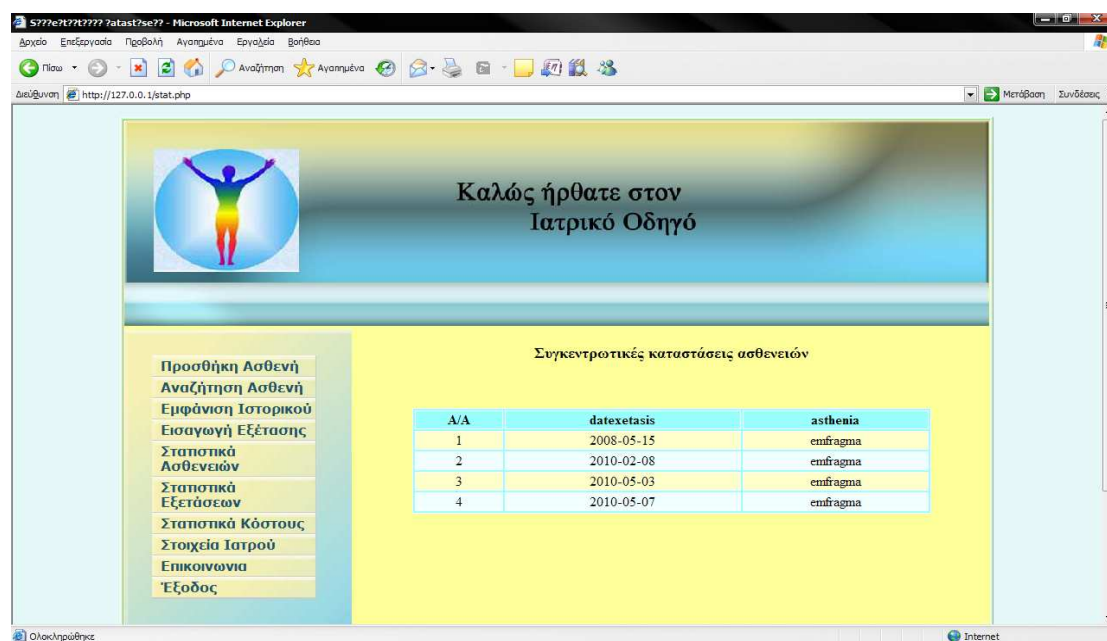
Εικόνα 37: Αναζήτηση περιστατικών ασθενειών

Η είσοδος της ημερομηνίας πραγματοποιείται από το εικονίδιο που βρίσκεται δεξιά τους, το οποίο με τη χρήση JavaScript επιτρέπει στο χρήστη να επιλέξει μια ημερομηνία από ένα ημερολόγιο. Το script που χρησιμοποιήθηκε είναι το jscalendar, το οποίο είναι ένα ανοικτού λογισμικό script. Το πλαίσιο διαλόγου για την επιλογή της ημερομηνίας φαίνεται στη παρακάτω εικόνα:



Εικόνα 38: Ημερολόγιο

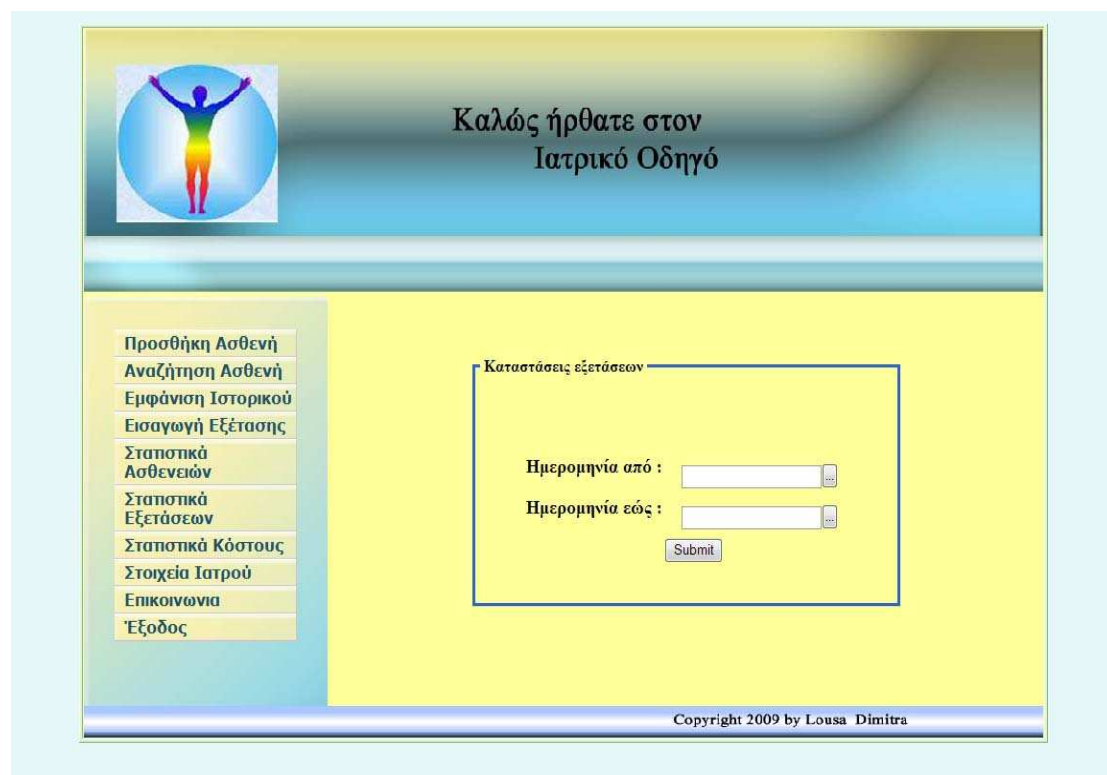
Αφού γίνει η εισαγωγή του χρονικού διαστήματος που επιθυμούμε να γίνει η αναζήτηση και αφού πληκτρολογήσουμε την ασθένεια που επιθυμούμε τα αποτελέσματα της αναζήτησης εμφανίζονται υπό μορφή πίνακα, όπως φαίνεται στην εικόνα που ακολουθεί:



Εικόνα 39 : Αποτέλεσμα αναζήτησης ασθενειών

10.8 Στατιστικά Εξετάσεων

Σε αυτή την σελίδα ο Ιατρός συμπληρώνοντας τα αντίστοιχα πεδία μπορεί να εμφανίσει συγκεντρωτικές καταστάσεις για το πόσα περιστατικά ασθενειών έλαβαν χώρα σε ένα συγκεκριμένο χρονικό διάστημα .



Εικόνα 40 : Αναζήτηση συνολικών περιστατικών

Τα αποτελέσματα εμφανίζονται υπό μορφή πίνακα όπως φαίνεται παρακάτω :

Καλώς ήρθατε στον Ιατρικό Οδηγό

Συγκεντρωτικές καταστάσεις ασθενειών

A/A	datexetasis	asthenia	timiexetasis
1	2008-05-15	emfragma	50€
2	2008-11-26	kardiako episod	75€
3	2009-02-02	asthna	100€
4	2009-10-23	broxitida	90€
5	2009-08-20	skolikoiditida	80€
6	2009-09-29	akmi	80€
7	2010-03-02	ipatotida	50€
8	2010-04-12	test pap	60€
9	2010-04-05	ipatiida	35€
10	2010-03-17	ponokefalos	30€
11	2010-01-19	kolitida	100€
12	2010-03-30	erithra	55€
13	2010-04-09	imikrania	70€
14	2010-04-03	thiroidis	65€
15	2010-05-04	skolikoiditida	80€

Copyright 2009 by Lousa Dimitra

Εικόνα 41 : Πίνακας συνολικών ασθενειών

10.9 Στατιστικά κόστους Εξετάσεων

Με αυτή την επιλογή ο χρήστης μπορεί να εξαγει οικονομικά αποτελέσματα για το συνολικό κόστος όλων των εξετάσεων που πραγματοποιήθηκαν σε ένα συγκεκριμένο διάστημα.

Καλώς ήρθατε στον Ιατρικό Οδηγό

Καταστάσεις συνολικού κόστους

Ημερομηνία από :

Ημερομηνία έως :

Submit

« May 2010 »

Mo	Tu	We	Th	Fr	Sa	Su
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

Today

Copyright 2009 by Lousa Dimitra

Εικόνα 42 : Αναζήτηση καταστάσεων συνολικού κόστους Εξετάσεων

Το αποτέλεσμα της αναζήτησης φαίνεται στην εικόνα που ακολουθεί και μας δίνει για κάθε περιστατικό ασθένειας το συνολικό κόστος από τις εξετάσεις που πραγματοποιήθηκαν στο διάστημα που δώσαμε.

Καλώς ήρθατε στον
Ιατρικό Οδηγό

Καταστάσεις κόστους εξετάσεων

A/A	asthenia	timiexetasis
1	akmi	80
2	asthma	100
3	broxitida	90
4	emfragma	600
5	erithra	55
6	imikrania	70
7	ipatiida	35
8	ipatotida	50
9	kardiako episod	75
10	kolitida	100
11	ponokefalos	100
12	skolikoiditida	160
13	test pap	60
14	thiroidis	65

Copyright 2009 by Lousa Dimitra

Εικόνα 43 : Πίνακας συνολικού κόστους

10.10 Στοιχεία Γιατρού

Με αυτή την επιλογή ο Γιατρός μπορεί να προβεί σε επεξεργασία των προσωπικών του στοιχείων που είναι καταχωρημένα στη Βάση Δεδομένων.

Στην παρακάτω εικόνα εμφανίζεται η φόρμα με τα στοιχεία που είναι ήδη καταχωρημένα στη Βάση Δεδομένων μας. Τα πεδία είναι ενεργά να τα επεξεργαστεί ο Γιατρός και πατώντας το κουμπί υποβολή να ενημερωθεί με τα καινούργια στοιχεία η Βάση. Αφού γίνει η τροποποίηση εμφανίζεται μήνυμα με τα νέα στοιχεία που μας ενημερώνει ότι πραγματοποιήθηκε με επιτυχία η ενέργειά μας.

Καλώς ήρθατε στον
Ιατρικό Οδηγό

Επεξεργασία στοιχείων γιατρού Δημητρίου Μάριος

Επίθετο :	Δημητρίου
Όνομα :	Μάριος
Αρ.Ταυτότητας:	4321
Κωδ.Κλινικής :	15
Κωδ.Εξέτασης:	1367
Ειδικότητα:	Καρδιολόγος
Διεύθυνση:	Νίκης 89
Τηλέφωνο :	69774737
Όνομα χρήστη :	dim
Κωδικός Πρόσβασης :	1981

Υποβολή

Copyright 2009 by Lousa Dimitra

Εικόνα 44: Επεξεργασία στοιχείων Γιατρού

10.11 Επικοινωνία

Στη σελίδα αυτή ο χρήστης μπορεί να επικοινωνήσει με το Διαχειριστή της Βάσης σε περίπτωση που παρουσιαστεί κάποιο πρόβλημα στέλνοντάς του ένα μήνυμα έτσι ώστε να προβεί στις ανάλογες ενέργειες για την σωστή λειτουργία τόσο της Βάσης όσο και της εφαρμογής.

Ο Διαχειριστής της Εφαρμογής είναι αυτός που διαχειρίζεται και την σωστή λειτουργία της Βάσης και δίνει αρχικά τους κωδικούς πρόσβασης στους εξουσιοδοτημένους γιατρούς.

Η φόρμα επικοινωνίας φαίνεται στην εικόνα που ακολουθεί.

Καλώς ήρθατε στον
Ιατρικό Οδηγό

Επικοινωνία με Administrator

Επικοινωνία

Αποστολή Μηνύματος

Όνομα:

* Διεύθυνση Email:

* Μήνυμα:

*Απαραίτητα πεδία

Αποστολή

Copyright 2009 by Lousa Dimitra

Εικόνα 45: Φόρμα επικοινωνίας

Και ακολουθεί το μήνυμα επιβεβαίωσης της αποστολής.

Καλώς ήρθατε στον
Ιατρικό Οδηγό

Ευχαριστούμε Δημητρίου Μάριος .

Το παρακάτω μήνυμα στάλθηκε:

Παρακαλώ επικοινωνήστε μαζί μου.Υπάρχει πρόβλημα στην προσθήκη εξέτασης

Copyright 2009 by Lousa Dimitra

Εικόνα 46: Μήνυμα επιβεβαίωσης

Παράρτημα Α

SQL κώδικας για την δημιουργία των πινάκων της βάσης .

Δομή Πίνακα για τον Πίνακα `doctor`

```
CREATE TABLE `doctor` (  
  `Lname` varchar(20) NOT NULL default "",  
  `Fname` varchar(20) NOT NULL default "",  
  `doctorid` varchar(10) NOT NULL default '0',  
  `idkliniki` varchar(7) NOT NULL default "",  
  `idexetasis` varchar(7) NOT NULL default "",  
  `idikotita` varchar(20) NOT NULL default "",  
  `phone` int(10) NOT NULL default '0',  
  `address` varchar(20) NOT NULL default "",  
  `username` varchar(20) NOT NULL default "",  
  `password` varchar(20) NOT NULL default "",  
  PRIMARY KEY (`doctorid`),  
  FULLTEXT KEY `Lname` (`Lname`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `ergastiriakosdoctor`

```
CREATE TABLE `ergastiriakosdoctor` (  
  `doctorid` varchar(8) NOT NULL default "",  
  `eidos` varchar(15) NOT NULL default "",  
  PRIMARY KEY (`doctorid`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `exetasi`

```
CREATE TABLE `exetasi` (  
  `idexetasis` varchar(10) NOT NULL default "",  
  `asthenia` varchar(20) NOT NULL default "",  
  `timiexetasis` varchar(10) NOT NULL default "",  
  `antidrastirio` varchar(10) NOT NULL default "",  
  `datexetasis` date NOT NULL default '0000-00-00',  
  `perigrafi` text NOT NULL,  
  PRIMARY KEY (`idexetasis`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `istoriko`

```
CREATE TABLE `istoriko` (  
  `patientid` varchar(10) NOT NULL default "",  
  `asthenia` varchar(15) NOT NULL default "",
```

```
`datediagnvsis` date NOT NULL default '0000-00-00',  
PRIMARY KEY (`patientid`,`asthenia`,`datediagnvsis`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `kliniki`

```
CREATE TABLE `kliniki` (  
  `idkliniki` varchar(20) NOT NULL default "",  
  `idmanager` varchar(7) NOT NULL default "",  
  `nbed` int(11) NOT NULL default '0',  
  PRIMARY KEY (`idkliniki`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `klinikosdoctor`

```
CREATE TABLE `klinikosdoctor` (  
  `doctorid` varchar(7) NOT NULL default "",  
  `years` int(5) NOT NULL default '0',  
  PRIMARY KEY (`doctorid`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `nosilia`

```
CREATE TABLE `nosilia` (  
  `patientid` varchar(7) NOT NULL default "",  
  `idkliniki` varchar(7) NOT NULL default "",  
  `startdate` date NOT NULL default '0000-00-00',  
  `enddate` date NOT NULL default '0000-00-00',  
  PRIMARY KEY (`patientid`,`idkliniki`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `patient`

```
CREATE TABLE `patient` (  
  `lname` varchar(20) NOT NULL default "",  
  `fname` varchar(20) NOT NULL default "",  
  `patronimo` varchar(30) NOT NULL default "",  
  `id` varchar(7) NOT NULL default "",  
  `address` varchar(40) NOT NULL default "",  
  `bdate` date NOT NULL default '0000-00-00',  
  `sex` varchar(10) NOT NULL default "",  
  `insurance` varchar(10) NOT NULL default "",  
  `history` text NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `pragmatopoiisiexet`

```
CREATE TABLE `pragmatopoiisiexet` (  
  `patientid` varchar(7) NOT NULL default "",  
  `idexetasis` varchar(7) NOT NULL default "",  
  `apotelesma` varchar(20) NOT NULL default "",  
  PRIMARY KEY (`patientid`,`idexetasis`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Δομή Πίνακα για τον Πίνακα `usertable`

```
CREATE TABLE `usertable` (  
  `id` varchar(6) NOT NULL default "",  
  `username` varchar(10) NOT NULL default "",  
  `password` varchar(11) NOT NULL default '0'  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Παράρτημα Β

Κώδικας login.html

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Login form</title>
<link rel="stylesheet" type="text/css" href="css/mystyles.css">
<SCRIPT language="javascript">
    function validate()
    {
        if      ((document.forms[0].username.value=="")      ||
(document.forms[0].password.value==""))
        {
            alert("Παρακαλώ συμπληρώστε το Όνομα Χρήστη και τον
Κωδικό Πρόσβασης");
            return false;
        }
        return true;
    }
</script>
<style type="text/css">
<!--
body {

    background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="550" border="1" bgcolor="#CCFFCC"
align="center">
    <tr>
        <td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">
            <p class="style1"></p>
            <p class="style1">&nbsp;</p>
            <p class="style1">
                <p>
                    </p>
                </p>
            <pre><h3> Καλώς ήρθατε στην ηλεκτρονική υπηρεσία διαχείρισης
ιατρικών δεδομένων .</h3></pre>
            <pre><h3> Για να εισέλθετε στην εφαρμογή θα πρέπει πρώτα να
δώσετε το όνομα χρήστη
                <p> και τον κωδικό πρόσβασης .</h3></pre>
            <p>&nbsp;</p>
            <p>&nbsp;</p>
            </div>
            <form action="login4.php" method="POST" onsubmit="return
validate();" > <center>
            <table width="300" cellspacing="2" border="2" bgcolor="#DADADA"
bordercolor="#0099FF" align="center">
            <tr>
                <td bgcolor="#949494"><font face="verdana" size="2">
                    <b>Είσοδος χρήση :</b>&nbsp;</font>
            </td></tr><tr> <td>
                <div align = center>

```

```

        <font face="verdana" size="2">
            <b> Όνομα Χρήστη:</b>&nbsp;&nbsp;&nbsp;</br></font>
        <input type="text" name="username"></br>
        <div align = center>
            <font face="verdana" size="2">
                <b>Κωδικός Πρόσβασης:</b>&nbsp;&nbsp;&nbsp;</br></font></div>
            <input type="password" name="password"></br>
            <br><br><div align=center>
                <input type="submit" value="Είσοδος" align =
center><br></div>
            </font>
        </td>
</tr> </table> </center> </form>
        
</table> </body> </html>

```

Κώδικας login.php

```

<?php
session_start( );
include("config.inc.php");
$con=mysql_connect($dbServer,$dbUser,$dbPass) or exit("Cannot connect
to database!");
$username=$_POST['username'];
$password=$_POST['password'];
if ($_POST['username'] == admin && $_POST['password']==admin)
{
    echo          '<META                HTTP-EQUIV="Refresh"
CONTENT="0;URL=http://127.0.0.1/doctor.html">';
}
$_SESSION['username'] =$username;
    $sql="SELECT * FROM doctor WHERE username='$username' AND
password='$password'";
    $res=mysql_db_query($dbDatabase , $sql);
if (($res!=null) && (mysql_num_rows($res)>0))
    {
        $_SESSION['username'] = $_POST['username'];
        $_SESSION['password'] = $_POST['password'];
        echo          '<META                HTTP-EQUIV="Refresh"
CONTENT="0;URL=http://127.0.0.1/second.html">';
        echo " kalos hr8es  $_SESSION[username]";
    }
else
{
    echo          '<META                HTTP-EQUIV="Refresh"
CONTENT="0;URL=http://127.0.0.1/login.html">';
}
?>

```

Κώδικας in.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Εισαγωγή Ασθενή</title>
<link rel="stylesheet" type="text/css" href="css_3.css">
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;

```

```
width: 70%;}
legend{color:black;
font-weight:bold;}

body {

    background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
    <tr>
        <td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">
            
        </div>
        <div id="leftnav">
            <div id="pageNav">
            <div id="sectionLinks">
            <a href="pros.html">Προσθήκη Ασθενή</a>
            <a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
            <a href="istoriko.html">Εμφάνιση Ιστορικού</a>
            <a href="exetasi.html">Εισαγωγή Εξέτασης</a>
            <a href="statistic.html">Στατιστικά Ασθενειών</a>
            <a href="statistic23.html">Στατιστικά Εξετάσεων</a>
            <a href="statistic3.html">Στατιστικά Κόστους</a>
            <a href="doct.php">Στοιχεία Ιατρού</a>
            <a href="send.html">Επικοινωνία</a>
            <a href="login.html">Εξοδος</a>
            </div></div></div>
            <div align = center>
            <p align="center" class="style3"><b>Η Εισαγωγή πραγματοποιήθηκε με
επιτυχία</b>
            <form action="in.php" method="post">
            <fieldset><legend > Στοιχεία Ασθενή</legend>
            <table border="0" width="70%" align="center">
            <table width="450" border="0" cellpadding="0" cellspacing="0" >
            <tr>
                <td><p align =right ><b>
Επίθετο :</b>
                </td>
                <td>
                <input type="text" name="pedio1" maxlength="10">
                </td>
            </tr><br>
            <tr>
                <td>
                <p align = right><b>Όνομα:</b>
                </td>
                <td>
                <input type="text" name="pedio2" maxlength="15">
                </td>
            </tr>
            <tr>
                <td><p align=right><b>
Όνομα πατρός :</b>
                </td>
                <td>
            </tr>
            <tr>
                <td><p align=right><b>
```

```

                <input type="text" name="pedio3" maxlength="20">
            </td>
        </tr><br>
    <tr>
        <td><p align =right><b>Αρ.ταυτότητας :</b>
    </td>
    <td>
        <input type="text" name="pedio4" maxlength="10">
    </td>
    </tr>
    <tr>
        <td><p align=right><b>Διεύθυνση :</b>
    </td>
    <td>
        <input type="text" name="pedio5" maxlength="30">
    </td>
    </tr>
    <tr>
        <td><p align =right><b>Ημ.γέννησης :</b>
    </td>
    <td>
    <input type="text" name="pedio6" maxlength="10">
    </td>
    </tr>
    <tr>
    <td>
    <p align =right><b> Φύλο :</b>
    </td>
    <td>
        <INPUT TYPE="radio" NAME="pedio7" VALUE="Male" CHECKED>
        Άνδρας
        <INPUT TYPE="radio" NAME="pedio7" VALUE="Female"> Γυναίκα<BR>
    </td>
    </tr>
    <tr>
        <td><p align=right><b>Ασφαλιστικός Οργανισμός :</b>
    </td>
    <td>
        <select name="pedio8" size="1">
            <option>Δημόσιο</option>
            <option>ΙΚΑ</option>
            <option>ΤΣΑ</option>
            <option>ΤΕΒΕ</option>
            <option>ΤΑΠ-ΟΤΕ</option>
            <option>ΔΕΗ</option>
            <option>ΤΣΜΕΔΕ</option>
            <option>Ιδιωτικός τομέας</option>
        </select>
    </td>
    </tr><td>
    <label for="history"><p align =right><b> Ιστορικό:</label></td><td>
    <textarea name="pedio9" id="history" rows="4" cols="20">
    </textarea><br /></td></tr>
    </tr>
    <tr> <td colspan="2" align="center">
    <input type="submit" name="jghg" value="Εισαγωγή" >
    </td></tr>
</form>
</table></fieldset>
<p>&nbsp;</p>

```



```

      </div></td>
    </tr>
  </table>
<p></p>
  <?
$metavliti1=$_POST['pedio1'];
$metavliti2=$_POST['pedio2'];
$metavliti3=$_POST['pedio3'];
$metavliti4=$_POST['pedio4'];
$metavliti5=$_POST['pedio5'];
$metavliti6=$_POST['pedio6'];
$metavliti7=$_POST['pedio7'];
$metavliti8=$_POST['pedio8'];
$metavliti9=$_POST['pedio9'];
$query="insert                                     into
patient (lname, fname, patronimo, id, address, bdate, sex, insurance, history)
values ('$metavliti1', '$metavliti2', '$metavliti3', '$metavliti4', '$meta
vlti5', '$metavliti6', '$metavliti7', '$metavliti8', '$metavliti9')";
$conn=mysql_connect("localhost", "root", "");
mysql_select_db("medico", $conn);
$result=mysql_query($query, $conn) or die(mysql_error());
?>
</table>
</body>
</html>
```

Κώδικας post.php

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Στοιχεία Ασθενή</title>
<link rel="stylesheet" type="text/css" href="css_3.css">
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 60%;
}
legend{color:black;
font-weight:bold;}
body {

        background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
  <tr>
    <td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">
      
    </div>
<div id="leftnav">
  <div id="pageNav">
```

```

<div id="sectionLinks">
<a href="pros.html">Προσθήκη Ασθενή</a>
<a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
<a href="istoriko.html">Εμφάνιση Ιστορικού</a>
<a href="exetasi.html">Εισαγωγή Εξέτασης</a>
<a href="statistic.html">Στατιστικά Ασθενειών</a>
<a href="statistic23.html">Στατιστικά Εξετάσεων</a>
<a href="statistic3.html">Στατιστικά Κόστους</a>
<a href="doct.php">Στοιχεία Ιατρού</a>
<a href="send.html">Επικοινωνία</a>
<a href="login.html">Εξοδος</a>
</div></div></div>
<p>&nbsp;</p>
<center>
<?
$metavlitil10=$_POST['pedio10'];
$query="select fname, lname , patronimo , id, bdate, sex, insurance, address
from patient where id ='$metavlitil10'";
$conn=mysql_connect("localhost", "root", "");
mysql_select_db("medico", $conn);
$result=mysql_query($query, $conn) or die(mysql_error());
$number=mysql_num_rows($result);
if ($number==0)
{echo"Δεν βρέθηκε εγγραφή με αυτό το Αρ. Ταυτότητας";}
else{
echo "<form action=\"del.php\" method=\"POST\" > <center>";
echo "<table border=\"2\"
width=\"50%\" bgcolor=\"#DBEFFB\" bordercolor=\"#0099FF\"
align=\"center\">";
while($newarray=mysql_fetch_array($result))
{
$fname=$newarray['fname'];
$lname=$newarray['lname'];
$patronimo=$newarray['patronimo'];
$bdate=$newarray['bdate'];
$id=$newarray['id'];
$sex=$newarray['sex'];
$insurance=$newarray['insurance'];
$address=$newarray['address'];
echo"<b><h2> <pre> Επεξεργασία στοιχείων ασθενή $fname
$lname</pre></h2></b>";
echo"<p>&nbsp;</p>";
echo "<tr>
<td>
<b> Επίθετο :</b>
</td>
<td>";
echo "<input type=\"text\" name=\"lname\"
value=\"$lname\" align=\"center\">";
echo "<tr>
<td>
<b> Όνομα :</b>
</td>
<td>";
echo "<input type=\"text\" name=\"fname\"
value=\"$fname\">";
echo "<tr>
<td>
<b> Πατρώνυμο:</b>

```

```

        </td>
    <td>";
        echo    "<input    type=\"text\"    name=\"patronimo\"
value='\$patronimo'>";

    echo "<tr>
        <td>
        <b>    Αρ.Ταυτότητας :</b>
        </td>
        <td>";
            echo    "<input    type=\"text\"    name=\"myid\"
value='\$id'>"; //"<textarea name='myid' label='myid' value='\$id'>";
        echo "<tr>
        <td>
        <b>    Ημ.Γέννησης:</b>
        </td>
        <td>";
            echo    "<input    type=\"text\"    name=\"bdate\"
value='\$bdate'>";

            echo "<tr>
        <td>
        <b>    Διεύθυνση:</b>
        </td>
        <td>";
            echo    "<input    type=\"text\"    name=\"address\"
value='\$address'>";

        echo "<tr>
        <td>
        <b>    Φύλο:</b>
        </td>
        <td>";
            echo    "<input    type=\"text\"    name=\"sex\"
value='\$sex'>";

        echo "<tr>
        <td>
        <b>    Ασφαλιστικός Οργανισμός :</b>
        </td>
        <td>";
            echo    "<input    type=\"text\"    name=\"insurance\"
value='\$insurance'>";
        }
        echo "</table>";
        echo"<p></p>";
        echo"<b> Επέλεξε ενέργεια:</b>";
            echo"<input
value='edit' >Επεξεργασία";
            echo"<input
value='del'>Διαγραφή";

        echo    "<input    type=\"submit\"    name=\"Submit1\"    value=\"Υποβολή\"
align = center><br></div>";
        echo "</form>";
    }
    ?>
    
        </td>
    </tr>

```

```
</table>
</body>
</html>
```

Κώδικας inxet.php

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Προσθήκη Εξέτασης</title>
<link rel="stylesheet" type="text/css" href="css_1.css">
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 70%;}
legend{color:black;
font-weight:bold;}

body {

    background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
<tr>
<td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">

</div>
<div id="leftnav">
<div id="pageNav">
<div id="sectionLinks">
<a href="pros.html">Προσθήκη Ασθενή</a>
<a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
<a href="istoriko.html">Εμφάνιση Ιστορικού</a>
<a href="exetasi.html">Εισαγωγή Εξέτασης</a>
<a href="statistic.html">Στατιστικά Ασθενειών</a>
<a href="statistic23.html">Στατιστικά Εξετάσεων</a>
<a href="statistic3.html">Στατιστικά Κόστους</a>
<a href="doct.php">Στοιχεία Ιατρού</a>
<a href="send.html">Επικοινωνία</a>
<a href="login.html">Εξοδος</a>
</div></div></div><p align="center" class="style3"><b>Η Εισαγωγή
πραγματοποιήθηκε με επιτυχία</b>
<form action="inxet.php" method="post">
<div align="center">
<fieldset><legend > Προσθήκη Εξέτασης </legend>
<table border="0" width="60%" align="center">
<table width="450" border="0" cellpadding="0" cellspacing="0" >
<tr>
<td><p align =right ><b>
Κωδικός Εξέτασης:</b>
```

```

        </td>
        <td>
        <input type="text" name="pedio1" maxlength="15">
        </td>
    </tr><br>

<tr>
    <td><p align=right><b>
    Ασθένεια:</b>
    </td>
    <td>
    <input type="text" name="pedio2" maxlength="20">
    </td>
</tr><br>

<tr>
    <td><p align =right><b>Τιμή Εξέτασης :</b>
</td>
<td>
    <input type="text" name="pedio3" maxlength="10">
    </td>
</tr>

<tr>
    <td><p align=right><b>Αντιδραστήριο :</b>
</td>
<td>
    <input type="text" name="pedio4" maxlength="30">
    </td>
</tr>

<tr>
    <td><p align =right><b>Ημ.Επίσκεψης:</b>
</td>
<td>    <p>
<input type="text" name="pedio5" maxlength="10"></p>
    </td>
</tr>

<tr><td>
<label for="history"><p align =right><b> Περιγραφή:</label></td><td>
<textarea name="pedio6" id="history" rows="4" cols="20">
</textarea><br /></td></tr>
<tr> <td colspan="2" align="center">
<input type="submit" name="jghg" value="Εισαγωγή" >
</td></tr>
</form></div>
    </table></fieldset>

<p></p>
<p></p>

    
    </div></td>
</tr></table>
<?
$metavliti1=$_POST['pedio1'];
$metavliti2=$_POST['pedio2'];
$metavliti3=$_POST['pedio3'];
$metavliti4=$_POST['pedio4'];
$metavliti5=$_POST['pedio5'];
$metavliti6=$_POST['pedio6'];
$query="insert
into
exetasi(idexetasis,asthenia,timixetasis,antidrastirio,datexetasis,pe

```

```

rigrafi)
values('$metavliti1','$metavliti2','$metavliti3','$metavliti4','$meta
vliti5','$metavliti6')");
$conn=mysql_connect("localhost","root","");
mysql_select_db("medico",$conn);
$result=mysql_query($query,$conn) or die(mysql_error());
?>
</td>
</tr></table>
</body> </html>

```

Κώδικας statistic.html

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Συγκριτικές καταστάσεις</title>
<link rel="stylesheet" type="text/css" href="css_1.css">
<script src="../src/js/jscal2.js"></script>
<script src="../src/js/lang/en.js"></script>
<link rel="stylesheet" type="text/css"
href="../src/css/jscal2.css" />
<link rel="stylesheet" type="text/css" href="../src/css/border-
radius.css" />
<link rel="stylesheet" type="text/css"
href="../src/css/gold/gold.css" />
</head>
<style type="text/css">
<!--
fieldset{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 70%;}
legend{color:black;
font-weight:bold;}
body {
background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
<tr>
<td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">

</div>
<div id="leftnav">
<div id="pageNav">
<div id="sectionLinks">
<a href="pros.html">Προσθήκη Ασθενή</a>
<a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
<a href="istoriko.html">Εμφάνιση Ιστορικού</a>
<a href="exetasi.html">Εισαγωγή Εξέτασης</a>
<a href="statistic.html">Στατιστικά Ασθενειών</a>
<a href="statistic23.html">Στατιστικά Εξετάσεων</a>
<a href="statistic3.html">Στατιστικά Κόστους</a>
<a href="doct.php">Στοιχεία Ιατρού</a>
<a href="send.html">Επικοινωνία</a>

```

```

<a href="login.html">Εξοδος</a>
</div></div></div>
<form action="stat.php" method="post">
<div align = center><p>&nbsp;</p>
<fieldset><legend > Περισσότερα ασθενειών</legend>
<table border="0" width="50%" align="center">
<table width="350" border="0" cellpadding="0" cellspacing="0" >
<tr>
<td>
<b> <h3> Ημερομηνία από : </h3>
</td>
<td>
<input type="text" name = "f_date1"
maxlength="15"><button id="f_btn1">...</button><br />
<script type="text/javascript">/*! [CDATA[
var cal = Calendar.setup({
onSelect: function(cal) { cal.hide() }
});
cal.manageFields("f_btn1", "f_date1", "%Y-%m-%d");
cal.manageFields("f_btn2", "f_date2", "%b %e, %Y");
cal.manageFields("f_btn3", "f_date3", "%e %B %Y");
cal.manageFields("f_btn4", "f_date4", "%A, %e %B, %Y");
//]]></script>
</body>
</html>
</td>
</tr><br>
<tr>
<td>
<b><h3>Ημερομηνία εώς :</h3>
</td>
<td>
<input type="text" name = "f_date2" maxlength="15"><button
id="f_btn2">...</button><br />
<script type="text/javascript">/*! [CDATA[
var cal = Calendar.setup({
onSelect: function(cal) { cal.hide() }
});
cal.manageFields("f_btn1", "f_date1", "%Y-%m-%d");
cal.manageFields("f_btn2", "f_date2", "%Y-%m-%d");
cal.manageFields("f_btn3", "f_date3", "%e %B %Y");
cal.manageFields("f_btn4", "f_date4", "%A, %e %B, %Y");
//]]></script>
</td>
</tr>
<tr>
<td><b><h3> Ασθένεια:</h3>
</td>
<td>
<input type="text" name="pedio3" maxlength="10">
</td>
</tr>
<tr> <td colspan="2" align="center">
<input type="submit" name="jghg" value="Υποβολή" >
</td></tr>
</form>
<p>&nbsp;</p>
<p>&nbsp;</p>
</table>
<p>&nbsp;</p></fieldset>


```

```

        </div></td>
    </tr></table>
</body> </html>

```

Κώδικας stat.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>S???e?t??t???? ?atast?se??</title>
<link rel="stylesheet" type="text/css" href="css_4.css">
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 60%;
}
legend{color:black;
font-weight:bold;}
body {
        background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
    <tr>
        <td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">
            
        </div>
        <div id="leftnav">
            <div id="pageNav">
            <div id="sectionLinks">
<a href="pros.html">Προσθήκη Ασθενή</a>
<a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
<a href="istoriko.html">Εμφάνιση Ιστορικού</a>
<a href="exetasi.html">Εισαγωγή Εξέτασης</a>
<a href="statistic.html">Στατιστικά Ασθενειών</a>
<a href="statistic23.html">Στατιστικά Εξετάσεων</a>
<a href="statistic3.html">Στατιστικά Κόστους</a>

<a href="doct.php">Στοιχεία Ιατρού</a>
<a href="send.html">Επικοινωνία</a>
<a href="login.html">Εξόδο</a>
</div></div></div>
<?
$a=1;
$b=0;
$day1=$_POST['f_date1'];
$day2=$_POST['f_date2'];
$metavliti3=$_POST['pedio3'];
$query="select * from exetasi where asthenia ='$metavliti3' and
(datexetasis between '$day1' and '$day2') ORDER BY datexetasis ASC";
$conn=mysql_connect("localhost","root","");
mysql_select_db("medico",$conn);
$result=mysql_query($query,$conn) or die(mysql_error());

```



```
$number=mysql_num_rows($result);
echo"<b><h3 align=\"center\">Συγκεντρωτικές καταστάσεις
ασθενειών</b></h3>";
echo"<p>&nbsp;</p>";
echo"<table border=\"1\"
width=\"60%\" bgcolor=\"#99FFFF\" bordercolor=\"#FFFFFF\"
align=\"center\">";
echo "<th align=\"center\">A/A</th><th align=\"center\"> datexetasis
</th><th align=\"center\"> asthenia</th>";
while($newarray=mysql_fetch_array($result))
{
$datexetasis=$newarray['datexetasis'];
$asthenia=$newarray['asthenia'];
$b=$b+$a;
if($b% 2 == 0){
echo "<tr align=\"center\" ><td style='background-
color:#F2FFFF'>".$b."</td> <td style ='background-color:#F2FFFF'>".
$datexetasis."</td> <td style='background-color:F2FFFF'>".
$asthenia."</td> ";
echo "</tr>";
}
else
{
echo "<tr align=\"center\" ><td style='background-
color:#FFFFCC'>".$b."</td> <td style ='background-color:#FFFFCC'>".
$datexetasis."</td> <td style='background-color:#FFFFCC'>".
$asthenia."</td> ";
echo "</tr>";
}}
echo "</table>";
//echo"<b> <pre><b> synolo $b peristatika </b></pre></b>";
//echo"</td></tr>";
//echo"</td>";
?>

</table>
</body>
</html>
```

Κώδικας statistic2.html

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Συγκεντρωτικές καταστάσεις</title>
<link rel="stylesheet" type="text/css" href="css_1.css">
<style type="text/css">
body, table {
font-family:verdana;
font-size:10px;
}
td.date input {
width:100px;
}
td.calendar {
padding-left:2px;
vertical-align:middle;
}
td.calendar img {
width:16px;
```

```

        height:16px;
        border:none;
    }
    .calendar div.button-wrapper {
        background-image:url(images/calendar-gray.gif);
        font-size:1px;
    }
    .disabled .calendar a:link, .disabled .calendar a:visited {
        visibility:hidden;
    }
</style>
<script src="calendar.js" type="text/javascript"
charset="windows-1253"></script>
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 60%;
}
legend{color:black;
font-weight:bold;}
body {
    background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
    <tr>
        <td height="350" align="left" valign="top" bgcolor="#FFFFFF"><div
align="left">
            
            </div>
            <div id="leftnav">
            <div id="pageNav">
            <div id="sectionLinks">
            <a href="eisagogi.html">Προσθήκη Ασθενή</a>
            <a href="anazitisi2.html">Αναζήτηση Ασθενή</a>

            <a href="reviews.html">Εμφάνιση Ιστορικού</a>
            <a href="exetasi.html">Εισαγωγή Εξέτασης</a>
            <a href="statistic.html">Στατιστικά Ασθενειών</a>
            <a href="statistic2.html">Στατιστικά Εξετάσεων</a>
            <a href="doct.php">Στοιχεία Ιατρού</a>
            <a href="login.html">Εξοδος</a>
            </div></div></div>
            <form action="stat2.php" method="post">
            <div align = center><p>&nbsp;</p>
            <fieldset><legend > Καταστάσεις εξετάσεων</legend>
            <table border="0" width="50%" align="center">
            <table width="350" border="0" cellpadding="0" cellspacing="0" >
            <tr>
                <td>
                    <b> <h3> Ημερομηνία από : </h3>
                    </td>
                <td>

```

```

                <td class="date"><input name="date1" value=""
onchange="this.value=Calendar.fixDate(this.value)"></td>
                <td class="calendar"><div class="button-wrapper"><a
id="button1" href="javascript:void(0)" onclick="Calendar.open(this,
document.forms[0].date1, true);return false"></a></div></td>
        </tr><br>
<tr>
<td>
<b><h3>Ημερομηνία έως :</h3>
</td>
<td>
                <td class="date"><input name="date2" value=""
onchange="this.value=Calendar.fixDate(this.value)"></td>
                <td class="calendar"><div class="button-wrapper"><a
id="button2" href="javascript:void(0)"
onclick="Calendar.open(document.forms[0].date2,
document.forms[0].date2);return false"></a></div></td>
        </td>
</tr>
<tr> <td colspan="2" align="center">
<input type="submit" name="jghg" value="Submit" >
</td></tr>
</form>
<p>&nbsp;</p>
<p>&nbsp;</p>
</table>
<p>&nbsp;</p></fieldset><div id="footer">

</div></td>
</tr></table>
</body> </html>

```

Κώδικας stat2.php

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Συγκεντρωτικές καταστάσεις</title>
<link rel="stylesheet" type="text/css" href="css_4.css">
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 60%;
}
legend{color:black;
font-weight:bold;}
body {
background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">

```

```
|  |
| --- |
| <div align="left"> |

```

```

</table>
</body>
</html>
```

Κώδικας doctor.php

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>St???e?a G?at???
|  |
| --- |
|  |
| <? $_session['username']; $query="select ,doctorid,idkliniki,idexetasis,idikotita,address,phone, username,password from doctor where username=$t"; Fname,Lname //$query="select ,doctorid,idkliniki,idexetasis,idikotita,address,phone, username,password from doctor where $_SESSION['username'] = $_POST['username']"; Fname,Lname $conn=mysql_connect("localhost","root",""); mysql_select_db("medico",$conn); $result=mysql_query($query,$conn) or die(mysql_error()); $number=mysql_num_rows($result); echo"<table width="50%" bgcolor="#DADADA" bordercolor="#0099FF" align="center">"; border="3" //echo"<table bgcolor="#DADADA">"; | |

```

```

while ($newarray=mysql_fetch_array($result))
{
$Fname=$newarray['Fname'];
$Lname=$newarray['Lname'];
$doctorid=$newarray['doctorid'];
$idkliniki=$newarray['idkliniki'];
$idexetasis=$newarray['idexetasis'];
$idikotita=$newarray['idikotita'];
$address=$newarray['address'];
$phone=$newarray['phone'];
$username=$newarray['username'];
$password=$newarray['password'];
echo"<b> <pre><b>                epexergasia stoixeiw n giatrou    $Fname
$Lname</b></pre></b>";
echo"<p>&nbsp;</p>";
echo "<tr>
        <td>
            <b>  epi8eto :</b>
        </td>
        <td>";
echo            "<input            type=\"text\"            name=\"Lname\"
value='$Lname'>";

echo "<tr>
        <td>
            <b>  onoma :</b>
        </td>
        <td>";
echo            "<input            type=\"text\"            name=\"Fname\"
value='$Fname'>";

echo "<tr>
        <td>
            <b>    id:</b>
        </td>
        <td>";
echo            "<input            type=\"text\"            name=\"doctorid\"
value='$doctorid'>";

echo "<tr>
        <td>
            <b>    idkliniki :</b>
        </td>
        <td>";
echo            "<input            type=\"text\"            name=\"idkliniki\"
value='$idkliniki'>";    //"<textarea    name='myid'    label='myid'
value='$id'>";
echo "<tr>
        <td>
            <b>    idexetasis:</b>
        </td>
        <td>";
echo            "<input            type=\"text\"            name=\"idexetasis\"
value='$idexetasis'>";

        echo "<tr>
        <td>
            <b>    idikotita:</b>
        </td>
        <td>";

```

```
        echo      "<input      type=\"text\"      name=\"idikotita\"
value='\$idikotita'>";

    echo "<tr>
        <td>
            <b>      address:</b>
        </td>
        <td>";
        echo      "<input      type=\"text\"      name=\"address\"
value='\$address'>";

    echo "<tr>
        <td>
            <b>      phone :</b>
        </td>
        <td>";
        echo      "<input      type=\"text\"      name=\"phone\"
value='\$phone'>";
    echo "<tr>
        <td>
            <b>      username :</b>
        </td>
        <td>";
        echo      "<input      type=\"text\"      name=\"username\"
value='\$username'>";
    echo "<tr>
        <td>
            <b>      password :</b>
        </td>
        <td>";
        echo      "<input      type=\"text\"      name=\"password\"
value='\$password'>";

    }
    echo "</table>";
    echo      "<input      type=\"submit\"      value=\"Update\"      align =
center><br></div>";
    echo "</form>";
    //}
    ?>
    
        </td>
    </tr>
</table>
</body>
</html>
```

Κώδικας send.html

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<link rel="stylesheet" type="text/css" href="css_3.css">
<title>Επικοινωνία</title>
<style type="text/css">
label,input, textarea, select {      display: block;      width:      180px;
float: left;      margin-bottom: 10px;}
label { text-align: right;      width: 95px; padding-right: 20px; font-
weight:bold; font-size:93%;}
br {clear: left;}
```

```

#subbut {width:100px; margin-top:10px; border: solid grey;
position:relative; left:125px; font-size:80%;}
.req {color:#a00;}
#note {width:180px; color:#a00; margin-left:115px; font-size:83%;}
fieldset table{margin-left:1em;
}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 65%;}
legend{color:black;
font-weight:bold;}
</style>
</head>
<body>
<table width="1000" height="673" border="1" bgcolor="#CCFFCC"
align="center">
<tr>
<td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">

<div id="leftnav">
<div id="pageNav">
<div id="sectionLinks">
<a href="pros.html">Προσθήκη Ασθενή</a>
<a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
<a href="istoriko.html">Εμφάνιση Ιστορικού</a>
<a href="exetasi.html">Εισαγωγή Εξέτασης</a>
<a href="statistic.html">Στατιστικά Ασθενειών</a>
<a href="statistic23.html">Στατιστικά Εξετάσεων</a>
<a href="statistic3.html">Στατιστικά Κόστους</a>
<a href="doct.php">Στοιχεία Ιατρού</a>
<a href="send.html">Επικοινωνία</a>
<a href="login.html">Εξοδος</a>
</div></div></div>
<div align = center>
<p><b><h2>Επικοινωνία με Administrator</h2></b></p>
<form method="post" action="process.php">
<fieldset id="contact">
<legend>Επικοινωνία</legend>
<h3>Αποστολή Μηνύματος</h3>
<div align="center">
<label for="name">Όνομα:</label><input type="text" name="name"
id="name"><br />
<label for="email"><span class="req">*</span>Διεύθυνση Email:</label>
<input type="text" name="email" id="email"><br />
<label for="message"><span class="req">*</span> Μήνυμα:</label>
<textarea name="message" id="message" rows="5" cols="35">
</textarea><br />
</div>
<br/ >
<div id="note">*Απαραίτητα πεδία</div>
<input id="subbut" type="submit" value="Αποστολή">
</form>
<p>&nbsp;</p>
</fieldset>
</form>
<p>&nbsp;</p>

</div></td>
</tr>

```



```
</table> </body> </html>
```

κώδικας process.php

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-
1253">
<title>Επικοινωνία</title>
<link rel="stylesheet" type="text/css" href="css_3.css">
</head>
<style type="text/css">
<!--
fieldset table{margin-left:1em;}
fieldset{margin-left:5.3em;
border:3px solid #3366CC;
width: 70%;}
legend{color:black;
font-weight:bold;}
body {
        background-color: #E4F8F8;
}
.style1 {color: #FFFFFF}
-->
</style></head>
<body>
<table width="1000" height="623" border="1" bgcolor="#CCFFCC"
align="center">
  <tr>
    <td height="350" align="left" valign="top" bgcolor="#FFFF99"><div
align="left">
      
    </div>
    <div id="leftnav">
      <div id="pageNav">
        <div id="sectionLinks">
          <a href="pros.html">Προσθήκη Ασθενή</a>
          <a href="anazitisi2.html">Αναζήτηση Ασθενή</a>
          <a href="istoriko.html">Εμφάνιση Ιστορικού</a>
          <a href="exetasi.html">Εισαγωγή Εξέτασης</a>
          <a href="statistic.html">Στατιστικά Ασθενειών</a>

          <a href="statistic23.html">Στατιστικά Εξετάσεων</a>
          <a href="statistic3.html">Στατιστικά Κόστους</a>
          <a href="doct.php">Στοιχεία Ιατρού</a>
          <a href="send.html">Επικοινωνία</a>
          <a href="login.html">Εξοδος</a>
        </div></div></div>
        <?php
$name = $_POST["name"];
$email = $_POST["email"];
$message = $_POST["message"];
$message= stripslashes($message);
if (empty($message) && empty($email)){
    print " <br>ΕΙΣΑΓΕΤΕ email και μήνυμα";
}
elseif (empty($message)){
    print "<br>Παρακαλώ εισάγετε μήνυμα.<br />";
}
elseif (empty($email)){
    print "<br>Παρακαλώ εισάγετε email. <br />";
}
}
```

```
else {
if (!empty($name)) {
    echo "<b><pre>                                Ευχαριστούμε $name.</b><br />";
}
print "<p><b><pre>                                Το παρακάτω μήνυμα στάλθηκε
</b>:<pre> <br>                                $message<br /></p>";
}
?>
</body>
</html>
```

Κώδικας config.php

```
<?php
$servername='localhost';      // Your MySQL Server Name or IP address
here
$dbusername='root';           // Login user id here
$dbpassword='';               // Login password here
$dbname='medico';             // Your database name here
connecttodb($servername,$dbname,$dbusername,$dbpassword);
function connecttodb($servername,$dbname,$dbuser,$dbpassword)
{
s
global $link;
$link=mysql_connect ("$servername","$dbuser","$dbpassword");
if(!$link){die("Could not connect to MySQL");}
mysql_select_db("$dbname",$link) or die ("could not open
db".mysql_error());
}
?>
```

Κώδικας css_1.css

```
/* CSS Document */
#leftnav
{
float: left;
width: 250px;
height:390px;

padding: 2em;
background-image:url(box8.jpg);
}
#pageNav{
float:left;
width:190px;
padding: 0px;
border-right: 1px solid #cccccc;
border-bottom: 1px solid #cccccc;
font: small Verdana,sans-serif;
}
#sectionLinks{
margin: 0px;
padding: 0px;
background-image:url(box8.jpg);
}

#sectionLinks h3{
padding: 10px 0px 2px 10px;
border-bottom: 1px solid #cccccc;
}
```

```
#sectionLinks a:link, #sectionLinks a:visited {
    display: block;
    border-top: 1px solid #ffffff;
    border-bottom: 1px solid #cccccc;
    background-image: url(box8.jpg);
    font-weight: bold;
    padding: 3px 0px 3px 10px;
    color: #21536A;
    text-decoration: none;
}

#sectionLinks a:hover{
    border-top: 1px solid #cccccc;
    background-color: #DDEEFF;
    background-image: none;
    font-weight: bold;
    text-decoration: none;
}
```

Βιβλιογραφία

1. **Welling,Luke** . *Ανάπτυξη Web Εφαρμογών με PHP και MySQL*. ΑΘΗΝΑ : ΚΛΕΙΔΑΡΙΘΜΟΣ, 2008.
2. **Julie C. Meloni**. *Μάθετε PHP, MySQL και Apache – Όλα σε ένα* . Εκδόσεις Μ. Γκιούρδας,2004.
3. **Musciano, Chuck** . *HTML and XHTML*. 2002
4. **Αποστολάκης Ιωάννης** .*Πληροφοριακά Συστήματα Υγείας* .ΑΘΗΝΑ: Παπαζήσης ,2002.
5. **Μπότσης ,Ταξιάρχης** . *Πληροφορική Υγείας* .ΑΘΗΝΑ: Δίαυλος ,2005

WEB SITES

<http://www.php.net>

<http://www.mysql.com/>

<http://el.Wikipedia.org/wiki/PHP>

<http://www.w3schools.com/PHP/Default.as>

<http://dide.flo.sch.gr/php/>

<http://en.wikipedia.org/wiki/PhpMyAdmin>