

Ατομική Διπλωματική Εργασία

ΗΛΕΚΤΡΟΝΙΚΟΣ ΦΑΚΕΛΟΣ ΑΣΘΕΝΗ

Γεώργιος Νικολάου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ιούνιος 2008

Ηλεκτρονικός Φάκελος Ασθενή

Όνομα: Νικολάου Γεώργιος

Επιβλέπων Καθηγητής: Κώστας Παττίχης

Ευχαριστίες

Όταν φτάνεις επιτέλους στο τέλος ενός μεγάλου στόχου και κοιτάς πίσω μόνο τότε συνειδητοποιείς την συνολική σου πορεία και τους ανθρώπους που την επηρέασαν. Σε αυτούς θέλω να αναφερθώ σε αυτές τις λίγες γραμμές και να τους ευχαριστήσω θερμά για την σημαντική βοήθεια που μου πρόσφεραν στα διάφορα στάδια αυτής της πορείας.

Αρχικά θέλω να ευχαριστήσω τον επιβλέποντα καθηγητή μου Κώστα Παττίχη του Πανεπιστημίου Κύπρου γιατί στάθηκε η αφορμή να ασχοληθώ με τον Ηλεκτρονικό Ιατρικό Φάκελο και μου επέτρεψε να εκπονήσω αυτή την ατομική διπλωματική εργασία. Ξεχωριστά οφείλω να ευχαριστήσω τον Δρ. Αντώνη Ιωσήφ που ως ειδικός βοήθησε στην κατανόηση των εννοιών που χρειάστηκαν για την εκπόνηση της διπλωματικής αυτής εργασίας. Στο ίδιο ύφος οφείλω να ευχαριστήσω τον Γεώργιο Τσούλουπα μεταπτυχιακό φοιτητή Πληροφορικής για την δική του βοήθεια όσο αφορά προβλήματα που τυχόν συνάντησα. Ένα ακόμα ξεχωριστό ευχαριστώ στα ακόλουθα άτομα που βοήθησαν στην διόρθωση του κειμένου. Στην αδερφή μου την Ελένη Νικόλαου απόφοιτος της Αγγλικής Φιλολογίας του Πανεπιστημίου Κύπρου και τον φίλο και συμφοιτητή Νικόλα Εξαρχάκο τεταρτοετή φοιτητή της Πληροφορικής στο Πανεπιστήμιο Κύπρου

Τέλος ευχαριστώ περισσότερο από όλους την οικογένεια μου για την συμπαράσταση τους, την κατανόηση τους και την υπομονή τους σε όλη την διάρκεια αυτής της εργασίας και ιδιαίτερα στις δύσκολες στιγμές της. Είμαι ιδιαίτερα ευγνώμων για όσα μου έχουν προσφέρει όλα αυτά τα χρόνια και σίγουρα σε αυτούς χρωστάω τα πάντα. Θα ήθελα να τους ευχαριστήσω για όλες τις θυσίες που έχουν κάνει ώστε να μου δώσουν την δυνατότητα να ασχοληθώ απερίσπαστος με κάτι τόσο συναρπαστικό όσο η αναζήτηση της γνώσης.

Νικολάου Γεώργιος Ιούνιος 2008

Περίληψη

Στην διπλωματική αυτή σκοπός μας είναι η δημιουργία ενός συστατικού για ενσωμάτωση με το ανοικτό λογισμικό FreeMed. Στόχος του συστατικού είναι η διαχείριση της κωδικοποίησης ασθενειών με βάση το International Disease Coding που είναι ο διεθνής τρόπος κωδικοποίησης ασθενειών. Στο πρώτο κεφάλαιο αναφέρονται γενικά εισαγωγικά στοιχεία που αφορούν τον ηλεκτρονικό ιατρικό φάκελο. Στην συνέχεια στο κεφάλαιο δυο μελετούνται εκτενώς εννέα συστήματα ανοικτού λογισμικού που αφορούν τον ηλεκτρονικό ιατρικό φάκελο. Στο τρίτο κεφάλαιο με χρήση περιήγησης τονίζουμε τις δυνατότητες που παρέχει το FreeMed. Στο τέταρτο κεφάλαιο αναφέρουμε ιστορικά στοιχεία που αφορούν την εξέλιξη της διεθνής κωδικοποίησης ασθενειών. Ακολουθεί το πέμπτο κεφάλαιο όπου αναφέρεται η προεργασία που έγινε πριν την υλοποίηση αλλά και η ίδια η υλοποίηση. Κλείνοντας αναφέρουμε μερικά συμπεράσματα στο τελευταίο κεφάλαιο.

Περιεχόμενα

Κεφάλαιο 1 Εισαγωγή	7
1.2 Ο Ιατρικός Φάκελος Ασθενή Σήμερα	8
1.3 Ο Ηλεκτρονικός Ιατρικός Φάκελος Ασθενούς (ΗΙΦΑ).....	10
1.4 Η είσοδος δεδομένων στον ΗΙΦΑ	12
1.4.1 Επεξεργασία φυσικής γλώσσας(NLP)	12
1.4.2 Δομημένη είσοδος δεδομένων	14
1.4.3 Κωδικοποίηση και standardization	14
1.5 Διαφορές μεταξύ του κλασσικού ΙΦΑ και του ΗΙΦΑ	15
Κεφάλαιο 2 Μελέτη υπαρχόντων συστημάτων ιατρικού φακέλου ανοικτού κώδικα	16
2.1 Σύστημα OSCAR.....	16
2.2 Σύστημα CIASI.....	17
2.3 Σύστημα CHIRIS (Canada Health Infoway Reference Implementation Suite)	31
2.3 Σύστημα CHITS.....	34
2.4 Σύστημα Cottage Med.....	37
2.5 Σύστημα ePHR.....	43
2.6 Σύστημα GNUMED.....	46
2.7 Σύστημα Hospitals	48
2.8 Σύστημα ndivoHealth.....	65
2.9 Σύστημα openEHR.....	73
2.10 Σύστημα FreeMed.....	74
Κεφάλαιο 3 Ηλεκτρονικός Ιατρικός Φάκελος Ασθενή FreeMed.....	76
3.1 Τι είναι το FreeMed;	76
3.2 Το FreeMed και το πρωτόκολλο HL7.....	77
3.3 Περιήγηση στο FreeMed.....	78
Κεφάλαιο 4 International Disease Coding (IDC)	83
4.1 Αρχικά ιστορικά στοιχεία	83
4.2 Υιοθέτηση του International List of Causes of Death.....	85
4.3 Η Πέμπτη Αναθεώρηση	88

4.4 Προηγούμενες ταξινομήσεις των ασθενειών για τις στατιστικές νοσηρότητας.....	90
4.5 Ενιαία Επιτροπή για κοινές αιτίες θανάτου	92
4.6 Έκτη αναθεώρηση των διεθνών καταλόγων	93
4.7 Οι έβδομη και όγδοη αναθεώρηση.....	95
4.8 Η ένατη αναθεώρηση	96
4.9 Προετοιμασίες για τη δέκατη αναθεώρηση	97
Κεφάλαιο 5 Υλοποίηση.....	99
5.1 Θεωρία γράφων.....	99
5.1.1 Ορισμοί	100
5.1.2 Αναπαράσταση γράφων με πίνακες γειτνίασης (adjacency matrix)	101
5.1.3 Αναπαράσταση γράφων με λίστες γειτνίασης (adjacency lists)	101
5.1.4 Τοπολογική ταξινόμηση.....	102
5.1.5 Αλγόριθμοι; Τυπολογικής αναζήτησης.....	103
5.1.6 Διάχυση γράφων	104
5.2 Θεωρία δέντρων	106
5.2.1 Ορισμοί	107
5.2.2 Ύψος και βάθος δέντρου.....	107
5.2.3 Αναπαράσταση δέντρων στην μνήμη	108
5.2.4 Δυαδικά δέντρα.....	109
5.3 Πρώτη προσπάθεια με IDC.....	112
5.4 Δεύτερη προσπάθεια με IDC	118
5.5 Τρίτη προσπάθεια με IDC.....	122
5.6 Τετάρτη προσπάθεια με IDC	123
5.7 Ολοκλήρωση του IDC	124
Κεφάλαιο 6 Συμπεράσματα.....	128
Βιβλιογραφία	129
Παράρτημα Α – Αρχικός κώδικας Access.....	132
Παράρτημα Β – Τελικός κώδικας Access	146
Παράρτημα C – Αρχικός κώδικας Java.....	158
Παράρτημα Δ – Τελικός κώδικας Java.....	212

Κεφάλαιο 1 Εισαγωγή

Ο ιατρικός φάκελος ασθενή (ΙΦΑ) μας δίνει μία εικόνα της υγείας του ασθενή αφού αυτός έχει ζητήσει θεραπεία σε κάποια βαθμίδα περίθαλψης. Συνήθως τα δεδομένα μπαίνουν στον ΙΦΑ από τον ιατρό και από τις νοσοκόμες. Ο ΙΦΑ συνήθως περιέχει ευρήματα, αποτελέσματα εξετάσεων, και θεραπευτικές πληροφορίες. Ο ΙΦΑ ουσιαστικά άρχισε από την εποχή του Ιπποκράτη, τον 5ο αιώνα π.Χ. Ο Ιπποκράτης πιο συγκεκριμένα θεωρούσε ότι ο ΙΦΑ πρέπει να εξυπηρετεί δύο σκοπούς:

- Να αντανakλά επακριβώς την πορεία της ασθeneίας
- Να δίνει πληροφορίες σχετικά με τα αίτια της ασθeneίας

Αρχικά ο Ιπποκράτης καταχωρούσε τις παρατηρήσεις του με ένα τρόπο που στηριζόταν στην χρονολογική σειρά της παρατήρησης των συμπτωμάτων. Αυτό είναι και το πρώτο είδος του ΙΦΑ που είναι ο βασισμένος στον χρόνο ΙΦΑ. Ήταν δε, πολύ σημαντικό να χρησιμοποιούνται αυτές οι παρατηρήσεις για πρόγνωση και θεραπεία, κάτι που λίγο-πολύ είναι ο κύριος στόχος και στους σημερινούς ΙΦΑ.

Μέχρι τις αρχές του 19ου αιώνας οι ιατροί βάσιζαν τις παρατηρήσεις τους σε οτιδήποτε μπορούσαν να ακούσουν, να αισθανθούν και να δουν με μεγάλη βοήθεια από το στηθοσκόπιο που ανακαλύφθηκε το 1816 από τον Laennec. Με την χρήση της σύγχρονης τεχνολογίας ο ιατρός πια έχει πληθώρα δεδομένων, τα οποία δίνουν ναι μεν μεγαλύτερη πληροφορία, αλλά είναι πιο δύσκολο να τα επεξεργαστεί και να τα κατηγοριοποιήσει.

Μετά το 1880 ο Αμερικανός χειρουργός Mayo χρησιμοποίησε στο Mayo Clinic τον πρώτο πιο συστηματοποιημένο ιατρικό φάκελο. Στο σύστημα αυτό ο κάθε ιατρός κρατούσε σημειώσεις για τους ασθενείς που επισκεπτόταν σε έναν δερμάτινο φάκελο, και όλες οι παρατηρήσεις των ιατρών και των νοσοκόμων κρατούνταν σαν ένα σετ από χειρόγραφα που αφορούσαν την πορεία της υγείας του κάθε αρρώστου. Το πρόβλημα ήταν, ότι υπήρχε σοβαρή δυσκολία με την

συστηματοποίηση της επεξεργασίας των παρατηρήσεων του κάθε ιατρού, αφού ο κάθε ιατρός είχε τον δικό του τρόπο σκέψης και καταγραφής των παρατηρήσεων του. Έτσι λοιπόν το 1920 η κλινική του Mayo όρισε ένα ελάχιστο σετ από δεδομένα που ο κάθε ιατρός ήταν υποχρεωμένος να καταγράφει όταν εξέταζε τον ασθενή του. Αυτός ο ΙΦΑ ονομάζεται ασθενοκεντρικός ΙΦΑ, αφού στηρίζεται στις παρατηρήσεις που γίνονται από τους ιατρούς σχετικά με την κατάσταση ενός ασθενούς.

Μετά το 1960, για να αντιμετωπιστεί το πρόβλημα της ανομοιογένειας της πληροφορίας που υπήρχε στον ασθενοκεντρικό ΙΦΑ, ο Weed εισήγαγε τον προβληματοκεντρικό ΙΦΑ. Σε αυτόν τον ΙΦΑ στον κάθε άρρωστο χρεωνόταν κάποια προβλήματα. Σημειώσεις κρατούνταν για κάθε πρόβλημα βασισμένες στην δομή SOAP, που σημαίνει:

S = Subjective και είναι τα παράπονα του ασθενούς

O = Objective και είναι οι παρατηρήσεις του ιατρού

A = Assessment και είναι η αξιολόγηση όπως διάγνωση, ή αποτελέσματα τεστ

P = Plan και είναι η θεραπευτική αγωγή

Αυτό το είδος ΙΦΑ είχε σαν βασικό σκοπό να αντανakλά πιο καλά την λογική που ακολουθεί ο ιατρός όταν παρακολουθεί και θεραπεύει έναν ασθενή. Παρά το γεγονός όμως ότι αυτός ο ΙΦΑ έγινε αμέσως αποδεκτός, στην πράξη αποδείχθηκε ότι χρειαζόταν όλο το ιατρικό προσωπικό να ακολουθεί πολύ συγκεκριμένους κανόνες, ώστε να μπορεί αυτός ο ΙΦΑ να έχει καλά αποτελέσματα. Αποδείχθηκε ότι πολλές φορές δεδομένα που συνδέονται με ένα συγκεκριμένο πρόβλημα πρέπει να καταγράφονται περισσότερα από μία φορά.

1.2 Ο Ιατρικός Φάκελος Ασθενή Σήμερα

Οι πιο πολλοί ΙΦΑ σήμερα δεν έχουν σαν βάση τους τον χρόνο, διότι έτσι δεν είναι εύκολο να γίνει ανάλυση τάσεων (trend analysis). Έτσι λοιπόν τα περιεχόμενα ενός ΙΦΑ θα πρέπει να δομηθούν ανάλογα με την μέθοδο με την οποία αποκτήθηκαν, και τα δεδομένα αυτά θα πρέπει να καταχωρούνται σε κάθε τμήμα του ΙΦΑ με χρονολογική σειρά. Η μεθοδολογία SOAP επηρεάζει μόνο κλινικές σημειώσεις. Μια βασική ερώτηση, είναι με ποιον τρόπο και για

ποιούς σκοπούς χρησιμοποιείται σήμερα ο ΙΦΑ. Αυτοί οι σκοποί μπορούν να συνοψισθούν ως κάτωθι:

- Υποστήριξη της ιατρικής φροντίδας (αξιολόγηση και λήψη αποφάσεων, πληροφορία που μοιράζεται ανάμεσα σε λειτουργούς υγείας).
- Ο ΙΦΑ μπορεί να είναι μια νομικά αποδεκτή αναφορά ιατρικών πράξεων.
- Υποστήριξη έρευνας (κλινική έρευνα, επιδημιολογικές μελέτες, αξιολόγηση ποιότητας υγείας, αξιολόγηση φαρμάκων).
- Εκπαίδευση ιατρών.
- Βελτίωση υπηρεσιών υγείας σε διοικητικό και κοινωνικό επίπεδο.

Γενικότερα σήμερα ο ΙΦΑ θα έχει πολλά προβλήματα εάν συνεχίσει να ακολουθεί την δομή που είχε πριν από δεκαετίες. Υπάρχει πληθώρα δεδομένων και εξετάσεων τα οποία δύσκολα κρατούνται σε έναν ΙΦΑ και ακόμα πιο δύσκολα μπορεί να έχει πρόσβαση κάποιος ιατρός όταν έχει ανάγκη από αυτά τα δεδομένα. Η ιατρική γνώση με την βοήθεια της τεχνολογίας έχει εκτοξευθεί σήμερα και εάν ο ιατρός δεν έχει εύκολη και άμεση πρόσβαση σε ιατρική πληροφορία υπάρχει σοβαρό πρόβλημα περιθάλψης. Επίσης ο κλασσικός ΙΦΑ δεν βοηθά ιδιαίτερα την έρευνα και τον σχεδιασμό μιας πολιτικής για την υγεία του πληθυσμού. Γενικότερα, μπορούμε να πούμε ότι το χαρτί σαν μέσο αποθήκευσης έχει τα εξής προβλήματα:

- Ο ΙΦΑ μπορεί να είναι σε ένα μόνο μέρος την κάθε χρονική στιγμή. Μπορεί να μην το βρίσκει κάποιος όταν τον έχει ανάγκη, είτε γιατί χάθηκε, είτε γιατί τον χρησιμοποιεί ένας άλλος ιατρός.
- Τα περιεχόμενα του είναι ελεύθερο κείμενο, και άρα δεν είναι ομοιογενή, μπορεί να μην διαβάζονται, μπορεί να μην είναι πλήρη και να είναι διφορούμενα.

- Για επιστημονικές αναλύσεις τα περιεχόμενα πρέπει να μεταφερθούν σε κάποιο σύστημα ανάλυσης με αυξημένη την πιθανότητα λανθασμένης μεταφοράς.
- Ο κλασσικός ΙΦΑ δεν μπορεί να δώσει συμβουλές, υπενθυμήσεις ή προειδοποιήσεις.

Για τους παραπάνω λόγους λοιπόν είναι απαραίτητο να χρησιμοποιηθεί η τεχνολογία και να δημιουργηθεί ο ηλεκτρονικός ιατρικός φάκελος ασθενούς (ΗΙΦΑ).

1.3 Ο Ηλεκτρονικός Ιατρικός Φάκελος Ασθενούς (ΗΙΦΑ)

Η αλματώδης ανάπτυξη της τεχνολογίας και ειδικότερα της τεχνολογίας πληροφορικής προκάλεσε και το μεγάλο ενδιαφέρον για την ανάπτυξη του ΗΙΦΑ. Πιο συγκεκριμένα εδώ και 25 χρόνια υπάρχουν προσπάθειες για την ανάπτυξη του ΗΙΦΑ.

Στην αρχή τα πιο πολλά δεδομένα που έμπαιναν στους ΗΙΦΑ ήταν κείμενο, και όχι τόσο πολύ δομημένη είσοδος. Επιπλέον, οι ιατροί λόγω του ότι τα λειτουργικά περιβάλλοντα δεν είχαν τόσο αναπτυγμένη φιλικότητα στον χρήστη ήταν διστακτικοί στην χρήση τέτοιων συστημάτων. Επίσης, η εκπαίδευση των ιατρών στις τεχνολογίες πληροφορικής δεν ήταν διαδεδομένη. Τέλος, οι υπολογιστικές πλατφόρμες όπου αναπτύσσονταν οι ΗΙΦΑ ήταν ακριβές και συντηρούνταν με δυσκολία και υψηλό κόστος.

Παρ' όλα αυτά ένας μεγάλος αριθμός προχωρημένων ΗΙΦΑ αναπτύχθηκε, όπως τα COSTAR, TMR, RMIS, STOR και ELIAS. Τα συστήματα αυτά χρησιμοποιούσαν φόρμες, οι οποίες είχαν εκτυπωμένα κάποια δεδομένα (διοικητικά και άλλα εργαστηριακά αποτελέσματα) τις οποίες κατόπιν τις χρησιμοποιούσε το νοσηλευτικό προσωπικό για την είσοδο επιπλέον δεδομένων είτε με την μορφή αριθμητικών τιμών είτε με την μορφή ελεύθερου κειμένου. Σήμερα ο ΗΙΦΑ χρησιμοποιείται σε δύο επίπεδα παροχής υπηρεσιών υγείας:

- Την πρωτοβάθμια φροντίδα
- Την δευτεροβάθμια ή πιο ειδικευμένη φροντίδα

Αναφορικά με την πρωτοβάθμια φροντίδα, οι γενικοί ιατροί χρησιμοποιούν το ΗΙΦΑ πολύ παραπάνω από τους ειδικούς ιατρούς. Πιο συγκεκριμένα, σήμερα στην Ολλανδία το 90% των γενικών ιατρών χρησιμοποιεί ΗΙΦΑ για διάφορους σκοπούς όπως διοικητικούς, ιατρικούς και οικονομικούς. Από ιατρικής πλευράς ο ΗΙΦΑ στην πρωτοβάθμια φροντίδα είναι προβληματοκεντρικός. Χρησιμοποιείται standard ορολογία, όπως το ICD9-CM για κωδικοποίηση διαγνώσεων.

Όσον αφορά τους ειδικούς ιατρούς θα περίμενε κανείς ότι θα χρησιμοποιούσαν πιο πολύ τον ΗΙΦΑ από ότι οι γενικοί ιατροί. Ο λόγος είναι, ότι ενώ ο γενικός ιατρός έχει μία πιο μακροχρόνια σχέση με τον ασθενή του και άρα χρειάζεται απλά σαν ένα βοήθημα τον ΗΙΦΑ, ενώ ο ειδικός ιατρός αντιμετωπίζει περιστασιακά δεδομένα, και άρα λοιπόν διαφαίνεται να έχει μεγαλύτερη ανάγκη τον ΗΙΦΑ. Αυτό όμως στην πράξη είναι αναληθές για τους παρακάτω λόγους:

- Οι γενικοί ιατροί, δουλεύουν είτε μόνοι τους, είτε ανήκουν σε μικρές ομάδες ιατρών με αποτέλεσμα να ελέγχουν πολύ παραπάνω το διοικητικό μέρος της δουλειάς τους. Αντίθετα, οι ειδικοί ιατροί δουλεύουν σε περίπλοκα περιβάλλοντα τα οποία θα πρέπει να μετρήσουν την επιρροή του ΗΙΦΑ στην όλη δομή της διοίκησης, της μελλοντικής ανάπτυξης και των λογιστικών μεγεθών.
- Παρά το γεγονός, ότι οι γενικοί ιατροί αντιμετωπίζουν ένα ευρύ φάσμα παθολογικών καταστάσεων, οι σημειώσεις τους δεν είναι τόσο εκτεταμένες όσο αυτές ενός ειδικού ιατρού. Άρα ο ειδικός ιατρός θα αφιερώσει περισσότερο χρόνο στην χρήση του ΗΙΦΑ παρά ο ειδικός ιατρός.
- Τέλος οι διάφορες ειδικότητες έχουν διαφορετικές απαιτήσεις από έναν ΗΙΦΑ. Έτσι λοιπόν είναι εξαιρετικά δύσκολο να βρει κανείς έναν ΗΙΦΑ που να ικανοποιεί όλες τις ειδικότητες. Αντίθετα οι γενικοί ιατροί έχουν μια πιο συγκεκριμένη πρακτική και έτσι λοιπόν ένας ΗΙΦΑ μπορεί να εξυπηρετήσει τους πιο πολλούς από αυτούς.

1.4 Η είσοδος δεδομένων στον ΗΙΦΑ

Τα υπάρχοντα συστήματα ΗΙΦΑ υποστηρίζουν χρονοκεντρικές και προβληματοκεντρικές προσεγγίσεις στον τομέα παροχής υπηρεσιών υγείας. Ο πιο σημαντικός παράγοντας επιτυχίας στην εξαγωγή σωστής διάγνωσης και μακροχρόνιας αντιμετώπισης των προβλημάτων που έχει ένας ασθενής είναι η πιστότητα και πληρότητα των δεδομένων που υπάρχουν στον ΗΙΦΑ. Γίνεται έτσι φανερό, ότι ο τρόπος που εισάγονται τα δεδομένα στον ΗΙΦΑ καθορίζει εν πολλοίς και την επιτυχία του ΗΙΦΑ. Για τον λόγο αυτόν, έχουν γίνει συστηματικές προσπάθειες τα τελευταία χρόνια για την ανάπτυξη διεθνώς αποδεκτών κωδικοποιήσεων ασθενειών και ιατρικών όρων, ώστε να υπάρχει μία καλή βάση αναφορικά με την δομημένη είσοδο δεδομένων στον ΗΙΦΑ και γι' αυτό έχει γίνει πολύ έρευνα για την επεξεργασία φυσικής γλώσσας. Γενικότερα υπάρχουν δύο βασικοί τρόποι για την συλλογή δομημένων δεδομένων στον ΗΙΦΑ.

- Επεξεργασία δεδομένων με τρόπους επεξεργασίας φυσικής γλώσσας (natural language processing).
- Δομημένη εισαγωγή στοιχείων .

1.4.1 Επεξεργασία φυσικής γλώσσας(NLP)

Η επεξεργασία φυσικής γλώσσας μπορεί να εφαρμοστεί σε ελεύθερο κείμενο, και αυτό είναι το μεγάλο της πλεονέκτημα. Το ελεύθερο κείμενο μπορεί να αποκτηθεί με διάφορους τρόπους όπως εισαγωγή κειμένου μέσω κάποιου editor, ή μέσω συστημάτων αναγνώρισης φωνής, όπου οι ιατροί και οι νοσοκόμες έχουν πλήρη ελευθερία έκφρασης. Κατόπιν, ηλεκτρονικά λεξικά ιατρικών όρων χρησιμοποιούνται για την προεπεξεργασία του φυσικού κειμένου για περαιτέρω επεξεργασία που γίνεται συνήθως από το σύστημα στήριξης απόφασης. Πιο συγκεκριμένα, τα συστήματα επεξεργασίας φυσικής γλώσσας παράγουν συνήθως μία λίστα από όρους που χρησιμοποιούνται. Τέτοιες λίστες χρησιμοποιούνται για να εντοπίζονται τα μέρη των κειμένων όπου αυτοί οι όροι εμφανίζονται. Είναι όμως φανερό ότι η ύπαρξη ενός ιατρικού όρου όπως 'βήχας' σε κάποια πρόταση, δε σημαίνει πάντα ότι έχει σχέση με την διάγνωση, και άρα γίνεται αντιληπτό ότι κάτι τέτοιο μπορεί να προκαλέσει προβλήματα.

Για το λόγο αυτό, εκτός από την εννοιολογική γνώση που μας παρέχεται από αυτές τις κωδικοποιημένες λίστες ιατρικών όρων χρειαζόμαστε και την συντακτική γνώση που μας παρέχεται από την πιο ενδελεχή ανάλυση των προτάσεων που βρήκαμε να περιέχουν τους ιατρικούς όρους που μας ενδιαφέρουν.

Πιο συγκεκριμένα παραδείγματα εννοιολογικής γνώσης είναι τα παρακάτω:

- Το στομάχι είναι όργανο.
- Ο βήχας είναι ένα παράπονο του ασθενούς.
- Το AIDS είναι μία ασθένεια.
- Η δύσπνοια είναι συνώνυμο με δυσκολία στην αναπνοή.
- Ο πόνος μπορεί να περιγραφεί από το που συμβαίνει, πόσο δυνατός είναι, από την εξέλιξη του κλπ.

Αυτού του είδους η εννοιολογική γνώση συνδέεται με την συντακτική γνώση που υπάρχει στο ελεύθερο κείμενο και με την χρήση του λογισμικού επεξεργασίας φυσικής γλώσσας μπορεί να μας δώσει πολύτιμες πληροφορίες, για την παραγωγή κάποιας ιατρικής απόφασης.

Για παράδειγμα, έστω και αν οι λέξεις βήχας και πτύσμα χωρίζονται από αρκετές λέξεις σε μία πρόταση δεν υπάρχει αμφιβολία ότι ανήκουν μαζί αφού το πτύσμα είναι αποτέλεσμα βήχα. Από την άλλη μεριά όμως, υπάρχουν πολλές διαφορούμενες περιπτώσεις που προκύπτουν από την επεξεργασία κειμένου με μεθόδους επεξεργασίας φυσικής γλώσσας. Ας πάρουμε σαν παράδειγμα την πρόταση 'Μέγιστη περιπατητή απόσταση χωρίς πόνο είναι 200 μ'. Οι μέθοδοι NLP θα συνδέσουν κυκλοφοριακό πρόβλημα στα πόδια με την μέγιστη απόσταση 200 μ. Εδώ πρέπει να πούμε ότι το κυκλοφοριακό πρόβλημα στα πόδια, δεν είναι το μόνο που θα μπορούσε να οριοθετήσει την μέγιστη περιπατητή απόσταση στα 200 μ, αλλά θα μπορούσε ένας πόνος που υπάρχει κάπου αλλού να προκαλέσει πόνο μετά την διάνυση των 200 μ. Γενικότερα τέτοια προβλήματα που υπάρχουν στην επεξεργασία NLP, προκαλούν συνήθως και την ανθρώπινη επέμβαση για την σωστή ιατρική εξήγηση των φαινομένων.

Γενικότερα η NLP επεξεργασία δεν περιορίζει τον κλινικό ιατρό στην έκφραση των ευρημάτων του, παρόλα αυτά έχει και ένα μειονέκτημα. Δεν μπορεί να υποχρεώσει τον ιατρό να είναι πιο πλήρης και πιο συγκεκριμένος στις εκφράσεις του. Εδώ ίσως η γνώση από τους ιατρούς κωδικοποιημένων όρων και εκφράσεων και η χρήση δομημένης εισόδου δεδομένων να μπορέσει μαζί με την NLP επεξεργασία να δώσει καλύτερα αποτελέσματα στην ανάλυση των δεδομένων που υπάρχουν στον ΗΙΦΑ.

1.4.2 Δομημένη είσοδος δεδομένων

Η άλλη τακτική για την είσοδο δεδομένων είναι η δομημένη είσοδος δεδομένων. Αυτού του είδους η είσοδος δεδομένων δίνει πολύ αξιόπιστα δεδομένα στον ΗΙΦΑ, αλλά διαφέρει σημαντικά από τον κλασσικό τρόπο εισόδου δεδομένων στον ΙΦΑ, ο οποίος στηρίζεται πιο πολύ σε ελεύθερο κείμενο. Σε αυτή την περίπτωση όλη η κωδικοποιημένη πληροφορία πρέπει να είναι διαθέσιμη στον χρήστη όταν εισάγει δεδομένα στον ΗΙΦΑ. Τέτοιου είδους ΗΙΦΑ είναι ιδιαίτερα χρήσιμοι με την έννοια ότι υπάρχει μία κοινή βάση σε όποιον χρησιμοποιεί τον ΗΙΦΑ.

Γενικότερα, ένας ΗΙΦΑ πρέπει να χρησιμοποιεί όσο το δυνατόν δυναμικές και προσαρμοζόμενες φόρμες προς τις ανάγκες και το επίπεδο εκπαίδευσης του ιατρικού προσωπικού που χρησιμοποιεί τον ΗΙΦΑ. Το σημερινό παραθυρικό περιβάλλον που υπάρχει στους Η/Υ και στο διαδίκτυο, βοηθά αποφασιστικά στην αποδοχή του ΗΙΦΑ από το νοσηλευτικό προσωπικό.

1.4.3 Κωδικοποίηση και standardization

Μεγάλα ερευνητικά και κλινικά προγράμματα τα οποία υλοποιούνται σε παγκόσμιο επίπεδο και εμπλέκουν πολλά ιδρύματα (νοσοκομεία και πανεπιστήμια) δεν έχουν ανάγκη μονό από δομημένες και συγκεκριμένες μορφές δεδομένων, αλλά χρειάζονται και τρόποι ανάλλαγης δεδομένων σε εννοιολογικό επίπεδο. Συστήματα κωδικοποίησης όπως το International Classification of Diseases (ICD), το Systematized Nomenclature of Diseases (SNOMED) και άλλα παρόμοια συστήματα όπως το HL7 έχουν σαν σκοπό την αρμονία και την δυνατότητα να

ανταλλάσσονται δεδομένα σε ομογενή περιβάλλοντα τόσο σε γλώσσα όσο και σε πρακτικές. Γενικότερα ο ΗΙΦΑ προς το παρόν δίνει την δυνατότητα στον ιατρό να παράγει ένα τελικό report όπου κάνει διαγνωστικές παρατηρήσεις, αλλά δεν χρησιμοποιεί τα δεδομένα στα οποία στηρίζεται όπως εικόνες, σήματα και άλλα. Έτσι λοιπόν τελευταία αναπτύσσονται ΗΙΦΑ βασισμένοι σε πολυμέσα, που λέγονται πολυμεσικοί ΗΙΦΑ.

1.5 Διαφορές μεταξύ του κλασσικού ΙΦΑ και του ΗΙΦΑ

Πλεονεκτήματα ΙΦΑ:

- Μπορούν να περιφερθούν εύκολα στους ιατρικούς χώρους.
- Δίνουν ελευθερία στην έκφραση.
- Εύκολα ανασκοπεί κανείς τα δεδομένα.
- Δεν χρειάζεται ειδική εκπαίδευση του νοσηλευτικού προσωπικού για την χρήση του.
- Δεν διακόπτεται η λειτουργία του όπως μπορεί να γίνει με τον Η/Υ.

Από την άλλη μεριά ο ΗΙΦΑ έχει τα παρακάτω πλεονεκτήματα:

- Μπορούν να χρησιμοποιηθούν ταυτόχρονα από πολλούς χρήστες.
- Δεν υπάρχει πρόβλημα αναγνωσιμότητας.
- Υπάρχουν διάφορες όψεις των δεδομένων.
- Υποστήριξη δομημένης εισόδου δεδομένων.
- Στήριξη ιατρικής απόφασης.
- Δυνατότητα περίπλοκης ανάλυσης δεδομένων.

Κεφάλαιο 2 Μελέτη υπαρχόντων συστημάτων ιατρικού φακέλου

ανοικτού κώδικα

Σε αυτό το κεφάλαιο σκοπός μας είναι να κάνουμε μια εμπεριστατωμένη μελέτη των συστημάτων ηλεκτρονικού ιατρικού φακέλου που υπάρχουν και διέπονται από τις αρχές του ανοικτού κώδικα[1]. Υπάρχουν πολλές παραλλαγές της άδειας χρήσης ανοικτού κώδικα (GPL). Η πηγή που χρησιμοποιείται αναφέρει την τρίτη έκδοση. Πιο κάτω μελετούμε εννιά συστήματα ηλεκτρονικού ιατρικού φακέλου. Στο κεφάλαιο αυτό δεν θα γίνει εκτενείς αναφορά του συστήματος που επιλέχτηκε σε αυτή την ατομική διπλωματική εργασία με σκοπό η αναφορά και η εκτενείς ανάλυση του να γίνει στο επόμενο κεφάλαιο.

2.1 Σύστημα OSCAR

Το Σύστημα OSCAR[2] είναι μια συμπερίληψη προγράμματος λογισμικού που αποτελείται από τρία σημαντικά συστατικά:

1. Oscar-McMaster (ή απλά γνωστό και ως OSCAR) είναι ένας ηλεκτρονικός ιατρικός φάκελος που δημιουργήθηκε αρχικά για τους παθολόγους. Εντούτοις, τώρα επίσης χρησιμοποιείται από πολλούς ειδικούς, και ιατρικές υπηρεσίες μέσα στις κοινοτικές κλινικές και τις οργανώσεις που βοηθούν τους αστέγους. Το φθινόπωρο του 2005, το Oscar-McMaster πιστοποιήθηκε από OntarioMD στο πλαίσιο του προγράμματος TP των παθολόγων. Υπάρχουν τώρα γνωστοί χρήστες και προμηθευτές του OSCAR στο Οντάριο, , και άλλα μέρη του κόσμου.[2,3]
2. Το OSCAR-Resource είναι ένα ελεύθερο κλινικό σύστημα διαχείρισης περιεχομένου. Είναι ενσωματωμένο μέσα στο OSCAR έτσι ώστε να μπορεί να υπάρχει πρόσβαση στους χρήσιμους κλινικούς πόρους ενώ παρακολουθούμε τους ασθενείς. Είναι επίσης ένα ανοικτό σύστημα έτσι ώστε η συμβολή μας να μπορεί να μοιραστεί με άλλους χρήστες του OSCAR.[3].

3. Το OSCAR-Citizens αλλιώς MyOSCAR είναι ένα προσωπικά ελεγχόμενο αρχείο υγείας. Αποτελεί ένα πραγματικό αρχείο υγείας που ελέγχεται από τους ασθενείς. Αυτό είναι πολύ διαφορετικό από την άποψη ότι είναι ένα προσωπικό αρχείο υγείας που δίνει την ελευθερία στους ασθενείς να δουν μια μερίδα των αρχείων του ιατρού που τους παρακολουθεί. Το MyOSCAR παρέχει επίσης ένα ασφαλές σύστημα μηνυμάτων για τους ασθενείς ώστε να μπορούν να επικοινωνήσουν με τους ιατρούς που τους παρακολουθούν.

Όλες αυτές οι δυνατότητες είναι βασισμένες σε εφαρμογές διαδικτύου εύκολα προσβάσεως όπου υπάρχει ένας υπολογιστής και το διαδίκτυο. Φέρνουν τους ασθενείς και τους ιατρούς που τους παρακολουθούν ακόμα και τα εργαλεία υποστήριξης πληροφοριών και λήψης αποφάσεων υγείας πιο κοντά μεταξύ τους. Το όραμά για αυτό το πρόγραμμα είναι ότι πρόκειται να αναπτύξει εφαρμογές λογισμικού προηγμένης τεχνολογίας για τις ελεύθερες κλινικές βασισμένες στα στοιχεία και τους πόρους προς χρήση από τους ιατρικούς επαγγελματίες και τους πολίτες όλου του κόσμου που θα βελτιώσει τελικά την υγεία της παγκόσμιας κοινότητάς.

2.2 Σύστημα CIASI

Το σύστημα CIASI είναι ένα σύστημα ηλεκτρονικού ιατρικού φάκελου που βασίζεται σε ένα σύστημα με όνομα Oscar που αναφέρθηκε και πιο πάνω αλλά προσφέρει πολύ περισσότερες λειτουργίες σε σχέση με το προαναφερόμενο σύστημα. Σκοπός του συστήματος είναι να προσφέρει μια ολοκληρωμένη λύση όσο αφορά την παροχή ιατρικών υπηρεσιών για τους αστέγους. [2,4]

Το Toronto Client Access to Integrated Services and Information (CAISI) Project σκοπό έχει να μειώσει το πρόβλημα παροχής υπηρεσιών σε αστέγους με την ενσωμάτωση ενός συστήματος μεταξύ υπηρεσιών χρησιμοποιώντας ένα ηλεκτρονικό πληροφοριακό σύστημα. Το έργο περιλαμβάνει τα εξής:

- 1) Την ανάπτυξη ενός ανοικτού λογισμικού [5]

2). Την δημιουργία μιας κοινότητας από άτομα και υπηρεσίες που να χρησιμοποιούν το σύστημα με σκοπό την δημιουργία συστήματος συνεργασίας για την φροντίδας άστεγων μεταξύ των ατόμων και υπηρεσιών αυτών. [5]

Στο ατομικό επίπεδο το έργο περιλαμβάνει την άμεση αξιολόγηση ατόμων / πελατών, την καθοδήγηση των ατόμων / πελατών στα κατάλληλα καταφύγια ή υπηρεσίες, την διαχείριση λιστών αναμονής, την υποστήριξη για διαχείριση πολλαπλών περιπτώσεων μεταξύ υπηρεσιών που τελικά οδηγεί στην αναφορά των ατόμων / πελατών στην κατάλληλη υπηρεσία.[5]

Σε επίπεδο πληθυσμού το έργο ενδυναμώνει την ικανότητα της κοινότητας να συλλέγει δεδομένα που μπορούν να χρησιμοποιηθούν από άτομα στα κέντρα λήψης αποφάσεων έτσι ώστε να βοηθήσουν την κοινότητα να αλλάξει συμπεριφορά και να μειώσει το πρόβλημα φροντίδας άστεγων. Πιο συγκεκριμένα προσφέροντας μια πιο οικονομική λύση στέγασης με την κατάλληλη υποδομή π.χ. ένα πρόγραμμα χρόνιας φροντίδας για άτομα που η κοινωνία δεν μπορεί ή δεν θέλει να φροντίσει.[5]

Το σύστημα είναι πελατοκεντρικό. Οι πελάτες δίνουν την δυνατότητα σε κάποιες υπηρεσίες να βελτιωθούν οι ίδιες οι υπηρεσίες υγείας που οι πελάτες ή τα άτομα λαμβάνουν. Οι ίδιοι οι πελάτες βοηθούν στην βελτίωση της λειτουργίας του συστήματος.[5]

Συμπερασματικά ο στόχος του συστήματος είναι να εστιάσει την προσοχή του στα άτομα που είναι άστεγα για να βελτιώσουν τελικά τη μετακίνησή των ατόμων αυτών από τις οδούς και τα καταφύγια σε μια κατοικία. Αυτή η μετακίνηση μπορεί να διευκολυνθεί μέσω:

1. Της άμεσης εισαγωγής και της κατάλληλης παραπομπής - αυτό σημαίνει για τον άστεγο γρήγορη αντιμετώπιση προβλημάτων υγείας και αντιμετώπιση των κοινωνικών προβλημάτων και την τοποθέτηση του στα κοινοτικά προγράμματα και τα καταφύγια που είναι καλύτερα εξοπλισμένα για να βοηθήσουν με εκείνα τα προβλήματα. Αυτό είναι πολύ καλύτερο, σε αντίθεση με την αναμονή, αυτά τα προβλήματα να φθάσουν σε ένα σημείο που η αντιμετώπιση πλέον γίνεται επάγουσα ανάγκη.[5]

2. Της διαχείρισης περίπτωσης πολλαπλών υπηρεσιών - αυτό σημαίνει ότι δίνεται στους άστεγους πελάτες η ευκαιρία να τύχουν προσοχής από πολλαπλές υπηρεσίες που λειτουργούν με ένα ενιαίο διοικητικό σχέδιο περίπτωσης. Αυτό μπορεί να διευκολυνθεί μέσω ενός κοινού συστήματος πληροφοριών που παρακολουθεί με προσοχή την φροντίδα που παρέχεται από τις πολλαπλές υπηρεσίες έτσι ώστε να μην είναι ανάγκη να εφαρμόζεται ένα νέο σχέδιο περίπτωσης κάθε φορά που μπαίνει ένας πελάτης σε μια νέα υπηρεσία είτε πρόκειται για ένα καταφύγιο, νοσοκομείο κτλ.[5]

3. Της υποστήριξης μετακίνησης από τις οδούς και τα καταφύγια στην κατοικία - αυτό σημαίνει ότι η ολοκλήρωση στο ατομικό επίπεδο καθορίζεται από τον τελευταίο στόχο δηλαδή την έλλειψη στέγης για το άτομο. Η έλλειψη στέγης τελειώνει για ένα άτομο όταν είναι σε θέση που να μπορεί να έχει το δικό του σπίτι όπου συμβάλλουν προς την πληρωμή οι υπηρεσίες (μέσω των αποδοχών ή της κοινωνικής βοήθειας) και εφόσον οι βασικές ανάγκες τους καλύπτονται [5]

4. Της γνώσης της πραγματικότητας - αξιολόγηση των χρονικών αναγκών - αυτό σημαίνει ότι με το σύστημα CAISI οι συμμετέχουσες υπηρεσίες ξέρουν με το πάτημα ενός κουμπιού πόσα άστεγα πρόσωπα ζουν στην οδό ή στα καταφύγια με σοβαρές διανοητικές ασθένειες, με εθισμούς και με ειδικές ανάγκες. Αυτό και άλλα είδη πληροφοριών υπό μορφή εκθέσεων που αφορούν τα επίπεδα πληθυσμού στην υπηρεσία, το πρόγραμμα και τα κοινοτικά επίπεδα θα επιτρέψουν στις υπηρεσίες και τους αρμόδιους να κάνουν σωστό σχεδιασμό βασισμένοι στην αξιολόγηση των αναγκών που θα είναι πλέον αμέσως διαθέσιμα. Με αυτό το σύστημα, οι υπηρεσίες και οι αρμόδιοι δεν θα στηρίζονται πλέον μόνο στη διαθέσιμη έρευνα για το σχεδιασμό και την κάλυψη των άστεγων μιας και έχουν πρόσβαση σε αυτού του είδους των στοιχείων ανά πάσα στιγμή. [5]

Το λογισμικό αποτελεί μια βελτιώσει και μια επέκταση του συστήματος OSCAR McMaster[2], ένα σύστημα ανοικτού λογισμικού ηλεκτρονικού φάκελου ασθενή. Περιλαμβάνει την ενσωμάτωση κοινοτήτων σε πληροφοριακά δίκτυα ανάμεσα σε υπηρεσίες που νοιάζονται για τα άστεγα άτομα. Τέτοια συστήματα περιλαμβάνονται σε ειδικές στέγες, δημοσιά κέντρα υγείας και νοσοκομεία. [5]

Δοκιμαστική ιστοσελίδα του συστήματος για το CAISI - OSCAR IT

Για να δει κανείς μια δοκιμαστική ιστοσελίδα του συστήματος μπορεί να πάει στο:

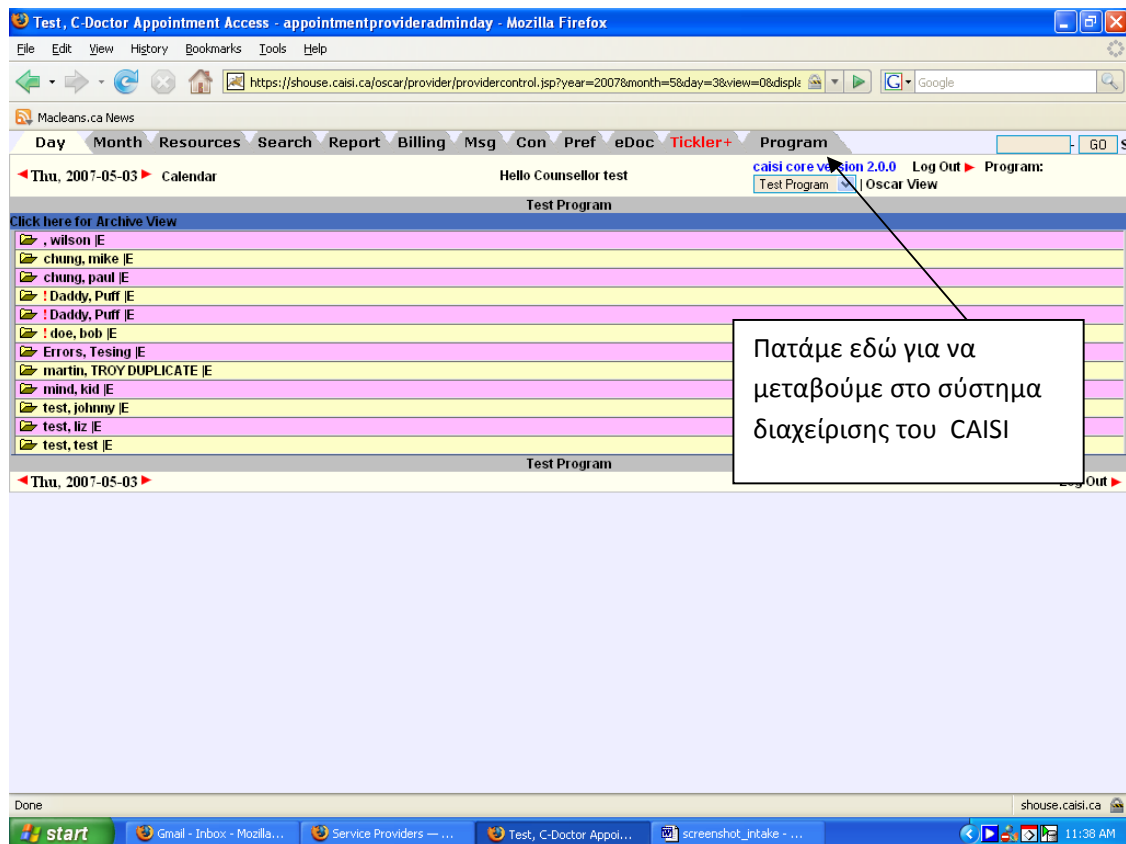
<http://demo.caisi.ca/oscar/> [6]

username: caisi

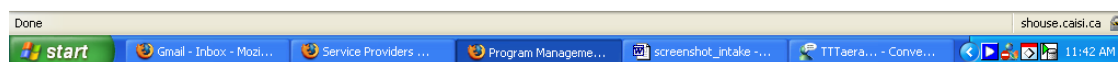
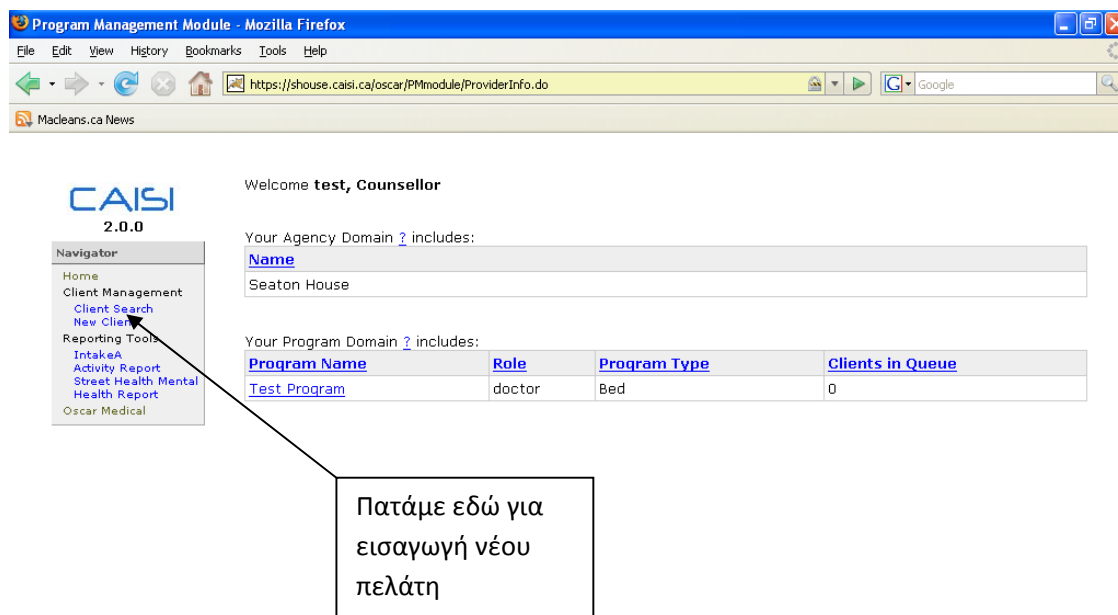
password: caisi

PIN: 1234

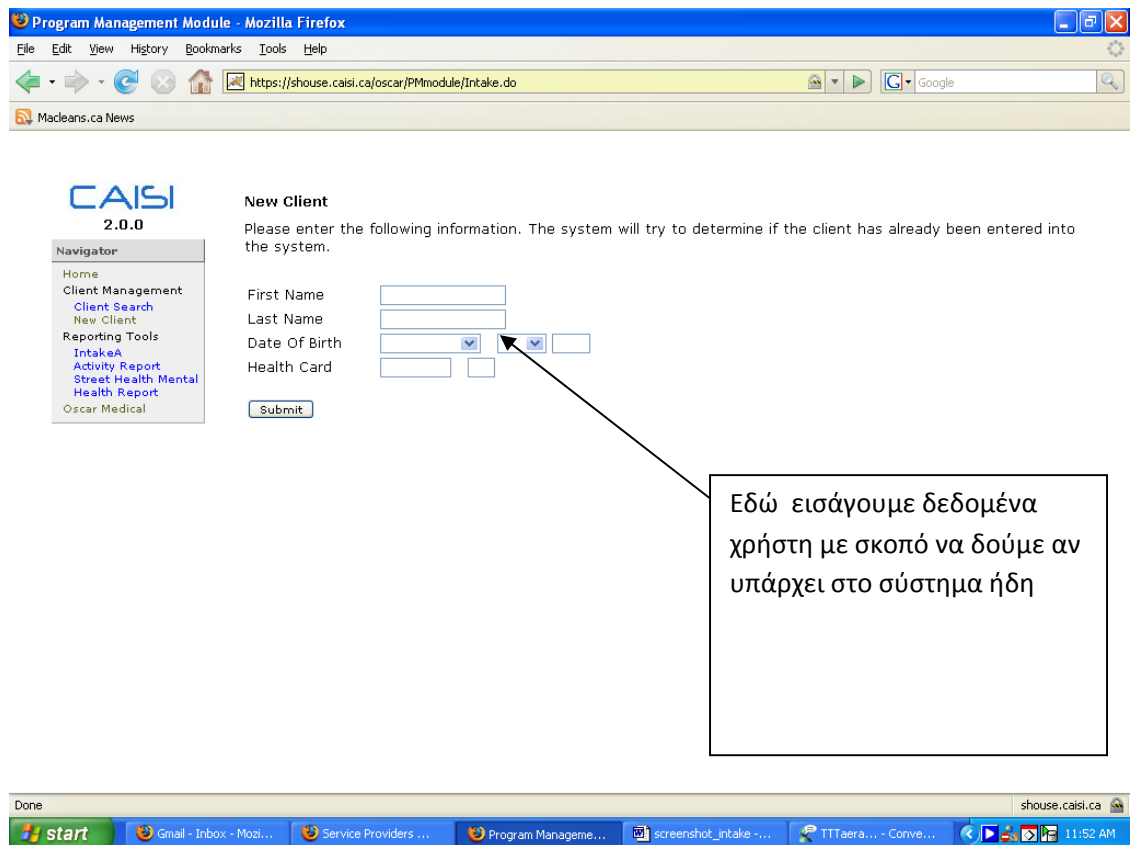
Το σύστημα παρέχει δυνατότητες διαχείρισης ραντεβού, σύστημα χρέωσης και ένα πολύ καλό σύστημα διαχειριστή για άμεση διαχείριση του συστήματος, των χρηστών του αλλά και των λειτουργιών που προσφέρει. Πιο κάτω ακολουθεί μια μικρή παρουσίαση του συστήματος.



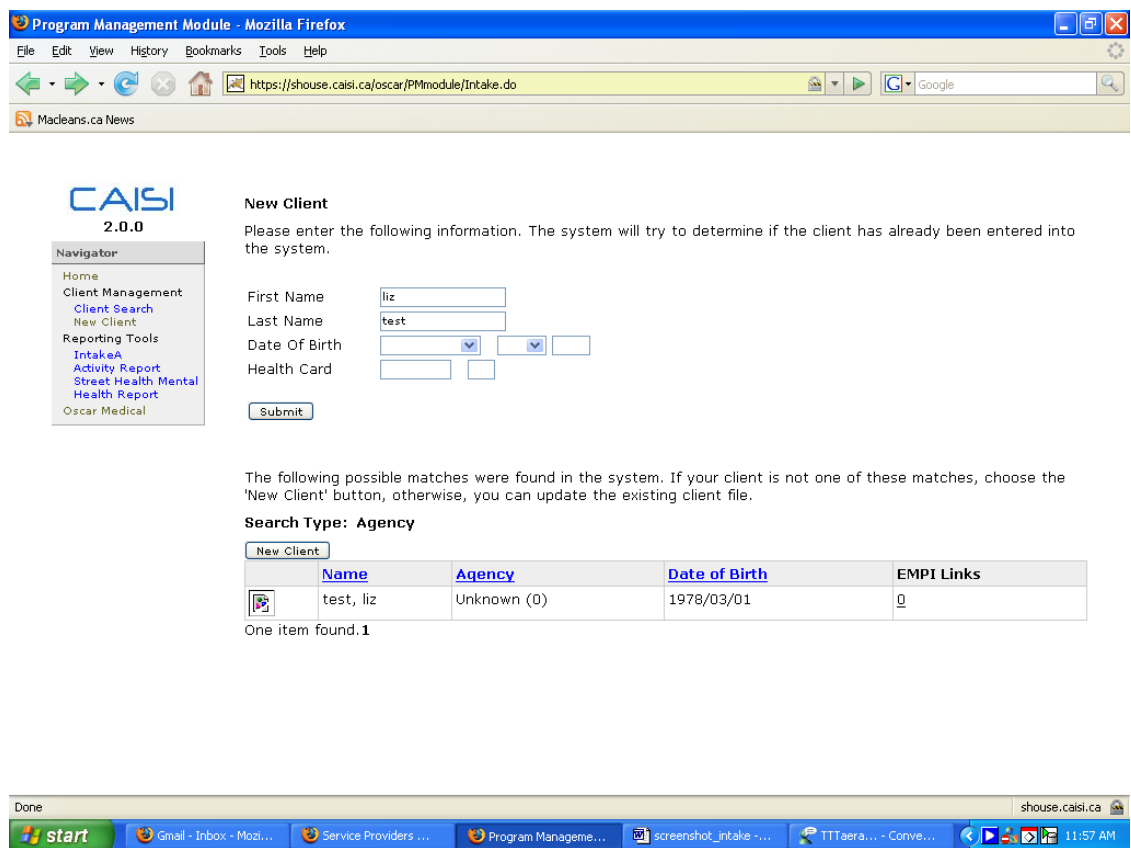
Σχήμα 2.1 – Εικόνα μετά το Login στο CAISI



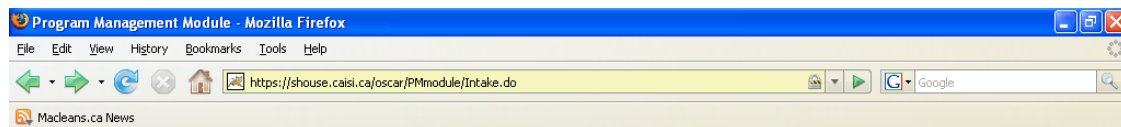
Σχήμα 2.2 – Εισαγωγή νέου πελάτη στο CAISI



Σχήμα 2.3 – Εισαγωγή νέου πελάτη στο CAISI (συνεχεία)



Σχήμα 2.4 – Αναζήτηση πελάτη στο CAISI (συνεχεία)



New Client

Please enter the following information. The system will try to determine if the client has already been entered into the system.

First Name
Last Name
Date Of Birth
Health Card

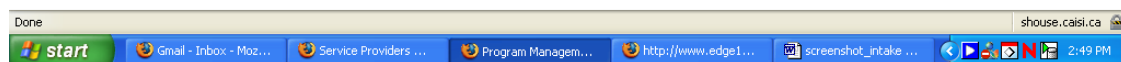
The following possible matches were found in the system. If your client is not one of these matches, choose the 'New Client' button, otherwise, you can update the existing client file.

Search Type: Agency

Name	Agency	Date of Birth	EMPI Links
test, liz	Unknown (0)	1978/03/01	0

One item found.1

Πατάμε εδώ για να οριστικοποιήσουμε την δημιουργία του νέου πελάτη



Σχήμα 2.5 – Εισαγωγή νέου πελάτη στο CAISI μετά από αναζήτηση

INTAKE A. RECEPTION ASSESSMENT [Back](#)

Assessment Date (YYYY/MM/DD): 2007/05/03	Assessment start time: am / pm	
Date client entered agency (YYYY/MM/DD)	☐ New Client	☐ Re-admission. Give date of last admission (YYYY/MM/DD).

Notes for completing this intake:

Reception Staff: ☐

1. Please complete every section of this intake. If you are unable to complete a section - use refused to answer, interview cut short).

2. Before beginning this intake, please read this paragraph to the client: "We would like to as getting the health care that you need and want while at this agency. Any answers you give us numbers from the answers that we get from everyone on a regular basis to try and make the system better at this agency and in the City but in this case we would not use your name. You don't have to answer these questions. Your stay at this agency will not change if you don't answer these questions"

Identifying Data

Client's Surname	Client's First Name	Date of Birth (MM/DD/YYYY)

Language(s) Spoken: ☐ English ☐ French ☐ Other:

Sex: ☐ Male ☐ Female ☐ Transgendered ☐ Declined

1. Reason for Admission

What is your reason for coming to this agency? (state briefly in client's own words if possible)

Have you ever stayed at this agency before? (give dates)

Εδώ γίνεται η εισαγωγή
στοιχείων του πελάτη

Σχήμα 2.6 – Εισαγωγή δεδομένων πελάτη στο CAISI

INTAKE A. RECEPTION ASSESSMENT - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://shouse.caaisi.ca/oscar/PMmodule/intake.do

Macleans.ca News

☐ Yes ☐ Uncertain ☐ No

I am concerned that this person is handicapped or disabled by severe behaviour problems

PRINT AND SIGN NAME:

9. ORIENTATION TO AGENCY

Tour given? ☐ Yes ☐ No (If no, give reason)

Information pamphlet issued? ☐ Yes ☐ No (If no, give reason)

10. COMMENTS/SUMMARY

COMPLETED BY

Completed by: (sign and print name)

Time Assessment Completed: am/pm

11. Admission

You must admit this client into a program. If you cannot admit into a program in your domain, the client will be admitted to the "holding tank" program by default.

Admit to bed program:

Done shouse.caaisi.ca

start Gmail - Inbox ... Service Provi... INTAKE A. RE... http://www.e... screenshot_in... iTunes 3:02 PM

Σχήμα 2.7 –Καταχώρηση πελάτη σε συγκεκριμένο πρόγραμμα στο CAISI

Εδώ επιλέγουμε το πρόγραμμα στο οποίο θέλουμε να εντάξουμε τον πελάτη ΠΡΙΝ να κάνουμε αποθήκευση.

Program Management Module - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://shouse.caais.ca/oscar/PMmodule/ClientManager.do?id=5367

Macleans.ca News

CAISI 2.0.0

test, johnny

Summary History Bed Reservation Forms Refer Discharge

Personal Information

Name test, johnny

Master File # 5367

Date of Birth 0001/01/01

Health Card

Resources

EMPI Unable to complete request

Bed/Reservation

Most Recent Staff Status Actions

2007-04-27 10:22 Murray, Elizabeth Update

Most Recent Staff Status Exclusions Actions

None found New Consent

Program Type Admission Date Days in Program Temporary Admission Admission Notes

Test Program Bed 2007-04-27 10:22 6 false Intake Based Admission

Done shouse.caais.ca

start Gmail - Inbox - Mo... Service Providers ... Program Manage... screenshot_intake... iTunes 3:26 PM

Πατώντας «Update» ενημερώνουμε τυχόν στοιχεία.

Από το σημείο αυτό μπορούμε να δώσουμε εξιτήριο και να αναθέτουμε ένα πελάτη σε κάποια άλλη υπηρεσία

Σχήμα 2.8 –Ανεύρεση / ενημέρωση υφιστάμενου πελάτη στο CAISI

Program Management Module - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://shouse.caasi.ca/oscar/PMmodule/Intake.do

Macleans.ca News

CAISI
2.0.0

Navigator

- Home
- Client Management
 - Client Search
 - New Client
- Reporting Tools
 - IntakeA
 - Activity Report
 - Street Health Mental
 - Health Report
 - Oscar Medical

New Client

Please enter the following information. The system will try to determine if the client has already been entered into the system.

First Name

Last Name

Date Of Birth

Health Card

The following possible matches were found in the system. If your client is not one of these 'New Client' button, otherwise, you can update the existing client file.

Search Type: Agency

	Name	Agency	Date of Birth
	Tekeste, Nahom	Unknown (0)	1973/1/22
	test, test	Unknown (0)	1978/01/01
	test, johnny	Unknown (0)	0001/01/01
	test, liz	Unknown (0)	1978/03/01
	Tookata, Michael	Unknown (0)	1972/06/29

5 items found, displaying all items.1

Done

start Gmail - Inbox (1) - ... Service Providers ... Program Manage... screenshot_intake... iTunes

shouse.caasi.ca

3:36 PM

Αν ο πελάτης υπήρχε τότε θα εμφανιστεί εδώ και πατώντας την εικόνα εμφανίζονται τα στοιχεία του

Σχήμα 2.9 –Ανεύρεση / ενημέρωση υφιστάμενου πελάτη στο CAISI (συνεχεία)

INTAKE A. RECEPTION ASSESSMENT - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://shouse.caaisi.ca/oscar/PMmodule/Intake.do

Maclean's.ca News

Answers from the answers that we get from everyone on a regular basis to try and make the system better at this agency and at the City but in this case we would not use your name. You don't have to answer these questions. Your stay at this agency will not change if you don't answer these questions"

Identifying Data

Client's Surname test Client's First Name liz Date of Birth (MM/DD/YYYY) 03 01 1978

Language(s) Spoken: ☐ English ☐ French ☐ Other: Sex: ☐ Male ☐ Female ☐ Transgendered ☐ Declined

1. Reason for Admission

What is your reason for coming to this agency? (state briefly in client's own words if possible)

Was just evicted from apartment.

Have you ever stayed at this agency before? (give dates)

2. Assistance Required

Do you require assistance with any of the following :

☐ Physical or Mental Health, including medication

☒ Obtaining Identification

☐ Addictions

☒ Housing issues

☐ Education Issues

☒ Employment issues

☒ Financial issues

☒ Legal issues

☐ Immigration issues

Done shouse.caaisi.ca

start Gmail - Inbox - Mo... Service Providers ... INTAKE A. RECEPTION screenshot_intake... iTunes 3:45 PM

Σχήμα 2.10 –Ανεύρεση / ενημέρωση υφιστάμενου πελάτη στο CAISI (συνεχεία)

Program Management Module - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://shouse.caaisi.ca/oscar/PMmodule/ProviderInfo.do

Macleans.ca News

CAISI
2.0.0

Welcome **Vlasic, Pamela**

Your Agency Domain ? includes:

Name
Seaton House

Your Program Domain ? includes:

Program Name	Role	Program Type	Clients in Queue
Hostel	doctor	Bed	1
Long Term Program	doctor	Bed	3
ONeill Program, Housing Transition Unit	doctor	Bed	4
ONeill Program, High Needs Referral Unit	doctor	Bed	1
Birchmount Residence	doctor	Bed	0
Downsview Delis	doctor	Bed	1
Annex Moderate Drinking Program	doctor	Service	0
Annex Harm Reduction Program	doctor	Bed	0
Rotary Club of Toronto Infirmary	doctor	Service	0
Fusion of Care Community Support	counsellor	Service	0
Housing Community	doctor	Service	2
Fort York	doctor	Bed	0
Fusion of Care Program	doctor	Service	0
Nursing Home	doctor	Service	0
Test Program	doctor	Bed	0
Duplicate Client Program	doctor	Bed	0
Annex Community Support	counsellor	Service	1

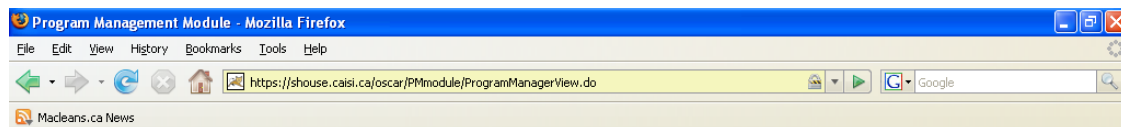
Done

shouse.caaisi.ca

start Gmail - Inbox - Mo... Service Providers ... Program Manage... screenshot_intake... iTunes 3:51 PM

Πατώντας σε κάποιο πρόγραμμα εδώ μπορούμε να δούμε ποιοι μεταφέρθηκαν στο πρόγραμμα αυτό

Σχήμα 2.11 –Ανεύρεση πελατών που ανήκουν σε ένα συγκεκριμένο πρόγραμμα στο CAISI



CAISI 2.0.0

Test Program

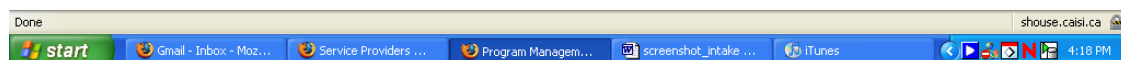
General Staff Function User Teams Clients Queue Access Bed Check

Queue

		Client Name	Referral Date	Referring Provider	Temporary Admission	Notes
Admit	Reject	test,liz	2007-05-03 16:17:54.0	Vlasic,Pamela	false	

Navigator

- Home
- Client Management
 - Client Search
 - New Client
- Reporting Tools
 - IntakeA
 - Activity Report
 - Street Health Mental Health Report
- Agency Management
 - Summary
 - Update Info
- Staff Management
 - Staff List
- Program Management
 - Program List
 - Add Program
 - Global Role Access
 - Caisi Roles
 - Oscar Medical



Σχήμα 2.12 –Ανεύρεση πελατών που ανήκουν σε συγκεκριμένο πρόγραμμα στο CAISI (συνεχεία)

2.3 Σύστημα CHIRIS (Canada Health Infoway Reference Implementation Suite)

Το σύστημα CHIRIS (Canada Health Infoway Reference Implementation Suite) είναι ένα σύστημα που παρέχει σύστημα ενδοεπικοινωνίας που δημιουργήθηκε από το Canada Health Infoway.[7]

Το Infoway Reference Implementation Suite επιδεικνύει πώς τα διαφορετικά συστήματα, σε αυτήν την περίπτωση συστήματα ληξιαρχείων πελατών που συνδέονται με τα κλινικά συστήματα, μπορούν να επικοινωνήσουν το ένα με το άλλο. Οι ασθενείς μέσα στο σύστημα υγειονομικής περίθαλψης εγγράφονται συχνά σε πολλούς διαφορετικούς καταλόγους (ληξιαρχεία), μέσα στις κλινικές, τα νοσοκομεία ή σε επαρχιακό επίπεδο. Για να παρουσιάσει την πλήρη εικόνα του ιατρικού ιστορικού ενός ασθενή, είναι απαραίτητο

να προσδιοριστούν οι ασθενείς στα ανεξάρτητα συστήματα για να είναι σε θέση να πάρει μια πλήρη άποψη των κλινικών πληροφοριών τους κάποιος γιατρός.[8]

Για να διευκολύνει τη δια-λειτουργικότητα, η Infoway επιδεικνύει πώς τα κοινά πρότυπα για την ανταλλαγή πληροφοριών υγείας μπορούν να λειτουργήσουν όταν συνδέονται τα συστήματα ληξιαρχείων πελατών με τα κλινικά συστήματα μιας αρμοδιότητας με ένα σύστημα άλλης αρμοδιότητας. Σύμφωνα με την αποστολή της για να επιταχύνει την ανάπτυξη μιας ηλεκτρονικής λύσης αρχείων υγείας, η Infoway του Καναδά λειτουργεί επιμελώς για να εξασφαλίσει ότι τα διαφορετικά συστήματα πληροφοριών υγείας μπορούν να ανταλλάξουν τις πληροφορίες το ένα με το άλλο. Χωρίς πρότυπα για να εξασφαλίσει δια-λειτουργικότητα, θα υπήρχε μόνο μια προσθήκη ανόμοιων συστημάτων σε ολόκληρη τη χώρα.[8]

«Ως εργαλείο αναφοράς για τους εφαρμοστές και τους προμηθευτές, η Infoway θα επιτρέψει τη γρηγορότερη λήψη των ηλεκτρονικών προτύπων δια-λειτουργικότητας αρχείων υγείας που επικυρώνονται από Infoway και τους ομοσπονδιακούς, επαρχιακούς και εδαφικούς συνεργάτες,» όπως ανέφερε ο Richard Alvarez, ο Πρόεδρος της Infoway. Το εργαλείο θα βοηθήσει επίσης να μετριάσει τον κίνδυνο για όλες τις αρμοδιότητες και τις οργανώσεις που επιθυμούν να υποβάλουν πληροφορίες στα δια-λειτουργικά συστήματα ληξιαρχείων πελατών. Για να καταστήσει το σύστημα διαθέσιμο στο ευρύτερο πιθανό ακροατήριο, η Infoway το έχει καταστήσει διαθέσιμο δωρεάν με άδεια ανοικτή λογισμικού σε όλους εκείνους που ενδιαφέρονται. Αυτό είναι πρώτη συμβολή Infoway στην κοινότητα ανοικτού λογισμικού.[1,8]

"Infoway Reference Implementation Suite" Τεχνική επισκόπηση

- Η Infoway Reference Implementation Suite βοηθά να καταδείξει και να αποδείξει ότι τα μηνύματα δια-λειτουργικότητας ληξιαρχείων πελατών χρησιμοποιώντας το HL7 πρότυπα έκδοσης 2.4 και έκδοσης 3.0 είναι μια πραγματικότητα. Το σύστημα εφαρμόζει το ελάχιστο υποσύνολο των δια-λειτουργικών ηλεκτρονικών υπηρεσιών και της λειτουργίας αρχείων υγείας πυρήνων (iEHR) που απαιτείται για να καταδείξει το κλινικό σύστημα στο ληξιαρχείο πελατών την δια-λειτουργικότητα ληξιαρχείων πελατών. Εφαρμόζει επίσης ένα ληξιαρχείο πελατών, ένα απλό ADT

(αναγνωρίστε, απαλλάξτε, μεταφορά) εφαρμογή και διάφορα βασικά συστατικά που βρίσκονται στα προϊόντα υλικό λογισμικού (στο σχεδιάγραμμα Infoway αναφερόμαστε σε αυτό το συστατικό ως HIAL - στρώμα πρόσβασης πληροφοριών υγείας). Η εστίαση της ακολουθίας είναι στο HIAL και τις αλληλεπιδράσεις μεταξύ του HIAL, του ληξιαρχείου πελατών και της εφαρμογής ADT. Επιπλέον, καταδεικνύει την εξαγωγή των κλινικών στοιχείων δειγμάτων από μια υπηρεσία αποθηκών EHR για να καταδείξει (να μιμηθεί) την κατασκευή που προέρχεται από τις διαφορετικές αρμοδιότητες υγείας στον Καναδά. Η Infoway Reference Implementation Suite θα δείξει ότι πραγματικά στο HL7 V2.4 και V3.0 τα μηνύματα ληξιαρχείων πελατών λειτουργούν σε ένα περιβάλλον επίδειξης. Οι προμηθευτές μπορούν να χρησιμοποιήσουν τον κώδικα πηγής για να επιταχύνουν τις εφαρμογές αυτών των μηνυμάτων τους που η Infoway Reference Implementation Suite προσφέρει αυτήν την περίοδο στο Sourceforge.net, μια περιοχή διανομής ανοικτού κώδικα. Η περιοχή επιτρέπει στους χρήστες για να έχει πρόσβαση στην ακολουθία λογισμικού συν την ακόλουθη ενισχυτική τεκμηρίωση:

- Εγχειρίδιο εγκαταστάσεις.
- Εγχειρίδιο διαχείρισης.
- Εγχειρίδιο Χρηστή.
- Παρουσίαση συστήματος.
- Τεχνικές Προδιαγραφές Αρχιτεκτονικής Συστήματος.
- Σχεδίαση εφαρμογής σε ψήλο επίπεδο. [8]

Ένας σύνδεσμος προς το Infoway Reference Implementation Suite στο Sourceforge.net είναι:

<http://sourceforge.net/projects/crrs/> [7]

Πιο κάτω ακολουθεί μια μικρή παρουσίαση του συστήματος με χρήση σχημάτων από την διεπαφή του συστήματος

2.3 Σύστημα CHITS

Το σύστημα CHITS είναι ένα κοινοτικό σύστημα πληροφοριών υγείας, ηλεκτρονικό ιατρικό αρχείο για τα κέντρα υγείας τοπικής κυβέρνησης στις αναπτυσσόμενες χώρες, ενεργός ανάπτυξη, Φιλιππίνες. [9]

Είναι ένα σύστημα από συστήματα εκείνων των υπαρχόντων πληροφοριών υγείας (κάθετα προγράμματα) στη δημόσια υγεία για να χρησιμεύσει ως μια αφετηρία για την εφαρμογή ολοκληρωμένων συστημάτων πληροφοριών. Μέσω του CHITS, οι πληροφορίες υγείας σε κοινοτικό επίπεδο τίθενται στην διάθεση όχι μόνο των οργανισμών δημόσιας υγείας που επιθυμούν τις πληροφορίες σε κοινοτικό επίπεδο αλλά και στην ίδια την κοινότητα που παράγει τις πληροφορίες. Επιτρέπει στην κοινότητα να χρησιμοποιήσει αυτές τις πληροφορίες για την τοπική λήψη αποφάσεων. [9]

Το CHITS σκοπό έχει την:

1. **Καινοτομία.** Οι υπάρχουσες πηγές πληροφοριών (στερεότυπες πληροφορίες υγείας). Αποκτούν αξία με την τοποθέτηση των πληροφοριών υγείας στο πλαίσιο όπου ανήκουν και οι πληροφορίες δεν είναι πλέον απομονωμένες στα κάθετα συστήματα στο κοινοτικό επίπεδο. Δημιουργείται έτσι το κοινωνικό κεφάλαιο στην κοινότητα με την ενίσχυση των ικανοτήτων των κοινοτικών εργαζομένων στον ιατρικό κλάδο και έτσι να διαχειριστούν τις πληροφορίες υγείας με την ποιότητα σαν κύριο στόχο. [9]
2. **Σύγκλιση.** η κοινότητα είναι σε μια θέση όπου όλοι οι συμμετέχοντες συναντιούνται, είτε είναι οι υπηρεσίες υγείας, η πληροφορική, η ιατρική πληροφορική, η πολιτική οργάνωση, η ακαδημαϊκή κοινότητα, άλλες ενδιαφερόμενες ομάδες και ο απλός κόσμος. Το σύστημα CHITS παρέχει ένα περιβάλλον όπου οι διαφορετικοί συμμετέχοντες μπορούν να ψάξουν με σκοπό να εφαρμόσουν το «bayanihan» ύφος. «Bayanihan» είναι μια παλαιά συνήθεια των

Φιλιππίνων που επιτρέπει στις κοινότητες να επιτύχει έναν μεγάλο στόχο μέσω της συνεργασίας και της εργασίας από κοινού. [9]

3. **Συνυπολογισμός.** Το σύστημα CHITS έχει κάτι για όλους. Εάν εκείνος ο κάποιος ενδιαφέρεται για την φυματίωση, προωθώντας την ανοσοποίηση σε παιδική ηλικία, την ενίσχυση της μητρικής φροντίδας, που προωθεί την πρωτοβάθμια περίθαλψη, Το σύστημα CHITS μπορεί να παρέχει ένα περιβάλλον πληροφοριών για τις διαφορετικές ομάδες συνεργασίας έτσι ώστε να αναπτυχθούν. Το σύστημα CHITS στοχεύει να δημιουργήσει αυτό το «οικοσύστημα» πληροφοριών για να επιτρέψει την ροή των πληροφοριών από τον ένα συμμετέχοντα σε άλλον. Σαν εφαρμογή ανοικτού λογισμικού, είναι ευπρόσδεκτη ακόμη και η συμμετοχή των υπεύθυνων για την ανάπτυξη λογισμικού έτσι ώστε να μπορούν με το να δώσουν την κατάλληλη σημασία να παραγάγουν εργασία που να ωφελεί ολόκληρους πληθυσμούς, συμπεριλαμβανομένου των ίδιων. Το σύστημα CHITS και οι δημιουργοί του υποστηρίζουν την επιτήρηση υγείας μέσω των κυβερνητικών τομέων και των συστημάτων πληροφοριών (FHSIS) και ενσωμάτωση των σχετικών προγραμμάτων ως ενότητες το συστήματος CHITS. [9]
4. **Ίσες ευκαιρίες.** Το σύστημα CHITS δεν κάνει διακρίσεις από άποψη φύλο, ηλικίας και δυνατότητας. Η εργασία πληροφοριών μπορεί να γίνει σε «bayanihan» ύφος έτσι ώστε ακόμη και η απλούστερη εργασία πληροφοριών (π.χ., ζυγίζοντας τον ασθενή, που διαβάζει τις ετικέτες εμβολίων) κρίνεται σημαντική δομική μονάδα της υψηλής ποιότητας πληροφοριών υγείας που χρησιμοποιούνται για λήψη αποφάσεων. [9]
5. **Ικανότητα υποστήριξης.** Το σύστημα CHITS και οι δημιουργοί του στοχεύουν στο να έχουν την ικανότητα υποστήριξης του «οικοσυστήματος πληροφοριών» που προαναφέρθηκε. Το κάνουν αυτό με την παραγωγική συμμετέχει των αρμόδιων για να ληφθεί το όφελος από τις πληροφορίες υγείας διοικούμενες από το σύστημα CHITS . Το κάνουν αυτό μέσω της καθιέρωσης συνεργασιών με τις μονάδες τοπικής της κυβέρνησης, με όσους συμφωνούν με τους όρους και τις διατάξεις και το σύστημα CHITS εγκαθίσταται στις κοινότητες τους. Αυτές οι κοινότητες, στη

συνέχεια, έχουν όφελος από την άποψη ότι καθιστώντας τις πληροφορίες υγείας εύκολα διαθέσιμες, είτε είναι ένας ασθενής είτε ολόκληρης της κοινότητας είναι δυνατόν να σωθούν πολλές ζωές. [9]

Σαν σύστημα, το σύστημα CHITS αποτελείται από τα εξής:

1. Μια πλατφόρμα ανάπτυξης λογισμικού που αποκαλείται “GAME Engine” που ουσιαστικά σημαίνει Generic Architecture for A Modular Enterprise. Χρησιμοποιεί Linux, Apache, MySQL και PHP ως πλατφόρμα για την μηχανή και για την δημιουργία εφαρμογών και συστατικών που είναι επαναχρησιμοποιήσιμος κώδικας. Αυτός ο μηχανισμός επιτρέπει στους υπεύθυνους για την ανάπτυξη να δημιουργήσουν plug-in, uploadable ενότητες στο σύστημα CHITS σαν εφαρμογή ενδοδικτύου. [10]
2. Στο σύστημα CHITS η ίδια η εφαρμογή, είναι πολύ-μορφική, πολύ-γλωσσική και συγχρόνως ένα αυτοματοποιημένο αρχείο ασθενών. [9]
3. Γίνεται κατάρτιση σε ενότητες για τους διαφορετικούς συμμετέχοντες: κοινοτικοί εργαζόμενοι στον ιατρικό κλάδο, ανώτεροι υπάλληλοι δημοτικής/υγείας παθολόγων και ανώτεροι υπάλληλοι και προσωπικό υγείας πόλεων. [9]
4. Υπάρχουν ενότητες έτσι ώστε να ενσωματώσει τα διαφορετικά Φιλιππινέζικα κάθετα προγράμματα δημόσιας υγείας: Μητρικής και παιδικής υγείας EPI, εθνικό πρόγραμμα φυματίωσης, Philhealth άπορο πρόγραμμα, εθνικής δήλωσης ασθένειες, ηλεκτρονικές εκθέσεις / παραγωγή γραφικών παραστάσεων για τις διαφορετικές ενότητες και πολλά άλλα [9]
5. Σύστημα ημερολογίων και διορισμού, κλινικό σύστημα υπενθυμίσεων SMS [9]

Η εφαρμογή του προγράμματος CHITS πραγματοποιήθηκε από μια επιχορήγηση από το Panasia-ICT, μια κοινή προσπάθεια χρηματοδότησης από το Διεθνές ερευνητικό κέντρο ανάπτυξης από τον Καναδά και το Πρόγραμμα ανάπτυξης Ηνωμένων Εθνών. Για μια πλήρη και πολύ καλύτερη παρουσίαση σε σχέση με αυτή που θα μας έδεναν οι εικόνες που θα παραθέταμε εδώ μπορεί κάνεις να επισκεφτεί την πιο κάτω ιστοσελίδα

<http://movies.miu.ph/chits/> [9,11,12,24]

2.4 Σύστημα Cottage Med

Το σύστημα Cottage Med είναι λογισμικό ανοικτού κώδικα που διέπεται από τα εξής χαρακτηριστικά:

- Διαθέσιμος & προσιτός (ελεύθερος, στην πραγματικότητα)
- Εύκολα εξατομικεύσιμος και επεκτάσιμος
- Σχεδιασμένος από παθολόγους για παθολόγους
- Χρησιμοποιημένος από τους γιατρούς στις ΗΠΑ. και άλλα αναπτυσσόμενα κράτη
- Αποτελεί ένα πρότυπο για τις εκδόσεις FileMaker© 5-8 [13]

Για όσος φοβίζονται από τους υπολογιστές και για όσους το κόστος ενός ηλεκτρονικού ιατρικού αρχείου («EMR») είναι πέρα από την οικονομική τους δυνατότητα είναι η ιδανική λύση. Αν ο οργανισμός σας είναι μη κερδοσκοπική οργάνωση ή απλά είστε ένας ιδιώτη γιατρός που αγωνίζεται τότε το σύστημα αυτό είναι το ιδανικό και στην δική σας περίπτωση. Το επάγγελμά ενός γιατρού κατά την άποψη των υπευθύνων του συστήματος, υποφέρει με μια χιονοθύελλα ανταγωνισμού και σύγκρουσης των EMR. Φουσκωμένα με τα περιττά χαρακτηριστικά γνώρισμα, κοστίζουν πολλά λεφτά. Με το σύστημα Cottage Med μπορούμε γρήγορα να μάθουμε και να επαναπρογραμματίσουμε εύκολα το EMR ακριβώς όπως μας βολεύει με οπτικό τρόπο και βασικά χωρίς καμία εμπειρία στον προγραμματισμό. [13]

Η ιατρική έχει ένα μεγάλο πρόβλημα - κανένα ευρέως αποδεκτό, προσιτό πρότυπο EMR δεν υπάρχει. Το σύστημα Cottage Med γεφυρώνει το δύσκολο παρόν και άγνωστο μέλλον και με τη βοήθεια που προσφέρει όσο αφορά τον επαναπρογραμματισμό έτσι ώστε να είναι όπως ο κάθε χρήστης το ονειρευτέ τον βοήθα να έχει σε έναν πιο ήρεμο, ομαλό, παραγωγικό επαγγελματικό προγραμματισμό όπου διατηρείτε πλήρης έλεγχο των πληροφοριών, πλήρης πρόσβαση όπως, και η δυνατότητα να εξελιχθεί ομαλά σε οτιδήποτε μελλοντικά σύστημα ή πρότυπα επιλεγούν. [13]

Αποτελεί ένα εύκαμπτο, αξιόπιστο και HIPAA-ασφαλές σύστημα γραφείων μπορεί να κατασκευαστεί με το παλαιό υλικό των υπολογιστών που έχουμε στην διάθεση μας για μερικές μονό εκατοντάδες δολάρια με την ασφαλή και ασύρματη δικτύωση, την υποστήριξη PDA και πολλαπλών-γραφείων, το γράψιμο συνταγών και η συμβατότητα πολλαπλών-πλατφορμών που ενσωματώνονται. Δεν πρέπει να θεωρήσουμε τη διαφημιστική εκστρατεία του AAFP και των εκατοντάδων των επιχειρήσεων λογισμικού για EMR. Οι λύσεις ανοικτού κώδικα για EMR λειτουργούν αμέσως, σήμερα, με κανένα κόστος ακόμη και χωρίς αγορά FileMaker©. [13]

Το σύστημα είναι σχεδιασμένο για το PC, Apple και τα περιβάλλοντα Linux, αυτά τα πρότυπα τρέχουν στα πλαίσια του προγράμματος FileMaker© βάσεων δεδομένων (έτσι δεν είναι ένα «καθαρό» έργο ανοικτού κώδικα), αλλά το σύστημα Cottage Med εγκαθίσταται όπως σαν ένα απλό παιχνίδι, αναπτύσσει γρήγορα, και κάνει την ζωές του ιατρού απλή για να βοηθήσει να βελτιώσει την υγειονομική περίθαλψη.[13].

Συμπερασματικά αναφέρουμε τα θετικά στοιχεία του συστήματος:

- Κόστος είναι ελεύθερο, αλλά οι δωρεές εκτιμώνται πολύ και είναι μια τεράστια βοήθεια για να κρατήσουν το λογισμικό στην ανάπτυξη
- Διέπεται από άδεια ανοικτού κώδικα
- Συμβατός με τη MAC. PC, και Linux (μέσω του ενδοδικτύου μηχανής αναζήτησης Ιστού)
- Διεθνής διανομή και υποστήριξη
- Γρήγορη εκπαίδευση από την ομάδα Cottage Med - αποδεδειγμένη 1 ώρα για το προσωπικό γραφείων και 2 ώρες για τους κλινικούς χρήστες
- Εύκολα προσαρμόσιμος, επαναπρογραμματίσιμος από τους χρήστες χωρίς την ανάγκη της προγραμματιστικής εμπειρίας
- Σχέδιο διορισμού είναι εύκολος και εύκαμπτος στο σπίτι και γραφείο
- Εγγραφή ασθενών με 2 γρήγορες και απλές οθόνες
- Τιμολόγηση εγγράφου με την επιλογή να μετατρέψει τις ηλεκτρονικές αξιώσεις αναλόγως της περίπτωσης.

- Βασικές λογιστικές λειτουργίες που ενσωματώνονται
- Ισχυρός και εύκαμπτος στην υποβολή έκθεσης
- λειτουργίες αναζήτησης σε όλους τους τομείς
- Ενσωματώνει με άλλες εφαρμογές της επιλογής του χρηστή
- ASP επιλογή
- Ασύρματη και PDA υποστήριξη
- Όλο το υλικό υποστηρίζεται από το 1997 στην παρούσα χρήση του Filemaker 5/6 2001 που συστήνεται για FM 7/8
- Γράψιμο συνταγών είναι πλήρως αυτοματοποιημένος
- Επιδημιολογία ανάσυρση δεδομένων υποστηρίζεται
- Εξελικτικός σε πολλά χωριστά γραφεία και 250 ταυτόχρονους χρήστες
- Εύκαμπτος στο γράψιμο σημειώσεων - μπορούμε να εισαγάγουμε τις πληροφορίες με οποιοσδήποτε τρόπο θέλουμε ή χρειαζόμαστε
- Χειρόγραφες σημειώσεις υποστηρίζονται
- Υπαγόρευση από τις σημειώσεις και άμεση πρόσθεση στο διάγραμμα
- Κλινικές φωτογραφίες τοποθετημένος σε κάθε σημείωση όταν απαιτείται
- Δεινότητα αποθήκευσης βίντεο, ακτινογραφιών, και καρδιογραφημάτων. Οι μελέτες μπορούν να αποθηκευτούν και να αναχαραχτούν με τους μεταφραστές QuickTime [13]

Cottage_Med.fp5

? Help Sync Form List Report New Prior Find Sort Word Date Phone I-net Graph Home

Cottage Med v.2.0 c.2005

Patient Info Letters Reports Measures Notes Studies Blank Accounts Filing

Lee - SAMPLE Sarah 8.23.2002 3.4 y/o 423-324-2020 Vaccines Measures Print

Health Measures Date & action pap mam psa wt bp choi guiac sig a1c malb eye pod INR Dig Cr UA CXR

Algorithms:

Hospice
 Diabetes
 GAF Scale
 PCA dose
 opioid dose
 Poisoning
 GERD
 Stress Tests

Public Health:

Growth Charts
 Child Schedule
 Adult Schedule
 VIS's
 Declination
 Reportables

Diagnoses Details

hypertension - well controlled 1995
 hypochondriasis
 test
 Abnormal X-ray lung 793.1

PN scans

blank
 View Print

PN scan Labels

11.0 blank View

Ped. Screening...

Date:
 Medicaid?
 Uninsured?
 Amerindian?
 Private Ins?

Clinical Alerts:

RN Make Null
 DR Make Null

Please cut and paste any handwritten notes, graphs, charts, lab reports or other in THIS GRAPHIC FIELD...

Σχήμα 2.13 – Διεπαφή Cottage Med

Cottage_Med.fp5

Help Sync Form List Report New Prior Find Sort Word Date Phone I-net Graph Home Cottage Med v.2.0 c.2005

Patient Info Letters Reports Measures Notes Studies Blank Accounts Filing

Sarah Lee - SAMPLE 8.23.2002
 First Name M. Last Name Birthday
 111 Route 111
 Street Address Age 3.4 y/o w Gender
 (extra space) Add'l Address ☐ boarding home ☐ nursing home
 Charlene Manor Ma 01221
 City State ZIP Code
 423-324-2020 Home # Work # Pharmacy

Name: pending
 Phone:
 Address:
 City&State:
 Relation:

Emergency Contact
 Stephanie home Nurse 1.1.2006 Last Nurse Q3 years Nurse Due
 Stefan Topolski Physician 1.1.2006 Last Physician Q2 years Physician Due
 Dr. Joe Smith PCP Name 12.31.2005 Last H&P 14 days H&P Due
 1.1.03 Last Labs Q5 years Labs Due

Photo

Social Hx
 Domestic Relationship: pending Occupation
 Work: Lawn
 Pleasures: Employer
 Diet & Exercise: self
 Drug Use:
 Housing:
 Education:
 Religion:
 Childhood:
 Birthplace:

Family Hx
 Parents: pending
 Siblings:
 Children:
 Origin:

Allergy
 PENICILLIN ASE
 ASA
 TMX

Current Prescriptions
 pcn vk 500mg i po aid

Dx's
 hypertension - well controlled 1995
 hypochondriasis
 test
 Abnormal X-ray lung 793.1

Details

Alerts for... Community ☐ Nurse ☐ Doctor ☐ Records needed: ☒ Status Modified Created ID #
 Make Erase Make Erase Make Erase Mailmerge active 1.3.06 10.29.05 10

α.2.15 – Στοιχεία ασθενή σε Cottage Med

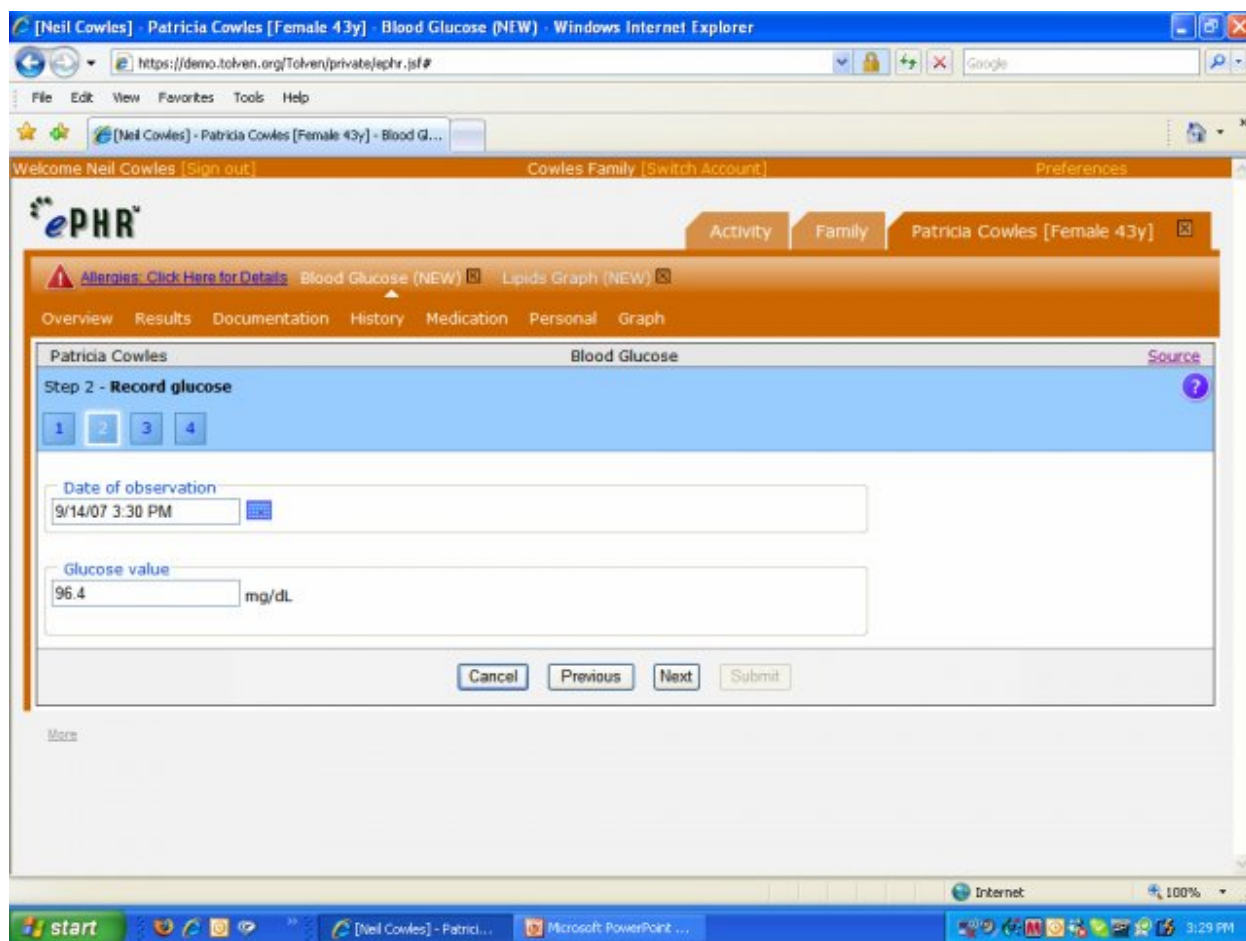
2.5 Σύστημα ePHR

Το ηλεκτρονικό προσωπικό αρχείο υγείας της Tolven επιτρέπει στους καταναλωτές να εξασφαλίσουν ότι οι ιατροί τους έχουν τις πιο πρόσφατες πληροφορίες για να τους καθοδηγήσουν στη λήψη των αποφασής τους. Ένα ηλεκτρονικό προσωπικό αρχείο υγείας παρέχει στον καταναλωτή μια διεπαφή βασισμένη σε WEB εφαρμογή για να δημιουργηθούν, να αντιμετωπισθούν, να αποθηκευτούν και να διαμοιραστούν οι πληροφορίες υγειονομικής περίθαλψης για τους ιδίους και τους εξαρτημένους που τους φροντίζουν π.χ. άτομα μεγάλης ηλικίας, συγγενείς και παιδιά για να επικοινωνήσουν με τους ιατρούς τους και να έχουν πρόσβαση σε αναγκαίες και σχετικές με την υγεία τους πληροφορίες μέσω της δύναμης του διαδικτύου και έτσι να κάνει άπλες κάποιες συχνές ανάγκες ασθενών, όπως το ξαναγέμισμα μιας συνταγής για τους ιδίους ή ενός από τους εξαρτημένους τους. Όλα αυτά γίνονται με μια ελάχιστη προσπάθεια. [14]

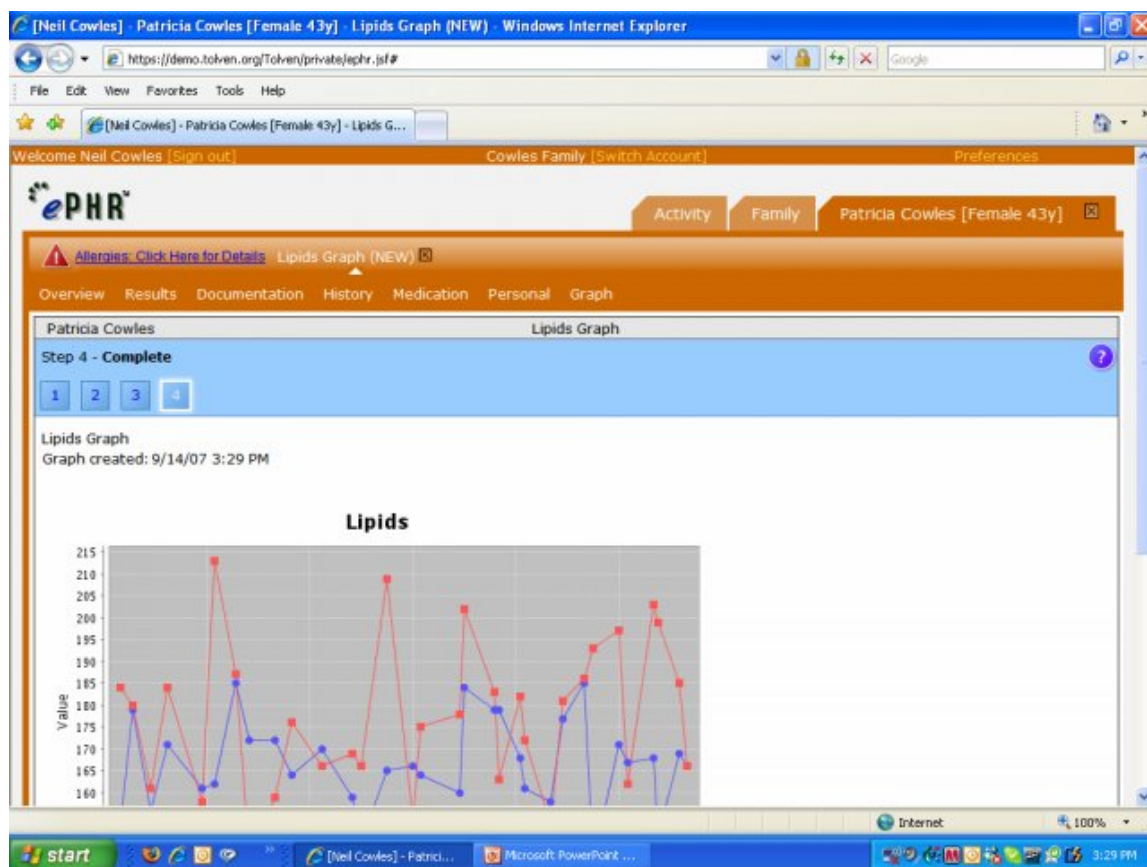
Ακόμα, το ηλεκτρονικό προσωπικό αρχείο υγείας δεν είναι απλά μια στατική συσκευή διατήρησης αρχείων. Το ηλεκτρονικό προσωπικό αρχείο υγείας επιτρέπει στον καταναλωτή να παίρνει επιλεκτικά τις πληροφορίες που ένας ιατρός έχει συλλέξει, επιτρέποντας κατά συνέπεια στον καταναλωτή με ευκολία να βλέπει τα στοιχεία αυτά και ασφαλώς να έχουν μια μερική ή περιεκτική άποψη των ιατρικών πληροφοριών τους διαθέσιμες σε άλλους ιατρούς. Με άλλα λόγια είναι ακριβώς όπως φανταζόμαστε το γραφείο του γιατρού μας αλλά χωρίς να πρέπει να συμπληρώσουμε οποιαδήποτε έντυπα επειδή όλες οι πληροφορίες που απαιτούνται είναι σε ανοικτή γραμμή ήδη διαθέσιμες. Ένας καταναλωτής που επιτρέπει την πρόσβαση φορέων παροχής υπηρεσιών έκτακτης ανάγκης στο προσωπικό αρχείο υγείας τους θα αποφύγουν ενδεχομένως τους αχρείαστους κινδύνους και καθυστερήσεις στην επεξεργασία των παθολόγων και των λειτουργούς έκτακτης ανάγκης που πλέον θα έχουν πρόσβαση στις ενημερωμένες κλινικές πληροφορίες των ασθενών αυτών.[14]



Σχήμα 2.16 – RIM πρόσβαση σε δεδομένα υγείας



Σχήμα 2.17 – ePHR εισαγωγή δεδομένων



Σχήμα 2.18 – ePHR γραφική αναπαράσταση

2.6 Σύστημα GNUMED

Το GNUmed είναι μια ομάδα εν ενεργεία γιατρών και προγραμματιστών από όλο τον κόσμο, δεσμευμένοι να παρέχουν μια καλύτερη, ελεύθερη λύση λογισμικού για την κοινοτική ιατρική. Χρησιμοποιώντας την τεχνολογία, το λογισμικό GNUmed θα αρχίσει την αρχειοθέτηση, αλλά θα καλύψει τελικά όλες τις πτυχές της ιατρικής πρακτικής, και θα διασυνδεθεί με το λογισμικό τρίτων άλλων εφαρμογών. Από τεχνικής άποψης, προσπαθεί να κάνει τα πράγματα «καθαρά» Αυτήν την περίοδο, το GNUmed προσεγγίζεται μέσω των επιχειρησιακών αντικειμένων που εφαρμόζονται σε Python που παρέχουν και κυβερνούν την άμεση πρόσβαση στο PostgreSQL RDBMS, αλλά το GNUmed θα είναι σε θέση επίσης να συνδέεται με διάφορους άλλους τύπους πρόσβασης βάσεων δεδομένων όπως RDBMS ή LDAP.[15]. Μια καταπληκτική παρουσίαση του συστήματος μπορεί να βρει κανείς στην

πιο κάτω ιστοσελίδα

<http://www.flickr.com/photos/gnumed/sets/72157594477873670/show/>

Το GNUmed χωρίζει καθαρά τις ιατρικές πτυχές (τήρηση αρχείων) από τις διοικητικές πτυχές (τιμολόγηση, αποθήκευση) μιας ιατρικής πρακτικής. Με αυτόν τον τρόπο επιτρέπει στον GNUmed να είχε ένα ευρύτερο κοινό από χρήστες σε διαφορετικές αρμοδιότητες. [15]

Το GNUmed είναι κατάλληλο για οποιοδήποτε παροχέα / λειτουργό υγειονομικής περίθαλψης που ενδιαφέρεται για την διατήρηση ενός υγιούς και περιεκτικού ιατρικού αρχείου. Είναι αυτήν την περίοδο σε λειτουργία με το PST και τους φυσιοθεραπευτές. Το GNUmed λειτουργεί ακίνδυνα στα δίκτυα, μερικά με πολλούς χρήστες και άλλα με πολύ ελάχιστους, και υποστηρίζει την ασφάλει, εξ' αποστάσεως πρόσβαση. Επίσης λειτουργεί σε έναν προσωπικό υπολογιστή, που το καθιστά πιθανό να εξεταστεί αρχικά το λογισμικό από τον πιθανό χρήστη, και έπειτα αν του αρέσει μπορεί να εφαρμοστεί και σε μικρής κλίμακας ιατρικές υπηρεσίες όπως είναι οι αγροτικές ή μειονεκτούσες περιοχές με περιορισμένη υποδομή. [15]

Το GNUmed δεν προορίζεται αυτήν την περίοδο σαν πληροφοριακό σύστημα νοσοκομείων. Διατίθεται δωρεάν και είναι ανοικτό λογισμικό (FOSS). Το GNUmed προορίζεται, εντούτοις, να διασυνδεθεί καλά με τέτοια συστήματα. Μπορούν επίσης να υπάρξουν μερικά τμήματα νοσοκομείων που να επιτρέπουν τη χρήση GNUmed. Όπως αναφέραμε δεν θεωρείτε όμως ικανό να καλύψει τουλάχιστον αυτή την στιγμή όλες τις ανάγκες ενός νοσοκομείου. [15]

Το GNUmed στοχεύει να επιτύχει την ευρύτερη πιθανή δυνατότητα χρήσης. Αυτό εξηγεί την επιθυμία να διατηρηθεί ο εννοιολογικός χωρισμός μεταξύ αυτού που σήμερα είναι πραγματικό σε σχέση με το αρχείο υγείας ενός ασθενή. Από την μια η υγεία και από την άλλη η διοίκηση και οι διαδικασίες εργασίας (συμπεριλαμβανομένης της τιμολόγησης) που την περιβάλλουν. [15]

Το GNUmed έχει εγκατασταθεί στο Unix, GNU/Linux, τα Windows και τα συστήματα Macs (εντούτοις η έκδοση της MAC εξαρτάται από ένα τεχνικό κομμάτι για να λειτουργήσει). Η χρήση των νεώτερων (πειραματικών) εκδόσεων προτείνεται μόνο για τους υπεύθυνους για την ανάπτυξη, και άλλους με λίγο περισσότερη προηγμένη εμπειρία με τους υπολογιστές ή/και με τον προγραμματισμό. Δεδομένου ότι η κοινότητα του GNU Med αυξάνεται, οι δημιουργία νέων εκδόσεων γίνεται τώρα με ένα σχετικά καλό ρυθμό. [15]

Το GNUmed, είναι ελεύθερο λογισμικό. Εάν κάποιος έχει πρόσβαση στον κώδικα θα μπορούσε να τον τροποποιήσει, να δημιουργήσει κάτι νέο και έπειτα να το κατοχυρώσει το νέο πρόγραμμα σαν δικό του. [15] Μια καταπληκτική παρουσίαση του συστήματος μπορεί να βρει κανείς στην πιο κάτω ιστοσελίδα [25]
<http://www.flickr.com/photos/gnumed/sets/72157594477873670/show/>

2.7 Σύστημα Hospitals

Το Hospitals είναι ένα πληροφοριακό σύστημα νοσοκομείων όσο αφορά τις διαδικασίες νοσοκομείων. Είναι ένα λογισμικό μοντέλου πελάτη εξυπηρετητή στο οποίο οι εργασίες των κεντρικών υπολογιστών είναι να λειτουργούν ως κεντρική μονάδα που αποθηκεύει όλες τις πληροφορίες και οι πελάτες είναι οι μονάδες που τροφοδοτούν τις πληροφορίες στον κεντρικό υπολογιστή. [16]

Ο κεντρικός υπολογιστής του Hospitals χρησιμοποιεί το λειτουργικό σύστημα Linux και το PostgreSQL ως σύστημα βάσης δεδομένων. Τα Linux και η PostgreSQL είναι προγράμματα ανοικτού κώδικα και διαθέσιμα για κατέβασμα στο διαδίκτυο. Το λογισμικό πελατών αναπτύσσεται με τη χρήση της JAVA και μπορεί να χρησιμοποιηθεί με τα windows98, windows ME, τα windows2000, windows, Linux και άλλα λειτουργικά συστήματα που έχουν εγκατεστημένη την εικονική μηχανή της JAVA. [16]

Η περιγραφή των χαρακτηριστικών γνωρισμάτων και των οθόνων έκδοσης Hospital OS 3.0 συλλαμβάνει μπορεί να χωριστούν στις πιο κάτω κατηγορίες. [16]

IR HospitalOS - HospitalOS Community Edition version 3.06.050406b

System View Tab Tool General Patient IPD Order Print Reverse Help

HN VN AN

Current Service Point Previous Service Point Insurance Plan Dx:

Submit To 01. Registration Doctor Send Unlock

Patient List by Service Point \ 1. Patient Data \ 2. Visit \ 3. Vital Sign \ 4. Order List \ 5. Diagnosis \ 6. Cashier \

General Data \ Other Data \

General Data

Title Name Last Name

PID

Gender male ☐ Real Birthdate Age Year

Status single Blood Type unknown

Occupation 000:unemployment

Nationality Austria Race Thailand

Religion Buddhism Education unknown

Father Name

Mother Name

Spouse Name

Contact Person Name

Address

Status ☒ Family Head ☐ Family Member

No. Group No. Street

Telephone No.

Province 10

District 01

Village 03

Contact Person Address

Relationship unknown Gender male

No. Group No. Street

Telephone No.

Province 10

District 01

Village 03

+ Search - Save

Tuesday 2 May 2006 12:13:50:49

Σχήμα 2.19 – Διεπαφή Χρηστή

Service Point 01. Registration Doctor

No. of Patients in Queue 3 person

Lock	Allergy	HN	VN	Name-Last Name	Time	Service Point
☒		000000010	049000011	Mr. Patrick Wilson	03 May 2549 16:54	01. Registrat
☒		000000011	049000012	Miss Emmy Rossum	03 May 2549 16:56	01. Registrat
☒		000000012	049000013	Mr. Gerk Poto	03 May 2549 16:57	01. Registrat

Σχήμα 2.20 – Διαχείριση ροής της δουλειάς

Patient List by Service Point | 1. Patient Data | 2. Visit | 3. Vital Sign | 4. Order List | 5. Diagnosis | 6. Cashier

Insurance Plan

All Insuranc...

Insurance Plan
Bupa
Compensation Fund (SSO)
Handicapped
Self Pay
Senior Citizen
Social Security
Student
UC Room Disc. 100 %
UC Room Disc. 50 %

List of Patient Available Plans

Card ID	Insurance Plan

Visit Data

Selected Insurance Plan

Card ID	Insurance Plan	Discount
323005612153	Compensation Fund (SSO)	Compensation Fund for ...

Insurance Plan Compensation Fund (SSO)

Card ID: Credit Limit: 35000 Baht
 Issue Date: Expiration Date:
 Primary Hospital:
 Secondary Hospital:

Σχήμα 2.21 – Διαχείριση ασφάλειας

Patient Referral

Refer Result | Refer In/Out

Patient Data

Refer No. HN
 Name
 Gender Age
 Hospital
 Near By Location

Referral Purpose

☐ Receive for Treatment ☐ Lab
☐ Observe ☐ Result

Patient Referral Type

☒ Refer Out ☐ Refer In

Other Details

Patient Treatment Data

Family History
 Current History
 Lab History
 Initial Diagnosis
 Curable Disease
 Reason of Referral
 Refer Doctor
☐ Infectious
☐ Reported ☐ Not Reported

Σχήμα 2.22 – Παραπομπή ασθενή

Patient Referral

Refer Results | Refer In/Out

Search Patient Referral

HN 0000012 Refer No.

From 03/05/2006 To 03/05/2006

HN	RN	Date	In/Out

Refer Result Form

Refer Date 03/05/2006

Refer No. R12 HN 000000012

Name Gerik Poto

Hospital HT, suwihans

Result Date 03/05/2006

Last Diagnosis xxxxx

Order Provided

Lab Results xxxxx

Follow Up Treatment xxxxxx

Analyzed by

Σχήμα 2.23 – Παραπομπή ασθενή(συνεχεία)

ER HospitalOS - HospitalOS Community Edition version 3.06.050406b

System View Tab Tool General Patient IPD Order Print Reverse Help

HN 0000002 VN AN 9000005 Mr. Peter Chan Age -509 Year 10 month 23 Day

Current Service Point 07. ER Previous Service Point 07. ER Insurance Plan Self Pay

Submit To 01. Registration Doctor Send Unlock

Patient List by Service Point \ 1. Patient Data \ 2. Visit \ 3. Vital Sign \ 4. Order List \ 5. Diagnosis \ 6. Cashier \

For Nurse

Vital Sign

Transaction Date and Time

Transaction Date and Time
24 May 2549 17:11:25

Description

Weight 67 Kg.
 Height 174 cm.
 Blood Pressure 120 / 80 m...
 Temperature 38 C
 Pulse Times/Minute
 Respiration Times/Minute
 BMI 22.13

☐ Pregnant Patient Order Hi...

General Symptom

Transaction Date and Time

Transaction Date and Time
24 May 2549 17:12

Primary Symptom

Headache with fever

Current Symptom

Fever, Coughing and Headache

Physical examination is completed

Σχήμα 2.24 – Τμήμα 1^{ον} βοηθειών

ER HospitalOS - HospitalOS Community Edition version 3.06.050406b

System View Tab Tool General Patient IPD Order Print Reverse Help

HN: 0000002 VN: 9000005 Mr. Peter Chan Age -509 Year 10 month 23 Day

Current Service Point: 07. ER Previous Service Point: 07. ER Insurance Plan: Self Pay Dxi: Fever

Submit To: 01. Registration Doctor: Send Unlink

Patient List by Service Point | 1. Patient Data | 2. Visit | 3. Vital Sign | 4. Order List | 5. Diagnosis | 6. Cashier

For Nurse

Vital Sign

Transaction Date and Time

Transaction Date and Time
24 May 2549 17:11:25

Description

Weight: 67 Kg
 Height: 174 cm
 Blood Pressure: 120 / 80 mmHg
 Temperature: 38 C
 Pulse: Times/Minute
 Respiration: Times/Minute
 BMI: 22.13

☐ **Pregnant Patient** Order HL...

General Symptom

Transaction Date and Time

Transaction Date and Time
24 May 2549 17:12

Primary Symptom: Headache with fever

Current Symptom: Fever, Coughing and Headache

For Doctor

Dxi: Fever

Note:

Physical examination:

Part of Body	Results
Throat	Slightly red

Drug Allergy Data

☐ Deny Drug Allergy

Drug Name	Allergic Symptom
Aspirin, -tab (300mg)	
Diazepam, -tab (10mg)	

Physical examination is completed

Wednesday 24 May 2006 17:13:59

Σχήμα 2.25 – Τμήμα 1^{ον} βοηθειών (συνεχεία)

Order History

Search Order History

Common Name

☒ Search by Group Group From To

Order Subgroup

VN	Order Date	Name	Qu...	Status	Dosage
049000...	24 May 2549	Amoxycilline .-Cap (2...	20	confir...	OR 1 Tablet TID PC 20...
049000...	24 May 2549	Paracetamol .-tab (50...	20	confir...	OR 2 Tablet Q46F 20 T...

Σχήμα 2.26 – Προηγούμενες αγωγές

Dr. Gregory House - HospitalOS Community Edition version 3.06.050406b

System View Tab Tool IPD Order Print Help

HN: 00000006 VN: AN: 49000002 Miss Christine Daee Age -528 Year 10 month 3 Day

Current Service Point: Cashier Previous Service Point: 05. Pharmacy Insurance Plan: Bupa Dx: Retu

Submit To: 01. Registration Doctor: Send Unlock

Patient List by Service Point \ 1. Patient Data \ 3. Vital Sign \ 4. Order List \ 5. Diagnosis \

Search Order Item

☐ Start with ☒ Contain Set Prev

Search

☐ Group Lab

Patient Order List

☒ All ☐ Between Date: 04/05/2006 To: 04/05/2006

Select All LAB ☐ List Cancelled Orders ☐ Continuous

#	Name	Unit	Price	Status	Req...	Ongoi...
1	Sugar	1	60.00	reported	☐	
2	Urine Sugar	1	20.00	reported	☐	
3	Fluid sugar	1	60.00	executed	☐	

Order :Urine Sugar

Confirm Execute Cancel

Thursday 4 May 2006 14:50:41

Σχήμα 2.27 – Αποτελέσματα αναλύσεων


Lab Results
✖

Result Reported \
Waiting for Lab Result \

List	Order Date	Name	Result	Unit	Norm...
CBC	03/05,'2549,10...	Hct.	20	%	
		Hb	10	gm%	
		WBC Count	5	/cubic mm.	
		RBC Count	4	/cumm	
		Platletet estima...	3	/Oil field	
		Platletet count	5	/cubic mm.	
		RBC Morphology	5		
		WBC Diff	8	-	
		Mp	5		
Fluid sugar	03/05,'2549,10...	Fluid sugar	23		
Fern test	03/05,'2549,10...	Fern test	67		

Save

Σχήμα 2.28 – Αποτελέσματα αναλύσεων (συνεχεία)

ER HospitalOS - HospitalOS Community Edition version 3.06.050406b

System View Tab Tool General Patient IPD Order Print Reverse Help

HN:00000002 VN:00000005 Mr. Peter Chan Age:509 Year 10 month 23 Day 0

Current Service Point:07. ER Previous Service Point:07. ER Insurance Plan:Self Pay Dx:

Submit To: 01. Registration Doctor: Send Unlock

Patient List by Service Point | 1. Patient Data | 2. Visit | 3. Vital Sign | 4. Order List | 5. Diagnosis | 6. Cashier

Search Order Item

Start with: Contain Set Prev

Para Search

Group: Lab

Name	Group Name
Paracetamol -inj (300mg/2ml)	Standard Drugs Expenses
Paracetamol -syr (50ml)	Standard Drugs Expenses
Paracetamol -tab (500 mg.)	Standard Drugs Expenses
Paracetamol+Orphenadine	Standard Drugs Expenses
Buffy coat preparation	Lab
Fluid preparation	Lab
India ink preparation	Lab
KOH Preparation	Lab
Sperm preparation	Lab

Order List

All Between Date: 24/05/2006 To: 24/05/2006

Select All Display All List Canceled Orders

B	Name	UNIT	Price	Status	Requested
0	Doctor Fee - Standard Hrs.	1	40.00	unconfirmed	
0	Paracetamol -tab (500 mg.)	20	10.00	unconfirmed	

Order: Paracetamol -tab (500 mg.)

Special Instruction

OR:Take 2 Tablet Warning

Q46F:Every 4-6 hrs, when having fever Description:Do not continuously use longer than 7 days

No. of Days: Day Dispense Quantity: 20 Tablet

Confirm Execute Dispense Save Cancel

Wednesday 24 May 2006 17:09:35

Σχήμα 2.29 – Εντολές του ιατρού για ένα ασθενή

Appointment

Search Appointment

All From: /05/2006 To: /05/2006

Current Patient: 02. Screening Search

Name	Service Point	Date
------	---------------	------

Appointment Details

HN: 000000012

Name: Mr. Gerik Poto

Appointment Date: 08/05/2006 To: 5/2006

Appointment Time: 8:00 e.g. 09:00

Appointment Type: Follow Up

Service Point: 02. Screening

Doctor Name: Dr. Gregory House

Detail:

Auto Visit: ☒

Appointment Status: 00000000

+ - Preview Print Card Save

Σχήμα 2.30 – Ραντεβού

Appointment

Search Appointment

☐ All ☒ From 24/05/20 To 30/05/20
☐ Current Patient 07. ER

HN	Name	Date	Service...
0000...	Mr. Peter Chan	29 May 2006,09:00	07. ER

Appointment Details

HN: 000000002
 Name: Mr. Peter Chan
 Appointment Date: /05/2006 To 5/2006
 Appointment Time: 9:00 e.g. 09:00
 Appointment Type: Follow Up
 Service Point: 07. ER
 Doctor Name: Dr. James Wilson
 Detail:
 Appointment Status: รอการนัด

มีผู้ป่วยอยู่ในคิวนัดทั้งสิ้น 1 คน

Σχήμα 2.31 – Ραντεβού (συνεχεία)


Visit History


Visit History

VN	เวลาเข้าสู่ระบบ	Discharge Time	Status
0490000005	23 April 2549 18:56		in process
0490000002	21 April 2549 17:42	23 April 2549 18:37	wait icd

Visit History Details

Service Poi...	Send to Doctor	Sending Time	Sta...	Start	End
07. ER		23 April 2549 18:56	2	18:56...	2549-...
07. ER		24 April 2549 11:18	2		17:17...
01. Registr...		24 May 2549 17:17	2	17:17...	

OK

Σχήμα 2.32 –Ιστορικό επισκέψεων

Order History

Search Order History

Common Name

☒ Search by Group Group From To

Order Subgroup

VN	Order Date	Name	Qu...	Status	Dosage
049000...	24 May 2549	Amoxycilline .-Cap (2...	20	confir...	OR 1 Tablet TID PC 20...
049000...	24 May 2549	Paracetamol .-tab (50...	20	confir...	OR 2 Tablet Q46F 20 T...

Σχήμα 2.33 –Ιστορικό εντολών ιατρού

Patient List by Service Point | 1. Patient Data | 2. Visit | 3. Vital Sign | 4. Order List | 5. Diagnosis | 6. Cashier

Search Order Item

☐ Start with ☒ Contain

☐ GroupGroup

Name	Group Name
CBC	Lob

Patient Order List

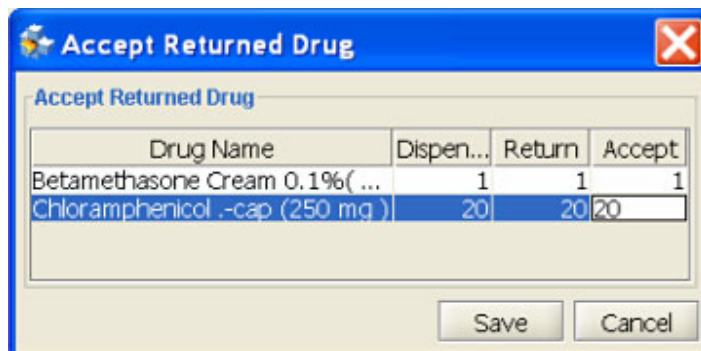
☒ All ☐ Between Date To ☐ List Cancelled Orders

#	Name	Unit	Price	Status	Requ...
1	Paracetamol .-tab (500 mg.)	20	10.00	unconfirmed	☐
2	Ibuprofen .-Syr (100mg./5ml.) , 60ml.*	1	8.00	unconfirmed	☐
3	Bromhexine .-tab (8 mg.)	20	2.00	unconfirmed	☐
4	CBC	1	60.00	unconfirmed	☐

ηλεκτρονική Order :CBC

1

Σχήμα 2.34 –Ιστορικό εντολών ιατρού (συνεχεία)

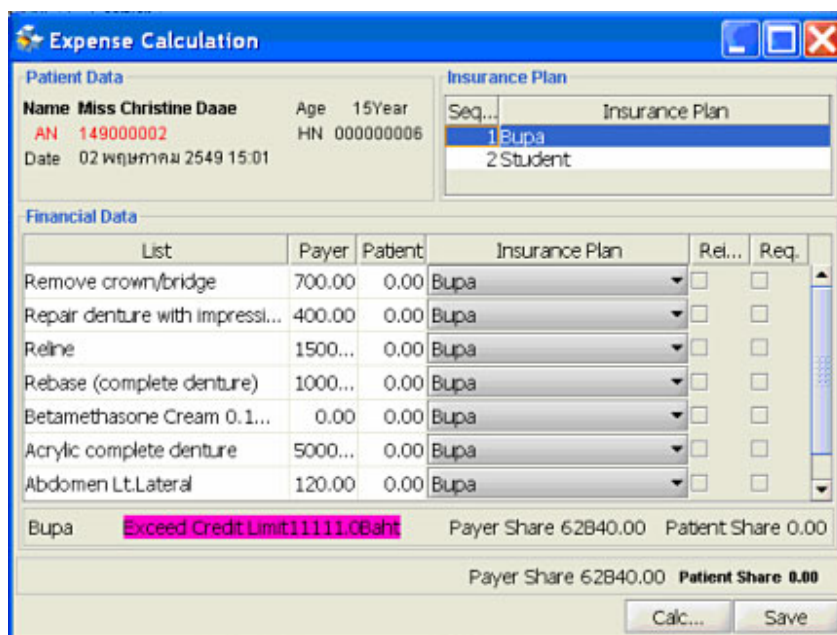


Accept Returned Drug

Drug Name	Dispen...	Return	Accept
Betamethasone Cream 0.1%(...	1	1	1
Chloramphenicol .-cap (250 mg)	20	20	20

Save Cancel

Σχήμα 2.35 – Αναπλήρωση συνταγής



Expense Calculation

Patient Data
 Name **Miss Christine Daae** Age 15Year
 AN 149000002 HN 000000006
 Date 02 Ιωημνηηηηη 2549 15:01

Insurance Plan
 Seq... Insurance Plan
 1 Bupa
 2 Student

Financial Data

List	Payer	Patient	Insurance Plan	Rel...	Req.
Remove crown/bridge	700.00	0.00	Bupa	☐	☐
Repair denture with impressi...	400.00	0.00	Bupa	☐	☐
Reline	1500...	0.00	Bupa	☐	☐
Rebase (complete denture)	1000...	0.00	Bupa	☐	☐
Betamethasone Cream 0.1...	0.00	0.00	Bupa	☐	☐
Acrylic complete denture	5000...	0.00	Bupa	☐	☐
Abdomen Lt.Lateral	120.00	0.00	Bupa	☐	☐

Bupa **Exceed Credit Limit11111.00aht** Payer Share 62840.00 Patient Share 0.00

Payer Share 62840.00 Patient Share 0.00

Calc... Save

Σχήμα 2.36 – Υπολογισμός εξόδων

Patient List by Service Point | 1. Patient Data | 2. Vital Sign | 4. Order List | 6. Cashier

Insurance Plan

Card ID	Insurance Plan	Discount
	Bupa	UC - within CUP
	Student	UC - within CUP

Insurance Plan:

Card ID:

Issue Date: Expiration Date:

Credit Limit: 111111 Baht

Secondary Hospital:

Patient Available Plans

Card ID	Insurance Plan	Discount
	Student	

Invoice Item

Invoice

Invoice No	Invoice Date	Invoice Time	Invoice Description
001	02 May 2549	17:04	Medical Supply Expenses
002	02 May 2549	17:04	Medical Supply Expenses
003	28 April 2549	17:01	Medical Supply Expenses

Insurance Plan	Payer	Patient
Student	51900.00	0.00
Bupa	10940.00	0.00

Payment Item

Payment Date	Paid	Balance
02 May 2549 17:04	0.00	0.00
02 May 2549 17:04	0.00	0.00
28 April 2549 17:01	0.00	0.00

Outstanding Balance: 0.0 Baht

Total Outstanding Balance: 0.0 Baht

Payment

Financial Disc...

Σχήμα 2.37 – Αποπληρωμή εξόδων

การชำระเงิน

HN 000000006 VN 149000002 Date 03 May 2549 13:54

Name Miss Christine Daee Age -5 ปี Dx:

Item	Insurance Plan	Payer	Patient
Medical Supply Expenses	Self Pay	0.0	48.00

Total 48.0

Paid Payment Balance Change

Dx in Receipt:

Preview Print Receipt Save

Σχήμα 2.38 – Αποπληρωμή εξόδων (συνεχεία)

Lab HospitalOS - HospitalOS Community Edition version 3.06.0504066

System View Tab Tool Print Help

HN\0000009 VNAN\9000010 Miss Lara Craft Age -513 Year 10 month 3 Day 0

Current Service Point 05, Pharmacy Previous Service Point 01, Registration Insurance Plan Compensation Fund (... Dxc)

Submit To 01, Registration Doctor Send Unlock

Locked by Pharmacy HospitalOS

Patient List by Service Point | 1. Patient Data | 3. Vital Sign | Lab

Lab Item

Item	History	Item	Order Date	Name	Result	Unit	Normal Value
Urine Sugar	Refer... Status	Urine Sugar	03/05/2549,16:21	Urine Sugar	10		
Blood sugar <FBS>	Refer... Status	Blood sugar <FBS>	03/05/2549,16:21	FBS	20	mg%	
Cholesterol (total)	Refer... Status	Cholesterol (total)	03/05/2549,16:21	Cholesterol	180	mg%	
LDL - cholesterol	Refer... Status	LDL - cholesterol	04/05/2549,11:01	LDL - cholesterol			

Refer... Cancel Report Suspe... Delete... Save Print

Σχήμα 2.39 – Αποτελέσματα εργαστήριου

X-ray HospitalOS - HospitalOS Community Edition version 3.06.050406b

System View Tab Tool Print Help

HN 00000009 / VN AN 90000010 Miss Lara Craft Age -513 Year 10 month 3 Day 0

Current Service Point 07. ER Previous Service Point 05. Pharmacy Insurance Plan Compensation Fund (... Dic:

Submit To 01. Registration Doctor Send Unlock Locked by Dr. Gregory House

Patient List by Service Point 1. Patient Data 3. Vital Sign X-ray

Item History

X-ray Item

Code	Name
0010553	Chest AP Upright

Description

Xn

Position Lateral

Lateral	X-ray Position	KV
Right	Mortise	1.0
		MA 2.0
		Second 3.0
		MAS .00
		FFD 5.0

Quantity 2

Price 0

X-ray staff details executed by Bill Randol

Result List

Code	Name
------	------

X-ray Results

Send Save

Σχήμα 2.40 – Αποτελέσματα ακτινογραφιών

Σχήμα 2.41 – Αναφορά ατυχήματος

2.8 Σύστημα ndivoHealth

Η τεχνολογία πίσω από το MyOSCAR είναι βασισμένη στο σύστημα Indivo έργο που ξεκίνησε από το Χάρβαρντ-MIT πρόγραμμα πληροφορικής Νοσοκομείων για Παιδιά (<http://www.chip.org/research/ping.htm>). Η προέλευση αυτού του προγράμματος βασίζεται στη δημοσίευση του 1994 της ομάδας (Guardian Angel manifest), στο οποίο προτάθηκε ότι πρέπει να γίνει κάποιο είδος αποκέντρωσης του λογισμικού που παρέχει τις πληροφορίες υγείας από τον πλήρη έλεγχο των γιατρών και των νοσοκομείων και να εστιάζοταν περισσότερο στο άτομο. Το μανιφέστο υποστήριξε ότι επειδή τα τρέχοντα συστήματα πληροφοριών υγείας «χτίστηκαν για διευκόλυνση των παροχών υγειονομικής περίθαλψης, «τα αρχεία του ασθενή έτειναν να τεμαχιστούν, και ήταν μερικές φορές ελλιπές, ανακριβές, ή απρόσιτα. Το όραμα ήταν ότι επρόκειτο να ανεπτυχθεί λογισμικό που

θα βοηθούσε να παρακολουθήσει, να διαχειριστεί, και να ερμηνεύσει το ιστορικό υγείας του κάθε ασθενή.

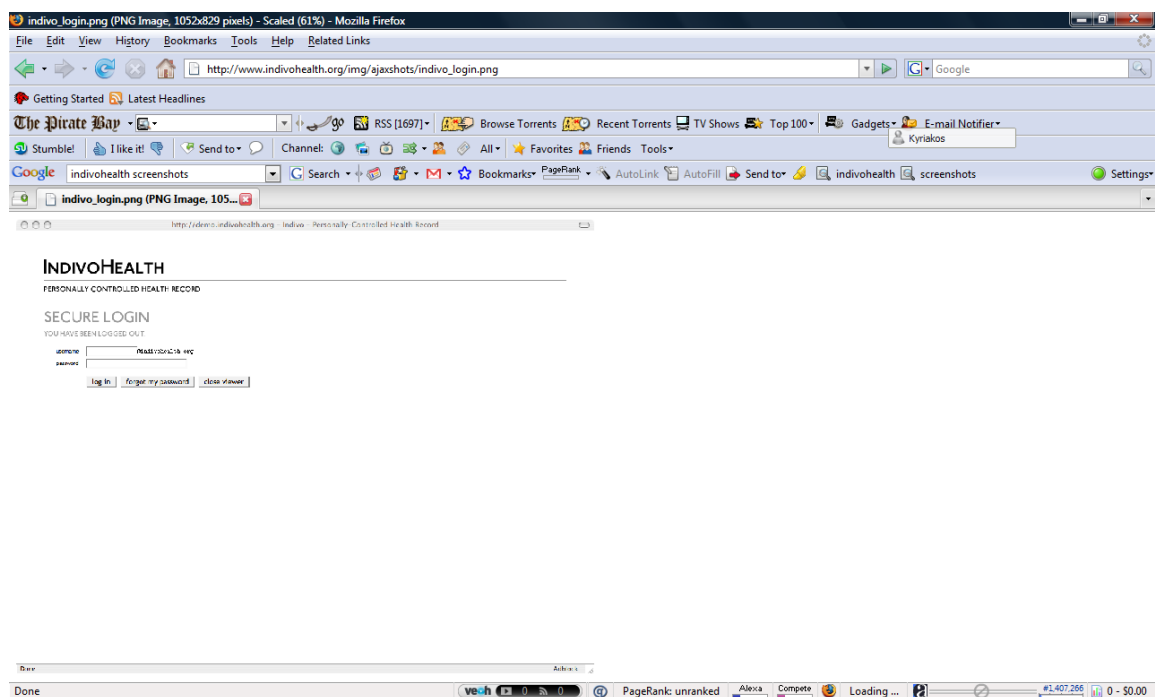
Αυτές οι έννοιες και ιδέες είναι που εξελίσσονταν κατά τη διάρκεια της επόμενης δεκαετίας, όπως και το καινοτόμο λογισμικό που αναπτύχθηκε κατά συνέπεια. Το 1998, η ομάδα άρχισε τα προγράμματα που επιτρέπουν στους ασθενείς να δουν κάποιο στοιχείο του ηλεκτρονικού ιατρικού αρχείου τους. Κατόπιν, το 2001, η εστίαση μετατοπίστηκε στο λογισμικό που επέτρεψε τον πιο συστηματικό έλεγχο. Ο αρχικά αποκαλούμενος IndivoHealth γίνεται το πρώτο σύστημα προσωπικά ελεγχόμενου αρχείου υγείας παγκοσμίως, μια ελεύθερη, εφαρμογή που χρίζετε με βάση τα πρότυπα που καθορίστηκαν από κοινότητα του ανοικτού λογισμικού.

Οι αρχές που διέπουν την ανάπτυξη του ηλεκτρονικού φάκελου ασθενή γράφτηκαν το 2001 και αναφέρονται στο Έγγραφο BMJ. Σημαντικά σημεία είναι :

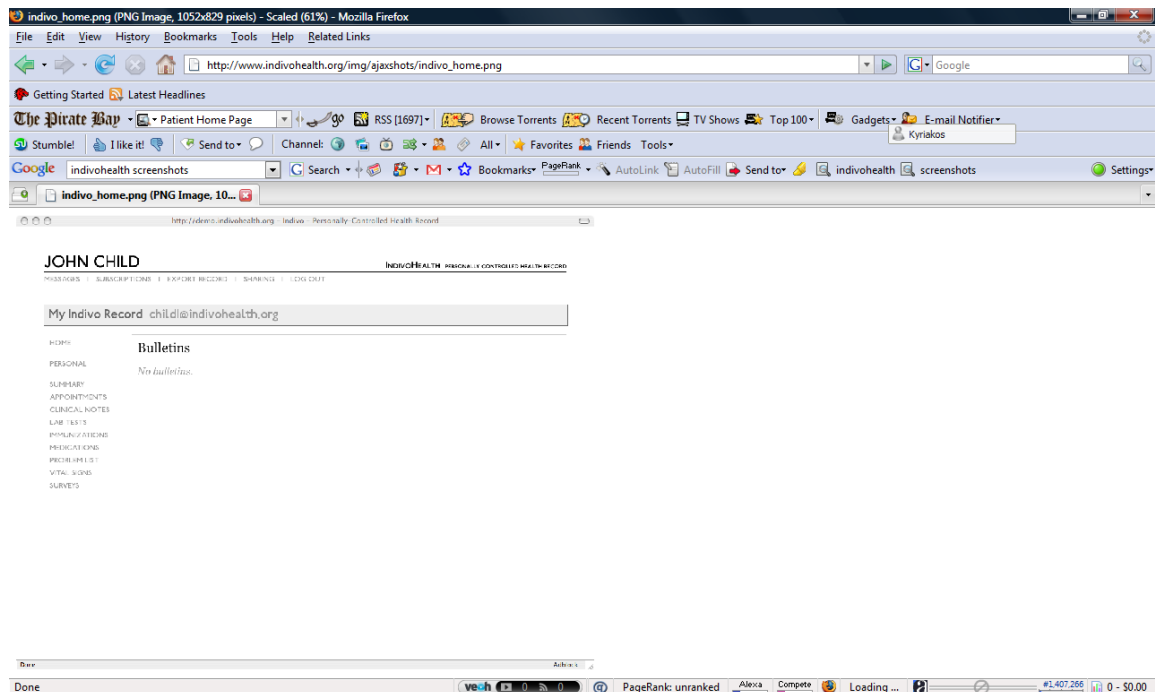
- Σε ένα ηλεκτρονικό ιατρικό αρχείο τα συστήματα πρέπει να σχεδιαστούν έτσι ώστε να μπορούν να ανταλλάξουν όλα τα αποθηκευμένα στοιχεία τους σύμφωνα με τα δημόσια πρότυπα
- Δίνοντας στους ασθενείς τον έλεγχο των δικαιωμάτων πρόσβασης όσο αφορά την επισκόπηση του αρχείου τους αλλά και δικαιώματα δημιουργίας, συλλογής, τροποποίησης και διαγραφής είναι ένας τρόπος για να εξασφαλιστεί η πρόσβαση των ασθενών στα δικά τους αρχεία μόνο αλλά εξασφαλίζεται και η διαφύλαξη της ιδιωτικής τους ζωής.
- Πολλά υπάρχοντα συστήματα ηλεκτρονικού φάκελου ασθενή διασκορπίζουν ιατρικά δεδομένα υιοθετώντας ασύμβατους τρόπους λήψης, επεξεργασίας, αποθήκευσης και επικοινωνίας δεδομένων.
- Συστήματα καταγραφής πρέπει να είναι σε θέση να δέχονται δεδομένα όπως ιστορικά, ακτινολογικά, εργαστηριακά κτλ. Από πολλαπλές πηγές που περιλαμβάνουν γραφεία ιατρών, υπολογιστικά συστήματα νοσοκομείων, εργαστήρια ακόμα και προσωπικούς υπολογιστές των ασθενών.

- Οι καταναλωτές σήμερα διαχειρίζονται τους λογαριασμούς τράπεζας τους, τις επενδύσεις αλλά και τις αγορές τους μέσω διαδικτύου. Ως αποτέλεσμα περιμένουν ότι θα μπορούν να στραφούν στο διαδίκτυο για ιατρικές πληροφορίες μέσω της διατήρησης ιατρικών πορτοφολιών

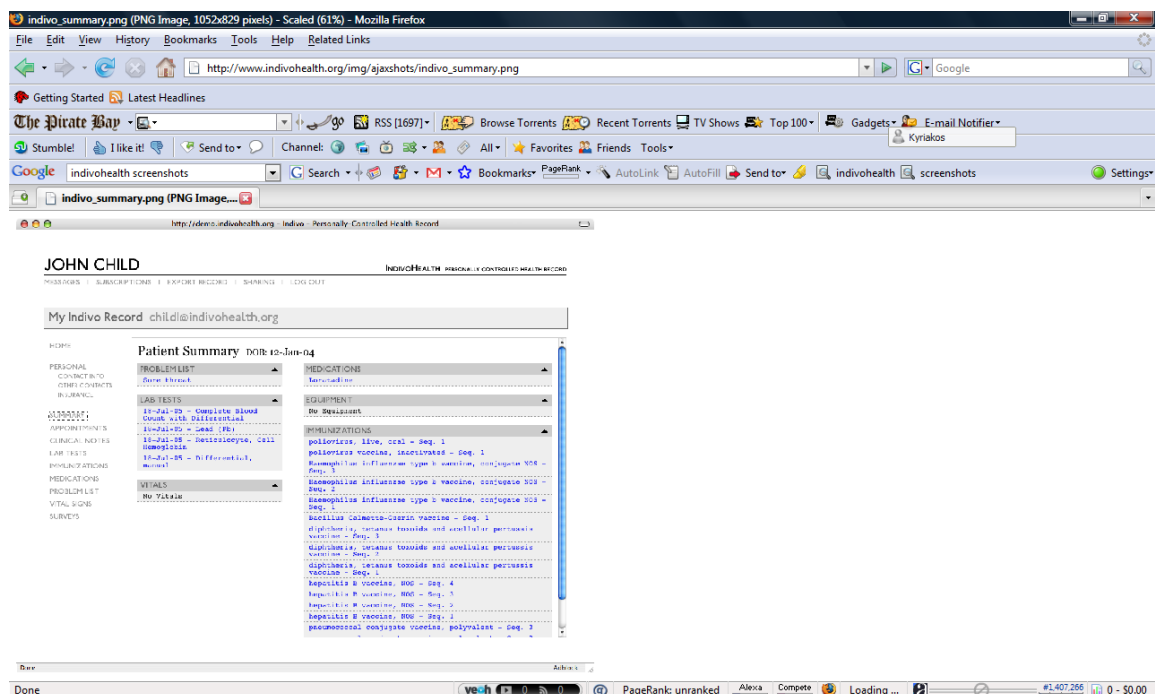
Η αρχιτεκτονική που εκπληρώνει τους στόχους που αναφέρονται πιο πάνω αναφέρεται στην ένα άρθρο του 2004 γνωστό ως «JAMIA paper». Μια επισκόπηση είναι διαθέσιμη και στο <http://indivohealth.org/pages/server>. [17,18,19,20,21,22]



Σχήμα 2.42 – Οθόνη Login



Σχήμα 2.43 – Οθόνη ασθενή



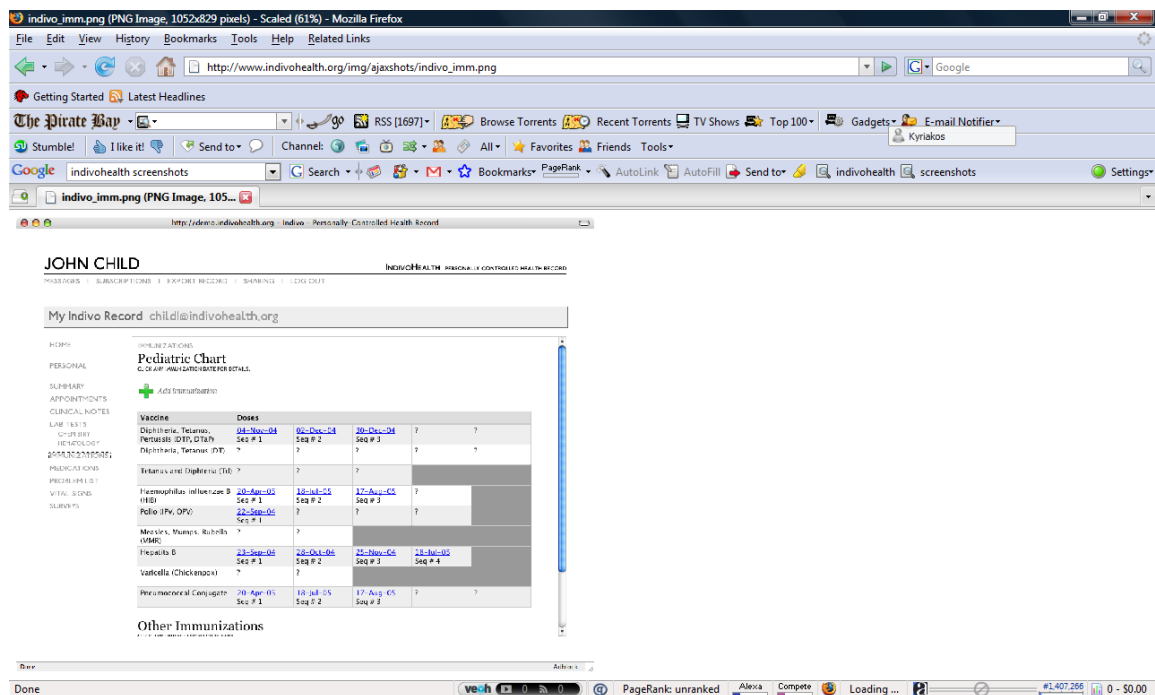
Σχήμα 2.44 – Περίληψη στοιχείων ασθενή

The screenshot displays a web browser window with the following elements:

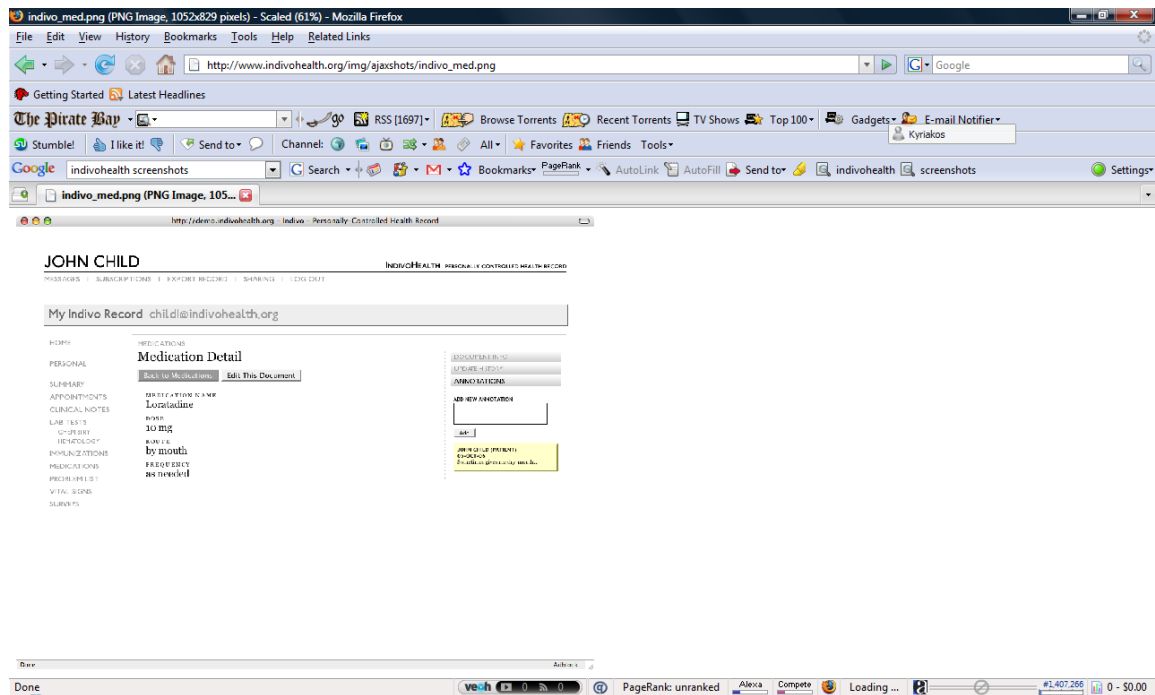
- Browser Title:** indivo_lab.png (PNG Image, 1052x829 pixels) - Scaled (61%) - Mozilla Firefox
- Address Bar:** http://www.indivohealth.org/img/ajaxshots/indivo_lab.png
- Page Content:**
 - Navigation:** Getting Started, Latest Headlines, RSS [1697], Browse Torrents, Recent Torrents, TV Shows, Top 100, Gadgets, E-mail Notifier, Kyriakos.
 - Search:** indivohealth screenshots
 - Page Title:** JOHN CHILD
 - Subheader:** indivoCHILDM - indivoCHILD - A CONTROLLED HEALTH RECORD
 - My Indivo Record:** child@indivohealth.org
 - Lab Tests:** Complete Blood Count With Differential
 - Table of Lab Results:**

Test Name	2005-07-14
Automated DIFF	NOT DONE null
WBC	8.84 k/uL
RBC	4.62
Hemoglobin	12.0
Hematocrit	35.0 %
MCV	24.5
RDW	7.9 %
MPV	24.9
PLT	11.4
RDW	28.0 %
RDW	2.49
MPV	16.2
Platelet	147.0 k/uL
- Footer:** Done, veeah, PageRank: unranked, Alexa, Compete, Loading ... #1,407,266, 0 - \$0.00

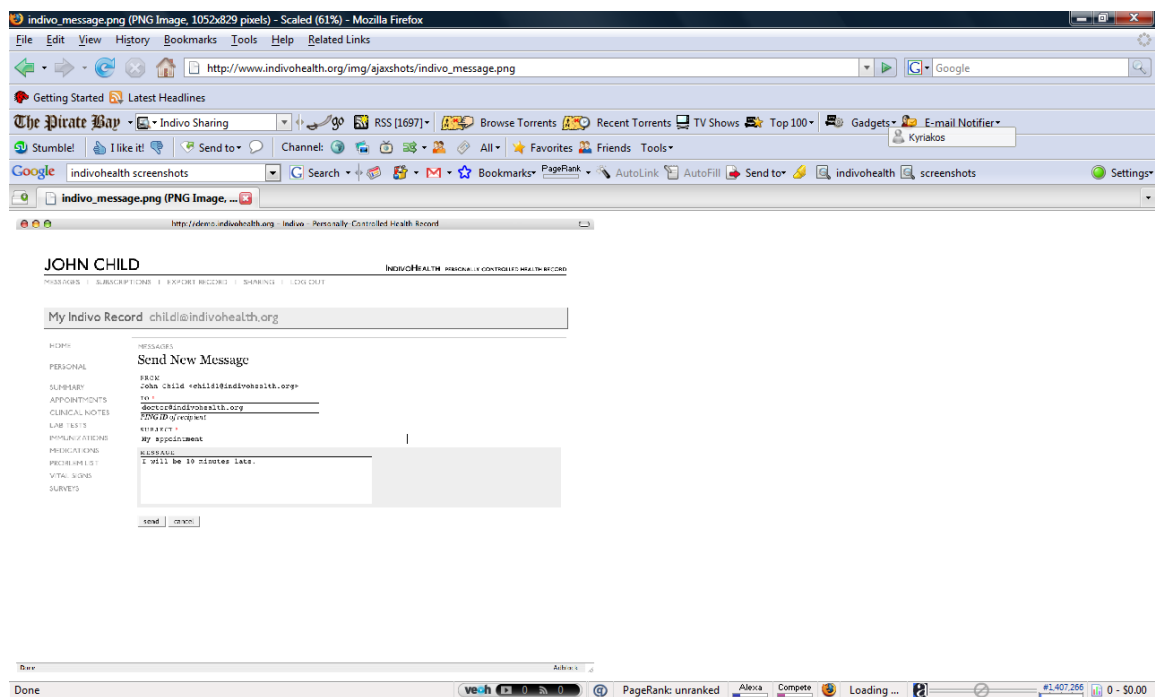
Σχήμα 2.45 – Παράδειγμα εργαστηριακών αναλύσεων



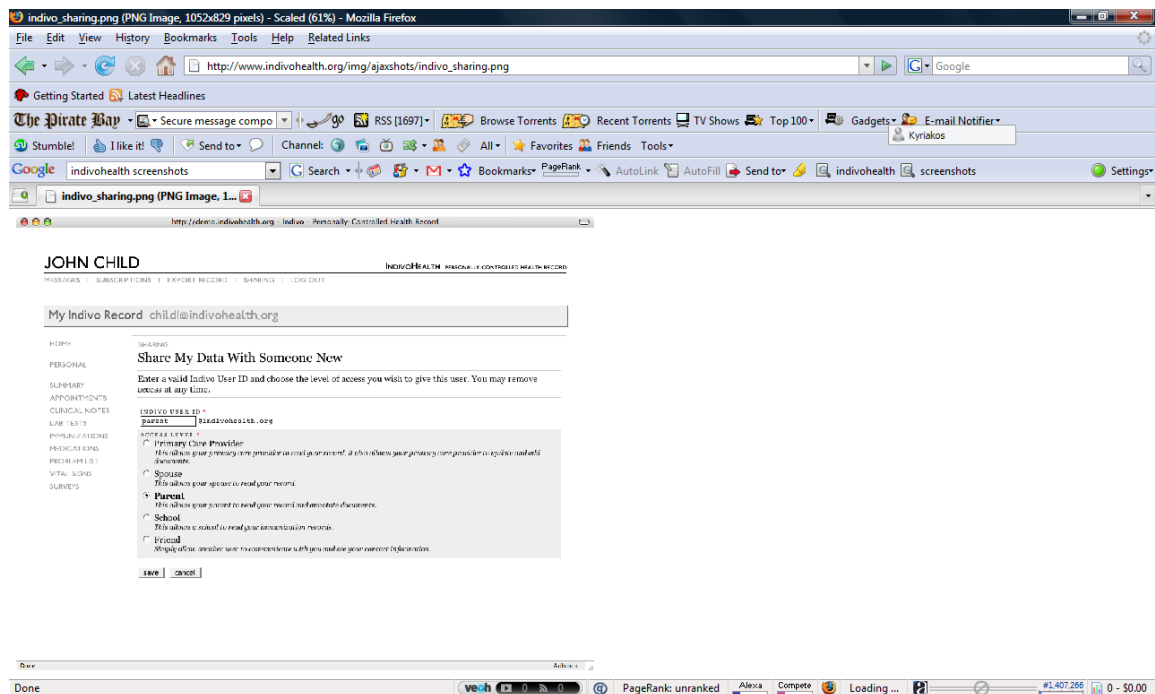
Σχήμα 2.46 – Παράδειγμα βρεφικού εμβολιασμού



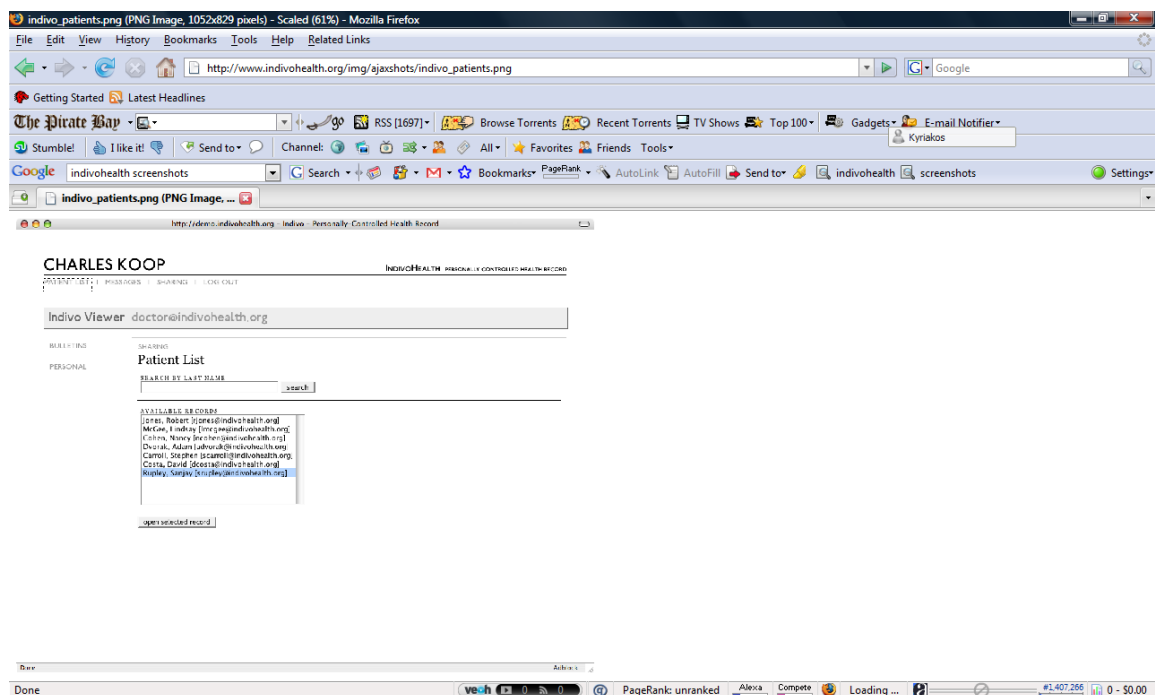
Σχήμα 2.47 – Στοιχεία φαρμακευτικής αγωγής ασθενή



Σχήμα 2.48 – Ασφαλής συγγραφή μηνυμάτων



Σχήμα 2.49 – Κοινή χρήση στοιχείων με άλλο χρήστη του σιτέματος



Σχήμα 2.50 – Λίστα ασθενών

2.9 Σύστημα openEHR

Το σύστημα openEHR είναι ένα σύνολο ανοικτών προδιαγραφών για μια ηλεκτρονική αρχιτεκτονική αρχείων υγείας (EHR) - αλλά δεν είναι μια εφαρμογή λογισμικού. Ο σκοπός του είναι να επιτρέψει τη σημασιολογική δια-λειτουργικότητα των πληροφοριών υγείας μεταξύ των συστημάτων EHR - όλα με ένα μη ειδικευμένο σχήμα, που θα βοηθήσει στην αποφεύγει του κλειδώματος των στοιχείων από τους ίδιους τους ιατρούς.

Όλες οι κλινικές έννοιες / γνώσης συλλαμβάνονται με έναν δομημένο τρόπο - γνωστό ως «αρχέτυπα» - εξωτερική όψη του λογισμικού. Οι τύποι αρχέτυπων υποστηρίζουν την καταγραφή που απαιτείται για τις κοινές κλινικές δραστηριότητες, με μερικά από τα βασικά αρχέτυπα δομικών μονάδων να περιλαμβάνουν τις παρατηρήσεις, τις αξιολογήσεις, τις οδηγίες και τις ενέργειες. Τα στοιχεία που χτίζονται σύμφωνα με αυτούς αποθηκεύονται στις EHR δομές της μεγαλύτερης σύνθεσης, οι οποίες έχουν τα αρχέτυπά τους.

Οι συνθέσεις είναι συγκρίσιμες και συνοδεύεται με ένα έγγραφο που προκύπτει από ένα κλινικό γεγονός π.χ. ένα αρχείο διαβουλεύσεων ή μια περίληψη απαλλαγής. Τα αρχέτυπα

μπορούν να είναι απλά, όπως η θερμοκρασία, η πίεση αίματος ή η διάγνωση, ή σύνθετος,. Τα αρχέτυπα περιέχουν ένα μέγιστο σύνολο στοιχείων για κάθε κλινική έννοια, συμπεριλαμβανομένων των συνοδευτικών στοιχείων που απαιτούνται όπως: το πρωτόκολλο που πρέπει να ακολουθηθεί, ή η μέθοδος μέτρησης για τα σχετικά γεγονότα και το πλαίσιο που απαιτείται για τα κλινικά στοιχεία που ερμηνεύονται ακριβώς.

Η δημιουργία των αρχέτυπων και των προτύπων είναι καθαρά ένας στόχος για τους νοσοκομειακούς γιατρούς – στο σύστημα openEHR

τα αρχέτυπα βάζουν τους νοσοκομειακούς γιατρούς στη θέση του οδηγού, επιτρέποντας σε αυτούς να δημιουργήσουν το εύρος, το βάθος και την πολυπλοκότητα του αρχείου υγείας για να ανταποκριθούν στις ανάγκες τους για την άμεση παροχή υγειονομικής περίθαλψης.

Οι συναθροίσεις των αρχέτυπων συνδυάζονται μέσα στο σύστημα openEHR με

«πρότυπα» προκειμένου να ληφθεί το σύνολο των δεδομένων που αντιστοιχεί σε έναν ιδιαίτερο κλινικό στόχο, όπως ένα συνοπτικό ή εμβρυικό αρχείο επίσκεψης/απαλλαγής. Όταν οι νοσοκομειακοί γιατροί εξετάζουν τα πρότυπα, οι πληροφορίες που λαμβάνονται μέσα σε δευτερόλεπτα έχουν νόημα και δεν απαιτούν σημαντική κατάρτιση για τους ενδιαφερόμενους νοσοκομειακούς γιατρούς να είναι σε θέση να δημιουργήσουν τα πρότυπα για τις δίκες τους ανάγκες - είτε πρόκειται για περιοχή, οργάνωση ή για κάποιο άλλο συγκεκριμένο σκοπό. Τα πρότυπα μπορούν να χρησιμοποιηθούν για να χτίσουν τις γενικές μορφές για να αντιπροσωπεύσουν το κατά προσέγγιση σχεδιάγραμμα EHR υπό μια πρακτική έννοια.

Τα αρχέτυπα και τα πρότυπα μπορούν να συνδεθούν με τις ορολογίες ή συμφραζόμενα για να βρεθούν τα κατάλληλα υποσύνολα ορολογίας που θα υποστηρίξουν την κατάλληλη επιλογή όρου από τους παροχής υγειονομικής περίθαλψης στο σημείο της εισαγωγής δεδομένων. [23].

2.10 Σύστημα FreeMed

Το σύστημα αυτό θα αναλυθεί σε βάθος στο κεφάλαιο που ακολουθεί γιατί αποτελεί το σύστημα της επιλογής μας για την εκπλήρωση αυτής της διπλωμάτης εργασίας. Οι

δυνατότητες αλλά και πω λόγοι της επιλογής του συγκεκριμένου συστήματος έναντι άλλων θα επεξηγηθεί εκτενώς στο κεφάλαιο που ακολουθεί.

Κλείνοντας αυτό το κεφάλαιο πρέπει να παραδεχτούμε ότι οι πιθανές επιλογές που υπάρχουν από πλευράς συστημάτων ηλεκτρονικού φάκελου ασθενή είναι παρά πολλές. Το κάθε σύστημα που αναλύθηκε στο παρόν κεφάλαιο έχει τα δικά του πλεονεκτήματα και μειονεκτήματα. Πρέπει να τονιστεί ότι υπάρχουν συστήματα με άπειρες δυνατότητες και άλλα με περιορισμένες δυνατότητες. Κάποια από τα συστήματα που αναλύθηκαν είναι γενικού σκοπού και άλλα είναι χρήσιμα μόνο για τον σκοπό που αυτά κατασκευάστηκαν. Οπωσδήποτε όμως το κάθε σύστημα μας έδωσε κάποιες πληροφορίες για τα απαραίτητα συστατικά ενός καλού λογισμικού ανοικτού κώδικα για σκοπούς ηλεκτρονικού φάκελου ασθενή.

Κεφάλαιο 3 Ηλεκτρονικός Ιατρικός Φάκελος Ασθενή FreeMed

Σε αυτό το κεφάλαιο σκοπός μας είναι να κάνουμε μια εμπεριστατωμένη μελέτη του ιατρικού φάκελου FreeMed Στο κεφάλαιο αυτό θα γίνει εκτενείς αναφορά του συστήματος που επιλέχτηκε.

3.1 Τι είναι το FreeMed;

Από το 1999 FreeMed Software Foundation δουλεύει πολύ σκληρά για να κάνει διαθέσιμο ένα ανοικτό λογισμικό που να είναι ένα σταθερό και αποδοτικό σύστημα διαχείρισης ιατρικών δεδομένων. Εδώ και 9 χρόνια το FreeMed εξελίσσεται συνεχώς με νέα χαρακτηριστικά και καλύτερη λειτουργικότητα. Μερικά από τα πράγματα που μας περιφέρονται από το FreeMed είναι

Διαχείριση ηλεκτρονικών εγγράφων → Σύστημα διαχείρισης εγγράφων χρησιμοποιώντας τεχνολογίες DjVu που αναπτύχθηκε από την AT&T.

HL7 interface – Το FreeMed έχει ένα HL7 interface που επιτρέπει την διαχείριση γεγονότων με βάση τον τύπο τους και για τον λόγο αυτό αναπτύσσεται και επεκτείνεται πολύ εύκολα.

Modular EMR - Το FreeMed έχει ένα πολύ εξελιγμένο σύστημα διάχυσης ηλεκτρονικού φάκελου που έχει την δυνατότητα να προσαρμόζεται πολύ εύκολα στις ανάγκες διαφορετικών χρηστών στην κάθε περίπτωση. Το σύστημα αυτό είναι γραμμένο σε συστατικά (modules / componement) έτσι ώστε να είναι δυνατή η εύκολη προσαρμογή του FreeMed για τον εκάστοτε χρηστή όπως αναφέραμε πιο πάνω.

Patient Management – Το FreeMed 0.9.x έχει ένα πολύ αναπτυγμένο σύστημα διαχείρισης ασθενών. Σε ένα χώρο ιατρικών εργασιών πολλές φορές ο φόρτος εργασίας μπορεί να οδηγήσει στην κατά λάθος τον διαχειριστή ενός συστήματος διαχείρισης ασθενών να επανεισαγάγει ένα ασθενή στο ηλεκτρονικό σύστημα. Το FreeMed επιτρέπει πολλών ειδών από ελέγχους που αποτρέπουν την διπλή εισαγωγή ασθενών. Οι ελέγχει περιλαμβάνουν ελέγχους ίδιου ονόματος ασθενή, ημερομηνίας γέννησης και ταχυδρομικού κώδικα κ.α. προς αποφεύγει της διπλής εισαγωγής ασθενών.

3.2 Το FreeMed και το πρωτόκολλο HL7

Μια ακόμα ερώτηση που μου ζητήθηκε να απαντήσω και προσήλθε από τον κύριο Παττίχη είναι αν το FreeMed ακολουθεί το πρωτόκολλο HL7 ή κάτι παραπλήσιο. Αυτό δεν χρειάστηκε σχεδόν καθόλου μελέτη από μέρος μου μιας και γίνεται αναφορά στις προδιαγραφές του FreeMed

<http://developer.freemedsoftware.org/wiki/index.php/HL7>

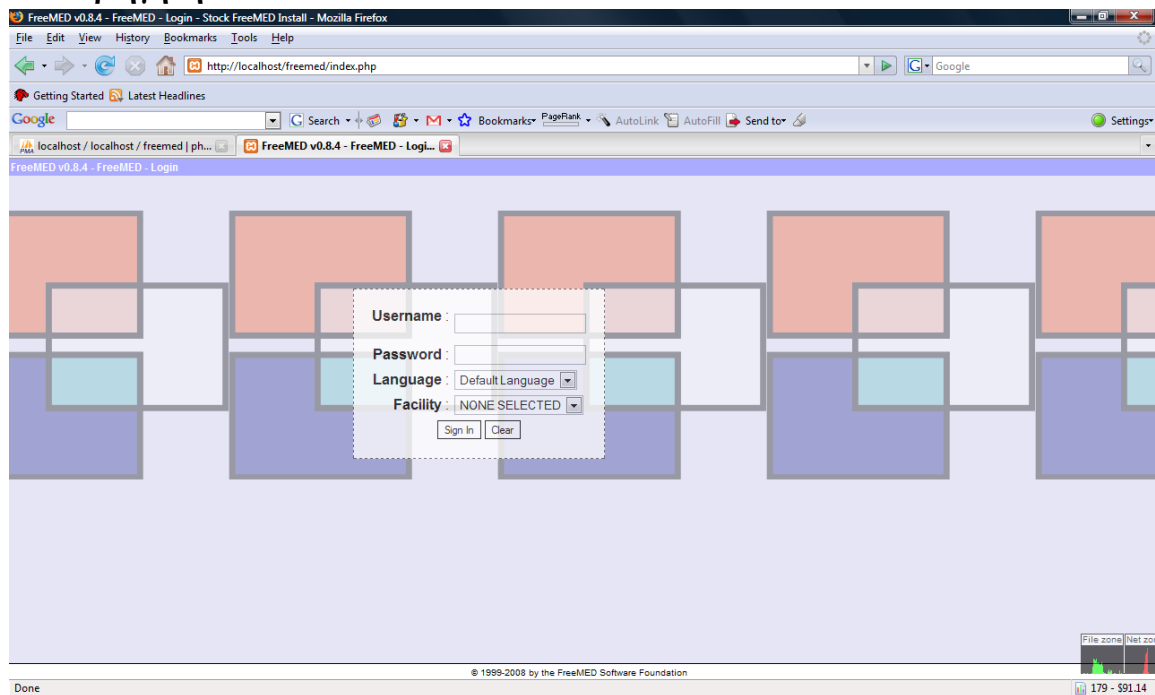
“Import of HL7 messages is handled by FreeMED.Parser_HL7v2. This is called like this:

```
$hl7 = CreateObject('_FreeMED.Parser_HL7v2', $message);
```

```
$hl7->Handle();
```

Note the use of the _FreeMED namespace, which is safe in any context, be it XML-RPC or standard FreeMED context.”[27]

3.3 Περιήγηση στο FreeMed



Σχήμα 3.1 – Εισαγωγή στο FreeMED

FreeMED v0.8.4 - FreeMED Main Menu - Stock FreeMED Install - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.freemed.info/main.php

Getting Started Latest Headlines

Google freemed demo Search

FreeMED v0.8.4 System Main User Patients Notify Help FreeMED Main Menu

You have 1 new message(s)

Patient Scheduler

You have 9 appointment(s) today and 46 appointment(s) this week.

Work Lists

Volker Bradley, M.D.			Irving J. Buchbinder, D.P.M.		
Name	Time	Status	Name	Time	Status
Fitch, Alfred	8:00	Checked out	Non-Patient Related	8:00	No Show
Non-Patient Related	9:00	Traveling to office	Buck, Anthony	9:00	Checked out
Barnes, Earline	9:30	Encounter/Procedure is complete	Federar, Samuel	9:30	Encounter/Procedure is complete
Ahias Portillo, Lopez Jose	10:00	Encounter/Procedure in progress	Rector, Joey	10:30	Encounter/Procedure in progress
Twain, Mark	10:25	No Show	Lopez, Oralia	10:45	Checked into Room 1
Bianchi, Camille	10:30	Checked into Room 1	Romney, Milt	11:15	Anticipating arrival at counter
Butterworth, Dillon	11:00		Nawaz, Ahmed	11:30	Anticipating arrival at counter
Maz, Aaron	11:15	Anticipating arrival at counter			
Orihuela Veraza, Luis	11:30	Anticipating arrival at counter			

Unread Faxes

There is currently 1 unread fax in the system. [Read](#)

© 1999-2008 by the FreeMED Software Foundation | User : Demo user

Done

Σχήμα 3.2 – Κατάσταση ραντεβού

FreeMED v0.8.4 - Manage Patient - Stock FreeMED Install - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.freemed.info/manage.php?id=833

Getting Started Latest Headlines

Google freemed demo Search PageRank AutoLink AutoFill Send to freemed demo Settings

FreeMED v0.8.4 System Main User Patients Notify Help Manage Patient

Alfred F. Fitch PAT833 62 year(s) old, DOB 23 Aug. 1945

Patient Information		Patient Appointments																	
Address : 4141 41st Street Albany, NY 12202 Home Phone : (518) 674-7611 Work Phone : (518) 672-311 SSN : 372-77-8254 PCP : Mark Belsky, M.D. Referring: Ronald P. Leemhuis, M.D.		<table border="1"> <thead> <tr> <th>Date</th> <th>Time</th> <th>Provider</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>25 May, 2008</td> <td></td> <td>Volker Bradley, M.D.</td> <td>Colonoscopy - one year follow up rectal carcinoma. At Ambulatory Surgical Center</td> </tr> <tr> <td>01 Dec, 2006</td> <td></td> <td>Roland E. Ugarte, M.D.</td> <td>Colonoscopy - one year follow up rectal carcinoma. At Ambulatory Surgical Center</td> </tr> <tr> <td>28 Dec, 2005</td> <td></td> <td>Roland E. Ugarte, M.D.</td> <td>Initial patient visit</td> </tr> </tbody> </table>		Date	Time	Provider	Description	25 May, 2008		Volker Bradley, M.D.	Colonoscopy - one year follow up rectal carcinoma. At Ambulatory Surgical Center	01 Dec, 2006		Roland E. Ugarte, M.D.	Colonoscopy - one year follow up rectal carcinoma. At Ambulatory Surgical Center	28 Dec, 2005		Roland E. Ugarte, M.D.	Initial patient visit
Date	Time	Provider	Description																
25 May, 2008		Volker Bradley, M.D.	Colonoscopy - one year follow up rectal carcinoma. At Ambulatory Surgical Center																
01 Dec, 2006		Roland E. Ugarte, M.D.	Colonoscopy - one year follow up rectal carcinoma. At Ambulatory Surgical Center																
28 Dec, 2005		Roland E. Ugarte, M.D.	Initial patient visit																

Progress Notes Box		Letters Box																																																							
<table border="1"> <thead> <tr> <th>Date</th> <th>Provider</th> <th>Description</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>10/06/2007</td> <td>Dr. Volker Bradley</td> <td>Cholelithiasis and biliary colic - imported template. Note not complete</td> <td></td> </tr> <tr> <td>12/04/2006</td> <td>Dr. Volker Bradley</td> <td>Colonoscopy with biopsy</td> <td></td> </tr> <tr> <td>12/04/2006</td> <td>Dr. Volker Bradley</td> <td>Colonoscopy with biopsy</td> <td></td> </tr> <tr> <td>06/28/2006</td> <td>Dr. Roland E. Ugarte</td> <td>Phone call from patient</td> <td></td> </tr> <tr> <td>12/28/2005</td> <td>Dr. Roland E. Ugarte</td> <td>Recent treatment for rectal adenocarcinoma</td> <td></td> </tr> </tbody> </table>		Date	Provider	Description	Action	10/06/2007	Dr. Volker Bradley	Cholelithiasis and biliary colic - imported template. Note not complete		12/04/2006	Dr. Volker Bradley	Colonoscopy with biopsy		12/04/2006	Dr. Volker Bradley	Colonoscopy with biopsy		06/28/2006	Dr. Roland E. Ugarte	Phone call from patient		12/28/2005	Dr. Roland E. Ugarte	Recent treatment for rectal adenocarcinoma		<table border="1"> <thead> <tr> <th>Date</th> <th>From</th> <th>To</th> <th>Subject</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>10/14/2006</td> <td>Roland E. Ugarte, M.D.</td> <td>Gretchen Allen, M.D., D.M.D.</td> <td>Patient seen for right inguinal hernia</td> <td>Written Print No Signature</td> </tr> <tr> <td>06/28/2006</td> <td>Volker Bradley, M.D.</td> <td>Mark Belsky, M.D.</td> <td>Regarding refused colonoscopy</td> <td>Written Print No Signature</td> </tr> <tr> <td>06/28/2006</td> <td>Volker Bradley, M.D.</td> <td>Mark Belsky, M.D.</td> <td>Regarding refused colonoscopy</td> <td>Written Print No Signature</td> </tr> <tr> <td>06/28/2006</td> <td>Volker Bradley, M.D.</td> <td>Mark Belsky, M.D.</td> <td>Regarding refused colonoscopy</td> <td>Written Print No Signature</td> </tr> <tr> <td>06/28/2006</td> <td>Volker Bradley, M.D.</td> <td>Mark Belsky, M.D.</td> <td>Regarding refused colonoscopy</td> <td>Written Print No Signature</td> </tr> </tbody> </table>		Date	From	To	Subject	Action	10/14/2006	Roland E. Ugarte, M.D.	Gretchen Allen, M.D., D.M.D.	Patient seen for right inguinal hernia	Written Print No Signature	06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature	06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature	06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature	06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature
Date	Provider	Description	Action																																																						
10/06/2007	Dr. Volker Bradley	Cholelithiasis and biliary colic - imported template. Note not complete																																																							
12/04/2006	Dr. Volker Bradley	Colonoscopy with biopsy																																																							
12/04/2006	Dr. Volker Bradley	Colonoscopy with biopsy																																																							
06/28/2006	Dr. Roland E. Ugarte	Phone call from patient																																																							
12/28/2005	Dr. Roland E. Ugarte	Recent treatment for rectal adenocarcinoma																																																							
Date	From	To	Subject	Action																																																					
10/14/2006	Roland E. Ugarte, M.D.	Gretchen Allen, M.D., D.M.D.	Patient seen for right inguinal hernia	Written Print No Signature																																																					
06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature																																																					
06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature																																																					
06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature																																																					
06/28/2006	Volker Bradley, M.D.	Mark Belsky, M.D.	Regarding refused colonoscopy	Written Print No Signature																																																					

© 1999-2008 by the FreeMED Software Foundation | User : Demo user

Σχήμα 3.3 –Στοιχεία και Κατάσταση Ασθενή

FreeMED v0.8.4 - Progress Notes - Stock FreeMED Install - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.freemed.info/module_loader.php?module=progressnotes&patient=833&action=display&id=1145&return=manage

Getting Started Latest Headlines

Google freemed demo Search PageRank AutoLink AutoFill Send to freemed demo Settings

Alfred F. Fitch PAT833 62 year(s) old, DOB 23 Aug. 1945

Progress Notes Manage Patient Select Patient Print

Relevant Date : 2007-10-06

Interval

Dr. - referred this patient to me for evaluation and treatment of cholelithiasis and biliary colic. The patient states that -_ has had nausea and vomiting. The patient states that -_ has had multiple episodes like this in the past.

Past medical history:

Operations: -
 Medications: -
 Allergies: -
 Chronic illnesses: -

Social history:

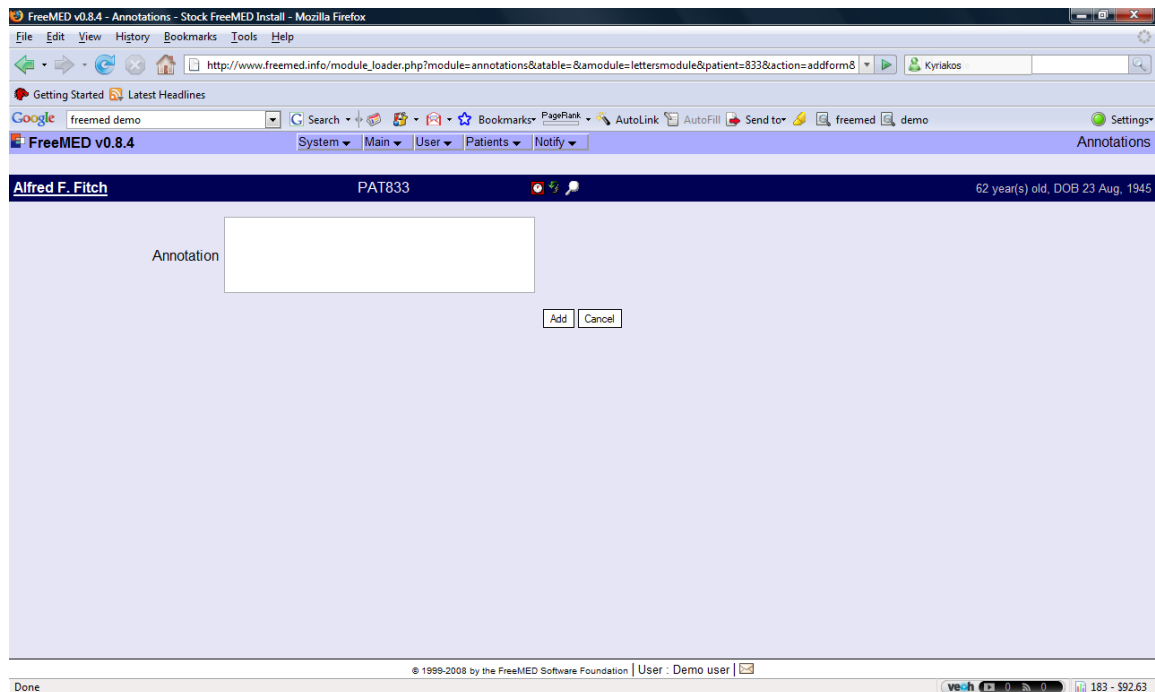
Review of systems:
 HEENT: No history of loss of consciousness, seizures, severe headaches, difficulty seeing or hearing.
 Respiratory: Denies chronic cough, sputum production, hemoptysis, dyspnea or history of asthma
 Cardiovascular: No history of hypertension, chest pain, heart murmurs, pedal edema
 GI: No history of unexplained weight loss, unexplained abdominal pain, change in bowel habits or melena. See present illness
 GU: No history of UTI's, genital lesions
 Extremities: WNL in the past

Physical examination:

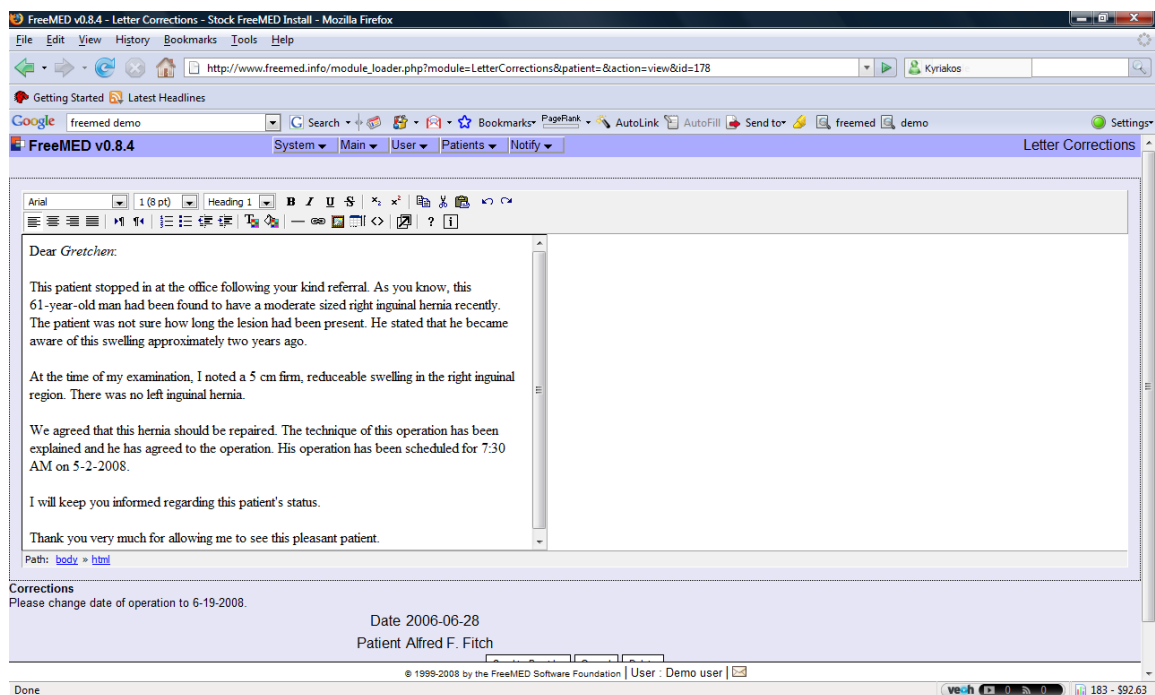
This reveals an awake, alert, oriented x3, well developed and well nourished -man who is 5'"" tall and who weighs - pounds. -e is in no apparent distress.
 HEENT: Normal.

Done

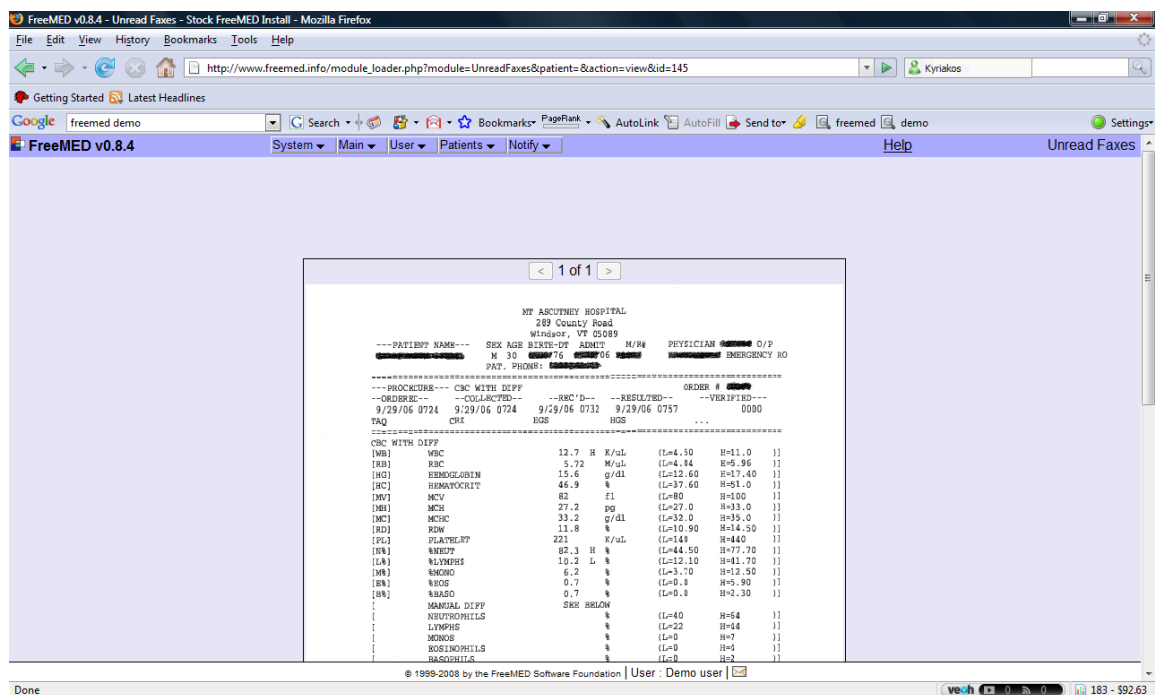
Σχήμα 3.4 –Εκτενείς αναφορά κάπου ιατρού που είδε τον ασθενή



Σχήμα 3.4 –Καταχώρηση σημείωσης ιατρού που είδε τον ασθενή



Σχήμα 3.5 –Δημιουργία και επεξεργασία επιτολών



Σχήμα 3.5 –Διαχείριση fax

Συμπερασματικά προφέρονται πολλές λειτουργίες. Διαχέαμε όμως ότι η διεθνής κωδικοποίηση ασθενή είναι ελλείψεις.

Κεφάλαιο 4 International Disease Coding (IDC)

Σε αυτό το κεφάλαιο σκοπός μας είναι να κάνουμε μια ιστορική ανάδρομη του International Disease Coding (IDC) που είναι ο τρόπος διεθνούς κωδικοποίηση ασθενών .

4.1 Αρχικά ιστορικά στοιχεία

Ο Sir George Knibbs, ένας Αυστραλός στατιστικολόγος, επαίνεσε τον François Bossier de Lacroix (1706-1777), γνωστός ως Sauvages, για την πρώτη προσπάθεια συστηματικής κατηγοριοποίησης ασθενειών [37]. Η ιδέα του Sauvages δημοσιεύτηκε κάτω από τον τίτλο *Nosologia methodica*. Ένας σύγχρονος συνεχιστής των ιδεών του Sauvages ήταν ο μεγάλος μεθοδολόγος Linnaeus (1707-1778), ένα σημαντικό του έργο έχει τίτλο *Genera morborum*. Στις αρχές του 19^{ου} αιώνα, η ομαδοποίηση ασθενειών στην πιο γενική της μορφή ορίστηκε από τον William Cullen (1710-1790), από το Εδιμβούργο και δημοσιεύτηκε το 1785 με τίτλο *Synopsis nosologiae methodicae*. Για όλους τους πρακτικούς λόγους, η στατιστική μελέτη των ασθενειών άρχισε ένα αιώνα νωρίτερα με τις μελέτες του John Graunt πάνω σε θέματα που αφορούσαν το London Bills of Mortality. Το είδος της ομαδοποίησης που όρισε αυτός ο πρωτοπόρος, αποδεικνύεται με τις προσπάθειες που έκανε να υπολογίσει τον αριθμό βρεφών που πέθαιναν πριν φτάσουν την ηλικία των έξι ετών μιας και μέχρι εκείνη την εποχή δεν υπήρχε κάποιο αρχείο, όπου να καταγράφονται στατιστικές θανάτων βάσει ηλικίας. Πηρέ όλους τους θανάτους και τους κατηγοριοποίησε σε thrush, convulsions, rickets, teeth, worms, abortives, chrysomes, infants και livergrown, έπειτα κατηγοριοποίησε τους μισούς θανάτους της εποχής σε smallpox, swinepox, measles, και worms without convulsions. Παρά την προχειρότητα της ομαδοποίησης υπολογίστηκε ότι το 36 % των θανάτων σε άτομα ήταν σε ηλικία κάτω των έξι ετών. Αργότερα αυτός ο αριθμός αποδείχθηκε ότι ήταν πολύ κοντά στα πραγματικά δεδομένα της εποχής.

Ενώ τρεις αιώνες έχουν συμβάλει στην επιστημονική ακρίβεια της ομαδοποίησης ασθενειών και στην ομαδοποίηση των αιτιών θανάτου, υπάρχουν ακόμα κάποιοι που αμφισβητούν την χρησιμότητα μιας τέτοιας ομαδοποίησης. Σε αυτούς μπορεί κανείς να πει τα λόγια που είπε κάποτε και ο Major Greenwood: "The scientific purist, who will wait for medical statistics until they are nosologically exact, is no wiser than Horace's rustic waiting for the river to flow away" [38]. Ευτυχώς για την πρόοδο της προληπτικής ιατρικής, το General Register Office της Αγγλίας και της Ουαλίας, κατά την περίοδο έναρξης του το 1837, βρήκαν στον William Farr (1807-1883) – τον πρώτο στατιστικολόγο - έναν άνθρωπο που όχι μόνο έκανε δυνατή την καλύτερη χρήση ενός ημιτελούς συστήματος κατηγοριοποίησης ασθενών που υπήρχε εκείνη την εποχή, αλλά εργάστηκε για την ασφαλή και καλύτερη κατηγοριοποίηση και για την διεθνή ομοιομορφία χρήσης. Ο Farr ανακάλυψε ότι η κατηγοριοποίηση του Cullen που εφαρμοζόταν στις δημόσιες υπηρεσίες της εποχής του, δεν έχει μεταβληθεί από τον ίδιο να ενσωματώνει την πρόοδο της ιατρικής επιστήμης, ούτε και θεωρήθηκε από τον ίδιο κάτι ικανοποιητικό για στατιστικούς σκοπούς.

Για τον λόγο αυτό στο πρώτη ετήσια αναφορά του Registrar General [39] συζήτησε ο ίδιος τις αρχές που πρέπει να διέπουν την στατιστική ομαδοποίηση των ασθενειών και τόνισε την ανάγκη για την υιοθέτηση μιας ενιαίας ομαδοποίησης ως ακολούθως : Το πλεονέκτημα της ονοματολογίας, όσο ημιτελής και αν είναι, είναι τόσο εμφανής που δημιουργεί έκπληξη το γεγονός ότι δεν δόθηκε προσοχή για την ενίσχυση του στο Bills of Mortality. Κάθε ασθένεια , σε πολλά της στιγμιότυπα, έχει καθοριστεί από τρεις ή τέσσερις όρους και κάθε όρος έχει εφαρμοστεί σε άλλες τόσες ασθένειες. Έτσι λοιπόν, λόγω της ασάφειας της ονοματολογίας, οι ιατροί πολλές φορές καταχωρούν μια διάγνωση περισσότερο δυσκολονόητη, αντί να καταχωρήσουν την κύρια ασθένεια. Η ονοματολογία λοιπόν, είναι εξίσου σημαντική στον τομέα αυτό, όσο σημαντικά είναι τα βάρη και οι μετρήσεις στην φυσική και κατ' επέκταση έπρεπε να ταχτοποιηθούν χωρίς καθυστερήσεις. Οι δυο μεθοδολογίες ομαδοποίησης είτε μιλάμε για την ονοματολογία είτε για τη στατιστική ομαδοποίηση, ήταν υπό διαρκή μελέτη από τον Farr στις ετήσιες του «Επιστολές» προς το Registrar General που δημοσιεύτηκαν στα Annual Reports του Registrar General. Η χρησιμότητα της ενιαίας ομαδοποίησης των αιτιών θανάτου είχαν τόσο έντονη αναγνώριση στο πρώτο International Statistical Congress, που διεξήχθη στις Βρυξέλλες το 1853, που το

Congress ζήτησε από τους William Farr και Marc d'Espine, από την Γένοβα, να ετοιμάσουν μια διεθνώς εφαρμόσιμη ενιαία ομαδοποίηση των αιτών θανάτου. Στο επόμενο Congress, στο Παρίσι το 1855, ο Farr και ο d'Espine παρουσίασαν δυο διαφορετικές λίστες που βασίζονταν σε πολύ διαφορετικές προσεγγίσεις. Η ομαδοποίηση του Farr ήταν χωρισμένη σε πέντε ομάδες: epidemic diseases, constitutional (general) diseases, local diseases arranged according to anatomical site, developmental diseases, and diseases that are the direct result of violence. Ο D'Espine ομαδοποίησε τις ασθένειες σύμφωνα με τη φύση τους (gouty, herpetic, haematic, κτλ.). Το Congress υιοθέτησε μια λίστα από 139 περιγραφές. Το 1864, αυτές οι ομαδοποιήσεις είχαν αναθεωρηθεί στο Παρίσι στην βάση του μοντέλου του Farr και κατά συνέπεια αναθεωρηθήκαν ξανά το 1874, 1880, και 1886. Παρ' όλο που οι ομαδοποιήσεις κατά γενική ομολογία δεν έγιναν αποδεκτές, οι γενικές αρχές που εισηγείται ο Farr, όσον αφορά τις ασθένειες κατά ανατομική τοποθεσία (anatomical site), επιβίωσαν τελικά και αποτέλεσαν τη βάση του International List of Causes of Death.

4.2 Υιοθέτηση του International List of Causes of Death

Το International Statistical Institute, ο πρόγονος του International Statistical Congress, κατά την συνάντησή τους στην Βιέννη το 1891, όρισαν μια επιτροπή με επικεφαλής τον, Jacques Bertillon (1851-1922), με την ευθύνη της δημιουργίας μιας λίστας που να ομαδοποιεί τις αιτίες θανάτου. Είναι ενδιαφέρον να τονιστεί, ότι ο Bertillon ήταν εγγονός του Achille Guillard, ενός αξιότιμου βοτανολόγου και στατιστικολόγου, που εισήγαγε την πρόταση προς τον Farr και τον d'Espine να ετοιμάσουν την πρώτη ομαδοποίηση αιτιών θανάτου στο πρώτο International Statistical Congress το 1853. Αυτή η αναφορά παρουσιάστηκε στην επιτροπή από τον Bertillon στην συνάντησή του International Statistical Institute στο Σικάγο το 1893 και υιοθετήθηκε. Η ομαδοποίηση που ετοιμάστηκε από την επιτροπή του Bertillon και βασίστηκε στην ομαδοποίηση με βάση την αιτία θανάτου που είχε χρησιμοποιηθεί στο Παρίσι, από την αναθεώρηση του 1885, παρουσιάστηκε μια Αγγλική, Γερμανική και Ελβετική εκδόσεων ομαδοποίησης. Η ομαδοποίηση βασίστηκε στην αρχή που προτάθηκε από τον Farr, που διαχώριζε μεταξύ γενικών ασθενειών και εκείνων που εντοπίζονται τοπικά και είναι συγκεκριμένες σε σχέση με τον τόπο προέλευσής τους ή την

ανατομική τοποθεσία στην οποία εμφανίζονται. Σε συνάρτηση με τις οδηγίες του Vienna Congress που εισηγήθηκε ο L. Guillaume, ο Director of the Federal Bureau of Statistics της Ελβετίας, Ο Bertillon συμπεριέλαβε τρεις ομαδοποιήσεις:

- Μια ελαχιστοποιημένη ομαδοποίηση σαράντα τεσσάρων τίτλων.
- Μια ομαδοποίηση ενενήντα εννέα τίτλων.
- Μια ομαδοποίηση εκατό εξήντα ενός τίτλων.

Αυτές οι τρεις ομαδοποιήσεις ονομάστηκαν Bertillon Classification of Causes of Death, έλαβαν έγκριση και έγιναν γενικώς αποδεκτές από πολλές χώρες.

Η ομαδοποίηση χρησιμοποιήθηκε για πρώτη φορά στην Βόρεια Αμερική από των Jesús E. Monjarás για την συλλογή στατιστικών για το San Luis de Potosí, στο Μεξικό [40]. Το 1898, το American Public Health Association, σε ένα συνέδριο τους στο Ottawa, του Καναδά εισηγήθηκαν την υιοθέτηση της Bertillon Classification από υπηρεσίες του Καναδά, του Μεξικού και των Ηνωμένων Πολιτειών Αμερικής. Η οργάνωση εισηγήθηκε, να γίνονται αναθεωρήσεις κάθε δέκα χρόνια

Στην συνεδρία του International Statistical Institute στο Christiania το 1899, Ο Bertillon παρουσίασε μια αναφορά της προόδου της ομαδοποίησης, συμπεριλαμβανόμενου της εισήγησης του American Public Health Association για αναθεώρηση. Το International Statistical Institute υιοθέτησε τα εξής [41]:

Το International Statistical Institute, είχε πειστεί για την ανάγκη χρήσης σε διαφορετικές χώρες με βάση το μοναδικό σύστημα ονοματολογίας της κάθε μιας. Εκεί γίνεται γνωστό προς ευχαρίστηση όλων ότι όλα τα γραφεία στατιστικής της Βόρειας Αμερικής, κάποια της Νότιας Αμερικής και κάποια γραφεία της Ευρώπης έχουν υιοθετήσει την μοναδική ονοματολογία των αιτιών θανάτου που είχαν προηγούμενος προταθεί το 1893. Κάτι τέτοιο οδήγησε το σύνολο των συνέδρων, να επιμένουν στην υιοθέτηση της μοναδικής ονοματολογίας άμεσα, από όλες τις στατιστικές υπηρεσίες της Ευρώπης. Τελικώς η συνεδρία εγκρίνει ένα γενικό πλαίσιο ονοματολογίας που προτείνεται από το American Public Health Association στην συνεδρία τους στην Ottawa το 1898, όπου και πάλι τονίζεται η ανάγκη υιοθέτησης του συστήματος ονοματολογίας από κάθε ευρωπαϊκή στατιστική υπηρεσία που ακόμα δεν το είχε υιοθετήσει, όπως επίσης και ότι θα πρέπει να

συμβάλλουν στην διαδικασία ονοματολογίας, έτσι ώστε να υπάρχει γενική συμβατότητα και αποδοχή στην διαδικασία της ονοματολογίας. Η Γαλλική κυβέρνηση για την λόγο αυτό διοργάνωσε στο Παρίσι, το πρώτο διεθνές συνέδριο για την αναθεώρηση του συστήματος ονοματολογίας Bertillon ή αλλιώς International List of Causes of Death το 1900.

Αντιπροσωπείες από είκοσι έξι χώρες έλαβαν μέρος στο συνέδριο. Το αποτέλεσμα ήταν η δημιουργία εκατό εβδομήντα εννέα ομάδων αιτιών θανάτου. Στις 21 Αύγουστου 1900 υιοθετήθηκε μια μικρότερη ομαδοποίηση που αποτελούνταν από τριάντα πέντε ομάδες αιτιών θανάτου. Η ανάγκη για αναθεωρήσεις κάθε δέκα χρόνια αναγνωρίστηκε, και ως εκ τούτου ζητήθηκε από την γαλλική κυβέρνηση να καλέσει την επόμενη συνεδρία το 1910. Στην πραγματικότητα η επόμενη συνεδρία έγινε το 1909 και η γαλλική κυβέρνηση κάλεσε τις επόμενες συνεδρίες το 1920, 1929 και 1938.

Το σύστημα του Bertillon συνέχισε να είναι η δύναμη που καθοδηγούσε την ανάπτυξη του International List of Causes of Death και οι αναθεωρήσεις του 1900, 1910, and 1920 έγιναν κάτω από τη δική του επίβλεψη. Ως γενικός γραμματέας του International Conference, έστειλε τις αναθεωρήσεις του 1920 σε περισσότερα από 900 άτομα, ζητώντας τα δικά τους σχόλια. Ο θάνατος του το 1922, άφησε το International Conference χωρίς καθοδήγηση. Στην συνεδρία του 1923 του International Statistical Institute, ο Michel Huber, ο διάδοχος του Bertillon στην Γαλλία, αναγνώρισε την λήψη ηγεσίας και εισηγήθηκε ένα τρόπο ξεδιαλύματος του θέματος για το International Statistical Institute να ανανεώσει τις διατάξεις του το 1893, σε σχέση με το International Classification of Causes of Death και επίσης εισηγήθηκε τη συνεργασία με άλλους οργανισμούς για την προετοιμασία της επόμενης αναθεώρησης. Το Health Organization of the League of Nations έδειξε πολύ ενδιαφέρον για τα ζωτικής σημασίας στατιστικά και διόρισαν το Commission of Statistical Experts για να μελετήσουν την ομαδοποίηση ασθενειών και αιτιών θανάτου όπως και για άλλα προβλήματα που απασχολούσαν την εποχή εκείνη, στον τομέα της ιατρικής στατιστικής. Ο E. Roselle, με τίτλο Chief of the Medical Statistical Service του γερμανικού Health Bureau και μέλος του Commission of Expert Statisticians, ετοίμασε ένα έγγραφο που απαριθμούσε την προέκταση των οδηγιών που προηγήθηκαν του International List of Causes of Death το 1920 που θα ήταν απαραίτητα, αν τα στατιστικά έπρεπε να χρησιμοποιηθούν σε συνάρτηση με στατιστικούς πίνακες. Η εμπεριστατωμένη αυτή μελέτη δημοσιεύτηκε από το Health Organization of the League of

Nations to 1928 [69]. Για να ήταν δυνατόν να συμπτυχθούν τα συμπεράσματα όλων των οργανισμών, ιδρύεται μια διεθνή επιτροπή, γνωστή ως "Mixed Commission", που αποτελείται από έναν ίσο αριθμό από αντιπροσώπους της International Statistical Institute και του Health Organization of the League of Nations. Αυτή η επιτροπή έγραψε τις αρχικές προτάσεις που αφορούσαν την τέταρτη (1929) και την πέμπτη (1938) αναθεώρηση του International List of Causes of Death.

4.3 Η Πέμπτη Αναθεώρηση

Το πέμπτο International Conference για αναθεώρηση του International List of Causes of Death, όπως και οι προηγούμενες έγιναν στο Παρίσι τον Οκτώβριο του 1938. Η συνεδρία ενέκρινε τρεις λίστες. Μια λεπτομερής λίστα διακοσίων τίτλων, μια ενδιάμεση λίστα από ογδόντα επτά τίτλους και μια τελική μειωμένη λίστα των σαράντα τεσσάρων τίτλων. Εκτός του ότι η λίστα ενημερώθηκε λαμβάνοντας υπόψη την επιστημονική εξέλιξη και πιο συγκεκριμένα στο κεφάλαιο που αφορούσε τις λοιμώξεις παρασιτικών ασθενών και επίσης στο κεφάλαιο που αφορούσε τις προγεννητικές παθήσεις και τα ατυχήματα, το Conference έκανε όσο λιγότερες αλλαγές μπορούσε στο περιεχόμενο της λίστας και στην αρίθμηση των αντικειμένων. Μια λίστα αιτιών που προκαλούν των γέννηση νεκρών βρεφών είχε επίσης δημιουργηθεί και εγκρίθηκε από το Conference. Όσο αφορά την ομαδοποίηση των ασθενειών για σκοπούς συλλογής στατιστικών θνησιμότητας το Conference, αναγνώρισε την αυξημένη ανάγκη για μια αντίστοιχη λίστα από ασθενείς που ικανοποιούν τις στατιστικές ανάγκες πολλών διαφορετικών οργανισμών, όπως οργανισμούς ασφάλισης υγείας, νοσοκομεία, στρατιωτικές υπηρεσίες υγείας και όμοια σώματα. Με βάση αυτό το δεδομένο υιοθετήθηκαν οι εξής λύσεις[70]:

1. Διεθνώς αποδεκτές λίστες ασθενειών

Λαμβάνοντας υπόψη την σημασία της σύνταξης διεθνών λιστών ασθενειών που να αντιστοιχούν σε λίστες αιτιών θανάτου:

Το Conference εισηγείται το Joint Committee να διοριστεί από το International Institute of Statistics και το Health Organization of the League of Nations ώστε να αναλάβουν, όπως

και το 1929, την προετοιμασία διεθνών λιστών ασθενειών, σε συνάρτηση με ειδικούς και εκπροσώπους των οργανισμών που επηρεάζονται.

Πριν την ολοκλήρωση των διεθνών λιστών ασθενειών το Conference προτείνει οι διάφορες τοπικές λίστες οι οποίες χρησιμοποιούνται, να προσαρμοστούν όσο το δυνατόν πιο πολύ με το International List of Causes of Death (οι αριθμοί των κεφαλαίων και οι επικεφαλίδες). Το Conference εισηγήθηκε ότι η κυβέρνηση των Ηνωμένων Πολιτειών να συνεχίσει την μελέτη όσο αφορά την λίστα ενοποιημένων ασθενών [46]:

2. Πιστοποιητικό θανάτου και επιλογή των αιτιών του θανάτου όπου περισσότερες από μια αιτίες δίνονται (Κοινές αιτίες)

Η διάσκεψη, εκτιμώντας ότι, το 1929, η κυβέρνηση των Ηνωμένων Πολιτειών ήταν αρκετά ικανή να αναλάβει τη μελέτη των μεθόδων, για επιλογή της κύριας αιτίας του θανάτου.

Αυτές οι αιτίες ταξινομούνται σε πίνακες στις περιπτώσεις εκείνες όπου δύο ή περισσότερες αιτίες αναφέρονται στο πιστοποιητικό θανάτου ενώ, οι πολυάριθμες έρευνες που ολοκληρώθηκαν ή βρίσκονταν στο στάδιο της προετοιμασίας σε διάφορες χώρες αποκάλυψαν τη σημασία αυτού του προβλήματος, το οποίο δεν έχει λυθεί ακόμα, Ωστόσο σύμφωνα με αυτές τις έρευνες, η διεθνής συγκρισιμότητα των ποσοστών θανάτου από τις διάφορες ασθένειες απαιτεί, όχι μόνο λύση του προβλήματος της επιλογής της κύριας αιτίας του θανάτου (με βάση τους ταξινομημένους πίνακες), αλλά και τη λύση διάφορων άλλων ερωτημάτων. Συμπερασματικά έγιναν τα εξής στο τέλος της συνεδρίας:

- (1) θερμές ευχαριστίες προς την κυβέρνηση των Ηνωμένων Πολιτειών για την εργασία που έχει ολοκληρώσει ή έχει προαγάγει άπαυτου
- (2) ζητείται από την κυβέρνηση των Ηνωμένων Πολιτειών να συνεχίσει τις έρευνές του κατά τη διάρκεια των επόμενων δέκα ετών, σε συνεργασία με άλλες χώρες και οργανώσεις, σε ελαφρώς ευρύτερη βάση, και
- (3) προτείνει ότι, για αυτές τις μελλοντικές έρευνες, η κυβέρνηση των Ηνωμένων Πολιτειών πρέπει να εμπεριέχει μια υποεπιτροπή περιλαμβάνοντας τους αντιπροσώπους από τις χώρες και τις οργανώσεις που συμμετέχουν στις έρευνες που αναλαμβάνονται άπαυτου.

4.4 Προηγούμενες ταξινομήσεις των ασθενειών για τις στατιστικές νοσηρότητας

Στη συζήτηση μέχρι τώρα, η ταξινόμηση της ασθένειας παρουσιάζεται σχεδόν εξολοκλήρου σε σχέση με τις στατιστικές των αιτιών θανάτου. Ο Farr, εντούτοις, αναγνώρισε ότι ήταν επιθυμητό «να επεκταθεί το ίδιο σύστημα της ονοματολογίας στις ασθένειες» και τώρα με τη χρήση των αριθμών στους πίνακες των ασθενειών στους στρατούς, στο ναυτικό, στα νοσοκομεία, στις φυλακές, στις δημόσιες υπηρεσίες κάθε είδους, καθώς και με την απογραφή των χωρών όπως η Ιρλανδία, όπου οι ασθένειες όλων των ανθρώπων απαριθμούνται [36].

Σε δική του Έκθεση σχετικά με την ονοματολογία και την στατιστική για την ταξινόμηση των ασθενειών, όπου παρουσιάστηκε στο δεύτερο διεθνές στατιστικό συνέδριο, περιέλαβε στο γενικό κατάλογο ασθενειών τις περισσότερες ασθένειες που έχουν επιπτώσεις στην υγεία καθώς επίσης και τις ασθένειες που είναι θανάσιμες. Στο τέταρτο διεθνές στατιστικό συνέδριο, που διοργανώθηκε στο Λονδίνο το 1860, η Florence Nightingale προώθησε την υιοθέτηση της ταξινόμησης Farr των ασθενειών σε πίνακα νοσηρότητας νοσοκομείων στο έγγραφο, *Προτάσεις για το ομοιόμορφο σχέδιο των στατιστικών νοσοκομείων*.

Στην πρώτη Διεθνή Διάσκεψη για να αναθεωρήσει την ταξινόμησης Bertillon των αιτιών θανάτου στο Παρίσι το 1900, υιοθετήθηκε η παράλληλη ταξινόμηση των ασθενειών για χρήση στις στατιστικές της ασθένειας. Ένας παράλληλος κατάλογος υιοθετήθηκε επίσης στην δεύτερη διάσκεψη το 1909. Οι πρόσθετες κατηγορίες για τις non-fatal ασθένειες διαμορφώθηκαν από την υποδιαίρεση της ταξινόμησης αιτία--θανάτου σε δύο ή τρεις ομάδες ασθενειών.

Οι εργασίες το 1910, είχαν όνομα *Διεθνής ταξινόμηση των αιτιών της ασθένειας και του θανάτου*. Αργότερα οι αναθεωρήσεις ενσωμάτωσαν μερικές από τις ομάδες στο λεπτομερή διεθνή κατάλογο αιτιών του θανάτου. Η τέταρτη Διεθνής Διάσκεψη υιοθέτησε μια ταξινόμηση της ασθένειας που διέφερε από το λεπτομερή διεθνή κατάλογο αιτιών θανάτου μόνο από την άποψη της προσθήκης των περαιτέρω υποδιαιρέσεων 12 τίτλων. Αυτές οι διεθνείς ταξινομήσεις οι ασθένειες, εντούτοις, απέτυχαν να λάβουν τη γενική αποδοχή, δεδομένου ότι παρείχαν μόνο μια περιορισμένη επέκταση του βασικού καταλόγου αιτιών θανάτου. Ελλείπει μιας ομοιόμορφης ταξινόμησης των ασθενειών που θα μπορούσε να χρησιμοποιηθεί ικανοποιητικά για τις στατιστικές της ασθένειας, οδήγησε πολλές χώρες να θεωρήσουν απαραίτητο να προετοιμάσουν τους δικούς τους καταλόγους. Ένας

τυποποιημένος κώδικας νοσηρότητας προετοιμάστηκε από την το Συμβούλιο υγείας του Καναδά και δημοσιεύτηκε το 1936. Οι κύριες υποδιαιρέσεις αυτού του κώδικα αντιπροσώπευσαν τα δεκαοχτώ κεφάλαια της αναθεώρησης του 1929 του διεθνούς καταλόγου αιτιών θανάτου, και αυτοί υποδιαιρέθηκαν σε μερικές ακόμα, 380 συγκεκριμένες κατηγορίες ασθενειών. Στην Πέμπτη Διεθνή Διάσκεψη το 1938, ο καναδικός εκπρόσωπος εισήγαγε το α τροποποίηση αυτού του καταλόγου για την εκτίμηση ως βάση για έναν διεθνή κατάλογο κωδικοποίησης ασθενειών. τα μέτρα λήφθηκαν όπως αναφέρει η εισήγηση στην πρόταση, η διάσκεψη ενέκρινε το ψήφισμα που αναφέρθηκε ανωτέρω. Το 1944, οι προσωρινές ταξινομήσεις των ασθενειών και οι τραυματισμοί δημοσιεύτηκαν και στο Ηνωμένο Βασίλειο και στις Ηνωμένες Πολιτείες της Αμερικής για τη χρήση στην ταξινόμηση σε πίνακα των στατιστικών νοσηρότητας. Και οι δύο ταξινομήσεις ήταν πιο εκτενείς από τον καναδικό κατάλογο, αλλά, ακολουθούσε τη γενική μορφή των ασθενειών στο διεθνή κατάλογο αιτιών θανάτου. Η βρετανική ταξινόμηση προετοιμάστηκε από την Επιτροπή στατιστικών νοσηρότητας νοσοκομείων του ιατρικού ερευνητικού συμβούλιο, το οποίο δημιουργήθηκε του Ιανουαρίου 1942. *Μια προσωρινή ταξινόμηση των ασθενειών και τραυματισμοί για τη χρήση στη σύνταξη των στατιστικών νοσηρότητας* [17]. Προετοιμάστηκε με το σκοπό ένα σχέδιο για συλλογή στατιστικών και καταγραφής των ασθενών που αναγνώρισαν σε νοσοκομείο στο Ηνωμένο Βασίλειο, χρησιμοποιώντας πρότυπα ταξινόμησης των ασθενειών και των τραυματισμών, και χρησιμοποιήθηκε σε όλη τη χώρα από τις κυβερνητικές και άλλες υπηρεσίες. Μερικά έτη νωρίτερα, τον Αύγουστο του 1940 ο γενικός χειρουργός της υπηρεσίας Ηνωμένης δημόσιας υγείας και ο διευθυντής του Ηνωμένου γραφείου της απογραφής δημοσίευσε έναν κατάλογο ασθενειών και τραυματισμών για ταξινόμηση σε πίνακα για στατιστικές νοσηρότητας [45]. Ο κώδικας προετοιμάστηκε από το τμήμα των μεθόδων δημόσιας υγείας της υπηρεσίας δημόσιας υγείας σε συνεργασία με μια επιτροπή των συμβούλων που διορίζονται από το γενικό χειρουργό. *Το εγχειρίδιο για την κωδικοποίηση αιτίες της ασθένειας σύμφωνα με έναν κώδικα διαγνώσεων για την ταξινόμηση σε πίνακα των στατιστικών νοσηρότητας*, αποτελείτο από τον κώδικα διαγνώσεων, ένας συνοπτικός κατάλογος συνυπολογισμών, και ένας αλφαβητικός δείκτης, δημοσιεύθηκαν το 1944. Ο κώδικας χρησιμοποιήθηκε σε αρκετά νοσοκομεία, σε έναν μεγάλο αριθμό εθελοντικών

ασφαλιστικών σχεδίων νοσοκομείων και σχεδίων ιατρικής περίθαλψης, και στις πρόσθετες μελέτες σε στενή συνεργασία με άλλες υπηρεσίες στις Ηνωμένες Πολιτείες.

4.5 Ενιαία Επιτροπή για κοινές αιτίες θανάτου

Σύμφωνα με το ψήφισμα της πέμπτης Διεθνούς Διάσκεψης, οι ΗΠΑ το 1945 διόρισε την Ηνωμένη Επιτροπή των κοινών αιτιών θανάτου υπό την προεδρία του Lowell J. Columbus, Καθηγητής της στατιστικής στο πανεπιστήμιο Johns Hopkins. Μέλη και σύμβουλοι αυτής της επιτροπής συμπεριλάμβαναν αντιπρόσωπους των κυβερνήσεων του Καναδά και του Ηνωμένου Βασιλείου και τα Ενωμένα Έθνη. Η επιτροπή αναγνώρισε τη γενική τάση της σκέψης όσον αφορά τους καταλόγους νοσηρότητας και θνησιμότητας σε επίπεδο στατιστικές, και αποφάσισαν ότι, πριν από την επανεξετάσει του θέματος κοινών αιτιών, θα ήταν συμφέρον να εξετάσει τις ταξινομήσεις από την άποψη της νοσηρότητας και της θνησιμότητας, από πλευράς του προβλήματος των κοινών αιτιών που αναφέρονται αλλά και από πλευράς των δύο τύποι στατιστικών. Η επιτροπή έλαβε υπόψη επίσης εκείνο το μέρος του ψηφίσματος σχετικά με τους διεθνείς καταλόγους ασθενειών της προηγούμενης Διεθνούς Διάσκεψης που συστήνει ότι οι διάφοροι «εθνικοί κατάλογοι» πρέπει να τεθούν σε λειτουργία, όσο το δυνατόν περισσότερο. Με βάση τον λεπτομερή διεθνή κατάλογο αιτιών θανάτου. Αναγνώρισε ότι η ταξινόμηση της ασθένειας και ο τραυματισμός συνδέεται στενά με την ταξινόμηση των αιτιών θανάτου. Η άποψη ότι τέτοιοι κατάλογοι είναι εντελώς διαφορετικοί προκύπτει από τη λανθασμένη πεποίθηση ότι ο διεθνής κατάλογος είναι μια ταξινόμηση των τελικών αιτιών, ενώ αυτός είναι στην πραγματικότητα βασισμένος στο morbid όρο που είναι ουσιαστικά η αρχή της αλυσίδας των γεγονότων που τελικά έχουν σαν συνέπεια το θάνατο. Η επιτροπή θεώρησε ότι, προκειμένου να χρησιμοποιηθούν πλήρως και οι στατιστικές νοσηρότητας και θνησιμότητας, όχι μόνο είναι η ταξινόμηση των ασθενειών και για τους δύο λόγους είναι συγκρίσιμη, αλλά είναι δυνατόν να υπάρξει ένας ενιαίος κατάλογος. Επιπλέον, ένας αυξανόμενος αριθμός στατιστικών οργανώσεων χρησιμοποιούσε τα ιατρικά αρχεία που περιλαμβάνουν και τα δύο στοιχεία ασθένεια και θάνατος. Ακόμη και στις οργανώσεις που συντάσσουν μόνο τις στατιστικές νοσηρότητας, για θανάσιμες καθώς επίσης και non-fatal περιπτώσεις πρέπει να είναι κωδικοποιημένες. Ένας ενιαίος κατάλογος, επομένως, διευκολύνει πολύ τις

διαδικασίες κωδικοποίησής τους. Παρέχει επίσης μια κοινή βάση για σύγκριση των στατιστικών νοσηρότητας και θνησιμότητας. Μια υποεπιτροπή επομένως διορίστηκε, η οποία προετοίμασε ένα σχέδιο μιας προτεινόμενης στατιστικής ταξινόμησης για ασθένειες, τραυματισμοί και αιτίες θανάτου. Ένα τελικό σχέδιο υιοθετήθηκε από την επιτροπή αφότου ήταν τροποποιημένο επάνω στην βάση των δοκιμών που αναλαμβάνονται από τις διάφορες υπηρεσίες στον Καναδά, το Ηνωμένο Βασίλειο και τις Ηνωμένες Πολιτείες της Αμερικής.

4.6 Έκτη αναθεώρηση των διεθνών καταλόγων

Η διεθνής διάσκεψη υγείας που έγινε στην πόλη της Νέας Υόρκης τον Ιούνιο και Ιουλίου 1946 [46] εμπιστεύτηκε το μεσοδιάστημα αυτό στην Επιτροπή της οργάνωσης παγκόσμιας υγείας με την ευθύνη: αναθεώρησης των υπάρχοντων μηχανισμών και της ανάληψης τέτοιας προπαρασκευαστικής εργασίας σχετικά με: (i) την επόμενη αναθεώρηση «των διεθνών καταλόγων αιτιών θανάτου (συμπεριλαμβανομένων των καταλόγων που υιοθετούνται κάτω από το διεθνή συμφωνία του 1934, σχετικά με τις στατιστικές των αιτιών θανάτου) και (ii) η καθιέρωση των διεθνών καταλόγων αιτιών της νοσηρότητας. Για να εκπληρώσει αυτήν την υποχρέωση, η προσωρινή Επιτροπή ανέθεσε στην ειδική Επιτροπή για την προετοιμασία της έκτης αναθεώρησης των διεθνών καταλόγων ασθενειών και αιτιών θανάτου. Αυτή η Επιτροπή, που λαμβάνει πλήρως υπόψη την επικρατούσα άποψη σχετικά με την ταξινόμηση νοσηρότητας και θνησιμότητας, αναθεώρησε την προαναφερθείσα προτεινόμενη ταξινόμηση που προετοιμάστηκε από την Επιτροπή των κοινών αιτιών θανάτου των Ηνωμένων Πολιτειών. Η προκύπτουσα ταξινόμηση κυκλοφόρησε στις εθνικές κυβερνήσεις που προετοιμάζουν τις στατιστικές νοσηρότητας και θνησιμότητας για σχόλια και προτάσεις υπό τον τίτλο, *Διεθνής ταξινόμηση των ασθενειών, των τραυματισμών, και των αιτιών θανάτου*. Η ειδική Επιτροπή εξέτασε τις απαντήσεις και προετοίμασε μια αναθεωρημένη έκδοση που ενσωματώνει τέτοιες αλλαγές όπως εμφανίζονται για να βελτιώνει της χρησιμότητας και την αποδοχή της ταξινόμησης. Η Επιτροπή σύνταξε επίσης έναν κατάλογο από διαγνωστικούς όρους για να εμφανιστεί κάτω από κάθε τίτλο της ταξινόμησης. Επιπλέον, μια υποεπιτροπή διορίστηκε που προετοίμασε έναν περιεκτικό αλφαβητικό δείκτη των διαγνωστικών

δηλώσεων που ταξινομούνται στην κατάλληλη κατηγορία Η Επιτροπή εξέτασε επίσης τη δομή και τις χρήσεις των πρόσθετων καταλόγων αιτιών για την ταξινόμηση σε πίνακα και τη δημοσίευση των στατιστικών νοσηρότητας και θνησιμότητας και μελέτησαν άλλα προβλήματα που αφορούσαν τη διεθνή συγκρισιμότητα των στατιστικών θνησιμότητας, όπως η μορφή ιατρικού πιστοποιητικού και κανόνων για την ταξινόμηση. Η Διεθνής Διάσκεψη για την έκτη αναθεώρηση των διεθνών καταλόγων ασθενειών και των αιτιών θανάτου δημιουργήθηκε στο Παρίσι από τις 26 έως τις 30 Απριλίου 1948 από την κυβέρνηση της Γαλλίας υπό τον όρο της συμφωνίας που υπογράφηκε στο πλαίσιο της πέμπτης διάσκεψης για αναθεώρηση το 1938. Η γραμματεία της εμπιστεύτηκε από κοινού στις ικανές Γαλλικές αρχές και στην οργάνωση παγκόσμιας υγείας, η οποία είχε εκτελέσει την προπαρασκευαστική εργασία υπό τον όρο της ρύθμισης που ολοκληρώνεται από τις κυβερνήσεις που αντιπροσωπεύονται στη διεθνή διάσκεψη υγείας το 1946 [46]. Η διάσκεψη υιοθέτησε την ταξινόμηση που προετοιμάστηκε από την ειδική Επιτροπή ως την έκτη αναθεώρηση [47]. Εξέτασε επίσης άλλες προτάσεις της ειδικής Επιτροπής σχετικά με τη σύνταξη, ταξινόμηση σε πίνακα και δημοσίευση των στατιστικών νοσηρότητας και θνησιμότητας. Η διάσκεψη ενέκρινε τη διεθνή μορφή του ιατρικού πιστοποιητικού της αιτίας θανάτου, δέχτηκε την ελλοχεύουσα αιτία του θανάτου ως κύρια αιτία που ταξινομείται σε πίνακες, και επικύρωσε τους κανόνες για την ελλοχεύουσα αιτία του θανάτου καθώς επίσης και τους πρόσθετους καταλόγους για την ταξινόμηση σε πίνακα σαν στοιχεία νοσηρότητας και θνησιμότητας. Περαιτέρω σύστησε ότι η συνέλευση παγκόσμιας υγείας πρέπει να εκδώσει τους κανονισμούς σύμφωνα με το άρθρο 21 (β) του συνταγματικού χάρτη της παγκόσμιας οργάνωσης υγείας για να καθοδηγήσει τα κράτη μέλη στη σύνταξη της νοσηρότητας και της θνησιμότητας σύμφωνα με τη διεθνή στατιστική ταξινόμηση. Το 1948, η πρώτη συνέλευση παγκόσμιας υγείας επικύρωσε την έκθεση της έκτης διάσκεψης αναθεώρησης και υιοθέτησε κανονισμούς, προετοιμασμένους βάσει των συστάσεων της διάσκεψης. Η διεθνής ταξινόμηση, συμπεριλαμβανομένου του συνοπτικού καταλόγου συνυπολογισμών που καθορίζουν το περιεχόμενο των κατηγοριών, ήταν ενσωματωμένος, μαζί με τη μορφή του ιατρικού πιστοποιητικού της αιτίας του θανάτου, οι κανόνες για την ταξινόμηση και πρόσθετοι κατάλογοι για την ταξινόμηση σε πίνακα, *Εγχειρίδιο της διεθνούς στατιστικής ταξινόμησης των ασθενειών, τραυματισμοί, και Αιτίες του θανάτου* [48]. Το εγχειρίδιο αποτελείται από δύο τόμους, ο τόμος 2 που είναι ένας αλφαβητικός

δείκτης διαγνωστικών όρων που κωδικοποιούνται στις κατάλληλες κατηγορίες. Η έκτη διάσκεψη αναθεώρησης χαρακτήρισε την αρχή μιας νέας εποχής ζωτικής σημασίας για τις στατιστικές υγείας. Εκτός από την έγκριση ενός περιεκτικού καταλόγου και για τη θνησιμότητα και για τη νοσηρότητα και τη συμφωνία σχετικά με διεθνή κανόνες για την ελλοχεύουσα αιτία του θανάτου, σύστησαν την υιοθέτηση ενός σημαντικού προγράμματος διεθνούς συνεργασία στον τομέα στατιστικών της υγείας. Ένα σημαντικό στοιχείο σε αυτό το πρόγραμμα ήταν η σύσταση ότι οι κυβερνήσεις καθιερώνουν τις εθνικές επιτροπές ζωτικής σημασίας και στατιστικών υγείας για να συντονίσουν στατιστικές δραστηριότητες στη χώρα, και για να χρησιμεύσει ως μια σύνδεση μεταξύ των εθνικών στατιστικών οργάνων και της οργάνωσης παγκόσμιας υγείας. Περαιτέρω προβλέφθηκε ότι τέτοιες εθνικές επιτροπές, είτε μεμονωμένα είτε σε συνεργασία με άλλες εθνικές επιτροπές, μελετήσουν τα σημαντικά στατιστικά προβλήματα της δημόσιας υγείας και να θέσουν τα αποτελέσματα τους στην διάθεση της οργάνωσης παγκόσμιας υγείας

4.7 Οι έβδομη και όγδοη αναθεώρηση

Η Διεθνής Διάσκεψη για την έβδομη αναθεώρηση της διεθνούς ταξινόμησης των ασθενειών έγινε στο Παρίσι υπό την αιγίδα της πρόγνωσης παγκόσμιας υγείας τον Φεβρουαρίου 1955 [49]. Σύμφωνα με μια σύσταση του εμπειρογνώμονα της οργάνωσης παγκόσμιας υγείας η Επιτροπή στατιστικών υγείας, σε αυτή η αναθεώρηση περιορίστηκε στις ουσιαστικές αλλαγές και τις τροποποιήσεις των λαθών και ασυνεπειών [50]. Η όγδοη διάσκεψη αναθεώρησης που συγκλήθηκε από το την οργάνωση παγκόσμιας υγείας συναντήθηκε στη Γενεύη, από τις 6 έως τις 12 Ιουλίου 1965 [51]. Αυτή η αναθεώρηση ήταν περισσότερο ριζοσπαστική από τον έβδομη αλλά που αφήνει αμετάβλητη τη βασική δομή της ταξινόμησης και την γενική φιλοσοφία να ταξινομήσει τις ασθένειες, όποτε είναι δυνατόν, σύμφωνα με την αιτιολογία τους παρά μιας ιδιότητας εκδήλωσης τους. Κατά τη διάρκεια των ετών οι έβδομες και όγδοες αναθεωρήσεις του ICD ήταν σε ισχύ, η χρήση του ICD για την ευρετηρίαση των ιατρικών αρχείων νοσοκομείων αυξήθηκαν ραγδαία και μερικές χώρες προετοίμασαν τις εθνικές προσαρμογές που παρείχαν λεπτομέρεια που απαιτείται επιπρόσθετα για την χώρα αυτήν στην εφαρμογή του ICD.

4.8 Η ένατη αναθεώρηση

Η Διεθνής Διάσκεψη για την ένατη αναθεώρηση της διεθνούς ταξινόμησης των ασθενειών, που συγκαλείται από την της οργάνωση παγκόσμιας υγείας, που συναντιέται στη Γενεύη από τις 30 Σεπτεμβρίου έως 6 Οκτωβρίου 1975 [52]. Στις συζητήσεις που καταλήγουν στη διάσκεψη, αυτή αρχικά τονίζουν ότι πρέπει να υπάρξει μικρή αλλαγή εκτός από την ενημέρωση της ταξινόμησης. Αυτό ήταν κυρίως λόγω της προσαρμογής των συστημάτων επεξεργασίας δεδομένων κάθε φορά που αναθεωρήθηκε η ταξινόμηση. Εκεί υπάρχει όντας μια τεράστια αύξηση του ενδιαφέροντος για το ICD και των τρόπων κωδικοποίησης έπρεπε να βρει λοιπόν τις απαντήσεις σε αυτό, εν μέρει τις βρήκε τροποποιώντας την ίδια την ταξινόμηση και εν μέρει με να εισαγάγει τις πρόσθετες κωδικοποιήσεις που έγιναν από τους ειδικούς οργανισμούς που είχαν γίνει πλέον οι ενδιαφερόμενοι για χρησιμοποίηση του ICD για τις στατιστικές τους. Μερικές θεματικές περιοχές στην ταξινόμηση θεωρήθηκαν πως έπρεπε να τακτοποιούνται και να υπάρξε ιδιαίτερη πίεση για περισσότερη λεπτομέρεια και για την προσαρμογή της ταξινόμησης για να το καταστήσει πιο σχετικό για την αξιολόγηση της ιατρικής φροντίδας, ταξινομώντας τους όρους στα ενδιαφερόμενα κεφάλαια για το μέρος του σώματος που επηρεάζεται παρά σε εκείνη της εξέτασης η κάτω από γενικευμένη ασθένεια. Από την άλλη πλευρά της κλίμακας, υπήρξαν αντιπροσωπεύσεις από τις χώρες και περιοχές όπου μια λεπτομερής και περίπλοκη ταξινόμηση ήταν άσχετη, αλλά που εντούτοις χρειάστηκαν μια ταξινόμηση με βάση το ICD προκειμένου να αξιολογηθεί η πρόοδός τους στην υγειονομική περίθαλψη και στον έλεγχο της ασθένειας. Οι τελικές προτάσεις που υποβλήθηκαν και που έγιναν αποδεκτές από τη διάσκεψη διατήρησαν και αποτελούν τη βασική δομή του ICD, αν και με πολλή πρόσθετη λεπτομέρεια στο επίπεδο των τετρανήφίων υποκατηγοριών, και μερικές προαιρετικές πέντε ψηφίων υποδιαίρέσεις. Προς όφελος των χρηστών που δεν απαιτούν τέτοια λεπτομέρεια, η προσοχή λήφθηκε για να εξασφαλίσει ότι οι κατηγορίες στο τρινήφιο επίπεδο ήταν αρκετός. Προς όφελος των χρηστών που επιθυμούν να παραγάγει τις στατιστικές και τους δείκτες προσανατολισμού προς την ιατρική φροντίδα, η ένατη αναθεώρηση περιέλαβε μια προαιρετική εναλλακτική μέθοδο τις διαγνωστικές δηλώσεις, συμπεριλαμβανομένων των πληροφοριών και για ελλοχεύουσα γενικής ασθένειας και μιας εκδήλωση σε ένα ιδιαίτερη όργανο ή μια περιοχή. Αυτό το σύστημα έγινε γνωστό ως το σύστημα στιλέτων και αστερίσκων και διατηρείται στη δέκατη αναθεώρηση. Διάφορες

άλλες τεχνικές καινοτόμες ήταν συμπεριλαμβανόμενες στην αναθεώρηση, που στοχεύει στην αύξηση της ευελιξίας του για τη χρήση σε ποικίλες καταστάσεις. Η εικοστή ένατη συνέλευση παγκόσμιας υγείας, που σημειώνει τις συστάσεις της Διεθνούς Διάσκεψης για την ένατη αναθεώρηση της διεθνούς ταξινόμησης των ασθενειών, ενέκρινε τη δημοσίευση, για δοκιμαστικούς λόγους, συμπληρωματικές ταξινομήσεις «Impairments» και των αναπηριών και των διαδικασιών στην ιατρική όπως συμπληρώματα, αλλά όχι ως αναπόσπαστα τμήματα, της διεθνούς ταξινόμησης των ασθενειών. Οι συστάσεις των διασκέψεων επίσης σχετικά με διάφορα σχετικά τεχνικά θέματα: οι κανόνες κωδικοποίησης για τη θνησιμότητα τροποποιήθηκαν ελαφρώς και κανόνες για την επιλογή από μια ενιαία αιτία για την ταξινόμηση σε πίνακα της νοσηρότητας εισήχθησαν για πρώτη φορά ορισμοί και συστάσεις για οι στατιστικές στον τομέα της προγενέθλιας θνησιμότητας τροποποιήθηκαν και επεκτάθηκαν και ένα πιστοποιητικό των αιτιών του προγενέθλιου θανάτου συστήθηκε και οι χώρες ενθαρρύνθηκαν για να κάνουν την περαιτέρω εργασία για την κωδικοποίηση πολλαπλού-όρου και την ανάλυση, αλλά και αρίθμηση ασθενειών. Οι επιστημονικές μέθοδοι συστήθηκαν και ένας νέος βασικός κατάλογος ταξινόμησης σε πίνακα παρήχθη.

4.9 Προετοιμασίες για τη δέκατη αναθεώρηση

Ακόμη και πριν από τη διάσκεψη για την ένατη αναθεώρηση, ο παγκόσμιος οργανισμός υγείας προετοιμαζόταν για τη δέκατη αναθεώρηση. Είχαν συνειδητοποιήσει ότι η μεγάλη επέκταση στη χρήση του ICD απαιτούσε μια λεπτομερή επανεξέταση της δομής της και προσπάθεια να επινοηθεί μια σταθερή και εύκαμπτη ταξινόμηση, η οποία δεν πρέπει να απαιτεί τη θεμελιώδη αναθεώρηση για πολλά χρόνια. Τα συνεργαζόμενα κέντρα της παγκόσμιας οργάνωσης υγείας για την ταξινόμηση των ασθενειών (δείτε τον τόμο 1, σελ. 7-8) ήταν συνεπώς κλιμένους για να πειραματιστεί με τα πρότυπα των εναλλακτικών δομών για icd-10. Είχε γίνει επίσης σαφές ότι το καθιερωμένο δεκαετές διάστημα μεταξύ των αναθεωρήσεων ήταν πάρα πολύ σύντομο. Εργασία για την διαδικασία αναθεώρησης έπρεπε να αρχίσει προτού η τρέχουσα έκδοση του ICD ήταν σε λειτουργία για αρκετό χρονικό διάστημα για να είναι λεπτομερώς αξιολογημένος, κυρίως επειδή η ανάγκη να ερωτηθούν τόσες πολλές χώρες και οργανώσεις έκανε τη διαδικασία πολύ χρονοβόρα. Ο Γενικός Διευθυντής του Παγκοσμίου Οργανισμού Υγείας επομένως έγραψε στα κράτη

μέλη και έλαβε τη συγκατάθεση τους ξαναβάλτε έως το 1989 τη δέκατη διάσκεψη αναθεώρησης, η οποία σχεδιάστηκε αρχικά για το 1985 και για να καθυστερήσει εισαγωγή της δέκατης αναθεώρησης που θα αναμενόταν το 1989. Εκτός από το να επιτρέψει τον πειραματισμό με τα εναλλακτικά πρότυπα για τη δομή του ICD, αυτός ήταν ο χρόνος για την αξιολόγηση icd-9, για το παράδειγμα μέσω των συνεδριάσεων που οργανώνονται από μερικά από τα περιφερειακά γραφεία παγκοσμίου οργανισμού υγείας και μέσω μιας έρευνας που οργανώνεται στην έδρα. Ένα εκτενές πρόγραμμα εργασίας ακολούθησε το οποίο κατέληξε στη δέκατη αναθεώρηση του ICD και περιγράφεται μέσα στην έκθεση της Διεθνούς Διάσκεψης για τη δέκατη αναθεώρηση της διεθνούς ταξινόμησης των ασθενειών, στον τόμο 1.

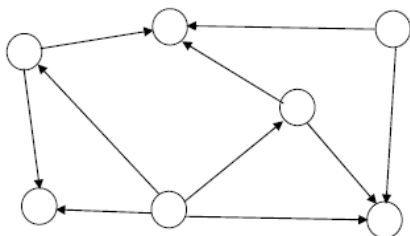
Κεφάλαιο 5 Υλοποίηση

5.1 Θεωρία γράφων

Η πιο γενική μορφή δομής δεδομένων, με την έννοια ότι όλες οι άλλες δομές μπορούν να θεωρηθούν ως περιπτώσεις γράφων.

Ένα γράφος αποτελείται από

- ένα σύνολο V κορυφών (vertices), ή σημείων, ή κόμβων , και
- ένα σύνολο E ακμών (edges), ή τόξων, ή γραμμών. Μια ακμή είναι ένα ζεύγος (u,v) από κορυφές.



Σχήμα 5.1 – Παράδειγμα γράφου

Οι γράφοι προσφέρουν μια χρήσιμη μέθοδο για τη διατύπωση και λύση πολλών προβλημάτων, σε

- δίκτυα και συστήματα τηλεπικοινωνιών (π.χ. το Internet),
- χάρτες - επιλογή δρομολογίων,
- προγραμματισμό εργασιών (scheduling),
- ανάλυση προγραμμάτων (flow charts).

Η θεωρία των γράφων θεωρείται ότι ξεκίνησε από τον Euler στις αρχές του 18ου αιώνα (1736).

5.1.1 Ορισμοί

Ένας γράφος ονομάζεται κατευθυνόμενος (directed graph, digraph) αν κάθε μια από τις ακμές του είναι προσανατολισμένη προς μία κατεύθυνση. Ένας γράφος ονομάζεται *μη-κατευθυνόμενος* (undirected) αν οι ακμές του δεν είναι προσανατολισμένες.

Αν (u,v) είναι ακμή τότε λέμε ότι οι κορυφές u και v είναι γειτονικές (adjacent) ή ότι γειτνιάζουν.

Μονοπάτι ή *διαδρομή* (path) ενός γράφου μήκους n , είναι μια ακολουθία κόμβων $v_0, v_1, \dots, v_n$, όπου για κάθε i , $0 \leq i < n$, (v_i, v_{i+1}) είναι ακμή του γράφου. *Μήκος* ενός μονοπατιού είναι ο αριθμός ακμών που περιέχει.

Μια διαδρομή ενός γράφου ονομάζεται *απλή* (simple) αν όλες οι κορυφές της είναι διαφορετικές μεταξύ τους, εκτός από την πρώτη και την τελευταία οι οποίες μπορούν να είναι οι ίδιες.

Κύκλος (cycle) ονομάζεται μια διαδρομή με μήκος >1 που ικανοποιεί $v_0 = v_n$.

Ένας γράφος που δεν περιέχει κύκλους ονομάζεται άκυκλος (acyclic). Έστω $G=(V,E)$ και $G'=(V', E')$ γράφοι, όπου $V' \subseteq V$ και $E' \subseteq E$. Τότε ο γράφος G' είναι υπογράφος (subgraph) του γράφου G . Η απόσταση δύο κορυφών είναι το μήκος της συντομότερης διαδρομής που οδηγεί από τη μια κορυφή στην άλλη. Ένας μη κατευθυνόμενος γράφος λέγεται συνεκτικός (connected) αν για κάθε ζευγάρι κορυφών υπάρχει διαδρομή που τις συνδέει. Ένας κατευθυνόμενος γράφος που ικανοποιεί την ίδια ιδιότητα ονομάζεται ισχυρά συνεκτικός (strongly connected). Αν ο μη-κατευθυνόμενος γράφος στον οποίο αντιστοιχεί είναι συνεκτικός, τότε ο γράφος ονομάζεται ελαφρά συνεκτικός (weakly connected).

Ένας γράφος λέγεται αραιός (sparse) αν ο αριθμός των ακμών του είναι της τάξης $O(n)$, όπου n είναι ο αριθμός κορυφών του, διαφορετικά λέγεται πυκνός (dense). Συχνά συσχετίζουμε κάθε ακμή ενός γράφου με κάποιο βάρος (weight). Τότε ο γράφος ονομάζεται γράφος με βάρη (weighted graph).

5.1.2 Αναπαράσταση γράφων με πίνακες γειτνίασης (adjacency matrix)

Ένας γράφος $G=(V,E)$ με n κορυφές μπορεί να αναπαρασταθεί ως ένας $n \times n$ πίνακας που περιέχει τις τιμές 0 και 1, και όπου

– αν η (i,j) είναι ακμή τότε $A[(i,j)]=1$, διαφορετικά $A[(i,j)]=0$.

Αν ο γράφος είναι γράφος με βάρη, και το βάρος κάθε ακμής είναι τύπου t , τότε για την αναπαράσταση του γράφου μπορεί να χρησιμοποιηθεί πίνακας τύπου t με

– $A[(i,j)] = \text{βάρος}(i,j)$, αν υπάρχει ακμή (i,j)

– $A[(i,j)] = \infty$, αν δεν υπάρχει ακμή (i,j)

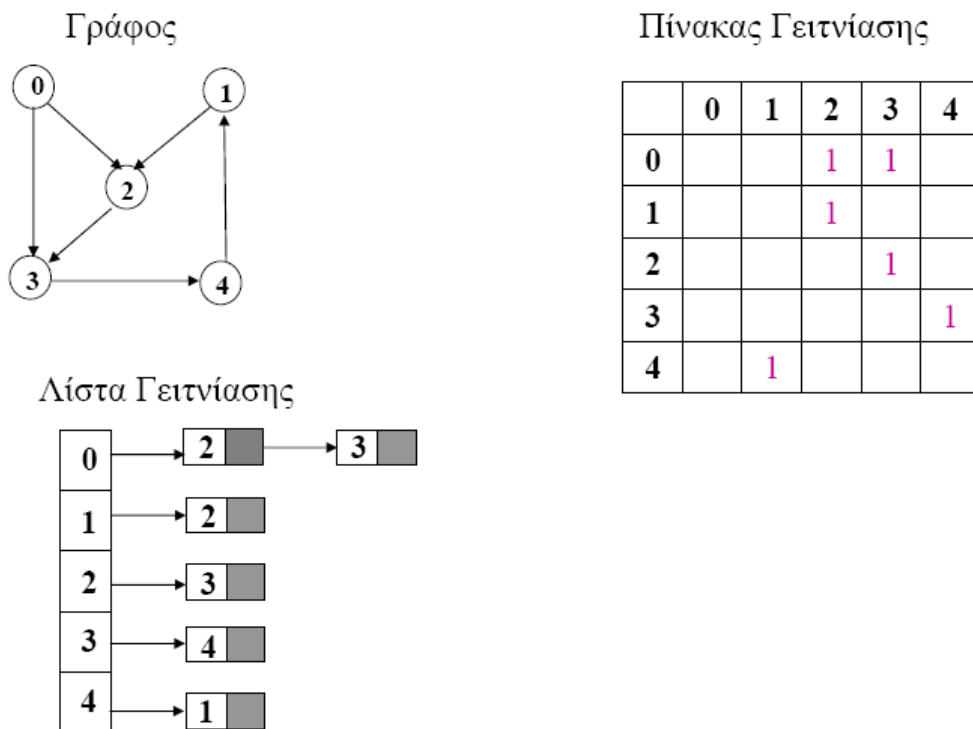
Αυτή η αναπαράσταση απαιτεί χώρο $O(n^2)$, όπου $n=|V|$. Αν ο γράφος είναι αραιός η μέθοδος οδηγεί σε σπάταλη χώρα.

5.1.3 Αναπαράσταση γράφων με λίστες γειτνίασης (adjacency lists)

Ένας γράφος $G=(V,E)$ αναπαρίσταται ως ένας μονοδιάστατος πίνακας

Για κάθε κορυφή w , $A[w]$ είναι ένας δείκτης σε μια συνδεδεμένη λίστα στην οποία αποθηκεύονται οι κορυφές που γειτνιάζουν με την w . Η μέθοδος απαιτεί χώρο $O(|V| + |E|)$.

- Επιτυγχάνεται εξοικονόμηση χώρου για αραιούς γράφους.
- Στην περίπτωση γράφων με βάρη στη λίστα γειτνίασης αποθηκεύουμε επίσης το βάρος κάθε ακμής.



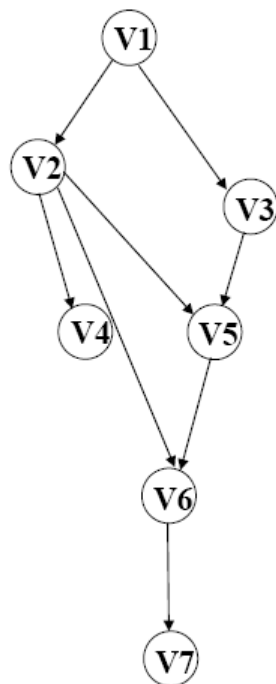
Σχήμα 5.2 – Παράδειγμα γράφου και γειτνιάσεων

5.1.4 Τοπολογική ταξινόμηση

Δίνεται ένα σύνολο εργασιών και θέλουμε να ορίσουμε τη σειρά με την οποία πρέπει να εκτελέσει τις εργασίες ένας επεξεργαστής, δεδομένης της ύπαρξης περιορισμών ως προς την προτεραιότητά τους. Κάθε εργασία έχει ένα σύνολο προαπαιτούμενων εργασιών, δηλαδή δεν μπορεί να εκτελεσθεί προτού συμπληρωθεί κάθε μια από τις προαπαιτούμενες. Μπορούμε να παραστήσουμε το πρόβλημα ως έναν κατευθυνόμενο γράφο:

- Οι κορυφές του γράφου αντιστοιχούν σε κάθε μια από τις εργασίες, και
- η ύπαρξη ακμής από την κορυφή A στην κορυφή B δηλώνει ότι η εργασία A πρέπει να εκτελεστεί πριν από τη B.

Τυπολογική ταξινόμηση του γράφου είναι μια σειρά των κορυφών του, $v_1, \dots, v_n$, ώστε αν (v_i, v_j) είναι ακμή του γράφου τότε $i < j$.



Τοπολογικές Ταξινομήσεις του γράφου:

V1, V2, V4, V3, V5, V6, V7

V1, V2, V3, V5, V6, V7, V4

V1, V3, V5, V2, V6, V4, V7

...

Σχήμα 5.3 – Παράδειγμα τοπολογικής ταξινόμησης γράφου

5.1.5 Αλγόριθμος; Τοπολογικής αναζήτησης

Ο βαθμός εισόδου (in-degree) ενός κόμβου είναι ο αριθμός των ακμών που καταλήγουν στον κόμβο. (Στο πρόβλημα μας, ο αριθμός των προαπαιτούμενων εργασιών) Για κάθε κορυφή u έστω $I[u]$ ο βαθμός εισόδου της u .

Επαναλαμβάνουμε τα εξής βήματα:

1. διαλέγουμε κορυφή A με $I[A]=0$,
2. τυπώνουμε την A ,
3. για όλες τις κορυφές B , με (A,B) μειώνουμε την τιμή $I[B]$ κατά 1.

5.1.6 Διάχυση γράφων

Αν θέλουμε να επισκεφτούμε όλους τους κόμβους ενός γράφου μπορούμε να χρησιμοποιήσουμε έναν από πολλούς τρόπους, οι οποίοι διαφέρουν στη σειρά με την οποία εξετάζουν τους κόμβους.

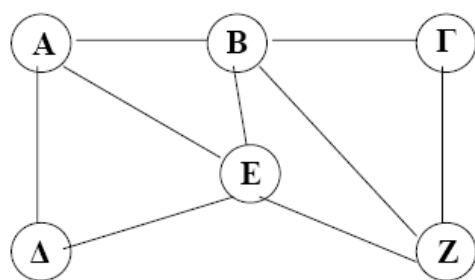
Διαδικασίες διάσχισης χρησιμοποιούνται για τη διακρίβωση ύπαρξης μονοπατιού μεταξύ δύο κόμβων κ.α.

Depth-First Search (Αναζήτηση κατά βάθος)

Γενίκευση της προθεματικής διάσχισης δένδρων: Ξεκινώντας από ένα κόμβο v , επισκεπτόμαστε πρώτα τον v και ύστερα καλούμε αναδρομικά τη διαδικασία στο καθένα από τα παιδιά του. Θα διατηρήσουμε ένα πίνακα Visited ο οποίος θα κρατά πληροφορίες ως προς το ποιους κόμβους έχουμε επισκεφθεί ανά πάσα στιγμή.

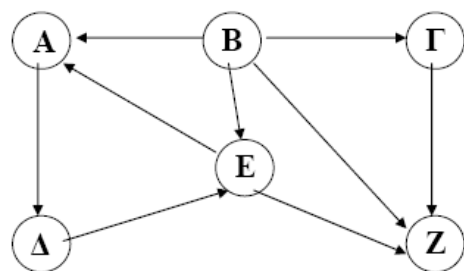
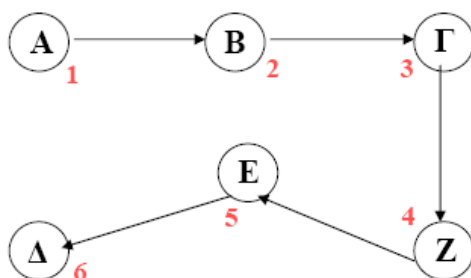
Προθεματική Διάσχιση

Αν ο γράφος δεν είναι συνεκτικός αυτή η στρατηγική πιθανό να αγνοήσει μερικούς κόμβους. Αν ο στόχος μας είναι να επισκεφθούμε όλους τους κόμβους τότε μετά το τέλος της εκτέλεσης του DFS(v) θα πρέπει να ελέγξουμε τον πίνακα Visited να βρούμε τους κόμβους που δεν έχουμε επισκεφθεί και να καλέσουμε σε αυτούς τη διαδικασία DFS:



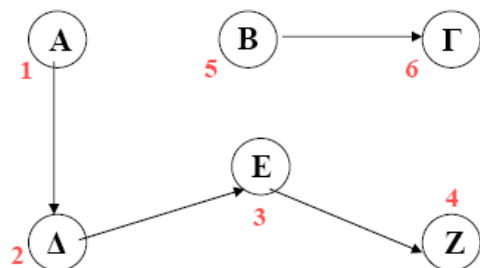
	A	B	Γ	Δ	E	Z	
A	0	1	0	1	1	0	A
B	1	0	1	0	1	1	B
Γ	0	1	0	0	0	1	Γ
Δ	1	0	0	0	1	0	Δ
E	1	1	0	1	0	1	E
Z	0	1	1	0	1	0	Z

DepthFirstSearch(G, A)



	A	B	Γ	Δ	E	Z	
A	0	0	0	1	0	0	A
B	1	0	1	0	1	1	B
Γ	0	0	0	0	0	1	Γ
Δ	0	0	0	0	1	0	Δ
E	1	0	0	0	0	1	E
Z	0	0	0	0	0	0	Z

DepthFirstSearch(G, A)



Σχήμα 5.4 – Παράδειγμα προθεματικής διάχυσης γράφου

Η διαδικασία καλείται σε κάθε κόμβο το πολύ μια φορά. Ο χρόνος εκτέλεσης: $O(|V| + |E|)$, δηλαδή γραμμικός ως προς τον αριθμό των ακμών και κορυφών.

Αρίθμηση DFS των κορυφών ενός γράφου ονομάζεται η σειρά με την οποία επισκέπτεται η διαδικασία DepthFirstSearch τις κορυφές του γράφου. Η διαδικασία μπορεί να κληθεί και για μη-κατευθυνόμενους και για κατευθυνόμενους γράφους. Αντί με αναδρομή η διαδικασία μπορεί να υλοποιηθεί, ως συνήθως, με τη χρήση στοιβών.

5.2 Θεωρία δέντρων

Στις λίστες η εύρεση κόμβου έχει χρόνο εκτέλεσης $O(N)$, όπου N το μήκος της λίστας. Στα δένδρα τα δεδομένα είναι ‘καλύτερα’ οργανωμένα ώστε οι διαδικασίες αναζήτησης, εισαγωγής και εξαγωγής κόμβου να είναι πιο αποδοτικές.

Τα δένδρο είναι ένα σύνολο κόμβων που συνδέονται από ακμές ή τόξα. Ορίζεται αναδρομικά ως εξής:

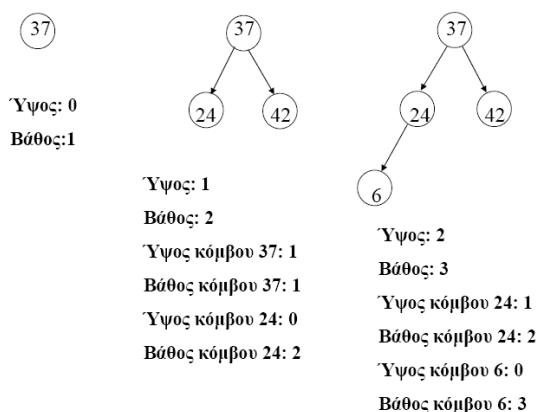
1. Ένα δένδρο είναι είτε κενό είτε
2. αποτελείται από μια ρίζα (root), δηλαδή ένα κόμβο στον οποίο δεν καταλήγουν αλλά μόνο ξεκινούν ακμές, και 0 ή περισσότερα υποδένω $T_1, T_2, \dots, T_k$, το καθένα ξεχωριστό από τα άλλα και από τη ρίζα. Υπάρχει μια ακμή από τη ρίζα στις ρίζες των $T_1, T_2, \dots, T_k$. Από κάθε κόμβο ξεκινούν 0 ή περισσότερες ακμές. Σε κάθε κόμβο εκτός της ρίζας καταλήγει μία μόνο ακμή. Στη ρίζα δεν καταλήγει καμιά ακμή. Οι κόμβοι περιέχουν δεδομένα. Οι ακμές επιβάλλουν μια ιεραρχική δομή στα δεδομένα.

5.2.1 Ορισμοί

Όταν υπάρχει ακμή από ένα κόμβο u σε ένα κόμβο v τότε ο u είναι ο πατέρας ή ο προκάτοχος (parent) του v , και ο v είναι παιδί (child) του u . Η ορολογία αυτή γενικεύεται ώστε να μιλούμε για προγόνους και απογόνους κόμβων. Κόμβοι που έχουν τον ίδιο προκάτοχο ονομάζονται αδέρφια. Κόμβοι που δεν έχουν παιδιά λέγονται φύλλα. Οι υπόλοιποι λέγονται εσωτερικοί. Βαθμός (degree) ενός κόμβου είναι ο αριθμός των παιδιών του. Βαθμός ενός δένδρου είναι ο μέγιστος από τους βαθμούς των κόμβων του. Δένδρα βαθμού n λέγονται n -άδικα. Δένδρα που χρησιμοποιούνται συχνά είναι τα δυαδικά (binary), τετραδικά (quad trees), ομαδικά (catters).

5.2.2 Ύψος και βάθος δέντρου

Επίπεδο ή βάθος ενός κόμβου ονομάζουμε τον αριθμό των προγόνων του + 1. π.χ. το βάθος της ρίζας ενός δένδρου είναι 1. Βάθος ενός δένδρου ονομάζουμε το μέγιστο επίπεδο κόμβων του δένδρου. Μονοπάτι ή διαδρομή (path) ενός δένδρου είναι μια ακολουθία κόμβων $v_1, v_2, \dots, v_k$, όπου κάθε v_i είναι πατέρας του v_{i+1} . Το μήκος του μονοπατιού $v_1, v_2, \dots, v_k$ είναι $k-1$. Ύψος ενός κόμβου ονομάζουμε το μήκος του μέγιστου μονοπατιού από τον κόμβο προς κάποιο φύλλο (μετρούμε ακμές). π.χ. το ύψος οποιουδήποτε φύλλου είναι 0. Ύψος ενός δένδρου είναι το ύψος της ρίζας του δένδρου. Διατεταγμένο δένδρο ονομάζεται ένα δένδρο όταν για κάθε παιδί u κάποιου κόμβου v γνωρίζουμε αν ο u είναι το k -οστό παιδί του v .

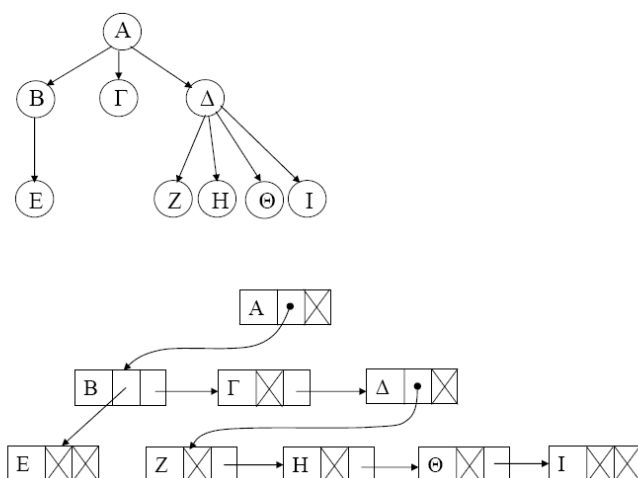


Σχήμα 5.5 – Παράδειγμα ύψους / βάθους γράφου

5.2.3 Αναπαράσταση δέντρων στην μνήμη

Προβλήματα αυτού του είδους αναπαράστασης είναι

1. αν το δένδρο δεν είναι πλήρες, τότε η μνήμη δεν αξιοποιείται αποτελεσματικά.
 2. πιθανόν ο βαθμός του δένδρου να είναι άγνωστος λύση: κάθε κόμβος u περιέχει τα δεδομένα (κλειδί), δείκτη στο πρώτο παιδί του u , δείκτη στον επί δεξιά αδελφό του u , αν υπάρχει.
- *Νέο πρόβλημα:* Πώς μπορούμε να επισκεφθούμε όλα τα παιδιά κάποιου κόμβου u ;



Σχήμα 5.6 – Αναπαράσταση δένδρων στην μνήμη

Αν θέλουμε να επισκεφθούμε όλους τους κόμβους ενός δένδρου, μπορούμε να χρησιμοποιήσουμε ένα από τους πιο κάτω τρόπους, οι οποίοι διαφέρουν στη

σειρά με την οποία εξετάζουν τους κόμβους. 1. Προθεματική Διάσχιση: (preorder traversal) επισκεπτόμαστε πρώτα τη ρίζα και ύστερα τα παιδιά της. Αναδρομικά η πράξη ορίζεται ως εξής:

```
Print_Preorder(*treenode u)
```

```
Output data at u;
```

```
for each child v of u
```

```
Print_Preorder(v)
```

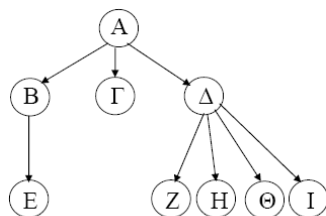
2. Επιθεματική Διάσχιση: (postorder traversal) επισκεπτόμαστε πρώτα τα παιδιά και ύστερα τη ρίζα του δένδρου. Αναδρομικά:

```
Print_Postorder(*treenode u)
```

```
for each child v of u
```

```
Print_Postorder(v)
```

```
output data at u;
```



Προθεματική Διάσχιση: A B E Γ Δ Z H Θ I

Μεταθεματική Διάσχιση: E B Γ Z H Θ I Δ A

Σχήμα 5.7 – Παράδειγμα προθεματικής / επιθεματικής διάχυσης γράφου

5.2.4 Δυαδικα δέντρα

Ένα δένδρο είναι δυαδικό αν όλοι οι κόμβοι του έχουν βαθμό ≤ 2 . Δυαδικό δένδρο λέγεται ένα δένδρο το οποίο είτε είναι κενό, είτε αποτελείται από μια ρίζα και δύο δυαδικά

υπόδενδρα, το καθένα διακριτό από το άλλο και από τη ρίζα. Αναφερόμαστε στα δύο υπόδενδρα ως το αριστερό και το δεξιό υπόδενδρο. Το ύψος ενός δυαδικού δένδρου με n κόμβους μπορεί να είναι το πολύ $n-1$ και το λιγότερο $\log n$.

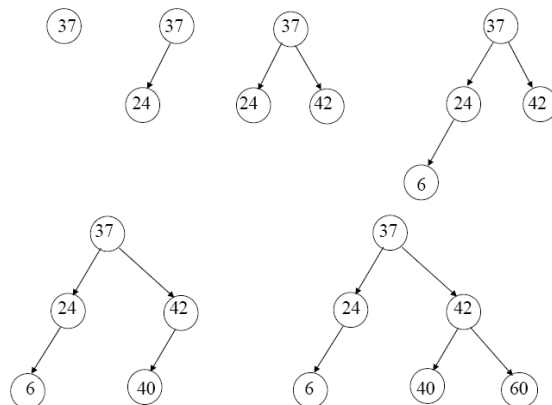
Ένα δυαδικό δένδρο είναι γεμάτο (full), αν κάθε εσωτερικός του κόμβος έχει δύο απογόνους.

Ένα δυαδικό δένδρο είναι τέλειο (perfect), αν είναι γεμάτο και όλοι του οι κόμβοι που βρίσκονται στο ίδιο επίπεδο έχουν το ίδιο βάθος.

Ένα δυαδικό δένδρο είναι πλήρες (full) αν

1. έχει ύψος 0 και ένα κόμβο,
2. έχει ύψος 1 και η ρίζα του έχει είτε δύο παιδιά είτε ένα αριστερό παιδί.
3. έχει ύψος h και η ρίζα του έχει ένα τέλειο αριστερό υπόδενδρο ύψους $h-1$ και ένα πλήρες δεξιό υπόδενδρο ύψους $h-1$, ή ένα πλήρες αριστερό υπόδενδρο ύψους $h-1$ και ένα τέλειο δεξιό υπόδενδρο ύψους $h-2$.

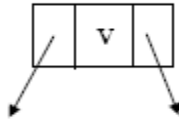
Παραδείγματα δυαδικών δένδρων



Σχήμα 5.8 – Παράδειγμα δυαδικών δέντρων

Αφού κάθε κόμβος σε ένα δυαδικό δένδρο έχει το πολύ δύο παιδιά, μπορούμε να κρατούμε δείκτες στο καθένα από αυτά. Δηλαδή, ένας κόμβος μπορεί να υλοποιηθεί ως μια εγγραφή BTreeNode με τρία πεδία (παρόμοια με κόμβο διπλά συνδεδεμένης λίστας).

1. **val**, όπου αποθηκεύουμε την πληροφορία του κόμβου,
2. **left**, τύπου pointer, ο οποίος δείχνει το αριστερό, υπόδενδρο που ριζώνει στον συγκεκριμένο κόμβο, και
3. **right**, τύπου pointer, ο οποίος δείχνει το δεξιό υπόδενδρο που ριζώνει στον συγκεκριμένο κόμβο.



Σχήμα 5.9 – Παράδειγμα αναπαράστασης κόμβου δυαδικών δέντρων

Έτσι, ένα δυαδικό δένδρο υλοποιείται ως ένας δείκτης στη ρίζα του δένδρου, δηλαδή δείκτης σε εγγραφή τύπου BTreeNode.

1. Ένα γεμάτο δυαδικό (μη άδειο) δένδρο με n εσωτερικούς κόμβους, έχει $n+1$ φύλλα.
2. Κάθε δυαδικό δένδρο με n κόμβους έχει $n+1$ null δείκτες

Το πιο σημαντικό πλεονέκτημα της χρήσης δυαδικών δένδρων η αποδοτική αναζήτηση σε ένα σύνολο στοιχείων.

Υποθέτουμε την ύπαρξη μιας σχέσης στο σύνολο των στοιχείων που επεξεργαζόμαστε, έστω τη σχέση $<$ πάνω στο σύνολο των ακεραίων. Ένα δυαδικό δένδρο αναζήτησης (ΔΔΑ) είναι ένα δυαδικό δένδρο κάθε κόμβος u του οποίου ικανοποιεί τα εξής:

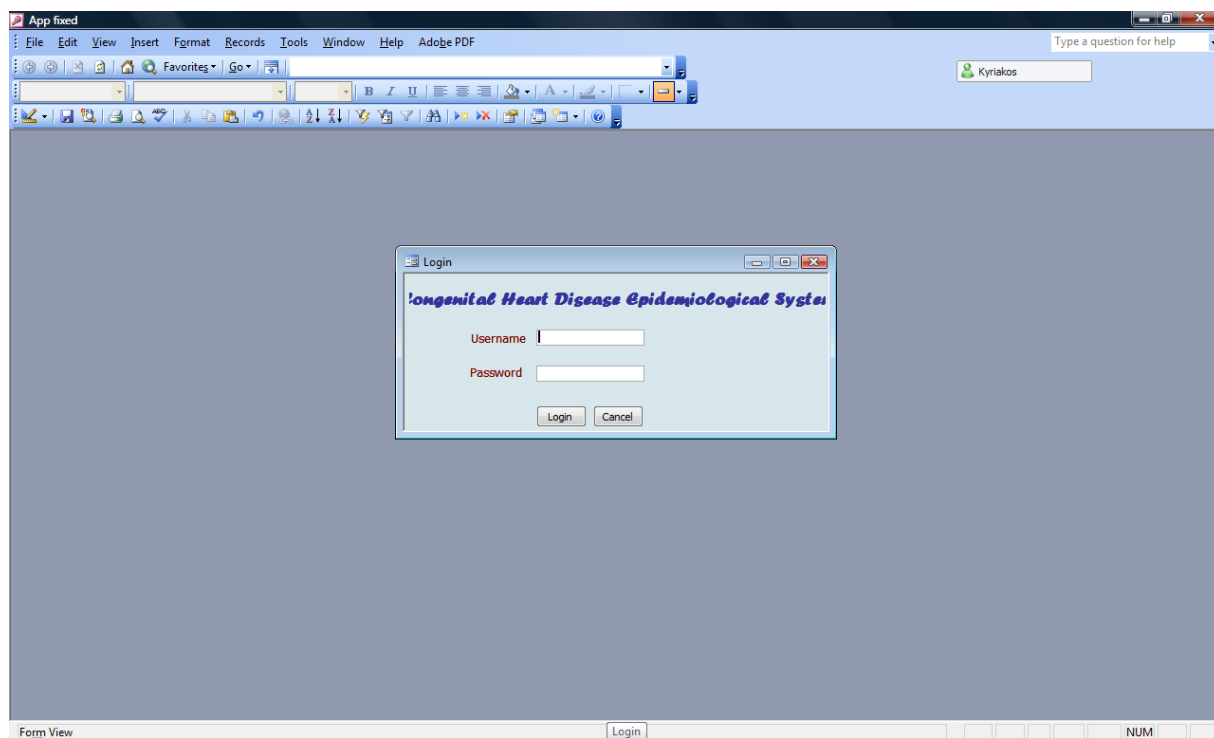
1. τα κλειδιά του αριστερού υποδένδρου του u είναι μικρότερα από το κλειδί του u

2. τα κλειδιά του δεξιού υποδένδρου του u είναι μεγαλύτερα από το κλειδί του u .

5.3 Πρώτη προσπάθεια με IDC

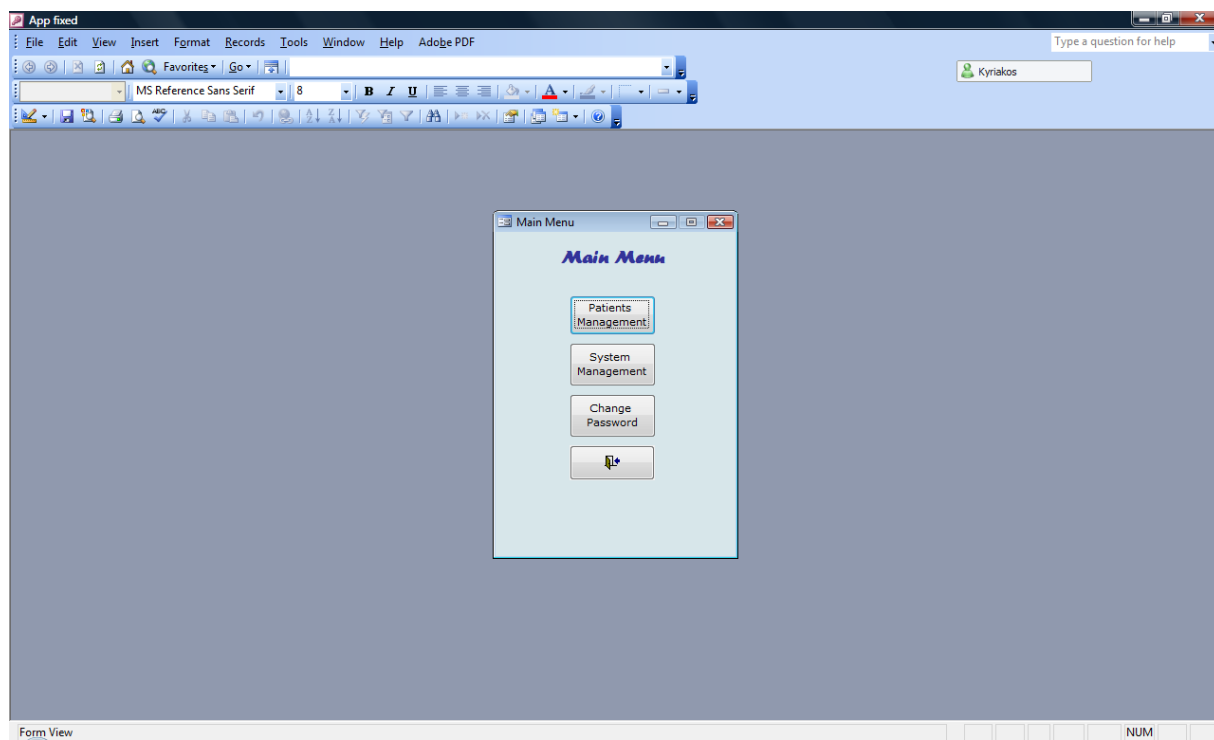
Η πρώτη επαφή μου με την κωδικοποίηση έρχεται με μια απαίτηση του πελάτη, στην περίπτωση μας ο Δρ. Αντώνης Ιωσήφ, να διορθώσω ένα υφιστάμενο πρόγραμμα σε Microsoft Access και να του προσθέσω την δυνατότητα κωδικοποίηση με βάση το IDC-10. Πιο κάτω με χρήση απεικονίσεων θα επεξηγήσω την αρχική κατάσταση του προγράμματος και την πορεία του μέχρι να επιτευχτεί ο στόχος. Επέτειο από πλευράς Freeman θα επεξηγηθεί η υλοποίηση ενός συστατικού που μπορεί να διαχειρίζεται την κωδικοποίηση ασθενών που υλοποιήθηκε σε Java και θα ενσωματωθεί σε αργότερο στάδιο στο FreeMed μιας και η ανάπτυξη του βρίσκεται σε ένα μεταβατικό στάδιο αυτή την στιγμή.

Ξεκινώντας την περιήγηση μας στο πρόγραμμα βλέπαμε τα εξής:



Σχήμα 5.10 – Login στο σύστημα

Μετά την επιτυχή εισαγωγή έχουμε την εξής οθόνη διαχείρισης



Σχήμα 5.11 – Οθόνη διαχείρισης

Μετά την επιλογή «Patient Management» που είναι αυτό που μας ενδιαφέρει φτάνουμε στην πιο κάτω οθόνη

PT ID	First Name	Last Name	ID Number	Birthdate	UN Num	CY ID	Father Name	Mother Name	Allocation	RS ID	Found
C8876	Nikoletta	Polikargou		3/11/2000	153872		Pantelis	Eleni Constan	Allocated	A	☐
C8891	Marina	Evripidou	1169477	10/9/2006	153405	116947	Filippos	Zoiro	Allocated	A	☐
C8890	Maria	Theodorou	93718	30/1/1995	153416	93718	Antreas	Skevi	Allocated	A	☐
C8889	Dimbriana	Ioannou		3/7/2004	153449		Tasos	Soulla	Allocated	A	☐
C8888	Nicolas	Fountoulis	1162244	29/4/2006	153451	116224	Michail	Irini Dimiriou	Allocated	A	☐
C8880	Giannakis	Andreou		2/10/2006	153591		Andreas	Antri Papetta	Allocated	A	☐
C8879	Gabrila	Neofitou		25/4/1997	153692		Panikos	Stella Nikolao	Allocated	A	☐
M7809	MYPANTA	KAKOYPOY		16/10/1999	115008				Allocated	G	☐
C8877	Melani	Charalambou		4/8/2006	153869		Antreas	Antri	Allocated	A	☐
C8894	Maria	Papadopoulou		27/7/2004	153349		Pavlos	Antonia	Allocated	A	☐
C8875	Constantina	Giangou		4/5/2006	153873		Loizos	Panagiota	Allocated	A	☐
C8874	Antreas	Kyriakou		13/8/2002	153904		Kyriakos	Pnagiota Kaki	Allocated	A	☐
C8873	Mariam	Michael		8/9/2005	153906		Iraklis	Antgoni Pana	Allocated	A	☐
C8872	Georgos	Adamidis	1015185	18/4/1988	153918	101518	Costas	Maritsa	Allocated	A	☐
C8871	Nektarios	Hadjipapa	66577108	16/5/2005	153968	665771	Antreas	Despo	Allocated	A	☐
C8870	ChrisaEkaterini	Kontou		25/11/2006	153977		Michalis	Antri	Allocated	A	☐
C8561	Maria	Cosma		27/5/1999	41104		Stefanos	Sotira	Allocated	A	☐
C8878	Ioannis	Kouallis		26/9/2006	153734		Pavlos	Panagiota	Allocated	A	☐
C8901	Ioanna	Michael		20/4/2006	153194		Efthimos	Panagiota	Allocated	A	☐

Σχήμα 5.12 – Αναζήτηση ασθενή

Όπου μπορούμε με βάση τα πεδία στα δεξιά να κάνουμε αναζήτηση και χρησιμοποιώντας τα πεδία στο πεδίο «Filters» να φιλτράρουμε περισσότερο την αναζήτηση μας. Όταν βρούμε τον ασθενή επιλέξουμε το ασθενή και πατάμε «Display Patient» Ως αποτέλεσμα εμφανίζεται η πιο κάτω οθόνη.

App fixed

File Edit View Insert Format Records Tools Window Help Adobe PDF

Type a question for help

Kyriakos

Tahoma 8

Patient

Name **Marina Evripidou** Birthdate **10/9/2006** Age **2 Years** Close

Patient ID **C8891** Allocation **Allocated** UN **153405**

Demographics Clinical Data Events

Demographics

First Name(Greek)	Last Name(Greek)	Demographics Completion
Mapiva	Eupimidou	Completed
Researcher ID A	Address No. 5 Menelaou Str.	PostalCode 4187
UN 153405	Town Or Village Ypsonas	District Limassol
Sex F	Home Phone 25390601	Mobile Phone 99561303
EN Last Name Evripidou	Fathers Work Phone	Mothers Work Phone 25821850
EN First Name Marina	Father Name Filippas	Mother Name Zoro
Middle Name Fi	Father ID Number 776671	Mother ID Number 813525
Date Of Birth 10/9/2006	Father Place Of Origin Limassol	Mother Place Of Origin Limassol
ID Number 1169477	Father Nationality CY	Mother Nationality CY
Patient Nationality CY-CY	Father Occupation Iektologos Michanik	Mother Occupation Secretary

Edit Save Cancel

Form View FLTR NUM

Σχήμα 5.13 – Στοιχεία ασθενή

Εδώ εμφανίζονται τα στοιχεία ασθενή. Αυτό που μας ενδιαφέρει ένα η οθονών που φαίνεται πιο κάτω και είναι η οθονή με τα κλινικά δεδομένα του ασθενή.

The screenshot shows a medical software application window titled 'App fixed'. The interface includes a menu bar (File, Edit, View, Insert, Format, Records, Tools, Window, Help, Adobe PDF) and a toolbar. The main area is divided into tabs: Demographics, Clinical Data, and Events. The 'Clinical Data' tab is selected, showing a form for patient Marina Evripidou (Patient ID: C8891, Birthdate: 10/9/2006, Age: 2 Years, Allocation: Allocated, UN: 153405). The form includes sections for Clinical Data (Date of Diagnosis: 24/11/2008, Time of Diagnosis: Postnatal, Birth Outcome: Alive, Twins: Single, Family History: No Fam Hist, Extracardiac Anomalies Existence: No, Patient Status: Alive, Cause Death), Narrative Diagnosis (No significant Heart disease, Epi: PFO), and Patient Coded Diagnosis (ICD-10, Other Code, Diagnosis Name). The Patient Coded Diagnosis section has buttons for Edit, Save, and Delete.

Σχήμα 5.14 – Οθονών καταχώρησης διάγνωσης

Στα αριστερά έχουμε τα βασικά κλινικά δεδομένα του ασθενή. Στην δεξιά στήλη έχουμε μια αρχική προσπάθεια για διάγνωση με βάση το IDC-10 με δυνατότητες τροποποίησης διαγραφής και αποθήκευσης μιας διάγνωσης ενός ασθενή. Η μεσαία στήλη βλέπουμε μια περιγραφής διάγνωσης. Πατώντας όμως το «edit» βλέπουμε ότι ενεργοποιούνται τα υπόλοιπα κουμπιά. Όπως πλέουμε όμως με την επιλογή μιας διάγνωσης από το combo box δεν εισάγεται σαν διάγνωση άρα αρχικά όπως μου δόθηκε ουσιαστικά δεν έχει καμία λειτουργικότητα όσο Αφώτα την διάγνωση. Ένα παράδειγμα της αδυναμίας είναι η πιο κάτω απεικόνιση

The screenshot shows a medical software interface titled 'App fixed'. The main window contains a form for 'Clinical Completion' which is set to 'Completed'. The form is organized into three main sections:

- Clinical Data:** Contains various dropdown menus and text fields. Key entries include 'Date Of Diagnosis' as 24/11/2006, 'Time of Diagnosis' as Postnatal, 'Birth Outcome' as Alive, 'Twins' as Single, 'Family History' as No Fam Hist, 'Extracardiac Anomalies Existence' as No, 'Patient Status' as Alive, and 'Cause Death' as an empty field. There are also buttons for 'Edit', 'Save', and 'Cancel' at the bottom of this section.
- Narrative Diagnosis:** A large text area for free-form text, currently empty.
- Patient Coded Diagnosis:** Includes fields for 'ICD-10', 'Other Code', and 'Diagnosis Name'. A list of diagnoses is visible, with 'Concordant AV co' selected. Buttons for 'Edit', 'Save', and 'Delete' are present.

At the bottom of the form, there are buttons for 'Display Clinical Data' and 'Add Coded Diagnosis'. The interface also features a standard menu bar (File, Edit, View, Insert, Format, Records, Tools, Window, Help) and a toolbar with various icons.

Σχήμα 5.15 – Καταχώρησης διάγνωσης

Ενώ έχω επιλέξει μια διάγνωση δεν φαίνεται να εισάγεται καν.

5.4 Δεύτερη προσπάθεια με IDC

Μέχρι το σχήμα 5.13 ακλουθούμε την ίδια διαδικασία. Στην οδονη βλέπουμε έχει προστεθεί το πεδίο qualifier.

App fixed

File Edit View Insert Format Records Tools Window Help Adobe PDF

Tahoma 8

Patient

Name **Nikoletta Polikarpou** Birthdate **3/11/2000** Age **8 Years**

Patient ID **C8876** Allocation **Allocated** UN **153872** **Close**

Demographics Clinical Data Events

Clinical Data

Date of Diagnosis **06/11/2006** Diagnostic Category **No significant Heart disease**

Time of Diagnosis **Postnatal** Certainty **Definitive**

Birth Outcome **Alive** Syndromes Existence **No**

Twins **Single** Syndromes

Family History **No Fam Hist**

Family Narrative History

Extracardiac Anomalies Existence **No** Miscellaneous

Extracardiac Anomalies

Patient Status **Alive** Date of Death

Cause Death

Patient Coded Diagnosis

ICD-10	Other Code	Diagnosis Name	Qualifier
		Trace MI	

Edit Save Delete

Form View Patient NUM

Σχήμα 5.16 – Καταχώρησης διάγνωσης qualifier

Μετά την επιλογή «edit» και μετά «Add Diagnosis» έχουμε μια οθόνη ομού με την πιο κάτω.

App fixed

File Edit View Insert Format Records Tools Window Help Adobe PDF

Type a question for help

Go Favorites Arial Greek 8 B I U

Patient

03.09 Clear

All Recently Used

Clinical Completion Completed

ICD-10	Other Cd	Diagnosis Name	Details
Q20.9	01.03.09	AV and/or VA connections abnormal,	AV and/or VA connections abnormal,
Q22.8	06.03.09	Straddling right AV valve in DIV,	Straddling tricuspid valve,
Q24.5	09.43.09	Intramural proximal coronary arterial course	Coronary artery: anomalous aortic origin /
Q40.8	14.03.09	Gastroesophageal reflux,	Non-cardiothoracic / vascular abnormality
I34.8	10.33.09	Mitral leaflet perforation,	Mitral valvar abnormality - acquired,
I27.8	10.13.09	Pulmonary vascular disease Heath & Edwards stage 1,	Pulmonary vascular disease,
D15.1	10.03.09	Benign heart tumour (DESCRIBE),	Heart tumour,

Search by Other Code

Search Display All Display Recently Used Deselect From Recently Used List

Narrative Diagnosis

Trace MI

Patient Coded Diagnosis

ICD-10	Other Code	Diagnosis Name	Qualifier
Q67.7	14.04.04	Pectus carinatum,	
Q24.5	04.04.04	Coronary sinus ostial stenosis,	
Q22.8	06.03.09	Straddling right AV valve in DIV,	

Edit Save Delete

Display Clinical Data Add Coded Diagnosis

Record: 14 2 of 3

Form View NUM

Σχήμα 5.17 – Καταχώρησης διάγνωσης qualifier

Στην πιο πάνω οθόνη βλέπουμε να λειτουργεί μετά από δίκες μ αλλαγές η εισαγωγή.

Δίστιχος όμως οι qualifiers δεν εμφανίζονται όπως αναμένεται με βάση τον αλγόριθμο μας.

The screenshot shows a medical software interface with a menu bar (File, Edit, View, Insert, Format, Records, Tools, Window, Help, Adobe PDF) and a toolbar. The main window is titled 'Patient' and displays patient information: Name (Nikoletta Polikarpou), Birthdate (3/11/2000), Age (8 Years), Patient ID (C8876), Allocation (Allocated), and UN (153872). Below this, there are tabs for Demographics, Clinical Data, and Events. The 'Clinical Data' tab is active, showing a list of diagnoses with columns for ICD-10, Other Cd, Diagnosis Name, and Details. A 'Patient Coded Diagnosis' section on the right shows a list of diagnoses with a dropdown menu for 'Qualifier' open, displaying options: - mild, - mid-moderate, - moderate, - moderate-severe, - severe, - critical, - ruled out, and unknown/ not rele. The 'Form View' button is visible at the bottom left, and 'FLTR' and 'NUM' buttons are at the bottom right.

ICD-10	Other Cd	Diagnosis Name	Details
Q32.4	03.03.05	Bronchial anatomy (DESCRIBE),	Position or morphology of thoraco-abdominal organs
Q89.0	03.07.00	Abnormal spleen,	Position or morphology of thoraco-abdominal organs
nc	09.15.23	Tricuspid aortic valve,	
Q22.4	06.01.92	Tricuspid valve stenosis,	Tricuspid stenosis,
Q23.3	06.02.91	Mitral regurgitation,	Mitral regurgitation,
not relevant		Arrangement of heart & organs, connections,	
not relevant		Tetralogy of Fallot and variants	

Σχήμα 5.18 – Καταχώρησης διάγνωσης qualifier αρχική διόρθωση

Τώρα οι qualifier δουλεύουν ως αναμένεται αλλά παρουσιάζεται το πρόβλημα που θα εξηγηθεί στην επόμενη προσπάθεια

5.5 Τρίτη προσπάθεια με IDC

The screenshot shows the 'App fixed' software interface. At the top is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Format', 'Records', 'Tools', 'Window', 'Help', and 'Adobe PDF'. Below the menu is a toolbar with various icons. The main window is titled 'Patient' and contains the following information:

- Name:** Nikoleta Polikarpou
- Birthdate:** 3/11/2000
- Age:** 8 Years
- Patient ID:** C8876
- Allocation:** Allocated
- UN:** 153872

Below this information are tabs for 'Demographics', 'Clinical Data', and 'Events'. The 'Clinical Data' tab is selected. It contains a 'Clear' button and a radio button for 'Recently Used'. Below this is a table with columns 'ICD-10', 'Other Cd', 'Diagnosis Name', and 'Details'. The table lists several diagnoses, including 'Bronchial anatomy (DESCRIBE)', 'Abnormal spleen', 'Tricuspid aortic valve', 'Tricuspid valve stenosis', and 'Mitral regurgitation'. A 'Search by' dropdown is at the bottom of the table.

To the right of the table is a 'Narrative Diagnosis' section with the text 'Trace MI'. Below this is a 'Patient Coded Diagnosis' section with a table that has columns 'ICD-10', 'Other Code', 'Diagnosis Name', and 'Qualifier'. The table lists 'Tricuspid valve stenosis' and 'Mitral regurgitation'. Below the table are 'Edit', 'Save', and 'Delete' buttons.

Σχήμα 5.19 – Καταχώρησης διάγνωσης qualifier αρχική διόρθωση

Ενώ όπως είδαμε πριν όλα δουλεύουν σωστά βλέπουμε ότι η πολλές διαγνώσεις προκαλούν πρόβλημα μιας και το combo box είναι bound στην γραμμή του πίνακα Selected Qualifier βλέπουμε ότι δεν παραμένουν οι σωστοί qualifiers σε σχέση με την επιλεγμένη διάγνωση. Το combo box ανάπαυσα στιγμή είναι γεμάτη με τους qualifier της τελευταίας διάγνωσης αν υπήρχα. Μετά την επιλογή όπως φαίνεται πιο πάνω εξαφανίζονται η προηγούμενα επιλεγμένοι qualifiers

5.6 Τετάρτη προσπάθεια με IDC

The screenshot shows the 'App fixed' software interface. At the top, there's a menu bar (File, Edit, View, Insert, Format, Records, Tools, Window, Help) and a search bar. Below the menu, a toolbar contains various icons. The main window is titled 'Patient' and displays patient information: Name (Nikoletta Polikarpou), Birthdate (3/11/2000), Age (8 Years), Patient ID (C8876), Allocation (Allocated), and UN (153872). There are tabs for Demographics, Clinical Data, and Events. The 'Clinical Data' tab is active, showing a list of diagnoses under 'Clinical Completion' and 'Completed'. The list includes ICD-10 codes, EPC codes, diagnosis names, and details. A 'Patient Coded Diagnosis' section on the right shows a selected diagnosis (Q22.4 06.01.92 Tricuspid valve stenosis) with a qualifier '- trace (functional)' and a qualifier code '2140.02'. There are buttons for Edit, Save, and Delete. A small inset window shows a preview of the patient's data.

ICD-10	EPC code	Diagnosis Name	Details
\	03.03.05	Bronchial anatomy (DESCRIBE),	Position or morphology of thoraco-abdominal organs
Q89.0	03.07.00	Abnormal spleen,	Position or morphology of thoraco-abdominal organs
nc	09.15.23	Tricuspid aortic valve,	
Q22.4	06.01.92	Tricuspid valve stenosis,	Tricuspid stenosis,
Q23.3	06.02.91	Mitral regurgitation,	Mitral regurgitation,
		not relevant Arrangement of heart & organs, connections,	
		not relevant Tetralogy of Fallot and variants	

Σχήμα 5.20 – Καταχώρησης διάγνωσης qualifier αρχική διόρθωση

Τώρα προκύπτει η ανάγκη να παρουσιάζουμε και κωδικό που δίνεται στο αντίστοιχο qualifier. Το πρόβλημα των πολλαπλών διαγνώσεων δεν επιλύθηκε ακόμα στο σημείο αυτό και ένα μειονέκτημα ακόμα είναι ότι ο κωδικός του qualifier εμφανίζεται αυτόματα αλλά πρέπει να επιλεγεί από το αντίστοιχο combo box.

5.7 Ολοκλήρωση του IDC

Πιο κάτω φαίνεται παράδειγμα πύου σηματοδότη την ολοκλήρωση της κωδικοποίησης ασθενής όπως ακριβώς ζητήθηκε από τον πελάτη

The screenshot displays the 'App fixed' software interface. At the top, there is a menu bar (File, Edit, View, Insert, Format, Records, Tools, Window, Help) and a toolbar. Below the menu bar, a search bar contains 'Tahoma' and a dropdown menu shows '8'. A user profile 'Kyriakos' is visible in the top right corner.

The main window is titled 'Patient' and shows the following information:

- Name: **Nikoletta Polikarpou**
- Birthdate: **3/11/2000**
- Age: **8 Years**
- Patient ID: **C8876**
- Allocation: **Allocated**
- UN: **153872**

Below this information, there are tabs for 'Demographics', 'Clinical Data', and 'Events'. The 'Clinical Data' tab is selected.

The 'Clinical Data' section is divided into two main areas:

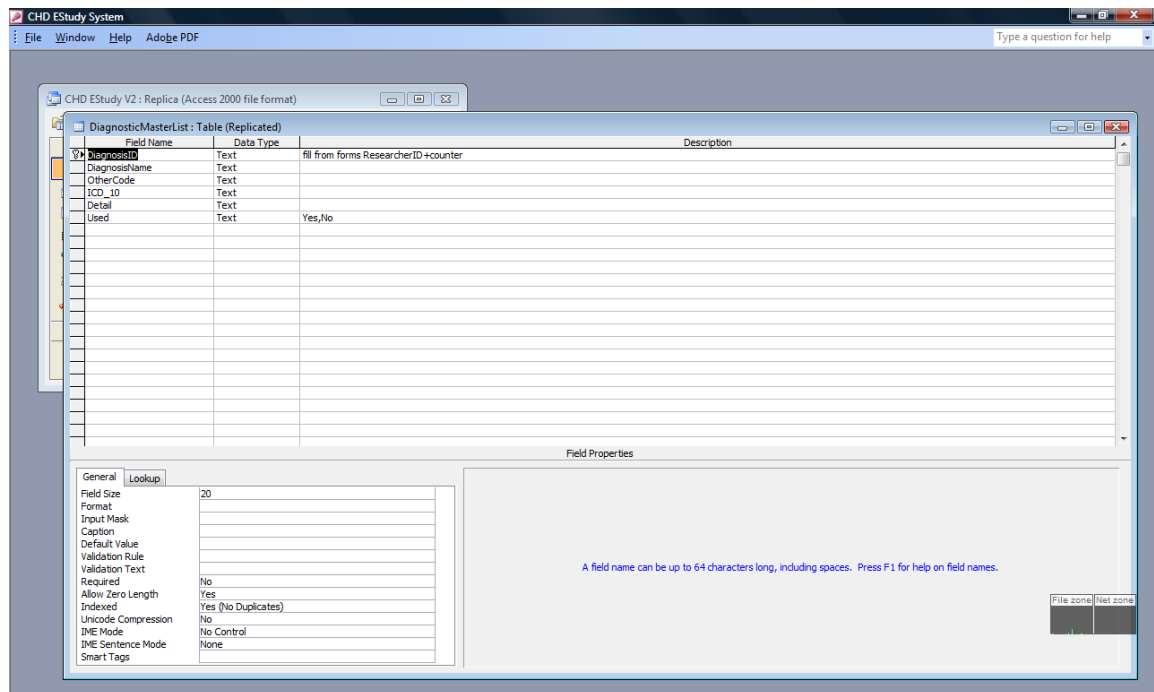
- Clinical Completion:** Includes fields for 'Date of Diagnosis' (06/11/2006), 'Time of Diagnosis' (Postnatal), 'Birth Outcome' (Alive), 'Twins' (Single), 'Family History' (No Fam Hist), 'Family Narrative History', 'Extracardiac Anomalies Existence' (No), 'Patient Status' (Alive), and 'Cause Death'.
- Narrative Diagnosis:** Includes a 'Diagnostic Category' (No significant Heart disease), 'Trace MI', and a 'Patient Coded Diagnosis' table.

The 'Patient Coded Diagnosis' table has the following columns: EPC Code, Diagnosis Name, and Qualifier. It contains two rows of data:

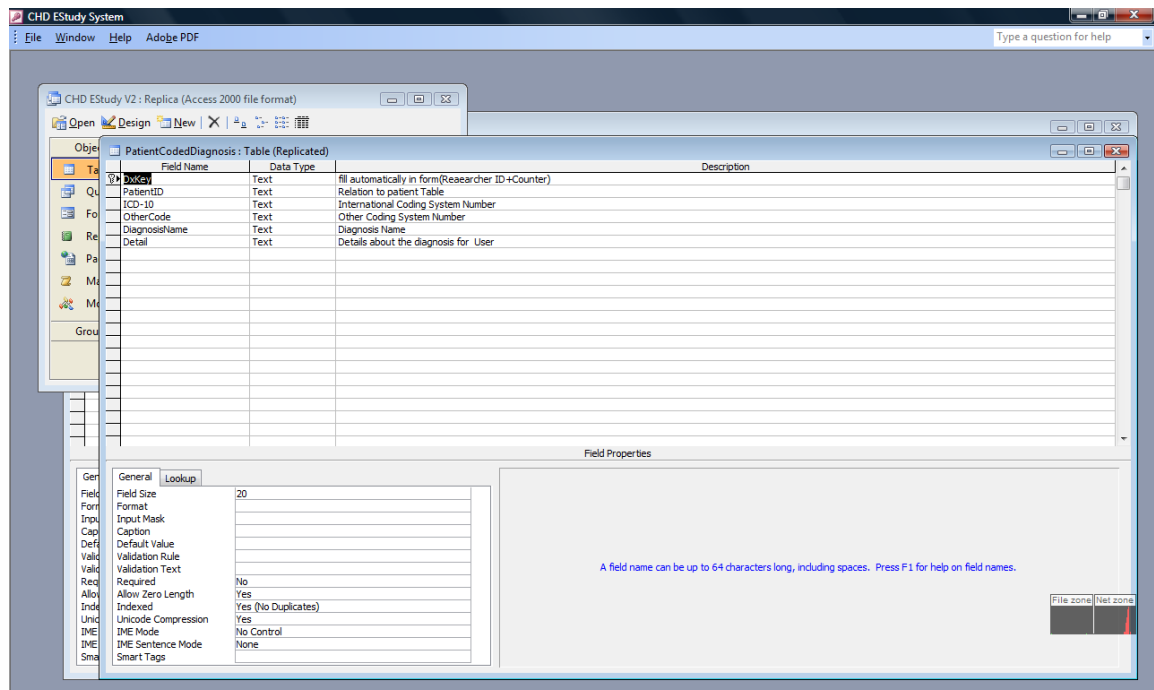
EPC Code	Diagnosis Name	Qualifier
03.07.01	Spleen left-sided,	
06.01.92	Tricuspid valve stenosis,	- trivial, Q1.40.00

At the bottom of the window, there is a 'Form View' button and a 'FLTR' button.

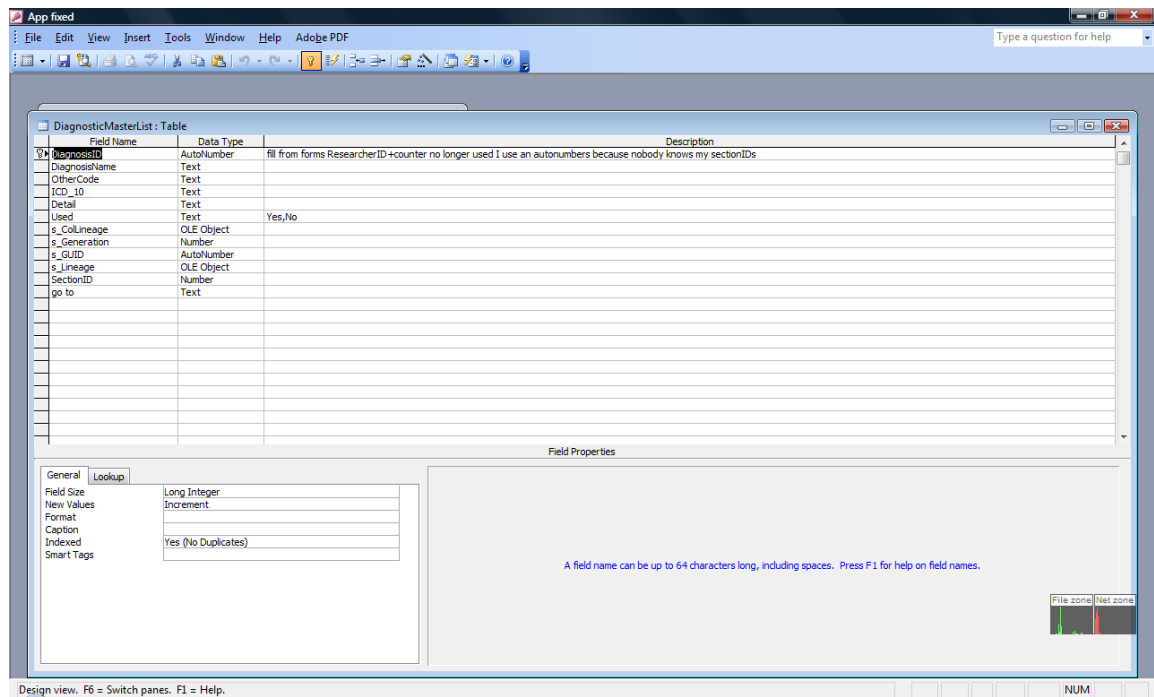
Σχήμα 5.21 – Καταχώρησης διάγνωσης qualifier τελική διόρθωση



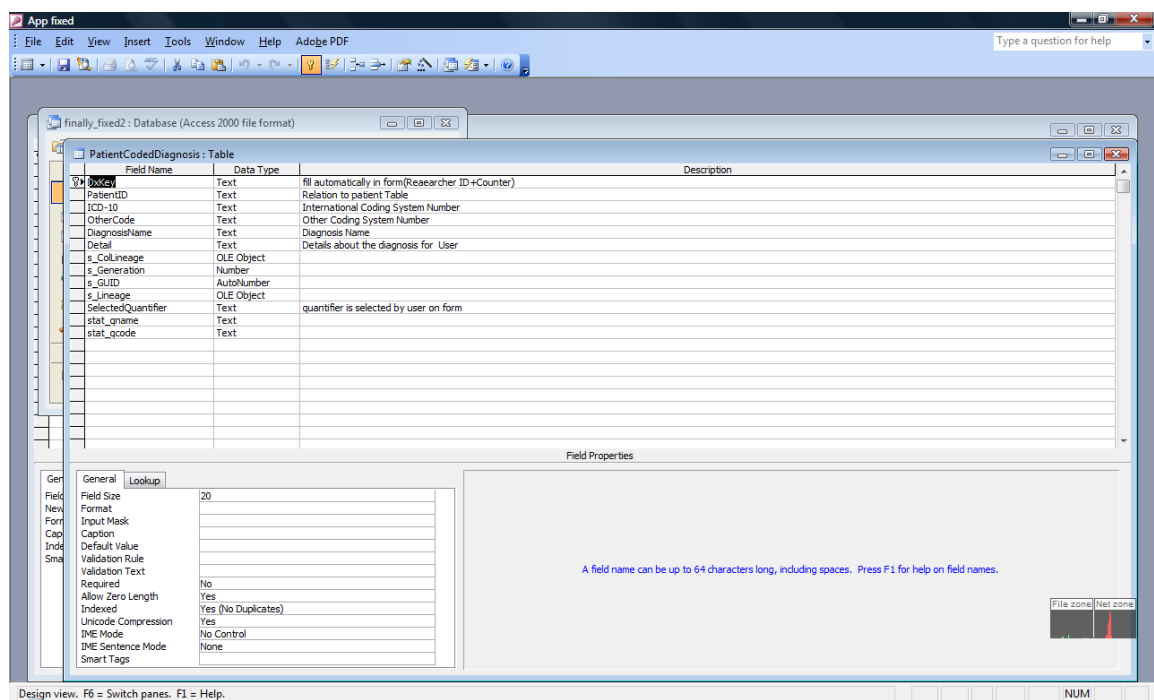
Σχήμα 5.22 – Σχήμα βάσης διαγνώσεων αρχικά



Σχήμα 5.23 – Σχήμα βάσης ασθενών αρχικά



Σχήμα 5.24 – Σχήμα βάσης διαγνώσεων τελικά



Σχήμα 5.25 – Σχήμα βάσης ασθενών τελικά

Database Application Example

File Help

The patient data is omitted so that we keep the implementation more general. Depending on how Pffidid determines the patient_id please change the value of lbpatientid_id or follow: lbpatient_id.setCrc(newvariable);

PatientID: 1

Search Clear

☒ All ☐ Recently Used

Diagnosis Name	IDC 10	EPC Code	Detail
Normal terms		not relevant	
Generic terms for no...		not relevant	
Generic cardiac terms		not relevant	
Congenital heart def...		not relevant	
Hereditary, fetal, ne...		not relevant	
Cardiothoracic acquir...		not relevant	
Functional and Pre-p...		not relevant	
Signs, Symptoms and...		not relevant	
Rhythm and conduct...		not relevant	
Qualifiers - diagnosti...		not relevant	
Cardiothoracic opera...		not relevant	
Post-operative and C...		not relevant	
Chest wall, pleural, t...		not relevant	
Transluminal interven...		not relevant	
Qualifiers - interven...		not relevant	
Post-procedural com...		not relevant	
Normal heart,	nc	01.01.00	Normal heart,
Normal atrial arrange...	nc	01.03.10	Normal atrial arrange...
Usual atrial arrangem...	nc	01.03.00	Usual atrial arrangem...
Concordant AV conn...	nc	01.04.00	
Concordant VA conn...	nc	01.05.00	Concordant VA conn...
Normal position/orien...	nc	02.01.00	
Laevocardia: heart p...	Q24.1	02.01.03	
Apex of heart to left...	Q24.1	02.02.00	
Normal position & mo...	nc	03.01.00	
Thymus present,	nc	03.01.07	
Lungs in normal arra...	nc	03.02.01	
Normal bronchial ana...	nc	03.03.01	
Spleen left-sided,	nc	03.07.01	
Liver right-sided,	nc	03.04.01	
Stomach left-sided,	nc	03.05.01	
Caecum right-sided,	nc	03.06.01	

Diagnosis Na...	IDC 10	EPC Code	Detail	Qualifier des...	Qualifier code
Normal terms		not relevant			
Generic terms...		not relevant			
Hereditary, f...		not relevant			
Cardiothoraci...		not relevant			
Stomach cent...	Q40.2	03.05.03	Position or m...		
Normal post...	nc	03.01.00			
Thymus pres...	nc	03.01.07			
Lungs in norm...	nc	03.02.01			
Normal bronc...	nc	03.03.01			
Spleen left-si...	nc	03.07.01			
Coronary sin...	Q24.5	04.04.04	Coronary sin...	- functional (...)	Q1.40.07
Mitral regurgi...	Q23.3	06.02.91	Mitral regurgi...	- functional (...)	Q1.40.16
Tricuspid ann...	Q22.8	06.01.06	Tricuspid valv...		
Left AV annul...	Q23.8	06.04.06	Mitral valvar ...		
Post-procedu...		not relevant			
LV dilation,	J51.7	07.06.02	LV abnormali...	- functional (...)	Q1.40.07

File zone Net zone

Edit Delete Patient Coded Diagnosis

Σχήμα 5.26 – IDC σε Java

Κεφάλαιο 6 Συμπεράσματα

Με βάση όσα υλοποιήθηκαν πρέπει να αναφέρω τα ακόλουθα συμπεράσματα. Παρά την γνώση που έχω όσο αφορά τις σχεσιακές βάσεις δεδομένων υπάρχει συγκεκριμένος λόγος που το σχήμα της βάσης παραβιάζει της αρχές της σχεσιακής βάσης δεδομένων. Αρχικά λαμβάνοντας υπόψη τον όγκο των πληροφοριών της κωδικοποίησης ασθενών αλλά και την μορφή στην οποία μας δίνονται τα δεδομένα πρέπει να τονίσουμε πως η μαζική εισαγωγή δεδομένων θα ήταν δύσκολη αν αλλάζαμε την δομή της βάσης. Ο φόρτος εργασίες με βάση την ανάκτηση του συνόλου των ασθενειών αφήνεται στο τέλος στον πελάτη και η βάση επιβαρύνεται μονό κατά την εισαγωγή η διαγραφή πλειάδων από την βάση.

Βιβλιογραφία

1. <http://www.gnu.org/licenses/gpl-3.0.txt>
2. <http://www.oscarcanada.org/>
3. <http://www.oscarcanada.org/front-page>
4. <http://sourceforge.net/projects/caisi>
5. <http://www.oscarcanada.org/caisi/caisi-project-histsory/caisi-history-and-background/>
6. <http://demo.caisi.ca/oscar/>
7. <http://sourceforge.net/projects/crrs/>
8. http://www.infoway-inforoute.ca/en/News-Events/InTheNews_long.aspx?UID=148
9. <http://chits.mudfish.info/HomePage>
10. <http://sourceforge.net/projects/game-engine/>
11. http://www.idrc.ca/en/ev-1-201-1-DO_TOPIC.html
12. <http://www.undp.org/>
13. <http://www.cottagemed.org/>
14. <http://www.tolven.org/ephr.html>
15. <http://en.wikipedia.org/wiki/GNUmed>
16. <http://www.hospital-os.com/en/>
17. <http://indivohealth.org/>
18. <http://www.chip.org/research/ping.htm>
19. <http://groups.csail.mit.edu/medg/projects/ga/manifesto/GAtr.html>
20. <http://bmj.bmjournals.com/cgi/reprint/322/7281/283.pdf>
21. <http://www.pubmedcentral.gov/picrender.fcgi?tool=pmcentrez&blobtype=pdf&artid=543826>
22. <http://indivohealth.org/pages/server>
23. <http://www.openehr.org/home.html>
24. <http://movies.miu.ph/chits/>
25. <http://www.flickr.com/photos/gnumed/sets/72157594477873670/show/>
26. <http://en.wikipedia.org/wiki/FreeMED>
27. <http://developer.freemedsoftware.org/wiki/index.php/HL7>
28. *International classification of diseases for oncology (ICD-O)*, second ed. Geneva, World Health Organization, 1990.

29. *Systematized nomenclature of medicine (SNOMED)*. Chicago, College of American Pathologists, 1976.
30. *Manual of tumor nomenclature and coding (MOTNAC)*. New York, American Cancer Society, 1968.
31. *Systematized nomenclature of pathology (SNOP)*. Chicago, College of American Pathologists, 1965.
32. *The ICD-10 classification of mental and behavioural disorders: clinical descriptions and diagnostic guidelines*. Geneva, World Health Organization, 1992.
33. *International classification of procedures in medicine (ICPM)*. Vols 1 and 2. Geneva, World Health Organization, 1978.
34. *International classification of impairments, disabilities, and handicaps. A manual of classification relating to the consequences of disease*. Geneva, World Health Organization, 1980.
35. *International Nomenclature of Diseases*. Geneva, Council for International Organizations of Medical Sciences and World Health Organization; for details of individual volumes, see text.
36. *Sixteenth annual report*. London, Registrar General of England and Wales, 1856, App. p. 73.
37. Knibbs G.H. The International Classification of Disease and Causes of Death and its revision. *Medical journal of Australia*, 1929, 1:2-12.
38. Greenwood M. *Medical statistics from Graunt to Farr*. Cambridge, Cambridge University Press, 1948.
39. *First annual report*. London, Registrar General of England and Wales, 1839, p. 99.
40. Bertillon J. Classification of the causes of death (abstract). In: *Transactions of the 15th International Congress on Hygiene Demography*. Washington, 1912.
41. *Bulletin of the Institute of International Statistics*, 1900, 12:280.
42. Roesle E. *Essai d'une statistique comparative de la morbidité devant servir à établir les listes spéciales des causes de morbidité*. Geneva, League of Nations Health Organization, 1928 (document C.H. 730)
43. *Nomenclatures internationales de causes de décès*. The Hague, International Statistical Institute, 1940.

44. Medical Research Council, Committee on Hospital Morbidity Statistics. *A provisional classification of diseases and injuries for use in compiling morbidity statistics*. London, Her Majesty's Stationery Office, 1944 (Special Report Series No. 248).
45. US Public Health Service, Division of Public Health Methods. *Manual for coding causes of illness according to a diagnosis code for tabulating morbidity statistics*. Washington, Government Publishing Office, 1944 (Miscellaneous Publication No. 32).
46. *Off.Rec. Wld. Hlth Org.*, 1948, 11, 23.
47. *Off.Rec. Wld. Hlth Org.*, 1948, 2, 110.
48. *Manual of the international statistical classification of diseases, injuries, and causes of death*. Sixth revision. Geneva, World Health Organization, 1949.
49. *Report of the International Conference for the Seventh Revision of the International Lists of Diseases and Causes of Death*. Geneva, World Health Organization, 1955 (unpublished document WHO/HA/7 Rev. Conf./17 Rev. 1; available on request from Strengthening of Epidemiological and Statistical Services, World Health Organization, 1211 Geneva 27, Switzerland).
50. *Third Report of the Expert Committee on Health Statistics*. Geneva, World Health Organization, 1952 (WHO Technical Report Series, No. 53).
51. *Report of the International Conference for the Eighth Revision of the International Classification of Diseases*. Geneva, World Health Organization, 1965 (unpublished document WHO/ICD9/74.4; available on request from Strengthening of Epidemiological and Statistical Services, World Health Organization, 1211 Geneva 27, Switzerland).
52. *Manual of the international statistical classification of diseases, injuries, and causes of death*, Volume 1. Geneva, World Health Organization, 1977.

Παράρτημα Α – Αρχικός κώδικας Access

Option Compare Database

```
Private Sub Combo22_AfterUpdate()
```

```
Dim SelectedID As String
```

```
Dim Characteristic As String
```

```
Dim TableName As String
```

```
Dim WhereClause As String
```

```
Dim finalstr As String
```

```
'Characteristic = "DiagnosisName"
```

```
'TableName = "DiagnosticMasterList"
```

```
'SelectedID = Me.Combo22.Value
```

```
'WhereClause = "DiagnosisID = " + SelectedID
```

```
'finalstr = "=" + "DLookup" + "(" + "[OtherCode]" + "," + "DiagnosticMasterList" + "," +  
"[DiagnosisID]=" + "" + SelectedID + "" + ")"
```

```
'MsgBox (finalstr)
```

```
'Me.Text53.ControlSource = finalstr
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
End Sub
```

```
Private Sub Combo22_BeforeUpdate(Cancel As Integer)
```

```
End Sub
```

```
Private Sub Combo22_Change()
```

```
'avoid write conflict
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
'end avoidance
```

```
'get data for update
```

```
Dim epccode As String
```

```
Dim patient_id As String
```

```
'get OtherCide of select diagnosis
```

```
epccode = Form_ListOfPatientCodedDiagnosis.OtherCode
```

```
'MsgBox (epccode)
```

```
'get patientid
```

```
patient_id = Form_Patient.PatientDemographicsID
```

```
'MsgBox (patient_id)
```

'get data for stat_qname

Dim varqname As String

'give textbox focus

Form_ListOfPatientCodedDiagnosis.Text57.SetFocus

'get qualifier name

varqname = Form_ListOfPatientCodedDiagnosis.Text57.Text

'give focus back to combo22

Form_ListOfPatientCodedDiagnosis.Combo22.SetFocus

'MsgBox (varqname)

'get data for stat_qcode

Dim varqcode As String

'give textbox focus

Form_ListOfPatientCodedDiagnosis.Text53.SetFocus

'get qualifier name

varqcode = Form_ListOfPatientCodedDiagnosis.Text53.Text

'give focus back to combo22

Form_ListOfPatientCodedDiagnosis.Combo22.SetFocus

'MsgBox (varqcode)

'end get dtata for update

'create update statement

Dim sqlstr As String

sqlstr = "UPDATE PatientCodedDiagnosis SET stat_qname = " + """" + varqname + """" +
", " + "stat_qcode = " + """" + varqcode + """" + " WHERE PatientID = " + """" + patient_id +
"""" + " AND OtherCode = " + """" + epccode + """" + ";"

'MsgBox (sqlstr)

DoCmd.RunCommand acCmdSaveRecord

DoCmd.RunSQL sqlstr

End Sub

Private Sub Combo22_Enter()

Combo22.Requery

End Sub

Private Sub Combo22_GotFocus()

On Error GoTo Err_Combo22_GotFocus

Dim strSql As String

Dim test As String

'Dim rs As DAO.Recordset

'Dim cn As DAO.Database

strSql = "SELECT [DiagnosticMasterList].[DiagnosisID],
[DiagnosticMasterList].[DiagnosisName], [DiagnosticMasterList].[OtherCode],
[DiagnosticMasterList].[go to] FROM DiagnosticMasterList where DiagnosisName = "" +
Me.DiagnosisName + "" ORDER BY DiagnosticMasterList.DiagnosisID;"

'Define Connection

Dim cnn As New ADODB.Connection

Set cnn = CurrentProject.Connection

'Define recordset

Dim RrecResearcher As New ADODB.Recordset

RrecResearcher.Open strSql, CurrentProject.Connection

'Check if something is there

If RrecResearcher.EOF Then

MsgBox "No diagnosis Please try again! ", vbDefaultButton1, "No diagnosis"

Else

'get goto

jump_id = RrecResearcher.Fields("go to")

'fix for qualifiers

'MsgBox ("just clicked!!!!")

Dim length As Integer

'get jump_id

'jump_id = Forms![Patient].[ListOfDiagnosisMasterList].Form![go to] 'looks like goto 5

'MsgBox (jump_id)

length = Len(jump_id)

```

'MsgBox (length)

'fixed jump_id

jump_id = Mid(jump_id, 4, (length - 3)) 'get just the last character

'MsgBox (jump_id)


'before jumping check is our destination is a qualifier

If jump_id >= "950" And jump_id <= "999" Then

FoundSectionID = jump_id

'MsgBox (FoundSectionID)

'Form_DblClick (0)

ElseIf jump_id >= "1600" And jump_id <= "1799" Then

FoundSectionID = jump_id

'MsgBox (FoundSectionID)

'Form_DblClick (0)

End If


'end fix for qualifiers

If FoundSectionID >= "950" And FoundSectionID <= "999" Then

'Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT
DiagnosticMasterList.DiagnosisID, DiagnosticMasterList.DiagnosisName,
DiagnosticMasterList.SectionID FROM DiagnosticMasterList where SectionID =" +
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"

```

```
Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
[DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],  
[DiagnosticMasterList].[OtherCode] FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
ElseIf FoundSectionID >= "1600" And FoundSectionID <= "1799" Then
```

```
'Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
DiagnosticMasterList.DiagnosisID, DiagnosticMasterList.DiagnosisName,  
DiagnosticMasterList.SectionID FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
[DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],  
[DiagnosticMasterList].[OtherCode] FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
Else
```

```
FoundSectionID = "0"
```

```
'Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
DiagnosticMasterList.DiagnosisID, DiagnosticMasterList.DiagnosisName,  
DiagnosticMasterList.SectionID FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
[DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],  
[DiagnosticMasterList].[OtherCode] FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
'Form_ListOfPatientCodedDiagnosis.Combo22.Requery
```

End If

End If

DoCmd.RunCommand acCmdSaveRecord

'MsgBox (test)

RrecResearcher.Close

cnn.Close

Exit_Combo22_GotFocus:

Exit Sub

Err_Combo22_GotFocus:

MsgBox Err.Description

Resume Exit_Combo22_GotFocus

End Sub

```
Private Sub DeleteButton_Click()
```

```
On Error GoTo Err_DeleteButton_Click
```

```
Dim sq As String
```

```
Me.AllowDeletions = True
```

```
Dim patient_id As String
```

```
Dim epccode As String
```

```
patient_id = Form_Patient.PatientDemographicsID
```

```
epccode = Form_ListOfPatientCodedDiagnosis.OtherCode
```

```
'MsgBox (epccode)
```

```
'MsgBox (patient_id)
```

```
sq = "DELETE * FROM PatientCodedDiagnosis WHERE
```

```
PatientCodedDiagnosis.DxKey=" + patient_id + "and PatientCodedDiagnosis.OtherCode=" + epccode + ";
```

```
'MsgBox (sq)
```

```
DoCmd.RunSQL sq
```

```
Me.Requery
```

Me.Refresh

Exit_DeleteButton_Click:

DoCmd.RunCommand acCmdSaveRecord

Exit Sub

Err_DeleteButton_Click:

'MsgBox Err.Description

Resume Exit_DeleteButton_Click

End Sub

Private Sub Ctl_DeleteButton_Click()

End Sub

Private Sub EditButton_Click()

'george code addres

Me.Combo22.Enabled = True

'george code addres

Me.AllowEdits = True

```
Me.SaveCodedDiagnosis.Enabled = True
```

```
Me.Command61.Enabled = True
```

```
Form_Patient.AddNewButton.Enabled = True 'George enabled add new button so we can  
add a diagnosis to patient
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
End Sub
```

```
Private Sub Form_Open(Cancel As Integer)
```

```
'george code addres
```

```
'VarSelectedQualifier = Form_ListOfPatientCodedDiagnosis.Combo22.Value
```

```
'MsgBox (VarSelectedQualifier)
```

```
Me.Combo22.Enabled = False
```

```
'george code addres
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
End Sub
```

```
Private Sub Label48_Click()
```

```
End Sub
```

```
Private Sub SaveCodedDiagnosis_Click()
```

```

Me.AllowEdits = False

Me.EditButton.SetFocus

Me.SaveCodedDiagnosis.Enabled = False

Me.Command61.Enabled = False

Me.Combo22.Enabled = False

End Sub

Private Sub Command58_Click()

On Error GoTo Err_Command58_Click

```

```

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

```

```

Exit_Command58_Click:

DoCmd.RunCommand acCmdSaveRecord

Exit Sub

```

```

Err_Command58_Click:

MsgBox Err.Description

Resume Exit_Command58_Click

```

End Sub

Private Sub Command61_Click()

On Error GoTo Err_Command61_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

Exit_Command61_Click:

Exit Sub

Err_Command61_Click:

' MsgBox Err.Description

Resume Exit_Command61_Click

End Sub

Παράρτημα Β – Τελικός κώδικας Access

Option Compare Database

```
Private Sub Combo22_AfterUpdate()
```

```
Dim SelectedID As String
```

```
Dim Charcteristic As String
```

```
Dim TableName As String
```

```
Dim WhereClause As String
```

```
Dim finalstr As String
```

```
'Charcteristic = "DiagnosisName"
```

```
'TableName = "DiagnosticMasterList"
```

```
'SelectedID = Me.Combo22.Value
```

```
'WhereClause = "DiagnosisID = " + SelectedID
```

```
'finalstr = "=" + "DLookup" + "(" + "[OtherCode]" + "," + "DiagnosticMasterList" + "," +  
"[DiagnosisID]=" + SelectedID + """)"
```

```
'MsgBox (finalstr)
```

```
'Me.Text53.ControlSource = finalstr
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
End Sub
```

```
Private Sub Combo22_BeforeUpdate(Cancel As Integer)
```

End Sub

Private Sub Combo22_Change()

'avoid write conflict

DoCmd.RunCommand acCmdSaveRecord

'end avoidance

'get data for update

Dim epccode As String

Dim patient_id As String

'get OtherCide of select diagnosis

epccode = Form_ListOfPatientCodedDiagnosis.OtherCode

'MsgBox (epccode)

'get patientid

patient_id = Form_Patient.PatientDemographicsID

'MsgBox (patient_id)

'get data for stat_qname

Dim varqname As String

'give textbox focus

Form_ListOfPatientCodedDiagnosis.Text57.SetFocus

'get qualifier name

```
varqname = Form_ListOfPatientCodedDiagnosis.Text57.Text
```

```
'give focus back to combo22
```

```
Form_ListOfPatientCodedDiagnosis.Combo22.SetFocus
```

```
'MsgBox (varqname)
```

```
'get data for stat_qcode
```

```
Dim varqcode As String
```

```
'give textbox focus
```

```
Form_ListOfPatientCodedDiagnosis.Text53.SetFocus
```

```
'get qualifier name
```

```
varqcode = Form_ListOfPatientCodedDiagnosis.Text53.Text
```

```
'give focus back to combo22
```

```
Form_ListOfPatientCodedDiagnosis.Combo22.SetFocus
```

```
'MsgBox (varqcode)
```

```
'end get dtata for update
```

```
'create update statement
```

```
Dim sqlstr As String
```

```
sqlstr = "UPDATE PatientCodedDiagnosis SET stat_qname = " + """" + varqname + """" + "," +
"stat_qcode = " + """" + varqcode + """" + " WHERE PatientID = " + """" + patient_id + """" + " AND
OtherCode = " + """" + epccode + """" + ","
```

```
'MsgBox (sqlstr)
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
DoCmd.RunSQL sqlstr
```

```
End Sub
```

```
Private Sub Combo22_Enter()
```

```
    Combo22.Requery
```

```
End Sub
```

```
Private Sub Combo22_GotFocus()
```

```
On Error GoTo Err_Combo22_GotFocus
```

```
Dim strSql As String
```

```
Dim test As String
```

```
'Dim rs As DAO.Recordset
```

```
'Dim cn As DAO.Database
```

```
strSql = "SELECT [DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],
[DiagnosticMasterList].[OtherCode], [DiagnosticMasterList].[go to] FROM DiagnosticMasterList
where DiagnosisName = """" + Me.DiagnosisName + """" ORDER BY
DiagnosticMasterList.DiagnosisID;"
```

```

'Define Connection

Dim cnn As New ADODB.Connection

Set cnn = CurrentProject.Connection


'Define recordset

Dim RrecResearcher As New ADODB.Recordset


RrecResearcher.Open strSql, CurrentProject.Connection


'Check if something is there

If RrecResearcher.EOF Then

MsgBox "No diagnosis Please try again! ", vbDefaultButton1, "No diagnosis"

Else


'get goto

jump_id = RrecResearcher.Fields("go to")


'fix for qualifiers

'MsgBox ("just clicked!!!!")

Dim length As Integer


'get jump_id

'jump_id = Forms![Patient].[ListOfDiagnosisMasterList].Form![go to] 'looks like goto 5

```

```

'MsgBox (jump_id)

length = Len(jump_id)

'MsgBox (length)

'fixed jump_id

jump_id = Mid(jump_id, 4, (length - 3)) 'get just the last character

'MsgBox (jump_id)


'before jumping check is our destination is a qualifier

If jump_id >= "950" And jump_id <= "999" Then

FoundSectionID = jump_id

'MsgBox (FoundSectionID)

'Form_DblClick (0)

ElseIf jump_id >= "1600" And jump_id <= "1799" Then

FoundSectionID = jump_id

'MsgBox (FoundSectionID)

'Form_DblClick (0)

End If


'end fix for qualifiers

If FoundSectionID >= "950" And FoundSectionID <= "999" Then

'Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT
DiagnosticMasterList.DiagnosisID, DiagnosticMasterList.DiagnosisName,
DiagnosticMasterList.SectionID FROM DiagnosticMasterList where SectionID =" + FoundSectionID
+ " ORDER BY DiagnosticMasterList.DiagnosisID;"

Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT
[DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],

```

```
[DiagnosticMasterList].[OtherCode] FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
ElseIf FoundSectionID >= "1600" And FoundSectionID <= "1799" Then
```

```
'Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
DiagnosticMasterList.DiagnosisID, DiagnosticMasterList.DiagnosisName,  
DiagnosticMasterList.SectionID FROM DiagnosticMasterList where SectionID =" + FoundSectionID  
+ " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
[DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],  
[DiagnosticMasterList].[OtherCode] FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
Else
```

```
FoundSectionID = "0"
```

```
'Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
DiagnosticMasterList.DiagnosisID, DiagnosticMasterList.DiagnosisName,  
DiagnosticMasterList.SectionID FROM DiagnosticMasterList where SectionID =" + FoundSectionID  
+ " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
Form_ListOfPatientCodedDiagnosis.Combo22.RowSource = "SELECT  
[DiagnosticMasterList].[DiagnosisID], [DiagnosticMasterList].[DiagnosisName],  
[DiagnosticMasterList].[OtherCode] FROM DiagnosticMasterList where SectionID =" +  
FoundSectionID + " ORDER BY DiagnosticMasterList.DiagnosisID;"
```

```
'Form_ListOfPatientCodedDiagnosis.Combo22.Requery
```

```
End If
```

```
End If
```

```
DoCmd.RunCommand acCmdSaveRecord
```

'MsgBox (test)

RrecResearcher.Close

cnn.Close

Exit_Combo22_GotFocus:

Exit Sub

Err_Combo22_GotFocus:

MsgBox Err.Description

Resume Exit_Combo22_GotFocus

End Sub

Private Sub DeleteButton_Click()

On Error GoTo Err_DeleteButton_Click

Dim sq As String

Me.AllowDeletions = True

Dim patient_id As String

Dim epccode As String

patient_id = Form_Patient.PatientDemographicsID

epccode = Form_ListOfPatientCodedDiagnosis.OtherCode

'MsgBox (epccode)

'MsgBox (patient_id)

sq = "DELETE * FROM PatientCodedDiagnosis WHERE PatientCodedDiagnosis.DxKey=" + patient_id
+ "and PatientCodedDiagnosis.OtherCode=" + epccode + ";"

'MsgBox (sq)

DoCmd.RunSQL sq

Me.Requery

Me.Refresh

Exit_DeleteButton_Click:

DoCmd.RunCommand acCmdSaveRecord

Exit Sub

Err_DeleteButton_Click:

'MsgBox Err.Description

Resume Exit_DeleteButton_Click

End Sub

Private Sub Ctl_DeleteButton_Click()

End Sub

```

Private Sub EditButton_Click()

'george code addres

Me.Combo22.Enabled = True

'george code addres

Me.AllowEdits = True

Me.SaveCodedDiagnosis.Enabled = True

Me.Command61.Enabled = True

Form_Patient.AddNewButton.Enabled = True 'George enabled add new button so we can add a
diagnosis to patient

DoCmd.RunCommand acCmdSaveRecord

End Sub

```

```

Private Sub Form_Open(Cancel As Integer)

'george code addres

'VarSelectedQualifier = Form_ListOfPatientCodedDiagnosis.Combo22.Value

'MsgBox (VarSelectedQualifier)

Me.Combo22.Enabled = False

'george code addres

DoCmd.RunCommand acCmdSaveRecord

End Sub

```

```

Private Sub Label48_Click()

End Sub

```

```
Private Sub SaveCodedDiagnosis_Click()
```

```
Me.AllowEdits = False
```

```
Me.EditButton.SetFocus
```

```
Me.SaveCodedDiagnosis.Enabled = False
```

```
Me.Command61.Enabled = False
```

```
Me.Combo22.Enabled = False
```

```
End Sub
```

```
Private Sub Command58_Click()
```

```
On Error GoTo Err_Command58_Click
```

```
DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
```

```
DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
```

```
Exit_Command58_Click:
```

```
DoCmd.RunCommand acCmdSaveRecord
```

```
Exit Sub
```

```
Err_Command58_Click:
```

```
MsgBox Err.Description
```

```
Resume Exit_Command58_Click
```

```
End Sub
```

```
Private Sub Command61_Click()
```

On Error GoTo Err_Command61_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

Exit_Command61_Click:

Exit Sub

Err_Command61_Click:

' MsgBox Err.Description

Resume Exit_Command61_Click

End Sub

Παράρτημα C – Αρχικός κώδικας Java

```
/*  
 * InternationalDiseaseCodingSystemView.java  
 */  
  
package internationaldiseasecodingsystem;  
  
import com.mysql.jdbc.Connection;  
import com.mysql.jdbc.Statement;  
import java.sql.SQLException;  
import java.util.logging.Level;  
import java.util.logging.Logger;  
import javax.swing.event.TableModelEvent;  
import org.jdesktop.application.Action;  
import org.jdesktop.application.ResourceMap;  
import org.jdesktop.application.SingleFrameApplication;  
import org.jdesktop.application.FrameView;  
import org.jdesktop.application.TaskMonitor;  
import org.jdesktop.application.Task;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.sql.DriverManager;  
import java.sql.ResultSet;  
import java.util.ArrayList;
```

```
import java.util.Collection;

import java.util.List;

import java.util.Vector;

import javax.persistence.EntityManager;

import javax.persistence.RollbackException;

import javax.swing.DefaultCellEditor;

import javax.swing.DefaultComboBoxModel;

import javax.swing.Timer;

import javax.swing.Icon;

import javax.swing.JComboBox;

import javax.swing.JDialog;

import javax.swing.JFrame;

import javax.swing.JTable;

import javax.swing.RowFilter;

import javax.swing.SwingUtilities;

import javax.swing.UIManager;

import javax.swing.event.ListSelectionEvent;

import javax.swing.event.ListSelectionListener;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableColumn;

import javax.swing.table.TableRowSorter;

import javax.xml.bind.annotation.XmlElementDecl.GLOBAL;

import org.jdesktop.beansbinding.AbstractBindingListener;

import org.jdesktop.beansbinding.Binding;

import org.jdesktop.beansbinding.PropertyStateEvent;
```

```

/**
 * The application's main frame.
 */
public class InternationalDiseaseCodingSystemView extends FrameView {

    public static String section_id = "";

    public InternationalDiseaseCodingSystemView(SingleFrameApplication app) {

        super(app);

        initComponents();

        // status bar initialization - message timeout, idle icon and busy animation, etc

        ResourceMap resourceMap = getResourceMap();
        int messageTimeout = resourceMap.getInteger("StatusBar.messageTimeout");

        messageTimer = new Timer(messageTimeout, new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                statusMessageLabel.setText("");

            }

        });

        messageTimer.setRepeats(false);

        int busyAnimationRate = resourceMap.getInteger("StatusBar.busyAnimationRate");

        for (int i = 0; i < busyIcons.length; i++) {

            busyIcons[i] = resourceMap.getIcon("StatusBar.busyIcons[" + i + "]");

        }

        busyIconTimer = new Timer(busyAnimationRate, new ActionListener() {

```

```

public void actionPerformed(ActionEvent e) {

    busylconIndex = (busylconIndex + 1) % busylcons.length;

    statusAnimationLabel.setIcon(busylcons[busylconIndex]);

}

});

idleIcon = resourceMap.getIcon("StatusBar.idleIcon");

statusAnimationLabel.setIcon(idleIcon);

progressBar.setVisible(false);


// connecting action tasks to status bar via TaskMonitor
TaskMonitor taskMonitor = new TaskMonitor(getApplication().getContext());
taskMonitor.addPropertyChangeListener(new java.beans.PropertyChangeListener() {

    public void propertyChange(java.beans.PropertyChangeEvent evt) {

        String propertyName = evt.getPropertyName();

        if ("started".equals(propertyName)) {

            if (!busylconTimer.isRunning()) {

                statusAnimationLabel.setIcon(busylcons[0]);

                busylconIndex = 0;

                busylconTimer.start();

            }

            progressBar.setVisible(true);

            progressBar.setIndeterminate(true);

        } else if ("done".equals(propertyName)) {

            busylconTimer.stop();

            statusAnimationLabel.setIcon(idleIcon);

```

```

        progressBar.setVisible(false);

        progressBar.setValue(0);
    } else if ("message".equals(propertyName)) {

        String text = (String)(evt.getNewValue());

        statusMessageLabel.setText((text == null) ? "" : text);

        messageTimer.restart();
    } else if ("progress".equals(propertyName)) {

        int value = (Integer)(evt.getNewValue());

        progressBar.setVisible(true);

        progressBar.setIndeterminate(false);

        progressBar.setValue(value);
    }
}

});

```

```

// tracking table selection

```

```

masterTable.getSelectionModel().addListSelectionListener(
    new ListSelectionListener() {

        public void valueChanged(ListSelectionEvent e) {

            firePropertyChange("recordSelected", !isRecordSelected(), isRecordSelected());

        }

    });

```

```

// tracking changes to save

```

```

bindingGroup.addBindingListener(new AbstractBindingListener() {

```

```

@Override

public void targetChanged(Binding binding, PropertyChangeEvent event) {

    // save action observes saveNeeded property

    setSaveNeeded(true);

}

});

// have a transaction started

entityManager.getTransaction().begin();

}

public boolean isSaveNeeded() {

    return saveNeeded;

}

private void setSaveNeeded(boolean saveNeeded) {

    if (saveNeeded != this.saveNeeded) {

        this.saveNeeded = saveNeeded;

        firePropertyChange("saveNeeded", !saveNeeded, saveNeeded);

    }

}

public boolean isRecordSelected() {

    return masterTable.getSelectedRow() != -1;

```

```
}
```

```
@Action
```

```
public void newRecord() {  
    internationaldiseasecodingsystem.Idc i = new internationaldiseasecodingsystem.Idc();  
    entityManager.persist(i);  
    list.add(i);  
    int row = list.size()-1;  
    masterTable.setRowSelectionInterval(row, row);  
    masterTable.scrollRectToVisible(masterTable.getCellRect(row, 0, true));  
    setSaveNeeded(true);  
}
```

```
@Action(enabledProperty = "recordSelected")
```

```
public void deleteRecord() {  
    int[] selected = masterTable.getSelectedRows();  
    List<internationaldiseasecodingsystem.Idc> toRemove = new  
    ArrayList<internationaldiseasecodingsystem.Idc>(selected.length);  
    for (int idx=0; idx<selected.length; idx++) {  
        internationaldiseasecodingsystem.Idc i =  
list.get(masterTable.convertRowIndexToModel(selected[idx]));  
        toRemove.add(i);  
        entityManager.remove(i);  
    }  
    list.removeAll(toRemove);  
}
```

```

        setSaveNeeded(true);
    }

```

```

@Action(enabledProperty = "saveNeeded")
public Task save() {
    return new SaveTask(getApplication());
}

```

```

private class SaveTask extends Task {

    SaveTask(org.jdesktop.application.Application app) {
        super(app);
    }

    @Override protected Void doInBackground() {
        try {
            entityManager.getTransaction().commit();

            entityManager.getTransaction().begin();
        } catch (RollbackException rex) {
            rex.printStackTrace();

            entityManager.getTransaction().begin();

            List<internationaldiseasecodingsystem.Idc> merged = new
ArrayList<internationaldiseasecodingsystem.Idc>(list.size());

            for (internationaldiseasecodingsystem.Idc i : list) {
                merged.add(entityManager.merge(i));
            }
        }
    }
}

```

```

        list.clear();

        list.addAll(merged);
    }

    return null;
}

@Override protected void finished() {
    setSaveNeeded(false);
}
}

/**
 * An example action method showing how to create asynchronous tasks
 * (running on background) and how to show their progress. Note the
 * artificial 'Thread.sleep' calls making the task long enough to see the
 * progress visualization - remove the sleeps for real application.
 */

@Action
public Task refresh() {
    return new RefreshTask(getApplication());
}

private class RefreshTask extends Task {
    RefreshTask(org.jdesktop.application.Application app) {
        super(app);
    }
}

```

```

}

@SuppressWarnings("unchecked")

@Override protected Void doInBackground() {

    try {

        setProgress(0, 0, 4);

        setMessage("Rolling back the current changes...");

        setProgress(1, 0, 4);

        entityManager.getTransaction().rollback();

        Thread.sleep(1000L); // remove for real app

        setProgress(2, 0, 4);


        setMessage("Starting a new transaction...");

        entityManager.getTransaction().begin();

        Thread.sleep(500L); // remove for real app

        setProgress(3, 0, 4);


        setMessage("Fetching new data...");

        java.util.Collection data = query.getResultList();

        for (Object entity : data) {

            entityManager.refresh(entity);

        }

        Thread.sleep(1300L); // remove for real app

        setProgress(4, 0, 4);


        Thread.sleep(150L); // remove for real app
    }
}

```

```

        list.clear();

        list.addAll(data);

    } catch (InterruptedException ignore) { }

    return null;
}

@Override protected void finished() {

    setMessage("Done.");

    setSaveNeeded(false);

}
}

```

```

@Action

public void showAboutBox() {

    if (aboutBox == null) {

        JFrame mainFrame =
InternationalDiseaseCodingSystemApp.getApplication().getMainFrame();

        aboutBox = new InternationalDiseaseCodingSystemAboutBox(mainFrame);

        aboutBox.setLocationRelativeTo(mainFrame);

    }

    InternationalDiseaseCodingSystemApp.getApplication().show(aboutBox);

}

```

```

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    bindingGroup = new org.jdesktop.beansbinding.BindingGroup();

    mainPanel = new javax.swing.JPanel();

    masterScrollPane = new javax.swing.JScrollPane();

    masterTable = new javax.swing.JTable();

    SearchField = new javax.swing.JTextField();

    RadioAll = new javax.swing.JRadioButton();

    RadioRecent = new javax.swing.JRadioButton();

    lblSearch = new javax.swing.JLabel();

    btnClear = new javax.swing.JButton();

    jScrollPane2 = new javax.swing.JScrollPane();

    jTextArea1 = new javax.swing.JTextArea();

    lblpatient_id = new javax.swing.JLabel();

    lblpatient = new javax.swing.JLabel();

    patientPane = new javax.swing.JScrollPane();

    tblPatientidc = new javax.swing.JTable();

    cmboQualifiers = new javax.swing.JComboBox();

```

```

menuBar = new javax.swing.JMenuBar();

javax.swing.JMenu fileMenu = new javax.swing.JMenu();

javax.swing.JMenuItem newRecordMenuItem = new javax.swing.JMenuItem();

javax.swing.JMenuItem deleteRecordMenuItem = new javax.swing.JMenuItem();

jSeparator1 = new javax.swing.JSeparator();

javax.swing.JMenuItem saveMenuItem = new javax.swing.JMenuItem();

javax.swing.JMenuItem refreshMenuItem = new javax.swing.JMenuItem();

jSeparator2 = new javax.swing.JSeparator();

javax.swing.JMenuItem exitMenuItem = new javax.swing.JMenuItem();

javax.swing.JMenu helpMenu = new javax.swing.JMenu();

javax.swing.JMenuItem aboutMenuItem = new javax.swing.JMenuItem();

statusPanel = new javax.swing.JPanel();

javax.swing.JSeparator statusPanelSeparator = new javax.swing.JSeparator();

statusMessageLabel = new javax.swing.JLabel();

statusAnimationLabel = new javax.swing.JLabel();

progressBar = new javax.swing.JProgressBar();

DeleteDiagnosis = new javax.swing.JButton();

btnEdit = new javax.swing.JButton();

entityManager = java.beans.Beans.isDesignTime() ? null :
javax.persistence.Persistence.createEntityManagerFactory("idcPU").createEntityManager();

org.jdesktop.application.ResourceMap resourceMap =
org.jdesktop.application.Application.getInstance(internationaldiseasesystem.InternationalD
iseaseCodingSystemApp.class).getContext().getResourceMap(InternationalDiseaseCodingSystemVi
ew.class);

query = java.beans.Beans.isDesignTime() ? null :
entityManager.createQuery(resourceMap.getString("query.query")); // NOI18N

```

```

list = java.beans.Beans.isDesignTime() ? java.util.Collections.emptyList() :
org.jdesktop.observablecollections.ObservableCollections.observableList(query.getResultList());

RadioGroup = new javax.swing.ButtonGroup();

idcQuery = java.beans.Beans.isDesignTime() ? null : entityManager.createQuery("SELECT i
FROM Idc i");

idcList = java.beans.Beans.isDesignTime() ? java.util.Collections.emptyList() :
idcQuery.getResultList();

patientIdcQuery = java.beans.Beans.isDesignTime() ? null :
entityManager.createQuery("SELECT p FROM PatientIdc p");

patientIdcList = java.beans.Beans.isDesignTime() ? java.util.Collections.emptyList() :
patientIdcQuery.getResultList();


mainPanel.setBackground(resourceMap.getColor("mainPanel.background")); // NOI18N

mainPanel.setName("mainPanel"); // NOI18N


masterScrollPane.setName("masterScrollPane"); // NOI18N


masterTable.setName("masterTable"); // NOI18N


org.jdesktop.swingbinding.JTableBinding jTableBinding =
org.jdesktop.swingbinding.SwingBindings.createJTableBinding(org.jdesktop.beansbinding.AutoBin
ding.UpdateStrategy.READ_WRITE, list, masterTable);

org.jdesktop.swingbinding.JTableBinding.ColumnBinding columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${id}"));

columnBinding.setColumnName("Id");

columnBinding.setColumnClass(Integer.class);

columnBinding.setEditable(false);

```

```

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${diagnosisName}
"));

        columnBinding.setColumnName("Diagnosis Name");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${icd10}"));

        columnBinding.setColumnName("Icd10");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${otherCode}"));

        columnBinding.setColumnName("Other Code");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${detail}"));

        columnBinding.setColumnName("Detail");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${used}"));

        columnBinding.setColumnName("Used");

        columnBinding.setColumnClass(String.class);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${sectionID}"));

        columnBinding.setColumnName("Section ID");

```

```

columnBinding.setColumnClass(Integer.class);

columnBinding.setEditable(false);

columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${goTo}"));

columnBinding.setColumnName("Go To");

columnBinding.setColumnClass(String.class);

columnBinding.setEditable(false);

bindingGroup.addBinding(jTableBinding);

jTableBinding.bind();

masterTable.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        masterTableMouseClicked(evt);

    }

});

masterScrollPane.setViewportViewView(masterTable);

masterTable.getColumnModel().getColumn(0).setMinWidth(3);

masterTable.getColumnModel().getColumn(0).setPreferredWidth(3);

masterTable.getColumnModel().getColumn(0).setMaxWidth(3);


masterTable.getColumnModel().getColumn(0).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title0")); // NOI18N


masterTable.getColumnModel().getColumn(1).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title1")); // NOI18N


masterTable.getColumnModel().getColumn(2).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title3")); // NOI18N


masterTable.getColumnModel().getColumn(3).setMinWidth(8);

```

```

masterTable.getColumnModel().getColumn(3).setPreferredWidth(8);

masterTable.getColumnModel().getColumn(3).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title2")); // NOI18N

masterTable.getColumnModel().getColumn(4).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title4")); // NOI18N

masterTable.getColumnModel().getColumn(5).setMinWidth(3);

masterTable.getColumnModel().getColumn(5).setPreferredWidth(0);

masterTable.getColumnModel().getColumn(5).setMaxWidth(3);

masterTable.getColumnModel().getColumn(5).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title5")); // NOI18N

masterTable.getColumnModel().getColumn(6).setMinWidth(0);

masterTable.getColumnModel().getColumn(6).setPreferredWidth(0);

masterTable.getColumnModel().getColumn(6).setMaxWidth(0);

masterTable.getColumnModel().getColumn(6).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title6")); // NOI18N

masterTable.getColumnModel().getColumn(7).setMinWidth(0);

masterTable.getColumnModel().getColumn(7).setPreferredWidth(0);

masterTable.getColumnModel().getColumn(7).setMaxWidth(0);

masterTable.getColumnModel().getColumn(7).setHeaderValue(resourceMap.getString("masterTa
ble.columnModel.title7")); // NOI18N

SearchField.setText(resourceMap.getString("SearchField.text")); // NOI18N

SearchField.setName("SearchField"); // NOI18N

SearchField.addActionListener(new java.awt.event.ActionListener() {

```

```

    public void actionPerformed(java.awt.event.ActionEvent evt) {
        SearchFieldActionPerformed(evt);
    }
});

```

```

RadioGroup.add(RadioAll);

RadioAll.setText(resourceMap.getString("RadioAll.text")); // NOI18N
RadioAll.setName("RadioAll"); // NOI18N

RadioAll.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        RadioAllActionPerformed(evt);
    }
});

```

```

RadioGroup.add(RadioRecent);

RadioRecent.setText(resourceMap.getString("RadioRecent.text")); // NOI18N
RadioRecent.setName("RadioRecent"); // NOI18N

RadioRecent.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        RadioRecentActionPerformed(evt);
    }
});

```

```

lblSearch.setText(resourceMap.getString("lblSearch.text")); // NOI18N
lblSearch.setName("lblSearch"); // NOI18N

```

```

btnClear.setText(resourceMap.getString("btnClear.text")); // NOI18N

btnClear.setName("btnClear"); // NOI18N

btnClear.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btnClearActionPerformed(evt);

    }

});

jScrollPane2.setName("jScrollPane2"); // NOI18N


jTextArea1.setColumns(20);

jTextArea1.setEditable(false);

jTextArea1.setRows(5);

jTextArea1.setText(resourceMap.getString("jTextArea1.text")); // NOI18N

jTextArea1.setName("jTextArea1"); // NOI18N

jScrollPane2.setViewportView(jTextArea1);


lblpatient_id.setText(resourceMap.getString("lblpatient_id.text")); // NOI18N

lblpatient_id.setName("lblpatient_id"); // NOI18N


lblpatient.setText(resourceMap.getString("lblpatient.text")); // NOI18N

lblpatient.setName("lblpatient"); // NOI18N


patientPane.setName("patientPane"); // NOI18N

```

```

tblPatientIdc.setName("tblPatientIdc"); // NOI18N

jTableBinding =
org.jdesktop.swingbinding.SwingBindings.createJTableBinding(org.jdesktop.beansbinding.AutoBin
ding.UpdateStrategy.READ_WRITE, patientIdcList, tblPatientIdc);

columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${id}"));

columnBinding.setColumnName("Id");

columnBinding.setColumnClass(Integer.class);

columnBinding.setEditable(false);

columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${diagnosisName}
"));

columnBinding.setColumnName("Diagnosis Name");

columnBinding.setColumnClass(String.class);

columnBinding.setEditable(false);

columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${icd10}"));

columnBinding.setColumnName("Icd10");

columnBinding.setColumnClass(String.class);

columnBinding.setEditable(false);

columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${otherCode}"));

columnBinding.setColumnName("Other Code");

columnBinding.setColumnClass(String.class);

columnBinding.setEditable(false);

```

```

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${detail}"));

        columnBinding.setColumnName("Detail");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${qname}"));

        columnBinding.setColumnName("Qname");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        columnBinding =
jTableBinding.addColumnBinding(org.jdesktop.beansbinding.ELProperty.create("${qcode}"));

        columnBinding.setColumnName("Qcode");

        columnBinding.setColumnClass(String.class);

        columnBinding.setEditable(false);

        bindingGroup.addBinding(jTableBinding);

        jTableBinding.bind();

        tblPatientidc.addMouseListener(new java.awt.event.MouseAdapter() {

            public void mouseClicked(java.awt.event.MouseEvent evt) {

                tblPatientidcMouseClicked(evt);

            }

        });

        patientPane.setViewportView(tblPatientidc);

        tblPatientidc.getColumnModel().getColumn(0).setMinWidth(0);

        tblPatientidc.getColumnModel().getColumn(0).setPreferredWidth(0);

        tblPatientidc.getColumnModel().getColumn(0).setMaxWidth(0);

```

```
tblPatientidc.getColumnModel().getColumn(0).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title0")); // NOI18N
```

```
tblPatientidc.getColumnModel().getColumn(1).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title1")); // NOI18N
```

```
tblPatientidc.getColumnModel().getColumn(2).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title2")); // NOI18N
```

```
tblPatientidc.getColumnModel().getColumn(3).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title3")); // NOI18N
```

```
tblPatientidc.getColumnModel().getColumn(4).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title4")); // NOI18N
```

```
tblPatientidc.getColumnModel().getColumn(5).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title5")); // NOI18N
```

```
tblPatientidc.getColumnModel().getColumn(6).setHeaderValue(resourceMap.getString("tblPatient  
idc.columnModel.title6")); // NOI18N
```

```
cmboQualifiers.setEnabled(false);  
  
cmboQualifiers.setName("cmboQualifiers"); // NOI18N  
  
cmboQualifiers.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        cmboQualifiersActionPerformed(evt);  
    }  
});
```

```

javax.swing.GroupLayout mainPanelLayout = new javax.swing.GroupLayout(mainPanel);

mainPanel.setLayout(mainPanelLayout);

mainPanelLayout.setHorizontalGroup(

    mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(mainPanelLayout.createSequentialGroup()

            .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(mainPanelLayout.createSequentialGroup()

                    .addGroup(mainPanelLayout.createSequentialGroup()

                        .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                            .addGroup(mainPanelLayout.createSequentialGroup()

                                .addGroup(mainPanelLayout.createSequentialGroup()

                                    .addGap(24, 24, 24)

                                    .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 341,
javax.swing.GroupLayout.PREFERRED_SIZE)

                                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                                    .addComponent(lblpatient)

                                    .addGap(18, 18, 18)

                                    .addComponent(lblpatient_id, javax.swing.GroupLayout.PREFERRED_SIZE, 56,
javax.swing.GroupLayout.PREFERRED_SIZE))

                                .addGroup(mainPanelLayout.createSequentialGroup()

                                    .addContainerGap()

                                    .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                                        .addGroup(mainPanelLayout.createSequentialGroup()

                                            .addGroup(mainPanelLayout.createSequentialGroup()

                                                .addComponent(masterScrollPane, javax.swing.GroupLayout.PREFERRED_SIZE,
498, javax.swing.GroupLayout.PREFERRED_SIZE)

                                                .addGap(35, 35, 35)

                                                .addComponent(patientPane, javax.swing.GroupLayout.PREFERRED_SIZE, 506,
javax.swing.GroupLayout.PREFERRED_SIZE)

                                                .addGap(41, 41, 41)

```

```

        .addComponent(cmboQualifiers, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGroup(mainPanelLayout.createSequentialGroup())

    .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(RadioAll)

        .addComponent(lblSearch, javax.swing.GroupLayout.PREFERRED_SIZE, 45,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(mainPanelLayout.createSequentialGroup()

            .addComponent(SearchField, javax.swing.GroupLayout.PREFERRED_SIZE,
264, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(18, 18, 18)

            .addComponent(btnClear))

            .addComponent(RadioRecent))))))

        .addGap(411, 411, 411))

    );

    mainPanelLayout.setVerticalGroup(

        mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(mainPanelLayout.createSequentialGroup()

            .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(mainPanelLayout.createSequentialGroup()

                    .addContainerGap()

                .addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

```

```

.addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(lblpatient)

    .addComponent(lblpatient_id, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(lblSearch)

    .addComponent(SearchField, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(btnClear))

    .addGap(20, 20, 20)

.addGroup(mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

    .addComponent(RadioAll)

    .addComponent(RadioRecent))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addComponent(masterScrollPane, javax.swing.GroupLayout.DEFAULT_SIZE, 406,
Short.MAX_VALUE))

    .addGroup(mainPanelLayout.createSequentialGroup())

    .addGap(193, 193, 193)

    .addComponent(cmboQualifiers, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addContainerGap())

    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
mainPanelLayout.createSequentialGroup())

```

```

        .addGap(162, 162, 162)

        .addComponent(patientPane, javax.swing.GroupLayout.DEFAULT_SIZE, 406,
Short.MAX_VALUE))

    );

    menuBar.setName("menuBar"); // NOI18N

    fileMenu.setText(resourceMap.getString("fileMenu.text")); // NOI18N

    fileMenu.setName("fileMenu"); // NOI18N

    javax.swing.ActionMap actionMap =
org.jdesktop.application.Application.getInstance(internationaldiseasescodingsystem.InternationalD
iseaseCodingSystemApp.class).getContext().getActionMap(InternationalDiseaseCodingSystemView
.class, this);

    newRecordMenuItem.setAction(actionMap.get("newRecord")); // NOI18N

    newRecordMenuItem.setName("newRecordMenuItem"); // NOI18N

    fileMenu.add(newRecordMenuItem);

    deleteRecordMenuItem.setAction(actionMap.get("deleteRecord")); // NOI18N

    deleteRecordMenuItem.setName("deleteRecordMenuItem"); // NOI18N

    fileMenu.add(deleteRecordMenuItem);

    jSeparator1.setName("jSeparator1"); // NOI18N

    fileMenu.add(jSeparator1);

    saveMenuItem.setAction(actionMap.get("save")); // NOI18N

    saveMenuItem.setName("saveMenuItem"); // NOI18N

```

```
fileMenu.add(saveMenuItem);

refreshMenuItem.setAction(actionMap.get("refresh")); // NOI18N
refreshMenuItem.setName("refreshMenuItem"); // NOI18N
fileMenu.add(refreshMenuItem);

jSeparator2.setName("jSeparator2"); // NOI18N
fileMenu.add(jSeparator2);

exitMenuItem.setAction(actionMap.get("quit")); // NOI18N
exitMenuItem.setName("exitMenuItem"); // NOI18N
fileMenu.add(exitMenuItem);

menuBar.add(fileMenu);

helpMenu.setText(resourceMap.getString("helpMenu.text")); // NOI18N
helpMenu.setName("helpMenu"); // NOI18N

aboutMenuItem.setAction(actionMap.get("showAboutBox")); // NOI18N
aboutMenuItem.setName("aboutMenuItem"); // NOI18N
helpMenu.add(aboutMenuItem);

menuBar.add(helpMenu);

statusPanel.setName("statusPanel"); // NOI18N
```

```
statusPanelSeparator.setName("statusPanelSeparator"); // NOI18N
```

```
statusMessageLabel.setName("statusMessageLabel"); // NOI18N
```

```
statusAnimationLabel.setHorizontalAlignment(javax.swing.SwingConstants.LEFT);
```

```
statusAnimationLabel.setName("statusAnimationLabel"); // NOI18N
```

```
progressBar.setName("progressBar"); // NOI18N
```

```
DeleteDiagnosis.setText(resourceMap.getString("DeleteDiagnosis.text")); // NOI18N
```

```
DeleteDiagnosis.setEnabled(false);
```

```
DeleteDiagnosis.setName("DeleteDiagnosis"); // NOI18N
```

```
DeleteDiagnosis.addMouseListener(new java.awt.event.MouseAdapter() {
```

```
    public void mouseClicked(java.awt.event.MouseEvent evt) {
```

```
        DeleteDiagnosisMouseClicked(evt);
```

```
    }
```

```
});
```

```
btnEdit.setText(resourceMap.getString("btnEdit.text")); // NOI18N
```

```
btnEdit.setName("btnEdit"); // NOI18N
```

```
btnEdit.addMouseListener(new java.awt.event.MouseAdapter() {
```

```
    public void mouseClicked(java.awt.event.MouseEvent evt) {
```

```
        btnEditMouseClicked(evt);
```

```
    }
```

```

});

javax.swing.GroupLayout statusPanelLayout = new javax.swing.GroupLayout(statusPanel);
statusPanel.setLayout(statusPanelLayout);
statusPanelLayout.setHorizontalGroup(
    statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(statusPanelSeparator, javax.swing.GroupLayout.DEFAULT_SIZE, 1529,
Short.MAX_VALUE)
        .addGroup(statusPanelLayout.createSequentialGroup()
            .addGap(29, 29, 29)
            .addComponent(statusMessageLabel)
            .addGap(208, 208, 208)
            .addComponent(progressBar, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(29, 29, 29)
            .addComponent(statusAnimationLabel)
            .addGap(29, 29, 29)
        )
);
statusPanelLayout.setVerticalGroup(
    statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(statusPanelLayout.createSequentialGroup()
            .addGap(29, 29, 29)
            .addComponent(statusMessageLabel)
            .addGap(29, 29, 29)
            .addComponent(statusAnimationLabel)
            .addGap(29, 29, 29)
        )
);

```

```

        .addComponent(statusPanelSeparator, javax.swing.GroupLayout.PREFERRED_SIZE, 2,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

        .addGroup(statusPanelLayout.createSequentialGroup())

.addGroup(statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(statusMessageLabel)

        .addComponent(statusAnimationLabel)

        .addComponent(progressBar, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(3, 3, 3))

.addGroup(statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(DeleteDiagnosis)

        .addComponent(btnEdit))))

);

btnEdit.getAccessibleContext().setAccessibleName(resourceMap.getString("btnEdit.AccessibleCon
text.accessibleName")); // NOI18N

```

```

setComponent(mainPanel);

setMenuBar(menuBar);

setStatusBar(statusPanel);

```

```

        bindingGroup.bind();

    }// </editor-fold>

private void SearchFieldActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    TableRowSorter sorter = new TableRowSorter(masterTable.getModel());

    RowFilter myFilter = RowFilter.regexFilter(SearchField.getText());

    sorter.setRowFilter(myFilter);

    masterTable.setRowSorter(sorter);

}

private void RadioAllActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    TableRowSorter sorter = new TableRowSorter(masterTable.getModel());

    //make sure textbox is empty

    SearchField.setText("");

    RowFilter myFilter = RowFilter.regexFilter(SearchField.getText());

    sorter.setRowFilter(myFilter);

    masterTable.setRowSorter(sorter);

}

private void RadioRecentActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO add your handling code here:

TableRowSorter sorter = new TableRowSorter(masterTable.getModel());

RowFilter myFilter = RowFilter.regexFilter("Yes");

sorter.setRowFilter(myFilter);

masterTable.setRowSorter(sorter);

}

private void masterTableMouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

if(evt.getClickCount() == 2)
{
Connection dbconn = null;

String sql = null;

int row = masterTable.getSelectedRow();

int col = 5; //select "Used field"

String patient_id; ////hold current patient_id

String dname; //hold selected diagnosis name

String idc_10; //hold selected diagnosis idc10 code

String epc_code; //hold selected epc_code

```

```

String detail; //hold selected detail field

masterTable.editCellAt(row, col);

masterTable.setValueAt("Yes", row, col);


entityManager.flush(); //sync database


//put clicked record in patient coded diagnosis
dname = (String) masterTable.getValueAt(row, 1); //gets DiagnosisName
patient_id = lblpatient_id.getText();
System.out.println(patient_id);
System.out.println(dname);
idc_10 = (String) masterTable.getValueAt(row, 2); //gets idc_10
System.out.println(idc_10);
epc_code = (String) masterTable.getValueAt(row, 3); //gets epc_code
System.out.println(epc_code);
detail = (String) masterTable.getValueAt(row, 4); //gets Detail
System.out.println(detail);


try {

    dbconn = (Connection)
DriverManager.getConnection("jdbc:mysql://localhost/idc","root","6jpeg*pb12");

    } catch (SQLException ex) {

Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

```

```

    }

    Statement select_stmt = null;

    try {

        select_stmt = (Statement) dbconn.createStatement();

    } catch (SQLException ex) {

        Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

    }

    try {

        ResultSet r = select_stmt.executeQuery("select * from patient_idc where patient_id = " +
patient_id + " and DiagnosisName = " + "" + dname + "" + ";"");

        if (!r.first()) //if nothing found

        {

            Statement stmt = null;

            try {

                stmt = (Statement) dbconn.createStatement();

            } catch (SQLException ex) {

                Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

```

```

    }

    sql = "insert into patient_idc (patient_id,DiagnosisName,OtherCode,ICD_10,Detail)Values(" +
    "" + patient_id + "" + "," + "" + dname + "" + "," + "" + epc_code + "" + "," + "" + idc_10 + "" +
    "," + "" + detail + "" + ");";

    try {

        stmt.executeUpdate(sql);

    } catch (SQLException ex) {

        Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
        ex);

    }

    entityManager.flush(); //sync database

    //update jtable view

    //Binding b = bindingGroup.getBinding("tblPatientidc");

    // System.out.println("b =" + b );

    //b.unbind();

    //patientIdcList.add(new TestClass(-1, "Minus one"));

    // internationaldiseasescodingsystem.PatientIdc i = new
    internationaldiseasescodingsystem.PatientIdc();

    //patientIdcList.clear();

```

```

        dbconn = (Connection)
DriverManager.getConnection("jdbc:mysql://localhost/idc","root","6jpeg*pb12");

Statement select_stmt2 = null;

        select_stmt2 = (Statement) dbconn.createStatement();

        ResultSet r2 = select_stmt2.executeQuery("select * from patient_idc;");

        while (!r2.next())

        {

            for (int m=0;m<=tblPatientidc.getModel().getColumnCount()-1;m++){

                patientIdcList.add(m,    (PatientIdc) r2.getObject(m));

            }

        }

```

```

//i.setDiagnosisName(dname);

// i.setDetail(detail);

// i.setIcd10(idc_10);

//i.setOtherCode(epc_code);

```

```
// entityManager.persist(i);
```

```
// patientIdcList.add(i);  
// int new_row = patientIdcList.size()-1;  
//tblPatientidc.setRowSelectionInterval(new_row, new_row);  
// tblPatientidc.scrollRectToVisible(tblPatientidc.getCellRect(new_row, 0, true));  
//setSaveNeeded(true);  
//entityManager.flush();
```

```
internationaldiseasecodingsystem.PatientIdc i = new  
internationaldiseasecodingsystem.PatientIdc();
```

```
// entityManager.persist(i);  
//patientIdcList.clear();  
//int new_row = 0;  
///tblPatientidc.setRowSelectionInterval(new_row, new_row);  
///tblPatientidc.scrollRectToVisible(tblPatientidc.getCellRect(new_row, 0, true));  
//entityManager.refresh(i);
```

```

// tblPatientIdc.getModel().setValueAt(dname, new_row, 1);
// tblPatientIdc.getModel().setValueAt(epc_code, new_row, 2);
//tblPatientIdc.getModel().setValueAt(idc_10, new_row, 3);
//tblPatientIdc.getModel().setValueAt(detail, new_row, 4);
//patientIdcList.addAll(patientIdcList);
//setSaveNeeded(true);

```

```

i.setDetail(detail);
i.setDiagnosisName(dname);
i.setIcd10(idc_10);
i.setOtherCode(epc_code);

```

```

patientIdcList.add(patientIdcList.size(), i);

```

```

for (int j=0; j <= patientIdcList.size() - 1; j++)
{
    System.out.println ( "size =" + patientIdcList.size());
    System.out.println(j);
}
System.out.println(patientIdcList.toString());

```

```

Binding b = bindingGroup.getBindings().get(1);

```

```
b.unbind();
```

```
b.bind();
```

```
b.saveAndNotify();
```

```
b.refreshAndNotify();
```

```
tblPatientidc.updateUI();
```

```
tblPatientidc.repaint();
```

```
System.out.println(sql);
```

```
}
```

```
} catch (SQLException ex) {
```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);
```

```
    }
```

```
}
```

```
}
```

```
private void btnClearActionPerformed(java.awt.event.ActionEvent evt) {
```

```
// TODO add your handling code here:
```

```
    SearchField.setText(""); //clear search text
```

```
}
```

```
private void tblPatientidcMouseClicked(java.awt.event.MouseEvent evt) {
```

```
// TODO add your handling code here:
```

```
//get selected row and create connection
```

```
    Connection dbconn = null;
```

```
    String sql = null;
```

```
    String dname; //gets selected DiagnosisName
```

```
    String jump_id = null; //hold section to jump to
```

```
    int jump_id_int = 0; //hold jump_id as an integer
```

```
int row = tblPatientidc.getSelectedRow();
```

```
String qname;
```

```

try {

    dbconn = (Connection)
    DriverManager.getConnection("jdbc:mysql://localhost/idc","root","6jpeg*pb12");

    } catch (SQLException ex) {

    Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
    ex);

    }

    Statement select_stmt = null;

    try {

        select_stmt = (Statement) dbconn.createStatement();

    } catch (SQLException ex) {

    Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
    ex);

    }

    try {

        dname = (String) tblPatientidc.getValueAt(row, 1); //gets DiagnosisName

        ResultSet r = select_stmt.executeQuery("select * from idc where DiagnosisName = " + "" +
        dname + "" + ";"");

        if(r.first())

        {

            System.out.println("dname before get string = " + dname);

```

```

        jump_id = r.getString(8); //looks like to 2 so i need to cut

    }

    } catch (SQLException ex) {

Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

    }

System.out.println(jump_id);

    int str_length = jump_id.length();

System.out.println(str_length);

jump_id = jump_id.substring(3, str_length - 1);

System.out.println(jump_id);

//cast jump_id to integer

jump_id_int = Integer.parseInt(jump_id);

//decide if diagnosis has qualifiers as fill the combox with the coorespondingf ones
if (jump_id_int >= 950 && jump_id_int <= 999){

System.out.println("is qualifier:" + jump_id);

section_id = jump_id;

```

```

        ResultSet r = null;

        try {

            r = select_stmt.executeQuery("select * from idc where SectionID = " + "" + jump_id + ""
+ ";"");

            } catch (SQLException ex) {

Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

        }

        //remove any item from combobox

        cmboQualifiers.removeAllItems();

        try {

            while (r.next()) {

                qname = r.getString(2); //get qname

                cmboQualifiers.addItem(qname);

            }

        } catch (SQLException ex) {

Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

        }

    }

```

```

else if (jump_id_int >= 1600 && jump_id_int <= 1799){

    System.out.println("is wualifier:" + jump_id);

    section_id = jump_id;

    ResultSet r = null;

    try {

        r = select_stmt.executeQuery("select * from idc where SectionID = " + "" + jump_id + ""
+ ";");

    } catch (SQLException ex) {

        Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

    }

    //remove any item from combobox

    cmboQualifiers.removeAllItems();

    try {

        while (r.next()) {

            qname = r.getString(2); //get qname

            cmboQualifiers.addItem(qname);

        }

    } catch (SQLException ex) {

```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

else{

    System.out.println("is not wualifier");
    cmboQualifiers.removeAllItems();

}

}
```

```

private void cmboQualifiersActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

    int row;

    int col;

    Connection dbconn = null;

    String sql = null;

    String qcode; //holds selected qnames qcode

    row = tblPatientidc.getSelectedRow();


//  String selected_qname = cmboQualifiers.getSelectedItem().toString();

    System.out.println("You selected qname =" + cmboQualifiers.getSelectedItem());

    //tblPatientidc.editCellAt(row, 5); //want to edit qname

    tblPatientidc.setValueAt(cmboQualifiers.getSelectedItem(), row, 5);

    //tblPatientidc.setValueAt(qcode, row, col)

    //tblPatientidc.editingStopped(evt.)

    entityManager.flush(); //sync database


    try {

        //find qcode for selected qname

        dbconn = (Connection)
DriverManager.getConnection("jdbc:mysql://localhost/idc","root","6jpeg*pb12");

    } catch (SQLException ex) {

```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);
```

```
}
```

```
Statement select_stmt = null;
```

```
try {
```

```
    select_stmt = (Statement) dbconn.createStatement();
```

```
} catch (SQLException ex) {
```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);
```

```
}
```

```
ResultSet r = null;
```

```
try {
```

```
    r = select_stmt.executeQuery("select * from idc where SectionID = " + "" + section_id + ""
+ " and DiagnosisName = " + "" + cmboQualifiers.getSelectedItemAt() + "" + "");
```

```
} catch (SQLException ex) {
```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);
```

```
}
```

```
try {
```

```
    if (r.first()) {
```

```

        qcode = r.getString(3); //get qcode which is in othercode colum

        System.out.println(qcode);

        tblPatientidc.setValueAt(qcode, row, 6);

        entityManager.flush(); //sync database

    }

    } catch (SQLException ex) {

        Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

    }

}

private void DeleteDiagnosisMouseClicked(java.awt.event.MouseEvent evt) {

```

```

// TODO add your handling code here:

//get selected row number
int row = tblPatientidc.getSelectedRow();
String patient_id = lblpatient_id.getText();
Connection dbconn = null;
String sql = null;
//get selected diagnosis name
String dname = tblPatientidc.getValueAt(row, 1).toString();
System.out.println("dname on click = " + dname);
System.out.println("patient_id on click = " + patient_id);
    try {

//do delete in sql

        dbconn = (Connection)
DriverManager.getConnection("jdbc:mysql://localhost/idc","root","6jpeg*pb12");

        } catch (SQLException ex) {

Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);

        }
Statement stmt = null;
    try {

        stmt = (Statement) dbconn.createStatement();

        } catch (SQLException ex) {

```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);
```

```
}
```

```
sql = "delete from patient_idc where DiagnosisName = " + "'" + dname + "'" + " and patient_id = " +
      "'" + patient_id + "'" + ";;";
```

```
System.out.println(sql);
```

```
try {
```

```
    stmt.executeUpdate(sql);
```

```
} catch (SQLException ex) {
```

```
Logger.getLogger(InternationalDiseaseCodingSystemView.class.getName()).log(Level.SEVERE, null,
ex);
```

```
}
```

```
entityManager.flush();
```

```
//handle jtable refresh
```

```
internationaldiseasecodingsystem.PatientIdc i = new
internationaldiseasecodingsystem.PatientIdc();
```

```
patientIdcList.remove(row);
```

```
for (int j=0; j <= patientIdcList.size() - 1; j++)  
{  
    System.out.println ( "size =" + patientIdcList.size());  
    System.out.println(j);  
}  
System.out.println(patientIdcList.toString());
```

```
Binding b = bindingGroup.getBindings().get(1);  
b.unbind();  
b.bind();  
b.saveAndNotify();  
b.refreshAndNotify();
```

```
tblPatientidc.updateUI();  
tblPatientidc.repaint();
```

```
// end jTable refresh
```

```
}
```

```
private void btnEditMouseClicked(java.awt.event.MouseEvent evt) {
```

```
// TODO add your handling code here:
```

```
    cmboQualifiers.setEnabled(true);
```

```
    DeleteDiagnosis.setEnabled(true);
```

```
}
```

```
// Variables declaration - do not modify
```

```
private javax.swing.JButton DeleteDiagnosis;
```

```
private javax.swing.JRadioButton RadioAll;
```

```
private javax.swing.ButtonGroup RadioGroup;
```

```
private javax.swing.JRadioButton RadioRecent;
```

```
private javax.swing.JTextField SearchField;
```

```
private javax.swing.JButton btnClear;
```

```
private javax.swing.JButton btnEdit;
```

```
private javax.swing.JComboBox cmboQualifiers;
```

```
private javax.persistence.EntityManager entityManager;
```

```
private java.util.List<internationaldiseasecodingsystem.Idc> idcList;
```

```
private javax.persistence.Query idcQuery;
```

```

private javax.swing.JScrollPane jScrollPane2;

private javax.swing.JSeparator jSeparator1;

private javax.swing.JSeparator jSeparator2;

private javax.swing.JTextArea jTextArea1;

private javax.swing.JLabel lblSearch;

private javax.swing.JLabel lblpatient;

private javax.swing.JLabel lblpatient_id;

private java.util.List<internationaldiseasecodingsystem.Idc> list;

private javax.swing.JPanel mainPanel;

private javax.swing.JScrollPane masterScrollPane;

public javax.swing.JTable masterTable;

private javax.swing.JMenuBar menuBar;

private java.util.List<internationaldiseasecodingsystem.PatientIdc> patientIdcList;

private javax.persistence.Query patientIdcQuery;

private javax.swing.JScrollPane patientPane;

private javax.swing.JProgressBar progressBar;

private javax.persistence.Query query;

private javax.swing.JLabel statusAnimationLabel;

private javax.swing.JLabel statusMessageLabel;

private javax.swing.JPanel statusPanel;

private javax.swing.JTable tblPatientIdc;

private org.jdesktop.beansbinding.BindingGroup bindingGroup;

// End of variables declaration

private final Timer messageTimer;

```

```
private final Timer busylconTimer;  
  
private final Icon idleIcon;  
  
private final Icon[] busylcons = new Icon[15];  
  
private int busylconIndex = 0;  
  
  
private JDialog aboutBox;  
  
  
private boolean saveNeeded;  
}
```

Παράρτημα Δ – Τελικός κώδικας Java

Εδώ δημιουργούμε μια κλάση IDC_handler με κάποιες απαιτούμενες μέθους μειώνοντας τον προηγούμενό κώδικα

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package internationaldiseasecodingsystem;

/**
 *
 * @author Guardian
 */
public class IDC_handler {
    public static String insert(String table_name,String col_names, String
    values) {

        java.sql.Connection conn = linktodata();

        String todo = ("INSERT into " + table_name +
            "(" + col_names + ") values (" + values + ")");

        try {
            java.sql.Statement s = conn.createStatement();
            int r = s.executeUpdate (todo);
        }
        catch (Exception e) {
            return ("Oh oops - code 003\n"+e);
        }

        return (todo);
    }

    public static String select(String table_name , String [] fields, String
    selector) {

        java.sql.Connection conn = linktodata();

        StringBuffer reply = new StringBuffer("<table border=1>");

        String todo = ("SELECT *"+
            " from" + table_name  + selector);

        try {
            java.sql.Statement s = conn.createStatement();
            java.sql.ResultSet r = s.executeQuery (todo);
            while(r.next()) {
                reply.append("<tr>");
                for (int i=0;i<fields.length;i++) {
```

```

        reply.append(tabit(r.getString(fields[i])));
    }
    reply.append("</tr>");
}
reply.append("</table>");
}
catch (Exception e) {
    return ("Oh oops - some crazy sql error in select\n"+e);
}

return (reply.toString());
}

private static String tabit(String box) {
    return ("<td>"+box+"</td>");
}

private static java.sql.Connection linktodata () {
    java.sql.Connection conn = null;

    try {
        conn = java.sql.DriverManager.getConnection(
            "jdbc:mysql://localhost/idc/root/6jpeg*pb12");
    }
    catch (Exception e) {
        return conn;
    }
    return conn;
}

public static String delete(String table_name,String col_names, String
value) {
    java.sql.Connection conn = linktodata();

    String todo = ("Delete from " + table_name +
        "where" + col_name + "=" + value");

    try {
        java.sql.Statement s = conn.createStatement();
        int r = s.executeUpdate (todo);
    }
    catch (Exception e) {
        return ("Oh oops - code 003\n"+e);
    }

    return (todo);
}
}

```