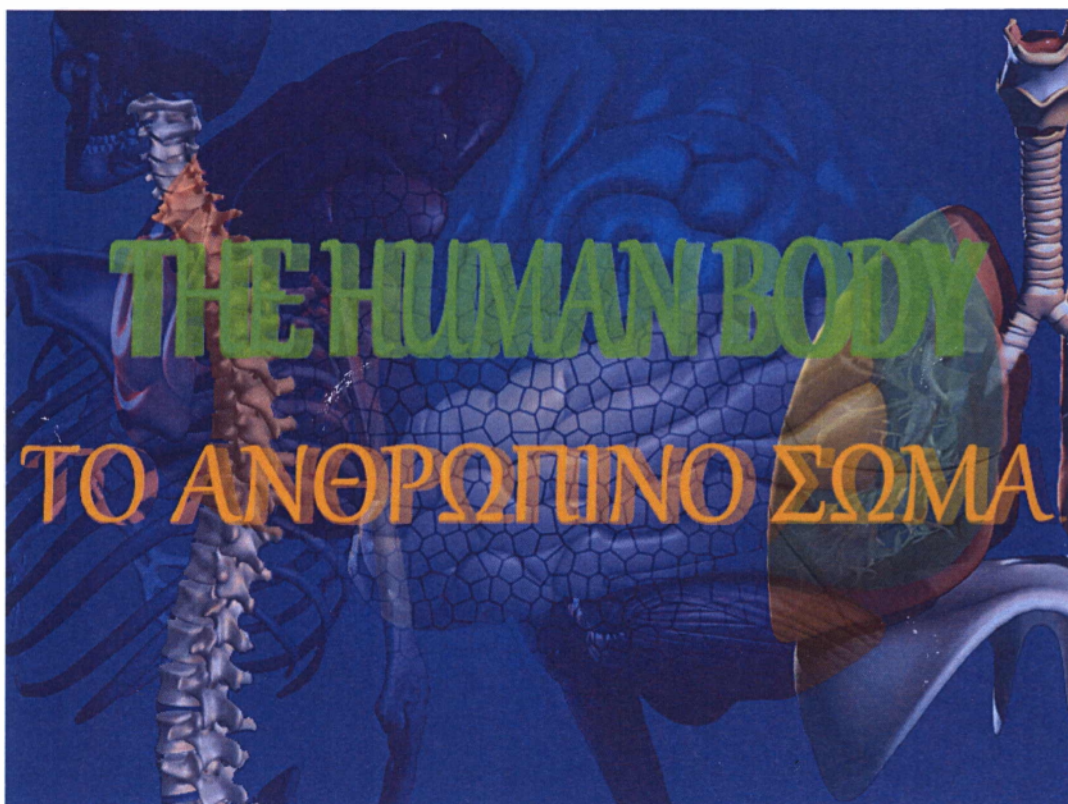




Α.Τ.Ε.Ι ΚΑΛΑΜΑΤΑΣ
ΠΑΡΑΡΤΗΜΑ ΣΠΑΡΤΗΣ
ΤΜΗΜΑ ΤΕΧΝΟΛΟΓΙΑΣ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΤΗΝ ΓΡΑΦΙΚΗ ΑΝΑΠΑΡΑΣΤΑΣΗ ΤΟΥ
ΑΝΘΡΩΠΙΝΟΥ ΣΩΜΑΤΟΣ ΜΕ ΥΠΟΣΤΗΡΙΞΗ ΠΟΛΛΑΠΛΗΣ
ΑΛΛΗΛΕΠΙΔΡΑΣΗΣ ΧΡΗΣΤΗ.**



ΓΚΟΥΒΑΣ ΣΩΤΗΡΙΟΣ

ΕΠΙΒΛΕΠΩΝ:

ΤΡΙΤΑΚΗΣ ΠΑΝΑΓΙΩΤΗΣ

ΣΠΑΡΤΗ, 2012

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΚΕΦΑΛΑΙΟ 1 – Η ΕΦΑΡΜΟΓΗ & Ο ΣΤΟΧΟΣ ΤΗΣ.....	5
1.1 ΕΙΣΑΓΩΓΗ	6
1.2 Η ΕΦΑΡΜΟΓΗ.....	8
1.3 ΔΟΜΗ ΑΝΑΦΟΡΑΣ.....	9
ΚΕΦΑΛΑΙΟ 2 – ΠΕΡΙΓΡΑΦΗ ΤΕΧΝΟΛΟΓΙΩΝ	11
2.1 .NET FRAMEWORK	12
2.1.1 CLR	13
2.1.2 .NET Class Libraries	14
2.1.3 .NET Γλώσσες Προγραμματισμού	15
2.1.4 ASP.NET.....	16
2.2 C#	17
2.3 WPF	19
2.4 XAML	20
2.5 MYSQL.....	21
2.6 ENTITY FRAMEWORK	22
2.7 LINQ	24
2.8 SPEECH RECOGNITION.....	26
ΚΕΦΑΛΑΙΟ 3 – ΔΟΜΗ & ΑΡΧΙΤΕΚΤΟΝΙΚΗ	29
3.1 ΠΛΗΡΟΦΟΡΙΑ-ΔΕΔΟΜΕΝΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	30
3.2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	33
3.3 ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΟΘΟΝΩΝ ΚΑΙ ΛΕΙΤΟΥΡΓΙΩΝ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	37
3.3.1 ΤΟ ΚΕΝΤΡΙΚΟ ΠΑΡΑΘΥΡΟ	37
3.3.2 ΠΑΡΑΘΥΡΟ ΠΡΟΣΘΗΚΗΣ ΑΣΘΕΝΗ	47
3.3.3 ΠΑΡΑΘΥΡΟ ΑΝΑΖΗΤΗΣΗΣ ΑΣΘΕΝΗ	49
3.3.4 ΦΩΝΗΤΙΚΕΣ ΕΝΤΟΛΕΣ.....	51
ΚΕΦΑΛΑΙΟ 4 – ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΛΕΙΤΟΥΡΓΙΩΝ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΜΕ ΠΑΡΑΔΕΙΓΜΑΤΑ	55
4.1 ΠΡΟΣΘΗΚΗ ΝΕΟΥ ΑΣΘΕΝΗ	56
4.2 ΑΝΑΖΗΤΗΣΗ ΑΣΘΕΝΗ	58
4.3 ΠΡΟΣΘΗΚΗ / ΕΠΕΞΕΡΓΑΣΙΑ / ΔΙΑΓΡΑΦΗ ΤΩΝ ΣΗΜΕΙΩΝ ΤΡΑΥΜΑΤΙΣΜΩΝ ΤΟΥ ΑΣΘΕΝΗ.....	61
4.4 ΜΕΤΑΚΙΝΗΣΗ ΣΤΟΥΣ ΤΡΑΥΜΑΤΙΣΜΟΥΣ ΤΟΥ ΑΣΘΕΝΗ ΜΕ ΧΡΗΣΗ ΤΟΥ ΠΑΡΑΘΥΡΟΥ ΓΡΗΓΟΡΗΣ ΠΡΟΣΒΑΣΗΣ ΣΤΑ ΣΗΜΕΙΑ ΤΡΑΥΜΑΤΙΣΜΩΝ.	65
ΕΠΙΛΟΓΟΣ.....	67

ΠΑΡΑΡΤΗΜΑ Α	69
1 ΟΙ ΠΙΝΑΚΕΣ ΤΗΣ MYSQL	70
1.1 Πίνακας Levels.....	70
1.2 Πίνακας Points	71
1.3 Πίνακας Patients.....	73
1.4 Πίνακας Medical Records	74
1.5 Πίνακας Photos.....	74
2 ΣΧΕΣΕΙΣ ΜΕΤΑΞΥ ΤΩΝ ΠΙΝΑΚΩΝ	76
ΠΑΡΑΡΤΗΜΑ Β	79
1 ΕΓΚΑΤΑΣΤΑΣΗ MYSQL SERVER.....	80
2 ΕΓΚΑΤΑΣΤΑΣΗ MYSQL WORKBENCH	84
3 ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ ADO.NET DRIVER FOR MYSQL (CONNECTOR/NET).....	86
4 ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ.....	87
ΒΙΒΛΙΟΓΡΑΦΙΑ	89

ΚΕΦΑΛΑΙΟ 1

Η ΕΦΑΡΜΟΓΗ ΚΑΙ Ο ΣΤΟΧΟΣ ΤΗΣ

Σε αυτό το κεφάλαιο γίνεται μια ιστορική αναδρομή στην εξέλιξη της ανατομίας του ανθρώπινου σώματος και των τεχνολογιών που ασχολούνται με αυτή. Επίσης εξηγείται ο σκοπός της πτυχιακής εργασίας και γίνεται μια σύντομη περιγραφή της εφαρμογής. Τέλος γίνεται λόγος για τη δομή της αναφοράς.

1.1 ΕΙΣΑΓΩΓΗ

Από τα αρχαία χρόνια ο άνθρωπος ένιωσε την ανάγκη να μελετήσει το ανθρώπινο σώμα. Στόχος του ήταν να μάθει τον τρόπο λειτουργίας του σώματος, να δομήσει τα διάφορα όργανα και να αντιμετωπίσει διάφορες ασθένειες. Την μελέτη αυτή την ονόμασε ανατομία του ανθρώπινου σώματος.

Η ανατομία αποτελεί ένα σημαντικό κλάδο της ιατρικής και γίνεται συνεχώς προσπάθεια για να εξελιχθεί. Είναι διαδεδομένη η αντίληψη ότι η Ανατομία είναι το μάθημα το οποίο καθορίζει την εγκυρότητα των όποιων ιατρικών σπουδών. Γι' αυτό και έχουν γραφτεί πολλά βιβλία και χρησιμοποιούνται διάφοροι τρόποι, ώστε να γίνονται οι πληροφορίες αυτής της επιστήμης, κατανοητές. Ένας από τους τρόπους είναι η χρήση τρισδιάστατων μοντέλων, η οποία είναι πολύ διαδεδομένη ακόμη και στις μέρες μας, όπου σε ένα αντίγραφο του ανθρώπινου σώματος έχουμε τη δυνατότητα να αφαιρέσουμε και να μελετήσουμε την δομή και την όψη των οργάνων.

Όμως η τεχνολογία έχει εξελιχθεί πολύ και έχουν γίνει διαθέσιμα πολλά προγράμματα ηλεκτρονικών υπολογιστών, τα οποία συνεχώς βελτιώνονται και εξελίσσονται για να παρέχουν στον χρήστη όσο το δυνατόν πιο λεπτομερείς πληροφορίες. Τα προγράμματα αυτά αρχικά περιείχαν μόνο εικόνες και γραπτές πληροφορίες αλλά πλέον περιλαμβάνουν και τρισδιάστατα μοντέλα με τα οποία ο χρήστης αλληλεπιδρά. Τα πιο σύγχρονα προγράμματα δίνουν στο χρήστη την δυνατότητα να περιστρέψει το μοντέλο, ή να εστιάσει σε συγκεκριμένα όργανα, ακόμα και να επιλέξει συγκεκριμένα συστήματα (πχ Νευρικό σύστημα).

Στόχος της πτυχιακής εργασίας είναι η ανάπτυξη μιας εφαρμογής η οποία να παρέχει στον χρήστη ένα απλό και φιλικό γραφικό περιβάλλον, από όπου θα μπορεί να αντλήσει πληροφορίες για την ανατομία του

ανθρώπινου σώματος. Το πρόγραμμα περιέχει γραφικές αναπαραστάσεις του ανθρώπινου σώματος, κάτι που επιτρέπει στο χρήστη να μπορεί εύκολα να εντοπίσει όποιο όργανο επιθυμεί να δει και να πάρει πληροφορίες σχετικά με αυτό.

Η δημιουργία αυτής της εφαρμογής είχε ακόμη ως στόχο την ενασχόληση με νέες τεχνολογίες όπως είναι το Entity Framework, το Linq και η δυνατότητα αλληλεπίδρασης με εφαρμογές μέσω φωνητικών εντολών, αλλά και η χρήση αυτών των τεχνολογιών για την ανάπτυξη μιας εφαρμογής.

1.2 Η ΕΦΑΡΜΟΓΗ

Για την δημιουργία του προγράμματος έγινε προσπάθεια να χρησιμοποιηθούν όσο το δυνατόν πιο σύγχρονες τεχνολογίες. Το πρόγραμμα είναι γραμμένο σε γλώσσα προγραμματισμού **C#** και χρησιμοποιήθηκε η **MySQL** ως σύστημα διαχείρισης της βάσης δεδομένων. Η εφαρμογή είναι φτιαγμένη ώστε να προσφέρει ένα δυναμικό περιβάλλον το οποίο διαμορφώνεται σύμφωνα με τα δεδομένα της βάσης δεδομένων. Επίσης δίνει στο χρήστη τη δυνατότητα να αλληλεπιδράσει με αυτή, με τη χρήση πληκτρολογίου και ποντικιού, περιλαμβάνει όμως και τη δυνατότητα χρήσης φωνητικών εντολών.

Το πρόγραμμα έχει δύο λειτουργίες. Η πρώτη είναι η λειτουργία παρουσίασης, όπου ο χρήστης έχει τη δυνατότητα με τη χρήση απλών μενού, να περιηγηθεί στα διάφορα συστήματα του ανθρώπινου σώματος, όπως είναι το μυϊκό, το σκελετικό και το πεπτικό και να πάρει πληροφορίες για αυτά. Επίσης μπορεί να κάνει μεγέθυνση, σμίκρυνση και να περιστρέψει το γραφικό μοντέλο με σκοπό να γνωρίσει τη δομή του ανθρώπινου σώματος και να πληροφορηθεί για τα διάφορα όργανα.

Το πρόγραμμα λειτουργεί και ως ιατρικό εργαλείο , το οποίο μπορεί να χρησιμοποιηθεί από γιατρούς και νοσοκομεία. Η σύνδεση της εφαρμογής με βάση δεδομένων δίνει στο χρήστη τη δυνατότητα εισαγωγής δεδομένων ασθενών και του ιστορικού τραυματισμών τους καθώς επίσης και να προβάλει αυτά τα δεδομένα πάνω στο γραφικό μοντέλο.

1.3 ΔΟΜΗ ΑΝΑΦΟΡΑΣ

Στο δεύτερο κεφάλαιο γίνεται αναφορά στις τεχνολογίες που χρησιμοποιήθηκαν για την κατασκευή της εφαρμογής. Αναλύεται ή γλώσσα προγραμματισμού C# η οποία σε συνδυασμό με τις τεχνολογίες WPF και Xaml συνθέτει το σκελετό του προγράμματος και του γραφικού περιβάλλοντος, εξηγείται η MySQL και πως μπορεί να βοηθήσει το LinQ και το Entity Framework τον προγραμματιστή, ώστε να συνδέσει το πρόγραμμα με τη βάση δεδομένων, εύκολα και απλά και γίνεται αναφορά στο Speech Recognition και τα πλεονεκτήματα που μας παρέχει.

Στο τρίτο κεφάλαιο αναλύεται η δομή και η αρχιτεκτονική του προγράμματος και γίνεται παρουσίαση των λειτουργιών και δυνατοτήτων του.

Τέλος στο τέταρτο κεφάλαιο δίνονται κάποια παραδείγματα λειτουργιών της εφαρμογής με σκοπό την εξοικείωση του χρήστη με το πρόγραμμα.

ΚΕΦΑΛΑΙΟ 2

ΠΕΡΙΓΡΑΦΗ ΤΕΧΝΟΛΟΓΙΩΝ

Στο κεφάλαιο αυτό γίνεται μια περιγραφή των τεχνολογιών που χρησιμοποιήθηκαν για την δημιουργία της εφαρμογής του ανθρώπινου σώματος. Εξηγείται η τεχνολογία .NET Framework καθώς και τα μέρη που την αποτελούν, δηλαδή το CLR, οι Class Libraries και οι διάφορες γλώσσες προγραμματισμού. Μια από αυτές τις γλώσσες είναι η C#, όπου μαζί με το WPF και τη γλώσσα XAML χρησιμοποιήθηκαν για τη δημιουργία της εφαρμογής και του γραφικού περιβάλλοντος αυτής. Επίσης γίνεται αναφορά στο σύστημα διαχείρισης βάσεων δεδομένων MYSQL και περιγράφεται πως το Entity Framework και η γλώσσα LINQ μπορούν να διευκολύνουν το χρήστη ώστε να επικοινωνεί πιο εύκολα με τη βάση δεδομένων. Τέλος αναλύεται η τεχνολογία φωνητικών εντολών που χρησιμοποιήθηκε και που παρέχει στο χρήστη τη δυνατότητα να αλληλεπιδρά με την εφαρμογή, με τη χρήση ομιλίας.

2.1 .NET FRAMEWORK

Το .NET Framework είναι ένας 'σκελετός' προγραμματισμού για υπολογιστές με λειτουργικό σύστημα Windows. Παρέχει τη δυνατότητα στους προγραμματιστές να χρησιμοποιήσουν διάφορες εξελιγμένες λειτουργίες έχοντας ως στόχο οι εφαρμογές τους να είναι συμβατές με όλα τα συστήματα που υποστηρίζουν και έχουν εγκατεστημένο το .NET Framework.

Η αρχιτεκτονική του .NET Framework αποτελείται από το Common Language Runtime (CLR) και ένα σύνολο από βιβλιοθήκες. Αυτά τα δύο στοιχεία αποτελούν την μηχανή εκτέλεσης και προγραμματισμού των APIs (Application Programming Interface) για την κατασκευή .NET εφαρμογών.

Η βιβλιοθήκη κλάσεων παρέχει υποστήριξη για ένα μεγάλο εύρος διεργασιών, συμπεριλαμβάνοντας ασφάλεια, είσοδο-έξοδο αρχείων, διαχείριση XML, υπηρεσίες web XML, επεξεργασία κειμένου, κατασκευή διεπαφής χρήστη, ASP.NET, και υπηρεσίες των Microsoft Windows.

Το .NET Framework σχεδιάστηκε ώστε να ικανοποιεί 3 στόχους. Αρχικά έπρεπε να κάνει τις εφαρμογές των Windows πιο αξιόπιστες, βελτιώνοντας το βαθμό της ασφάλειάς τους. Δεύτερον προοριζόταν για την απλούστευση της ανάπτυξης web εφαρμογών και υπηρεσιών (web services), οι οποίες θα έτρεχαν και σε φορητές συσκευές. Τρίτον, το Framework σχεδιάστηκε για να παρέχει ένα σύνολο βιβλιοθηκών που μπορούσαν να λειτουργήσουν με πολλές γλώσσες.

2.1.1 CLR

Οι γλώσσες προγραμματισμού συνήθως αποτελούνται από έναν compiler και ένα runtime περιβάλλον. Ο compiler μεταφράζει τον κώδικα σε εκτελέσιμο αρχείο, που μπορεί να εκτελεστεί από τους χρήστες. Το runtime περιβάλλον παρέχει ένα σύνολο υπηρεσιών του λειτουργικού συστήματος, στον εκτελέσιμο κώδικα. Οι υπηρεσίες αυτές είναι ενσωματωμένες σε ένα επίπεδο runtime (Runtime Layer), που επιτρέπει στον κώδικα να μην ασχολείται με λεπτομέρειες χαμηλού επιπέδου του λειτουργικού συστήματος. Τέτοιες λειτουργίες μπορεί να είναι η διαχείριση μνήμης, εγγραφή και ανάγνωση αρχείων κλπ.

Πριν το .NET Framework, κάθε γλώσσα είχε και το δικό της runtime περιβάλλον. Το περιβάλλον ενσωματωνόταν με τον εκτελέσιμο κώδικα και έπρεπε να εγκατασταθεί στο μηχάνημα του χρήστη. Το βασικό πρόβλημα με τα περιβάλλοντα αυτά, βρίσκεται στο ότι ήταν σχεδιασμένα για χρήση με μόνο μία γλώσσα. Δεν μπορούσαν να χρησιμοποιηθούν λειτουργίες από το περιβάλλον μιας γλώσσας, σε μία άλλη. Έτσι, ένας από τους βασικούς στόχους του .NET Framework ήταν να ενοποιήσει τα runtime περιβάλλοντα ώστε οι προγραμματιστές να μπορούν να χρησιμοποιούν μόνο ένα περιβάλλον. Έτσι δημιουργήθηκε το Common Language Runtime (CLR).

Οι εφαρμογές για .NET Framework μεταγλωττίζονται σε μια ενδιάμεση γλώσσα που ονομάζεται Microsoft Intermediate Language (MSIL). Όταν μια εφαρμογή πρόκειται να εκτελεστεί για πρώτη φορά, ο compiler του CLR μεταγλωττίζει τον MSIL κώδικα σε γλώσσα μηχανής. Το CLR όμως δεν είναι ένας απλός μεταγλωττιστής. Είναι επίσης υπεύθυνος για την εκτέλεση υπηρεσιών χαμηλού επιπέδου, όπως ο συλλέκτης απορριμμάτων, η διαχείριση λαθών και οι υπηρεσίες ασφαλείας και όλα αυτά για κάθε γλώσσα που δουλεύει με το .NET Framework.

Το CLR επίσης επιτρέπει στις γλώσσες να συνεργάζονται μεταξύ τους. Μπορεί για παράδειγμα να δεσμευτεί ένα κομμάτι μνήμης με κώδικα

γραμμένο στην Visual Basic .NET και το ίδιο κομμάτι να ελευθερωθεί με κώδικα γραμμένο σε άλλη γλώσσα όπως η C#.

2.1.2 .NET Class Libraries

Οι προγραμματιστές αρέσκονται στο να δουλεύουν με κώδικα που ήδη έχει δοκιμαστεί και φαίνεται να λειτουργεί, όπως για παράδειγμα το Win32 API και οι βιβλιοθήκες MFC. Η επαναχρησιμοποίηση κώδικα ήταν στόχος της προγραμματιστικής κοινότητας, από πολύ παλιά. Πολλές γλώσσες είχαν πρόσβαση σε κομμάτια κώδικα δοκιμασμένα, έτοιμα για εκτέλεση. Οι προγραμματιστές που χρησιμοποιούσαν την Visual C++ είχαν επωφεληθεί από βιβλιοθήκες όπως η Microsoft Foundation Classes (MFC) που τους επέτρεπαν να δημιουργήσουν εφαρμογές Windows εύκολα και γρήγορα. Ωστόσο, το ότι οι βιβλιοθήκες αυτές ήταν προορισμένες για μία μόνο γλώσσα σήμαινε ότι δεν μπορούσαν να χρησιμοποιηθούν με καμία άλλη γλώσσα.

Το .NET Framework παρέχει πολλές κλάσεις για να βοηθήσει τους προγραμματιστές στην επαναχρησιμοποίηση κώδικα. Οι κλάσεις είναι οργανωμένες σε namespaces. Τα πιο πολλά από τα ενσωματωμένα APIs είναι μέρος του System.*, ή Microsoft.* namespace. Αυτές οι δυο κλάσεις ενσωματώνουν ένα μεγάλο αριθμό κοινών λειτουργιών, όπως η ανάγνωση και εγγραφή αρχείων, επεξεργασία γραφικών, αλληλεπίδραση με βάσεις δεδομένων και επεξεργασία εγγράφων XML. Οι βιβλιοθήκη κλάσεων του .NET Framework χωρίζεται σε δύο τμήματα, τη βιβλιοθήκη βασικών κλάσεων και τη βιβλιοθήκη κλάσεων του Framework.

Η βιβλιοθήκη βασικών κλάσεων περιέχει ένα μικρό κομμάτι της βιβλιοθήκης κλάσεων και είναι ο πυρήνας των κλάσεων που χρησιμεύουν ως τα βασικά API του CLR. Η βιβλιοθήκη κλάσεων του Framework αναφέρεται σε όλες τις κλάσεις συγκεντρωτικά, συμπεριλαμβάνοντας τις βασικές κλάσεις. Πρόκειται δηλαδή για όλες τις βιβλιοθήκες κλάσεων που

υπάρχουν στο .NET Framework. Η βιβλιοθήκη κλάσεων του Framework περιλαμβάνει μεταξύ άλλων, βιβλιοθήκες όπως: Windows forms, ADO.NET, ASP.NET, Language Integrated Query, Windows Presentation Foundation (WPF) και Windows Communication Foundation. Το καλύτερο σημείο βέβαια είναι ότι οι βιβλιοθήκες είναι διαθέσιμες σε κάθε γλώσσα που λειτουργεί με το .NET Framework.

2.1.3 .NET Γλώσσες Προγραμματισμού

Η Microsoft παρέχει ένα σύνολο γλωσσών που είναι ήδη συμβατές με το .NET. Η C# είναι μία από αυτές. Επίσης δημιουργήθηκαν νέες εκδόσεις της Visual Basic και της Visual C++ όπως και μία νέα έκδοση της Jscript.NET. Ένα πολύ σημαντικό στοιχείο είναι ότι οι συμβατές γλώσσες με το .NET δεν είναι αποκλειστικά της Microsoft, αφού η εταιρία έχει δημοσιεύσει πλήρης τεκμηρίωση που δείχνει το πώς οι κατασκευαστές γλωσσών μπορούν να κάνουν τις γλώσσες τους συμβατές με το .NET, και διάφοροι κατασκευαστές το επιχείρησαν όπως η COBOL και η Perl. Υπάρχουν αυτή τη στιγμή πάνω από 20 γλώσσες τρίτων κατασκευαστών που μπορούν και λειτουργούν στο περιβάλλον .NET Framework.

Οι γλώσσες αυτές χρησιμοποιούν από κοινού τις βιβλιοθήκες κλάσεων του .NET Framework. Αυτό επιτυγχάνεται μέσω του CLR. Πως όμως αναγνωρίζει τη γλώσσα το CLR; Κατά την εκτέλεση της εφαρμογής, ο κώδικας, ασχέτως της γλώσσας προγραμματισμού που χρησιμοποιήθηκε, μεταγλωττίζεται σε ένα αρχείο που περιέχει έναν ειδικού τύπου κώδικα που ονομάζεται Microsoft Intermediate Language (MSIL). Αυτή η ενδιάμεση γλώσσα είναι χαμηλού επιπέδου και είναι σχεδιασμένη για να την αναγνωρίζει το CLR. Οπότε σε όποια γλώσσα και αν είναι γραμμένος ο κώδικας, ο compiler θα τη μετατρέψει σε MSIL. Για παράδειγμα αν έχουμε μια ακέραια μεταβλητή integer σε Visual Basic και μια int σε C#, στην MSIL αναπαριστώνται ως System.Int32.

2.1.4 ASP.NET

Το Internet αρχικώς δημιουργήθηκε για την παροχή στατικού περιεχομένου στους Web browsers. Το Active Server Pages (ASP) δημιουργήθηκε από τη Microsoft για να προσθέσει δυναμικές ιδιότητες στις web σελίδες. Αυτό το πέτυχε με προγραμματισμό πίσω από την web σελίδα, κυρίως σε VB Script. Όταν οι χρήστες επισκέπτονταν ένα site τους ζητούνταν να επιβεβαιώσουν κάποιες πληροφορίες (είτε χειροκίνητα είτε με χρήση cookies) και στη συνέχεια ένα script δημιουργούσε τη σελίδα ανάλογα με τα στοιχεία που εισήγαγε ο χρήστης.

Η ASP.NET βελτιώνει κατά πολύ την αρχική ASP. Με την ASP το HTML και το script βρίσκονταν στο ίδιο έγγραφο. Στην ASP.NET το script και το HTML βρίσκονται σε διαφορετικά έγγραφα.

Η ASP.NET υποστηρίζει το Web Forms. Το Web Forms επιτρέπει στον προγραμματιστή να σύρει και να αφήσει (drag and drop) controls των φορμών, και να δημιουργηθεί αυτόματα ο κώδικας από πίσω, όπως ακριβώς σε μία τυπική εφαρμογή Windows.

2.2 C#

Τον Ιανουάριο του 1999, ο Anders Hejlsberg σχημάτισε μια ομάδα με σκοπό τη δημιουργία μιας νέας γλώσσας αντικειμενοστραφούς προγραμματισμού με το όνομα COOL (C-like Object Oriented Language). Για λόγους δικαιωμάτων όμως, η Microsoft επέλεξε να μην κρατήσει αυτό το όνομα. Μέχρι τον Ιούλιο του 2000, που ανακοινώθηκε το .NET η γλώσσα είχε μετονομαστεί σε C# και είχαν εισαχθεί σε αυτή οι βιβλιοθήκες κλάσεων του .NET.

Η C# είναι μια απλή, μοντέρνα, γενικής χρήσης, αντικειμενοστραφής γλώσσα προγραμματισμού και μοιάζει πολύ με την Java και τη C++. Ο μεταγλωττιστής της γλώσσας παράγει ένα ενδιάμεσο κώδικα, ανάλογο με τον object code της Java, την Microsoft Intermediate Language (MSIL). Κατά την εκτέλεση της εφαρμογής, το Common Language Runtime, το περιβάλλον δηλαδή μέσα στο οποίο θα τρέξει το πρόγραμμα, διαβάζει τον ενδιάμεσο κώδικα (MSIL) και τον μεταγλωττίζει σε κώδικα Windows. Η γλώσσα περιλαμβάνει στοιχεία όπως ο συλλέκτης απορριμμάτων (garbage collector) και η διαχείριση μνήμης και έτσι επιτρέπει στους καινούργιους προγραμματιστές να μάθουν τη γλώσσα πιο εύκολα από τη C και τη C++. Επίσης περιλαμβάνει μια εκτενή βιβλιοθήκη κλάσεων και υποστηρίζει διαχείριση λαθών, πολλαπλούς τύπους πολυμορφισμού και διαχωρισμό των διεπαφών από τις υλοποιήσεις. Το 2012 με την παρουσίαση του νέου Visual Studio 2012, η Microsoft παρουσίασε και την 5η έκδοση της C# καθώς και το .NET Framework 4.5 .

Η C# ακολουθεί παρόμοιο συντακτικό με τη C++. Για να σημειώσουμε το τέλος μιας δήλωσης χρησιμοποιούμε το ελληνικό ερωτηματικό. Για να δηλώσουμε μια μεταβλητή χρησιμοποιούμε το ίσον (πχ `int x = 1;`), αλλά αν θέλουμε να συγκρίνουμε δύο μεταβλητές χρησιμοποιούμε το ίσον δύο φορές (πχ `x == y`). Οι αγκύλες '[']' χρησιμοποιούνται με πίνακες, για να τους δηλώσουμε αλλά και για να πάρουμε την τιμή μιας συγκεκριμένης θέσης του πίνακα. Τα άγκιστρα χρησιμοποιούνται για να ομαδοποιήσουμε

τις δηλώσεις. Οι δηλώσεις ομαδοποιούνται σε μεθόδους (συναρτήσεις), οι μέθοδοι σε κλάσεις και οι κλάσεις σε Namespaces.

2.3 WPF

Το Windows Presentation Foundation (WPF) είναι ένα σύστημα παρουσιάσεων επόμενης γενιάς, για τη δημιουργία εφαρμογών σε Windows με εκπληκτικά γραφικά περιβάλλοντα. Με το WPF μπορούμε να δημιουργήσουμε μια μεγάλη ποικιλία από εφαρμογές.

Ο πυρήνας του WPF είναι μια ανεξαρτήτου ανάλυσης και βασισμένης σε διανύσματα μηχανή, η οποία είναι φτιαγμένη ώστε να εκμεταλλεύεται τις σύγχρονες συσκευές γραφικών. Το WPF συμπεριλαμβάνεται στο Microsoft .Net Framework, έτσι ώστε να ενσωματώνει τα υπόλοιπα στοιχεία της κλάσης βιβλιοθηκών του .Net Framework. Επεκτείνεται με ένα σετ εκτενών ιδιοτήτων ανάπτυξης εφαρμογών, που περιλαμβάνουν την Extensible Markup Language (Xaml), στοιχεία ελέγχου, 2-D και 3-D γραφικά, animations, πολυμέσα και κείμενο.

Το WPF συστήνει μια νέα γλώσσα, την XAML. Η XAML είναι σχεδιασμένη ως μία πιο αποδοτική μέθοδος ανάπτυξης εφαρμογών γραφικού περιβάλλοντος. Το πλεονέκτημα που προσδίδει η XAML στο WPF είναι ότι πρόκειται για μια δηλωτική γλώσσα που επιτρέπει στον χρήστη να περιγράφει τις συμπεριφορές των στοιχείων χωρίς τη χρήση διαδικαστικού προγραμματισμού. Αυτό επιτρέπει τον διαχωρισμό της εφαρμογής σε δύο επίπεδα. Το μοντέλο και τον κώδικα.

2.4 XAML

Η XAML (Extensible Markup Language) είναι μια δηλωτική γλώσσα βασισμένη στην XML, που δημιουργήθηκε από τη Microsoft. Αρχικά το ακρωνύμιο σήμαινε Extensible Avalon Markup Language, όπου Avalon ήταν η κωδική ονομασία του Windows Presentation Foundation (WPF). Η σύνταξη της XAML εστιάζει στο User Interface και άρα είναι ξεχωριστή από τον κώδικα της εφαρμογής που βρίσκεται από πίσω. Με λίγα λόγια η XAML αναλαμβάνει το παρουσιαστικό κομμάτι της εφαρμογής, δηλαδή τα διάφορα κουμπιά, εικόνες κτλ. Η γλώσσα αυτή περιγράφει αντικείμενα, παραμέτρους, καθώς και τις σχέσεις που υπάρχουν μεταξύ τους, όπως και τις σχέσεις μεταξύ αντικειμένων και 'παιδιών'.

Η XAML χρησιμοποιείται εκτενώς στις τεχνολογίες .NET Framework 3.0, 4.0 και 4.5, ειδικά στο WPF, το Silverlight, το Windows Workflow Foundation (WF) και το Windows Runtime XAML Framework. Παρότι η XAML παρουσιάστηκε για χρήση στα Windows, η WPF/E (Windows Presentation Foundation/Everywhere) θα φέρει σταδιακά την XAML και σε άλλες πλατφόρμες και συσκευές. Η XAML επίσης χρησιμοποιείται στο Visual Studio 2012 για εφαρμογές Windows 8 Metro.

Ένα αρχείο XAML μπορεί να μεταγλωττιστεί σε αρχείο .BAML (Binary Application Markup Language), το οποίο μπορεί να εισαχθεί σαν πόρος σε ένα .NET Framework assembly. Κατά την εκτέλεση, η μηχανή του Framework εξάγει το .BAML αρχείο, το αναλύει και δημιουργεί ένα αντίστοιχο εικονικό δέντρο WPF ή μια ροή εργασίας. Όταν η XAML χρησιμοποιείται σε WPF, η XAML περιγράφει εικονικές διεπαφές χρήστη. Το WPF επιτρέπει τον προσδιορισμό 2D και 3D γραφικών, περιστροφές, animations και μια ποικιλία από άλλα εφέ και ιδιότητες.

2.5 MYSQL

Η MySQL είναι το πιο δημοφιλές, open-source σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων. Το όνομα προέκυψε από το όνομα της κόρης του συν-ιδρυτή Michael Widenius, My. Το ακρωνύμιο SQL προέρχεται από το Structured Query Language (Γλώσσα Δομημένων Αιτημάτων).

Η MySQL είναι κατά κύριο λόγο ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) και παρέχεται χωρίς γραφικά εργαλεία για τη διαχείριση των βάσεων δεδομένων, ή των δεδομένων που βρίσκονται μέσα σε αυτές. Αντιθέτως η αλληλεπίδραση γίνεται με τη χρήση γραμμής εντολών. Ωστόσο υπάρχουν και γραφικά εργαλεία τα οποία μπορούν να χρησιμοποιήσουν οι χρήστες και τα οποία παρέχουν μεταξύ άλλων, δυνατότητες δημιουργίας και επεξεργασίας βάσεων δεδομένων, δημιουργία αντιγράφων ασφαλείας και έλεγχο κατάστασης βάσεων. Το επίσημο σετ γραφικών εργαλείων, το MySQL Workbench έχει αναπτυχθεί από την Oracle και είναι διαθέσιμο δωρεάν. Άλλα προγράμματα που παρέχουν γραφική υποστήριξη είναι το phpMyAdmin, Adminer, OpenOffice.org Base και το Sequel Pro που απευθύνεται σε χρήστες λειτουργικού συστήματος Mac OS X.

Η MySQL είναι γραμμένη σε C και C++. Ο αναλυτής της SQL είναι γραμμένος σε yacc και ένα λεξιλογικό αναλυτή που έχει αναπτύξει η ίδια η εταιρεία. Πολλές γλώσσες προγραμματισμού χρησιμοποιούν βιβλιοθήκες για να αποκτήσουν πρόσβαση σε βάσεις δεδομένων της MySQL. Ανάμεσά τους είναι το MySQL Connector/Net που επιτρέπει τη σύνδεση του Visual Studio με τη MySQL.

2.6 ENTITY FRAMEWORK

Το Entity Framework συμπεριλήφθηκε με το .Net Framework 3.5 Service Pack 1 και το Visual Studio 2008 SP1. Αν και η πρώτη έκδοση κατακρίθηκε από το προγραμματιστικό κοινό οι επόμενες εκδόσεις που κυκλοφόρησαν ήταν αρκετά βελτιωμένες.

Στόχος του Entity Framework είναι να γεφυρώσει το κενό ανάμεσα στον αντικειμενοστραφή προγραμματισμό και τον παραδοσιακό προγραμματισμό, όταν πρόκειται για σύνδεση με κάποια βάση δεδομένων και την επεξεργασία των δεδομένων αυτής. Πρόκειται για έναν Object Relational Mapper (ORM) που επιτρέπει στους developers να δουλεύουν με σχεσιακές βάσεις δεδομένων χρησιμοποιώντας αντικείμενα. Έτσι καταργείται η ανάγκη κώδικα για την επικοινωνία με τη βάση δεδομένων.

Το Entity Framework παράγει αντικείμενα και entities σύμφωνα με τους πίνακες της βάσης δεδομένων και παρέχει μηχανισμούς για τις βασικές λειτουργίες CRUD (Create, Read, Update, Delete), τη δημιουργία και διαχείριση σχέσεων '1 προς 1', '1 προς πολλά' και 'πολλά προς πολλά', μεταξύ των πινάκων, καθώς και τη δυνατότητα να υπάρχουν σχέσεις κληρονομικότητας μεταξύ των entities.

Αυτή η σχετικά νέα τεχνολογία έχει φέρει αρκετά πλεονεκτήματα. Αρχικώς μειώνει την ανάγκη κώδικα για την σύνδεση με τη βάση δεδομένων. Μέσα από ένα απλοποιημένο μενού ο προγραμματιστής καθορίζει την βάση με την οποία θέλει να συνδεθεί και το Entity Framework κατασκευάζει ένα οπτικό μοντέλο στο οποίο απεικονίζονται οι πίνακες με τα πεδία τους και τις όποιες σχέσεις τους συνδέουν. Αυτομάτως το κάθε πεδίο έχει μετατραπεί σε αντικείμενο το οποίο ο χρήστης μπορεί να χρησιμοποιήσει στον κώδικά του σαν να το είχε ορίσει ο ίδιος. Επίσης στις πιο πρόσφατες εκδόσεις του Entity Framework δίνεται η δυνατότητα του Model First όπου ο προγραμματιστής μπορεί να σχεδιάσει πρώτα το μοντέλο με τη χρήση του Entity Framework και μετά με βάση αυτό να δημιουργήσει την βάση δεδομένων.

Το Entity Framework είναι φτιαγμένο ώστε να αλληλεπιδρά με τον SQL Server αλλά μπορεί να συνδεθεί και με άλλα συστήματα διαχείρισης βάσεων δεδομένων όπως είναι η MySQL. Για να επιτευχθεί αυτό απαιτείται η εγκατάσταση ενός 'connector' ο οποίος παρέχεται από την ιστοσελίδα του κατασκευαστή.

2.7 LINQ

Το LINQ (Language Integrated Query) είναι ένα στοιχείο του .NET Framework που προσθέτει δυνατότητες εκτέλεσης ερωτημάτων σε βάσεις δεδομένων από γλώσσες .NET (Visual Basic, C# κτλ) και παρουσιάστηκε μαζί με το .NET Framework 3.5 τον Νοέμβριο του 2007.

Το LINQ ορίζει ένα σύνολο από ονόματα μεθόδων που ονομάζονται χειριστές ερωτημάτων, μαζί με κανόνες μετάφρασης από τις λεγόμενες εκφράσεις ερωτημάτων σε εκφράσεις χρησιμοποιώντας αυτά τα ονόματα μεθόδων, τις εκφράσεις lambda και ανώνυμους τύπους. Αυτά μπορούν, για παράδειγμα, να χρησιμοποιηθούν για να προβληθούν και να φιλτραριστούν δεδομένα σε πίνακες, XML (LINQ to XML), σχεσιακές βάσεις, και άλλες πηγές δεδομένων.

Οι βασικοί χειριστές ερωτημάτων που περιέχει το LINQ είναι:

• Select	• Reverse
• Where	• GroupBy
• SelectMany	• Distinct
• Sum / Min / Max / Average	• Union / Intersect / Except
• Aggregate	• SequenceEqual
• Join / GroupJoin	• First / FirstOrDefault / Last / LastOrDefault
• Take / TakeWhile	• Single
• Skip / SkipWhile	• ElementAt
• ofType	• Any / All / Contains
• Concat	• Count
• OrderBy / ThenBy	

Το LINQ παρέχει στον προγραμματιστή έναν ενοποιημένο τρόπο για αποστολή ερωτημάτων προς πηγές δεδομένων. Έτσι ο προγραμματιστής μπορεί να αλληλεπιδρά με όλα τα συστήματα διαχείρισης βάσεων δεδομένων (SQL, MySQL), χωρίς να χρειάζεται να γνωρίζει τον τρόπο σύνταξης των ερωτημάτων για το κάθε σύστημα. Αρκεί να γνωρίζει τον τρόπο σύνταξης ερωτημάτων σε LINQ και το LINQ αναλαμβάνει να κάνει

τη μετάφραση στην αντίστοιχη γλώσσα ερωτημάτων που χρησιμοποιείται σε κάθε σύστημα.

2.8 SPEECH RECOGNITION

Η εφαρμογή που αναπτύχθηκε για τις ανάγκες της πτυχιακής εργασίας, παρέχει στο χρήστη εκτός από το ποντίκι και το πληκτρολόγιο και τη δυνατότητα αλληλεπίδρασης με φωνητικές εντολές.

Το πρόγραμμα χρησιμοποιεί τις βιβλιοθήκες της Microsoft, με τις οποίες λειτουργεί και η αναγνώριση ομιλίας των Windows και λειτουργεί μόνο με αγγλικές εντολές αφού η ελληνική γλώσσα δεν υποστηρίζεται από τη Microsoft. Οι εφαρμογές που παρέχουν δυνατότητα φωνητικών εντολών συνήθως έχουν δύο τρόπους αναγνώρισης φωνής. Υπάρχουν οι 'Speaker Dependent' και οι 'Speaker Independent' εφαρμογές. 'Speaker Dependent' λέγονται οι εφαρμογές που υποβάλλουν τον χρήστη σε μια σειρά από τεστ πριν την κανονική λειτουργία. Αυτά τα τεστ περιλαμβάνουν την ανάγνωση κάποιου κειμένου, κάποιων γραμμάτων και κάποιων λέξεων προκειμένου να ρυθμιστεί σωστά ή αναγνώριση ομιλίας για τον συγκεκριμένο χρήστη. Η εφαρμογή της πτυχιακής είναι μια 'Speaker Independent' εφαρμογή, δηλαδή για να λειτουργήσει δεν χρειάζεται πρώτα ο χρήστης να περάσει αυτά τα τεστ, καθώς έχει αρκετά περιορισμένο λεξιλόγιο.

Η αναγνώριση ομιλίας των Windows, προσφέρει μια βασική υποδομή για αναγνώριση ομιλίας, που ψηφιοποιεί ακουστικά σήματα και ανακτά λέξεις και στοιχεία ομιλίας από εισόδους ήχου (πχ μικρόφωνο). Σε μια εφαρμογή η ανάκτηση λέξεων και στοιχείων ομιλίας γίνεται από το λεξιλόγιο που έχει καθορίσει ο προγραμματιστής και το οποίο ονομάζεται γραμματική. Η γραμματική αποτελείται από ένα σύνολο κανόνων, για να προσδιορίζονται οι λέξεις και οι φράσεις που θα αναγνωρίζει η εφαρμογή. Εφόσον έχει οριστεί η γραμματική στη συνέχεια, οι μηχανές αναγνώρισης ομιλίας 'SpeechRecognizer' και 'SpeechRecognitionEngine' με τη χρήση της ορισθέντας γραμματικής παρέχουν πρόσβαση στις μηχανές αναγνώρισης ομιλίας της τεχνολογίας ομιλίας των Windows. Αφού γίνει η αναγνώριση της λέξης, ή των λέξεων, ή των προτάσεων τότε ανάλογα με

την εντολή του χρήστη, η εφαρμογή θα εκτελέσει και την ανάλογη λειτουργία.

Για να λειτουργήσουν οι φωνητικές εντολές, πρέπει να δηλωθεί μια μηχανή αναγνώρισης ομιλίας (*SpeechRecognitionEngine*) και καθορίζεται η συσκευή από την οποία θα εισάγεται ο ήχος. Παράλληλα, δημιουργείται η γραμματική του προγράμματος, οι λέξεις δηλαδή που θα μπορεί να χρησιμοποιήσει ο χρήστης για την αλληλεπίδρασή του με το πρόγραμμα. Στη συνέχεια γίνεται φόρτωση της γραμματικής στην *Speech Recognition Engine* και δημιουργείται ένας event handler που ενεργοποιείται κάθε φορά που η μηχανή αναγνωρίζει μία λέξη. Όταν γίνει αναγνώριση το πρόγραμμα εισέρχεται σε μια Switch και εκτελούνται οι αντίστοιχες εντολές που έχουν καθοριστεί για την λέξη που εκφώνησε ο χρήστης.

```
void SpeechRecognizer_Setup()
{
    recognizer = new SpeechRecognitionEngine();
    recognizer.SetInputToDefaultAudioDevice();

    Choices choices = new Choices();
    choices.Add("ExitProgram");
    choices.Add("ReturnHome");

    Grammar grammar = new Grammar(grbuilder);
    recognizer.LoadGrammar(grammar);
    recognizer.SpeechRecognized += new EventHandler<SpeechRecognizedEventArgs>(recognizer_SpeechRecognized);
}

void recognizer_SpeechRecognized(object sender, SpeechRecognizedEventArgs e)
{
    foreach (RecognizedWordUnit word in e.Result.Words)
    {
        switch (word.Text)
        {
            case "ExitProgram":
                if (bClose.IsEnabled == true)
                    bClose_Click(bClose, new RoutedEventArgs());
                else
                    MessageBox.Show("The function 'Exit' Program is not available at the moment.");
                break;

            case "ReturnHome":
                if (bHome.IsEnabled == true)
                    bHome_Click(bHome, new RoutedEventArgs());
                else
                    MessageBox.Show("The function 'Home' is not available at the moment.");
                break;
        }
    }
}
```

Εικόνα 1 - Ο υλοποιημένος κώδικας για τις φωνητικές εντολές.

Η τεχνολογία φωνητικών εντολών είναι πολύ σημαντική, καθώς επιτρέπει τον έλεγχο μιας εφαρμογής χωρίς να χρειάζεται να υπάρχουν άλλες συσκευές εισόδου και επιτρέπει τη χρήση της εφαρμογής ακόμη και από άτομα με προβλήματα κινητικότητας που δεν έχουν τη δυνατότητα χρήσης πληκτρολογίου και ποντικιού.

ΚΕΦΑΛΑΙΟ 3

ΔΟΜΗ & ΑΡΧΙΤΕΚΤΟΝΙΚΗ

Το κεφάλαιο αυτό πραγματεύεται την δομή και οργάνωση της πληροφορίας της εφαρμογής. Εξηγείται ο τρόπος που λειτουργεί το πρόγραμμα και ποιες πληροφορίες αντλεί από τη βάση δεδομένων αλλά και πως τις οργανώνει.

Επίσης αναλύεται η αρχιτεκτονική 3-tier και τα πλεονεκτήματα που μας παρέχει. Το πρόγραμμα ακολουθεί κατά κάποιο τρόπο την αρχιτεκτονική αυτή και τονίζονται οι αντιστοιχίες των επιπέδων της εφαρμογής με τα επίπεδα της αρχιτεκτονικής 3-tier.

Τέλος, γίνεται περιγραφή του προγράμματος και των διάφορων στοιχείων που περιέχει. Αναφέρεται ο τρόπος αρχικοποίησης του προγράμματος και ο τρόπος ανάκτησης των δεδομένων από τη βάση. Ακόμη γίνεται λόγος για τα παράθυρα, τα γραφικά, τα κουμπιά, τις φωνητικές εντολές και για όλα τα υπόλοιπα στοιχεία που αποτελούν την εφαρμογή.

3.1 ΠΛΗΡΟΦΟΡΙΑ-ΔΕΔΟΜΕΝΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Παρατηρώντας το ανθρώπινο σώμα από ανατομική άποψη, μπορούμε να διακρίνουμε διάφορα επίπεδα, ή συστήματα. Τα κύρια είναι το σκελετικό, το μυϊκό, το αναπνευστικό, το πεπτικό, το οδοντικό και το σύστημα του εγκεφάλου, τα οποία και εμφανίζονται στη μπάρα συστημάτων του προγράμματος. Εκτός από αυτά υπάρχουν και τα υπό-επίπεδα, τα οποία εμφανίζονται όταν ο χρήστης επιλέξει να κάνει zoom, ή να περιστρέψει το γραφικό μοντέλο. Για παράδειγμα, το κρανίο είναι υπό-επίπεδο του σκελετικού συστήματος.

Τα επίπεδα και οι πληροφορίες για αυτά βρίσκονται αποθηκευμένα στη βάση δεδομένων, που σημαίνει ότι υπάρχει η δυνατότητα προσθήκης και άλλων επιπέδων. Για να ξεχωρίζει το πρόγραμμα ποιά επίπεδα είναι κύρια και ποιά υπό-επίπεδα, γίνεται έλεγχος της τιμής του πεδίου *ChildrenOfLevel* στον πίνακα των επιπέδων. Σε αυτό το πεδίο βρίσκεται ο κωδικός του επιπέδου στο οποίο ανήκει κάποιο υπό-επίπεδο. Εάν όμως το πεδίο αυτό έχει την τιμή 0, τότε το συγκεκριμένο επίπεδο είναι κύριο επίπεδο και εμφανίζεται στη μπάρα συστημάτων του προγράμματος.

Επίσης στον πίνακα επιπέδων της βάσης δεδομένων, υπάρχει το πεδίο *PhotoID*. Το πεδίο αυτό περιέχει τον κωδικό μιας φωτογραφίας, ο οποίος αντιστοιχίζεται σε έναν κωδικό του πίνακα *Photos*. Ο πίνακας *Photos*, περιέχει τις διαδρομές, όπου βρίσκονται αποθηκευμένες οι εικόνες. Έτσι όταν ο χρήστης επιλέξει ένα κύριο επίπεδο, ή επιλέξει να κάνει zoom, ή περιστροφή σε κάποιο υπό-επίπεδο, στην εφαρμογή εμφανίζεται το γραφικό μοντέλο που αποτελείται από την αντίστοιχη εικόνα.

Επάνω στην εικόνα του επιπέδου, υπάρχει η δυνατότητα εμφάνισης σημείων πληροφοριών. Η ύπαρξη κάποιου σημείου πάνω στο γραφικό μοντέλο, ειδοποιεί το χρήστη ότι μπορεί να πάρει πληροφορίες για το συγκεκριμένο σημείο, να το περιστρέψει, να κάνει μεγέθυνση σε αυτό, ή

σμίκρυνση, αν βρίσκεται σε κάποιο υπό-επίπεδο. Τα σημεία πληροφοριών είναι καταχωρημένα στη βάση δεδομένων. Κατά την επιλογή ενός επιπέδου, τα σημεία αντλούνται από τη βάση και τοποθετούνται πάνω στην εικόνα. Τα σημεία αναπαριστώνται με κόκκινες ή κίτρινες ελλείψεις. Η τοποθέτηση γίνεται με χρήση συντεταγμένων x, y , οι τιμές των οποίων βρίσκονται αποθηκευμένες στη βάση δεδομένων. Το κάθε σημείο πληροφοριών παρέχει στο χρήστη πληροφορίες για το σημείο που αντιπροσωπεύει πάνω στην εικόνα. Οι πληροφορίες είναι αποθηκευμένες στη βάση δεδομένων σε αγγλική και ελληνική γλώσσα και εμφανίζονται μέσω του παραθύρου πληροφοριών στο χρήστη.

Επίσης τα σημεία δίνουν στο χρήστη την δυνατότητα να κάνει μεγέθυνση και σμίκρυνση του επιπέδου στο οποίο βρίσκεται. Στον πίνακα *Points* υπάρχει το πεδίο *ZoomInToLevelID* στο οποίο βρίσκεται καταχωρημένος ο κωδικός του επιπέδου-συστήματος που θα εμφανιστεί, αν επιλέξουμε να μεγεθύνουμε στο συγκεκριμένο σημείο. Ομοίως στον πίνακα *levels*, υπάρχει το πεδίο *ChildrenOfLevel* το οποίο εκτός από τον διαχωρισμό των επιπέδων σε κύρια και υπό-επίπεδα, ο κωδικός που περιέχει χρησιμοποιείται και για την εμφάνιση του αντίστοιχου επιπέδου, όταν ο χρήστης επιλέξει να κάνει σμίκρυνση.

Τέλος, για την περιστροφή του μοντέλου, υπάρχουν τα πεδία *RotatePlusToLevelID* και *RotateMinusToLevelID*, τα οποία περιέχουν τους κωδικούς των επιπέδων που θα εμφανιστούν, αν ο χρήστης επιλέξει δεξιά ή αριστερή περιστροφή του μοντέλου.

Εκτός από την γραφική παρουσίαση του ανθρώπινου σώματος, το πρόγραμμα μπορεί να χρησιμοποιηθεί και από ιατρούς και νοσοκομεία, καθώς παρέχει τη δυνατότητα προσθήκης και επεξεργασίας στοιχείων ασθενών αλλά και των ιατρικών τους αρχείων, τα οποία αποθηκεύονται στη βάση δεδομένων και εμφανίζονται γραφικώς πάνω στα μοντέλα του προγράμματος. Έτσι ο ιατρός μπορεί εύκολα να εντοπίσει τα όποια ιατρικά προβλήματα είχαν οι ασθενείς του, αλλά και να προσθέσει καινούργια ή να επεξεργαστεί τα υπάρχοντα δεδομένα.

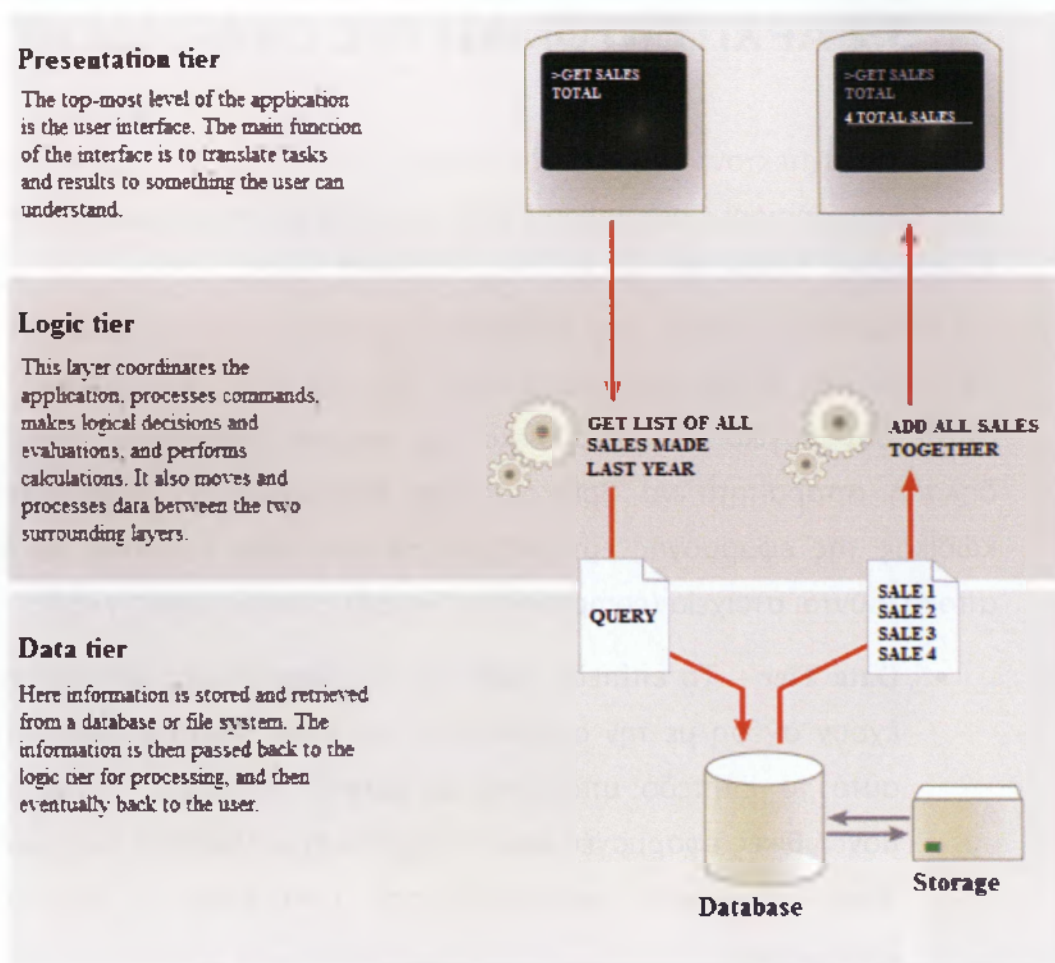
Όταν ο χρήστης επιλέξει έναν συγκεκριμένο ασθενή, τα προσωπικά του στοιχεία εμφανίζονται στη μπάρα στοιχείων ασθενή του προγράμματος. Τα στοιχεία αυτά βρίσκονται καταχωρημένα στη βάση δεδομένων και αποτελούνται από τον κωδικό του ασθενή, ο οποίος είναι μοναδικός για κάθε ασθενή, το επίθετο, το όνομα, το όνομα πατρός, το όνομα μητρός, την ημερομηνία γέννησης, το τηλέφωνό του και την κατάστασή του, δηλαδή αν είναι εν ζωή, ή αν έχει αποβιώσει. Στο πρόγραμμα ωστόσο, δεν εμφανίζεται η ημερομηνία γέννησης του ασθενή, αλλά υπολογίζεται και εμφανίζεται η ηλικία του.

Αν κάποιος ασθενής έχει υποστεί κάποιον τραυματισμό, τότε το πρόγραμμα μας πληροφορεί με την ύπαρξη κίτρινων ελλείψεων, πάνω στα αντίστοιχα σημεία του μοντέλου. Αυτά τα σημεία αντλούνται από τη βάση δεδομένων, και συγκεκριμένα από τον πίνακα *MedicalRecords*. Το πρόγραμμα αναζητεί στον πίνακα, όλες τις εγγραφές που περιέχουν τον κωδικό του επιλεγμένου ασθενή. Ύστερα αντλεί τους κωδικούς των σημείων στα οποία έχει υποστεί τραυματισμό ο ασθενής και μετατρέπει τα σημεία αυτά, πάνω στο γραφικό μοντέλο, από κόκκινα σε κίτρινα. Τέλος όταν ο χρήστης επιλέξει να πληροφορηθεί για ένα σημείο τραυματισμού, εμφανίζεται το παράθυρο πληροφοριών το οποίο περιέχει τα δεδομένα για τον τραυματισμό του ασθενή.

3.2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Η αρχιτεκτονική 3-tier αναπτύχθηκε από τον John J. Donovan στην Open Environment Corporation (OEC), μια εταιρεία που ίδρυσε ο ίδιος στο Cambridge. Πρόκειται για μία αρχιτεκτονική λογισμικού και ένα μοτίβο σχεδιασμού λογισμικού που βοηθάνε στην οργάνωση των δεδομένων. Η αρχιτεκτονική 3-tier αποτελείται από τρία επίπεδα. Τα επίπεδα αυτά αποτελούν λογικό διαχωρισμό και όχι φυσικό διαχωρισμό. Δεν είναι δηλαδή απαραίτητο να βρίσκονται σε διαφορετικούς υπολογιστές. Ο κώδικας της εφαρμογής εμπεριέχεται σε αυτόνομα κομμάτια τα οποία αποκαλούνται στοιχεία (components). Τα τρία επίπεδα είναι τα εξής:

- **Data Tier** – Το επίπεδο Data Tier αναφέρεται στις υπηρεσίες που έχουν σχέση με την ανάκτηση, ή την αποθήκευση δεδομένων. Σε αυτό το επίπεδο υπάγονται οι βάσεις δεδομένων, παλαιότερες μονολιθικές εφαρμογές που παρέχουν πρόσβαση σε δεδομένα και άλλες εξωτερικές εφαρμογές που αποτελούν οι ίδιες πηγές δεδομένων.
- **Logic Tier** - Η 'λογική' των εφαρμογών. Οι υπηρεσίες που προσφέρει η εφαρμογή. Σε αυτό το επίπεδο υπάγονται τα κομμάτια της εφαρμογής που περιέχουν τη λογική της εφαρμογής. Δηλαδή οι κανόνες και η λειτουργικότητα της εφαρμογής.
- **Presentation Tier** – Πρόκειται για τα δεδομένα που παρουσιάζονται στο χρήστη, δηλαδή τα διάφορα User Interfaces που συναντούμε στο πρόγραμμα. Αυτά χωρίζονται σε thin clients και rich clients. Τα thin clients είναι σχετικά απλά interfaces, όπως για παράδειγμα μια σελίδα HTML, την οποία βλέπουμε μέσω ενός browser. Τα rich clients από την άλλη, είναι πλούσια interfaces, όπως είναι για παράδειγμα μια 32-bit εφαρμογή των Windows. Υπάρχουν φυσικά και τα ενδιάμεσα στάδια, όπως είναι η χρήση ActiveX components και DHTML σε μια ιστοσελίδα.



Εικόνα 2 - Τα επίπεδα της αρχιτεκτονικής 3-tier. Πηγή:
http://en.wikipedia.org/wiki/Multitier_architecture

Οι εφαρμογές που έχουν βασιστεί στην αρχιτεκτονική 3-tier αποκαλούνται server-centric, διότι επιτρέπουν στα κομμάτια της εφαρμογής που ανήκουν στο Logic tier να τρέχουν σε ειδικούς sever που είναι εντελώς ανεξάρτητοι από το user interface ή την υλοποίηση των βάσεων δεδομένων που ανήκουν στο data tier. Αυτή η απεξάρτηση της λογικής της εφαρμογής επιφέρει πολλά πλεονεκτήματα. Προσφέρει υποστήριξη πολλαπλών προγραμματιστικών γλωσσών, όπου τα στοιχεία της εφαρμογής μπορούν να γραφτούν σε διαφορετικές προγραμματιστικές γλώσσες. Για παράδειγμα η λογική της εφαρμογής μπορεί να γραφτεί σε C++, ενώ το user interface θα μπορούσε να υλοποιηθεί σε Visual Basic. Ακόμη προσφέρει καλύτερη πρόσβαση στα δεδομένα, καθώς οι περιορισμοί που πιθανόν να υπήρχαν στη σύνδεση πολλαπλών χρηστών στη βάση δεδομένων εκμηδενίζονται, γιατί η βάση, πλέον 'μιλάει' μόνο με

κάποια συγκεκριμένα στοιχεία. Επίσης οι πόροι του συστήματος χρησιμοποιούνται πολύ καλύτερα, καθώς στην αρχιτεκτονική 3-tier οι συνδέσεις προς τη βάση δεδομένων γίνονται μόνο όταν χρειάζονται και απελευθερώνονται αμέσως μόλις εκπληρώσουν το σκοπό τους. Τέλος η αρχιτεκτονική 3-tier παρέχει βελτιωμένη ασφάλεια, διότι τα στοιχεία που ανήκουν στο Logic Tier μπορούν πλέον να διασφαλιστούν χρησιμοποιώντας κάποια ενοποιημένη υποδομή. Η πρόσβαση στο κάθε στοιχείο μπορεί να δοθεί ή να αρνηθεί ξεχωριστά, διευκολύνοντας έτσι το ρόλο των διαχειριστών.

Η εφαρμογή παρουσίασης του ανθρώπινου σώματος δεν ακολουθεί ακριβώς την αρχιτεκτονική 3-tier, αλλά υπάρχει μια αρκετά καλή διάκριση των επιπέδων. Στην εφαρμογή, το data tier υλοποιήθηκε με MySQL. Το πρόγραμμα είναι δυναμικό, συνεπώς τα δεδομένα αντλούνται από τη βάση δεδομένων για τη δημιουργία των γραφικών και των στοιχείων ελέγχου, όπως είναι τα διάφορα buttons. Ακόμη τα δεδομένα για τους ασθενείς τίθενται προς προβολή και επεξεργασία στον χρήστη.

Το Entity Framework και η γλώσσα ερωτημάτων LinQ, αναλαμβάνουν την επικοινωνία του προγράμματος με τη βάση δεδομένων, δημιουργώντας το Logic Tier. Έτσι η εφαρμογή δεν 'μιλάει' με πίνακες αλλά με αντικείμενα, τα οποία έχουν ιδιότητες. Για παράδειγμα, τα δεδομένα σχετικά με το ιστορικό τραυματισμών ενός ασθενή που εμφανίζονται στο παράθυρο γρήγορης πρόσβασης του προγράμματος, αντλούνται από τη βάση δεδομένων μέσω του entity framework και της Linq. Η αναζήτηση, όπως φαίνεται και στην εικόνα, γίνεται στα αντικείμενα medical records και points και δεν υπάρχει υλοποιημένος κώδικας επικοινωνίας του προγράμματος με τη βάση δεδομένων. Αντιθέτως το Entity Framework αναλαμβάνει να κάνει τη επικοινωνία με τη βάση.

```
int patid = int.Parse(tbID.Text.Trim().ToString());  
  
var results = from q in context.medicalrecords  
              from r in context.points  
              where (q.PatientID == patid) && (q.PointID == r.ID)  
              select r;
```

Εικόνα 3 - Κώδικας εύρεσης των σημείων τραυματισμού ενός ασθενή, όπου την επικοινωνία με τη βάση την αναλαμβάνει το Entity Framework

Τέλος το Presentation Tier υλοποιήθηκε με τη χρήση του WPF, της XAML και του Speech Recognition. Το WPF και η XAML αναλαμβάνουν να εμφανίσουν στον χρήστη το γραφικό User Interface (GUI), ενώ το Speech Recognition παρέχει έναν εναλλακτικό τρόπο επικοινωνίας του χρήστη με το πρόγραμμα.

3.3 ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΟΘΟΝΩΝ ΚΑΙ ΛΕΙΤΟΥΡΓΙΩΝ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Το πρόγραμμα παρουσίασης του ανθρώπινου σώματος έχει αναπτυχθεί έτσι ώστε το περιεχόμενό του να είναι δυναμικό. Γι αυτό το λόγο, έχει φτιαχτεί μια βάση δεδομένων σε MySQL, όπου βρίσκονται αποθηκευμένα όλα τα δεδομένα που χρειάζεται το πρόγραμμα για να αρχικοποιηθεί κατά την εκκίνησή του. Στη βάση δεδομένων βρίσκονται αποθηκευμένα τα επίπεδα, τα σημεία πληροφοριών για κάθε επίπεδο και οι πληροφορίες που περιγράφουν αυτά, οι διαδρομές για τις εικόνες που χρησιμοποιούνται για τα γραφικά μοντέλα αλλά και τα στοιχεία των ασθενών, καθώς και τα ιατρικά τους ιστορικά. Με βάση αυτά τα δεδομένα το πρόγραμμα παίρνει την τελική του μορφή κάθε φορά.

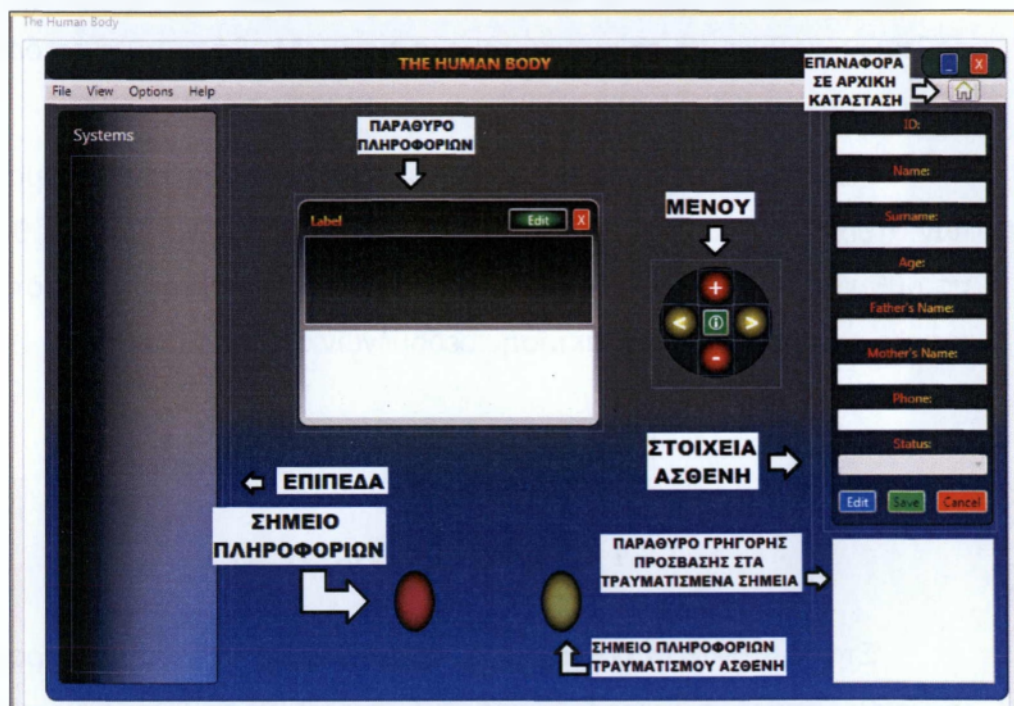
Παρακάτω γίνεται μια λεπτομερής ανάλυση του προγράμματος και των στοιχείων ελέγχου του. Ειδικότερα θα γίνει λόγος για τα παράθυρα και τις ρυθμίσεις τους, τα γραφικά, τα κουμπιά και τον τρόπο αρχικοποίησης του προγράμματος και ανάκτησης δεδομένων από τη βάση.

3.3.1 ΤΟ ΚΕΝΤΡΙΚΟ ΠΑΡΑΘΥΡΟ

Στο κεντρικό παράθυρο του προγράμματος γίνεται η παρουσίαση του ανθρώπινου σώματος. Ο χρήστης μπορεί να επιλέξει ποιο επίπεδο-σύστημα επιθυμεί να δει (σκελετικό, μυϊκό κτλ) και μπορεί να πάρει πληροφορίες για αυτό. Το παράθυρο αυτό περιέχει ένα μενού, από όπου ο χρήστης μπορεί να προσθέσει έναν ασθενή στη βάση ή να αναζητήσει κάποιον ήδη αποθηκευμένο. Από αυτό το μενού μπορεί επίσης, να αλλάξει τη γλώσσα του προγράμματος, να ενεργοποιήσει και να απενεργοποιήσει τη δυνατότητα φωνητικών εντολών και να προβάλει τη βοήθεια για το πρόγραμμα. Το παράθυρο αυτό είναι δυναμικό και τα

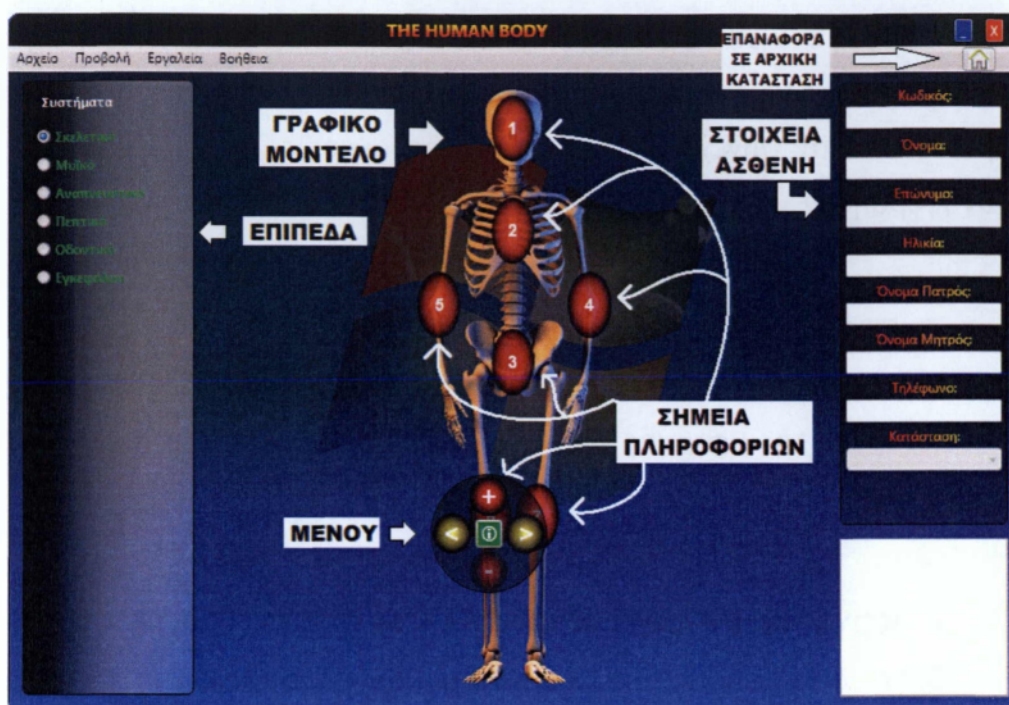
επίπεδα, τα γραφικά και οι πληροφορίες εισάγονται από τη βάση δεδομένων.

Στην παρακάτω εικόνα φαίνεται ο σκελετός του προγράμματος, πως δηλαδή είναι πριν εισαχθούν τα δεδομένα από τη βάση. Κάποια από τα στοιχεία ελέγχου που περιέχει δεν είναι ορατά ή τοποθετημένα όπως στην εικόνα όταν εμφανιστεί το πρόγραμμα στην οθόνη, αλλά εμφανίζονται με κάποιες συγκεκριμένες ενέργειες του χρήστη που θα εξηγηθούν στη συνέχεια. Τα στοιχεία αυτά είναι το παράθυρο πληροφοριών, το μενού και τα σημεία πληροφοριών.



Εικόνα 4 - Ο σκελετός του προγράμματος

Κατά την εκκίνηση του προγράμματος το πρόγραμμα συνδέεται με τη βάση δεδομένων και λαμβάνει τα επίπεδα τα οποία προσθέτει υπό τη μορφή radio buttons στην αριστερή πλευρά του παραθύρου. Στη συνέχεια επιλέγεται αυτόματα το πρώτο επίπεδο-σύστημα και φορτώνεται το γραφικό μοντέλο που του αντιστοιχεί. Πάνω στο μοντέλο προστίθενται τα σημεία πληροφοριών. Παρακάτω φαίνεται το αρχικό παράθυρο, που εμφανίζεται όταν ενεργοποιηθεί το πρόγραμμα.



Εικόνα 5 - Τα στοιχεία ελέγχου της εφαρμογής

Το παράθυρο έχει ρυθμιστεί ώστε το φόντο να είναι διαφανές εν μέρει.

Στην κορυφή του προγράμματος υπάρχει η γραμμή τίτλου με το περιεχόμενο "THE HUMAN BODY". Επίσης υπάρχουν τα κουμπιά ελαχιστοποίησης και τερματισμού του προγράμματος. Στη συνέχεια είναι η γραμμή μενού, όπου υπάρχει και το κουμπί επαναφοράς του προγράμματος στην αρχική κατάσταση. Στην αριστερή πλευρά υπάρχει η λίστα επιπέδων-συστημάτων, τα περιεχόμενα της οποίας είναι δυναμικά και φορτώνονται από τη βάση δεδομένων κατά την εκκίνηση του προγράμματος. Στο κέντρο εμφανίζεται το γραφικό μοντέλο, σύμφωνα με το επιλεγμένο σύστημα και πάνω σε αυτό εμφανίζονται τα σημεία πληροφοριών. Όταν ο χρήστης μετακινήσει το ποντίκι πάνω σε κάποιο από τα σημεία αυτά εμφανίζεται το μενού πληροφοριών. Το μενού δίνει στον χρήστη τη δυνατότητα να εστιάσει πιο κοντά ή πιο μακριά στο επιλεγμένο σημείο, να περιστρέψει το μοντέλο ή να πάρει πληροφορίες για το συγκεκριμένο σημείο. Στη δεξιά πλευρά εμφανίζονται τα στοιχεία του ασθενή και υπάρχει και το παράθυρο γρήγορης πρόσβασης στα σημεία

τραυματισμού. Σε αυτό το παράθυρο εμφανίζονται τα συστήματα-επίπεδα του ασθενή στα οποία υπάρχει κάποιος τραυματισμός και πατώντας πάνω σε κάποιο από αυτά, αυτομάτως εμφανίζεται το αντίστοιχο γραφικό μοντέλο. Στη συνέχεια αναλύονται λεπτομερώς όλα τα στοιχεία που αναφέρθηκαν.

- **ΓΡΑΜΜΗ ΤΙΤΛΟΥ**

Στη γραμμή τίτλου εμφανίζεται το όνομα του προγράμματος. Το περιεχόμενο της γραμμής είναι 'THE HUMAN BODY' και η ελληνική του μετάφραση είναι 'ΤΟ ΑΝΘΡΩΠΙΝΟ ΣΩΜΑ'.

- **ΚΟΥΜΠΙΑ ΕΛΑΧΙΣΤΟΠΟΙΗΣΗΣ ΚΑΙ ΤΕΡΜΑΤΙΣΜΟΥ**

Στην δεξιά μεριά του προγράμματος, υπάρχουν τα κουμπιά ελαχιστοποίησης και τερματισμού. Το κουμπί ελαχιστοποίησης επιτρέπει στον χρήστη να ελαχιστοποιήσει το πρόγραμμα, όταν χρειάζεται να δουλέψει με κάποιο άλλο πρόγραμμα. Το κουμπί τερματισμού επιτρέπει την έξοδο από το πρόγραμμα. Όταν ο χρήστης επιλέξει να τερματιστεί το πρόγραμμα, εμφανίζεται ένα παράθυρο διαλόγου που ρωτάει τον χρήστη αν είναι σίγουρος για την ενέργειά του. Έτσι αποφεύγεται ο κατά λάθος τερματισμός του προγράμματος.

- **ΓΡΑΜΜΗ ΜΕΝΟΥ**

Η γραμμή μενού προσφέρεται στα αγγλικά, αλλά και στα ελληνικά, ανάλογα με την γλώσσα προβολής που επιλέγει ο χρήστης. Η γραμμή μενού προσφέρει τις επιλογές, *File-Αρχείο*, *View-Προβολή*, *Options-Επιλογές*, *Help-βοήθεια* και το πλήκτρο 'επαναφορά στην αρχική κατάσταση του προγράμματος'.

Αρχείο Προβολή Επιλογές Βοήθεια



Εικόνα 6 - Η γραμμή μενού

Όταν ο χρήστης επιλέξει *Αρχείο (File)* εμφανίζεται ένα υπό-μενού, το οποίο παρέχει τις επιλογές, *Αναζήτηση Ασθενή (Find Patients)* και *Προσθήκη Ασθενή (Add Patient)*, οι οποίες αναλύονται παρακάτω και την επιλογή *Έξοδος (Exit)*, που τερματίζει το πρόγραμμα.

Στην επιλογή *Προβολή (View)*, εμφανίζεται το υπό-μενού *Σημεία Πληροφοριών (InfoPoints)*. Εκεί παρέχεται στο χρήστη η επιλογή να κάνει ορατά ή να εξαφανίσει τα σημεία πληροφοριών από το γραφικό μοντέλο, με σκοπό την καλύτερη προβολή του μοντέλου. Όταν τα σημεία πληροφοριών είναι ενεργά τότε στην αριστερή πλευρά της επιλογής εμφανίζεται ένα '✓'.

Δίπλα από την *Προβολή (View)*, υπάρχουν οι *Επιλογές (Options)*. Ο χρήστης έχει τη δυνατότητα να επιλέξει ανάμεσα στην αγγλική και την ελληνική γλώσσα ως γλώσσα προβολής και επίσης μπορεί να ενεργοποιήσει και να απενεργοποιήσει τις *Φωνητικές Εντολές (Voice Commands)*.

Στη συνέχεια υπάρχει το μενού *Βοήθεια (Help)*. Εκεί ο χρήστης μπορεί να πληροφορηθεί για τον τρόπο λειτουργίας του προγράμματος (*Program Functions*) και για τις φωνητικές εντολές (*Voice Commands*). Επίσης εμφανίζονται πληροφορίες για το πρόγραμμα, πχ ποιος είναι ο δημιουργός του προγράμματος, ποιες τεχνολογίες χρησιμοποιήθηκαν για την υλοποίησή του, πότε κατασκευάστηκε κτλ.

Τέλος, στη γραμμή μενού εμφανίζεται το *κουμπί επαναφοράς του προγράμματος στην αρχική κατάσταση*. Το κουμπί συμβολίζεται με ένα σπιτάκι και επιτρέπει στο χρήστη ανά πάσα στιγμή, σε οποιοδήποτε επίπεδο και αν είναι να επιστρέψει στο αρχικό γραφικό μοντέλο. Επίσης αν έχει επιλεγθεί κάποιος ασθενής, τότε με το κουμπί επαναφοράς το πρόγραμμα επιστρέφει σε λειτουργία παρουσίασης.

• ΣΥΣΤΗΜΑΤΑ

Η γραμμή συστήματα εμφανίζεται στην αριστερή πλευρά του προγράμματος. Εκεί εμφανίζονται όλα τα κύρια επίπεδα-συστήματα του

ανθρώπινου σώματος. Πρόκειται για μια δυναμική μπάρα, τα περιεχόμενα της οποίας, αντλούνται από τη βάση δεδομένων. Όλα τα επίπεδα είναι αποθηκευμένα στον πίνακα *Levels* της βάσης δεδομένων. Στον πίνακα αυτό βρίσκονται τα κύρια επίπεδα αλλά και τα υπό-επίπεδα, δηλαδή τα επίπεδα-συστήματα που εμφανίζονται στο πρόγραμμα όταν ο χρήστης κάνει zoom σε κάποιο από τα κύρια επίπεδα. Ο πίνακας περιέχει τα πεδία *ID*, *PhotoID*, *EnglishName*, *GreekName* και *ChildrenOfLevel*. Στο πεδίο *ChildrenOfLevel* υπάρχει ένας κωδικός που αντιστοιχίζεται με τον κωδικό του πεδίου *ID*. Έτσι το πρόγραμμα ξέρει πότε ένα επίπεδο είναι κύριο και πότε υπό-επίπεδο. Τα κύρια επίπεδα είναι αυτά που εμφανίζονται στη μπάρα συστημάτων. Για να ξεχωρίζουν από τα υπό-επίπεδα, στο πεδίο *ChildrenOfLevel* της βάσης δεν έχουν κάποιο κωδικό αλλά τον αριθμό μηδέν. Κατά την εκκίνηση του προγράμματος λοιπόν, το πρόγραμμα ελέγχει τη βάση για τα επίπεδα με αριθμό μηδέν στο πεδίο *ChildrenOfLevel* και τα προσθέτει στη μπάρα συστήματα με τη μορφή radio buttons. Έτσι όταν ο χρήστης επιλέξει ένα radio button στο κέντρο του προγράμματος εμφανίζεται το αντίστοιχο γραφικό μοντέλο.

Η επιλογή συστήματος μπορεί να γίνει και με τη χρήση φωνητικών εντολών. Ο χρήστης πρέπει να εκφωνήσει το επίπεδο-σύστημα που θέλει να μεταφερθεί και να προσθέσει τη λέξη *System. PX Respiratory System*.

• ΓΡΑΦΙΚΟ ΜΟΝΤΕΛΟ

Το γραφικό μοντέλο εμφανίζεται στο κέντρο του παραθύρου. Πρόκειται για εικόνες και εμφανίζονται όταν ο χρήστης επιλέξει κάποιο από τα επίπεδα. Όπως αναφέρθηκε και πιο πάνω, στη βάση, ο πίνακας *Levels* περιέχει το πεδίο *PhotoID*. Όταν ο χρήστης πατήσει σε ένα radio button της μπάρας Συστήματα, το πρόγραμμα ψάχνει στη βάση να βρει ποιο επίπεδο αντιστοιχεί στην επιλογή του χρήστη και ποιος είναι ο κωδικός του πεδίου *PhotoID* για αυτό το επίπεδο. Στη συνέχεια αντιστοιχεί το *PhotoID* με το *ID* του πίνακα *Photos*. Στον πίνακα *Photos* βρίσκονται οι διαδρομές όπου είναι αποθηκευμένες οι εικόνες στον υπολογιστή. Όταν βρεθεί η εικόνα, τότε εμφανίζεται στο πρόγραμμα.

- **ΣΗΜΕΙΑ ΠΛΗΡΟΦΟΡΙΩΝ**

Τα *σημεία πληροφοριών* είναι ελλείψεις, κόκκινες ή κίτρινες που εμφανίζονται πάνω στο γραφικό μοντέλο. Υποδηλώνουν ότι τα σημεία αυτά του γραφικού μοντέλου παρέχουν πρόσβαση στο μενού πληροφοριών.

Τα κίτρινα σημεία εμφανίζονται μόνο όταν ο χρήστης έχει επιλέξει κάποιον ασθενή. Υποδηλώνουν ότι στο συγκεκριμένο σημείο, ο ασθενής έχει υποστεί κάποιον τραυματισμό. Τα κόκκινα σημεία, υποδηλώνουν το ακριβώς αντίθετο. Όταν ο χρήστης τοποθετήσει το δείκτη του ποντικιού σε ένα κόκκινο ή ένα κίτρινο σημείο πληροφοριών, τότε εμφανίζεται το μενού πληροφοριών. Αυτό με τη σειρά του δίνει στο χρήστη τη δυνατότητα να δει τις πληροφορίες για το συγκεκριμένο σημείο, να περιστρέψει το μοντέλο και να κάνει zoom στο σημείο αυτό.

Τα σημεία πληροφοριών είναι αριθμημένα. Οι αριθμοί δίνουν την ικανότητα στο χρήστη να τα επιλέξει με τη χρήση φωνητικών εντολών, χωρίς να χρειάζεται να μετακινεί το ποντίκι. Το μόνο που πρέπει να κάνει είναι να πει τη λέξη 'Point' και να προσθέσει τον αριθμό του σημείου στα αγγλικά. *PX Point Two*.

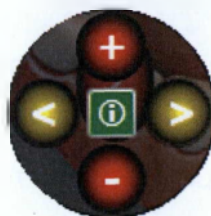


- **ΜΕΝΟΥ ΠΛΗΡΟΦΟΡΙΩΝ**

Το *μενού πληροφοριών* δεν είναι ορατό πάντα. Εμφανίζεται όταν ο χρήστης τοποθετήσει το δείκτη του ποντικιού πάνω σε ένα σημείο πληροφοριών. Τότε εμφανίζεται το μενού πληροφοριών. Μόλις ο χρήστης απομακρύνει το ποντίκι από το σημείο πληροφοριών τότε το μενού παραμένει ορατό για 3 δευτερόλεπτα και μετά εξαφανίζεται ξανά. Ωστόσο

αν ο χρήστης μετακινηθεί σε κάποιο άλλο σημείο πληροφοριών πριν το πέρας των τριών δευτερολέπτων, τότε το μενού μετακινείται αυτόματα από το παλιό στο καινούργιο σημείο πληροφοριών.

Το μενού πληροφοριών δίνει 5 επιλογές στον χρήστη. Του επιτρέπει να κάνει *zoom in* και *zoom out*, περιστροφή του μοντέλου προς τα δεξιά ή προς τα αριστερά και τέλος επιτρέπει την εμφάνιση του παραθύρου πληροφοριών σημείου. Οι επιλογές αυτές δεν είναι πάντα όλες ενεργοποιημένες. Εκτός από το *κουμπί εμφάνισης του παραθύρου πληροφοριών*, τα υπόλοιπα τέσσερα εξαρτώνται από το επίπεδο και το σημείο. Αν παράδειγμα υπάρχει καταχωρημένη στη βάση δεδομένων, μόνο η μπροστά πλευρά του μυϊκού συστήματος του ανθρώπινου σώματος, τότε τα κουμπιά περιστροφής του μοντέλου δεξιά και αριστερά είναι απενεργοποιημένα.

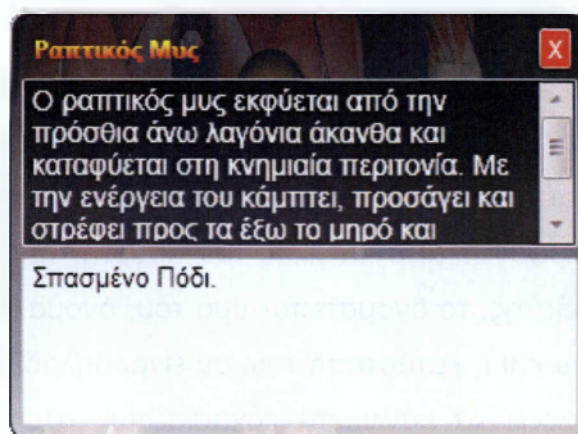


Εικόνα 7 - Το μενού πληροφοριών

Οι επιλογές του μενού πληροφοριών είναι διαθέσιμες και με τη χρήση φωνητικών εντολών. Για να γίνει μεγέθυνση στο επίπεδο-σύστημα ο χρήστης πρέπει να πει τη λέξη '*Zoom*', ενώ για να κάνει σμίκρυνση πρέπει να πει τη λέξη '*Unzoom*'. Η περιστροφή πραγματοποιείται με τις λέξεις '*Rotate Left*' και '*Rotate Right*' για αριστερή και δεξιά περιστροφή του γραφικού μοντέλου, αντίστοιχα. Τέλος για να εμφανιστεί το παράθυρο πληροφοριών σημείου, ο χρήστης πρέπει να πει τη λέξη '*InfoBox*'.

• ΠΑΡΑΘΥΡΟ ΠΛΗΡΟΦΟΡΙΩΝ ΣΗΜΕΙΟΥ

Στο παράθυρο πληροφοριών συστήματος εμφανίζονται οι γενικές πληροφορίες για το επιλεγμένο σημείο και οι πληροφορίες-σημειώσεις για τον τραυματισμό του ασθενή. Στην κορυφή του παραθύρου εμφανίζεται η ονομασία του επιλεγμένου σημείου. Στο πρώτο πεδίο εμφανίζονται οι γενικές πληροφορίες για το επιλεγμένο σημείο. Αν για παράδειγμα το επιλεγμένο σημείο είναι το ζυγωματικό οστό, τότε στο πρώτο πεδίο εμφανίζεται το μήνυμα 'Τα δύο ζυγωματικά οστά σχηματίζουν τα μάλα του προσώπου'. Το δεύτερο πεδίο είναι άδειο αν το πρόγραμμα βρίσκεται σε λειτουργία παρουσίασης. Αν όμως έχει επιλεχθεί κάποιος ασθενής, ο οποίος έχει υποστεί κάποιον τραυματισμό στο επιλεγμένο σημείο, τότε στο πεδίο εμφανίζεται το μήνυμα που έχει γράψει ο γιατρός, που περιγράφει το πρόβλημα του ασθενή.



Εικόνα 8 - Το παράθυρο πληροφοριών σημείου

Στο παράθυρο πληροφοριών σημείου, ο χρήστης δεν μπορεί να επεξεργαστεί τα δεδομένα του πρώτου πεδίου γιατί κάτι τέτοιο δεν υποστηρίζεται αυτή τη στιγμή από την εφαρμογή. Τα δεδομένα από το δεύτερο πεδίο, όμως μπορούν να αλλάξουν. Ο χρήστης μπορεί να αλλάξει και να διαγράψει το περιεχόμενο ή να προσθέσει πληροφορίες, όταν το πεδίο είναι άδειο. Για να ενεργοποιηθεί η επεξεργασία των πληροφοριών τραυματισμού του ασθενή, βασική προϋπόθεση είναι να έχει επιλεχθεί κάποιος ασθενής. Στη συνέχεια με το πάτημα του κουμπιού *Επεξεργασία* στη μπάρα όπου εμφανίζονται τα στοιχεία του ασθενή, ενεργοποιείται στο

παράθυρο πληροφοριών σημείου το κουμπί *επεξεργασίας*. Αν πατηθεί και αυτό τότε το δεύτερο πεδίο με τις πληροφορίες τραυματισμού ενεργοποιείται και ο χρήστης μπορεί να εισάγει τα δεδομένα που επιθυμεί. Όταν ο χρήστης πατήσει στο κουμπί *Επεξεργασία*, στο παράθυρο πληροφοριών σημείου, τότε το κουμπί αυτό αντικαθίσταται με το κουμπί *αποθήκευσης*. Όταν ο χρήστης εισάγει τα δεδομένα που επιθυμεί, αν πατήσει το κουμπί *αποθήκευσης*, τα δεδομένα που εισήγαγε αποθηκεύονται στη βάση δεδομένων. Τέλος πρέπει να επιστρέψει στη μπάρα στοιχείων ασθενή και να πατήσει *αποθήκευση*.

- **ΣΤΟΙΧΕΙΑ ΑΣΘΕΝΗ**

Η μπάρα που περιέχει τα στοιχεία του ασθενή βρίσκεται στη δεξιά πλευρά του προγράμματος. Κατά την εκκίνηση του προγράμματος η μπάρα αυτή δεν περιέχει κάποια πληροφορία. Όταν ο χρήστης επιλέξει κάποιον ασθενή, τότε τα στοιχεία του εισάγονται από τη βάση δεδομένων και εμφανίζονται σε αυτή τη μπάρα. Ταυτόχρονα, εμφανίζεται και ενεργοποιείται το κουμπί *επεξεργασίας (Edit)* των στοιχείων του ασθενή. Στη μπάρα εμφανίζεται ο κωδικός του ασθενή, τον οποίο δεν μπορεί να αλλάξει ο χρήστης, το ονοματεπώνυμό του, όνομα πατρός και μητρός, το τηλέφωνό του και η κατάστασή του, αν είναι δηλαδή εν ζωή ή όχι. Όταν ο χρήστης επιλέξει να κάνει επεξεργασία των στοιχείων αυτών, τότε το κουμπί *επεξεργασίας* εξαφανίζεται και εμφανίζονται τα κουμπιά *Αποθήκευση (Save)* και *Ακύρωση (Cancel)*.

- **ΠΑΡΑΘΥΡΟ ΓΡΗΓΟΡΗΣ ΠΡΟΣΒΑΣΗΣ ΣΤΑ ΤΡΑΥΜΑΤΙΣΜΕΝΑ ΣΗΜΕΙΑ**

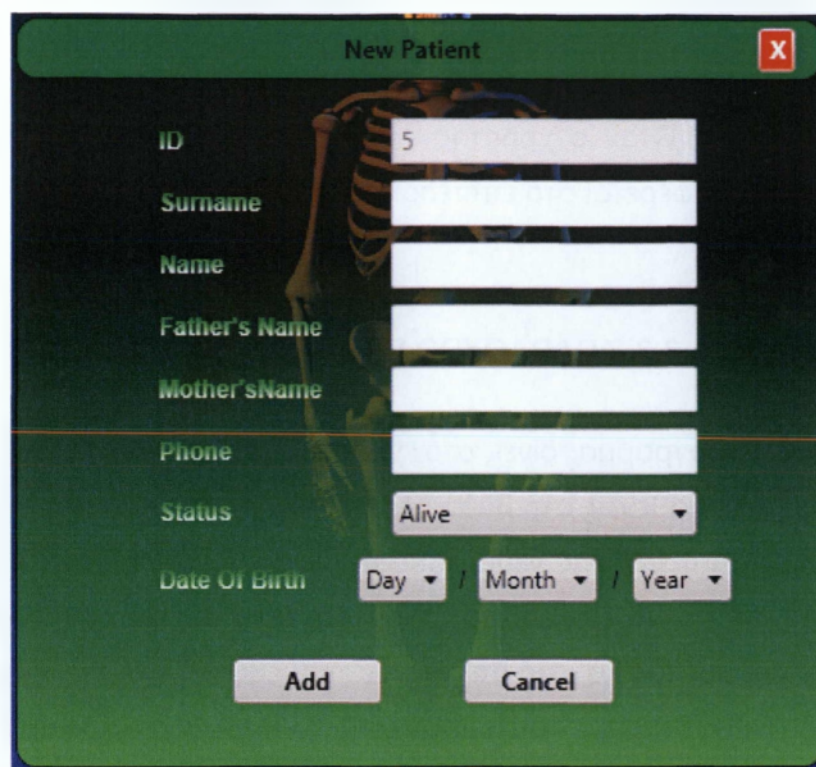
Το παράθυρο γρήγορης πρόσβασης παρέχει στο χρήστη τη δυνατότητα να βλέπει ποια επίπεδα-συστήματα έχουν υποστεί κάποιο τραυματισμό για τον επιλεγμένο ασθενή και να μεταφέρεται σε αυτά με ένα κλικ. Έτσι δεν χρειάζεται να ψάχνει να βρει αν και σε ποιο επίπεδο υπάρχει κίτρινο σημείο πληροφοριών, δηλαδή κάποιος τραυματισμός. Το πρόγραμμα αντλεί όλα τα ιατρικά αρχεία του επιλεγμένου ασθενή από τον

πίνακα Medical Records της βάσης δεδομένων και εμφανίζει το όνομα του σημείου στο οποίο υπάρχει τραυματισμός σε αυτό το παράθυρο υπό τη μορφή λίστας. Όταν ο χρήστης επιλέξει κάποια από τις εγγραφές, τότε αυτόματα μεταφέρεται στο επίπεδο όπου υπάρχει τραυματισμός.

3.3.2 ΠΑΡΑΘΥΡΟ ΠΡΟΣΘΗΚΗΣ ΑΣΘΕΝΗ

Το πρόγραμμα δίνει στο χρήστη τη δυνατότητα να προσθέσει καινούργιους ασθενείς στη βάση δεδομένων. Αυτό επιτυγχάνεται μέσω του παραθύρου προσθήκης νέου ασθενή. Για να εμφανιστεί το παράθυρο, ο χρήστης πρέπει να επιλέξει το μενού *Αρχείο (File)*, από τη γραμμή μενού του προγράμματος και στο υπό-μενού που εμφανίζεται επιλέγει *Προσθήκη Ασθενή (New Patient)*. Τότε εμφανίζεται το παράθυρο προσθήκης ασθενή. Η εμφάνιση αυτού του παραθύρου, αυτομάτως απενεργοποιεί όλες τις λειτουργίες του κύριου παραθύρου.

Το παράθυρο αυτό είναι φτιαγμένο ώστε να δέχεται όλα τα στοιχεία που χρειάζονται για να προστεθεί επιτυχώς ένας νέος ασθενής στη βάση δεδομένων. Στην κορυφή του παραθύρου εμφανίζεται ο τίτλος *New Patient*. Από κάτω υπάρχει το πεδίο κωδικού (*ID*) του νέου ασθενή. Αυτό το πεδίο είναι κλειδωμένο και δεν μπορεί να το επεξεργαστεί ο χρήστης. Αντιθέτως ο κωδικός δημιουργείται από το πρόγραμμα και έτσι αποφεύγονται οι διπλό-εγγραφές στους κωδικούς της βάσης δεδομένων. Στη συνέχεια υπάρχουν τα πεδία Επίθετο, Όνομα, Όνομα πατρός, Όνομα μητρός και Τηλέφωνο. Επίσης υπάρχει το πεδίο Κατάστασης (*Status*) του ασθενή, όπου ο χρήστης μέσα από το έτοιμο μενού, επιλέγει αν ο ασθενής είναι εν ζωή ή όχι. Τέλος ζητείται από τον χρήστη να εισάγει την ημερομηνία γέννησης του ασθενή. Για να αποφευχθούν τα λάθη και οι λάθος διατυπώσεις, η ημερομηνία επιλέγεται από τρία διαφορετικά μενού. Το πρώτο μενού περιέχει τις ημέρες, το δεύτερο τους μήνες και το τρίτο τα έτη.



Εικόνα 9 - Το παράθυρο προσθήκης νέου ασθενή

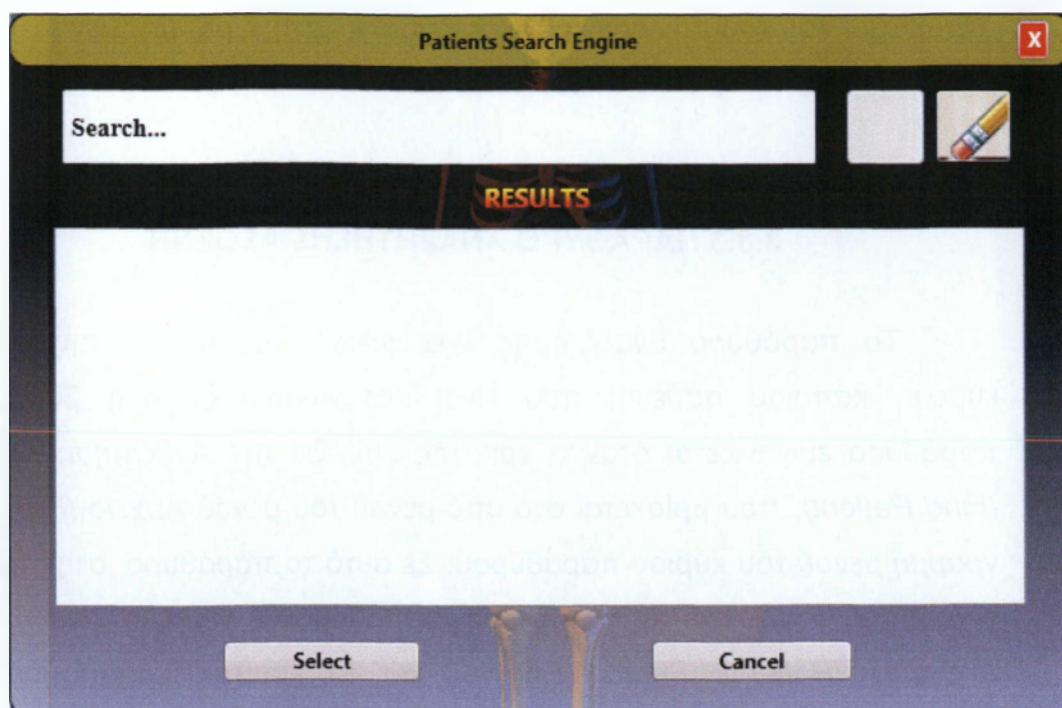
Όταν συμπληρωθούν όλα τα πεδία, τότε ο χρήστης μπορεί να πατήσει το κουμπί *Προσθήκη* (*Add*). Αν υπάρχει κάποιος λάθος, τότε εμφανίζεται ένα παράθυρο διαλόγου που ενημερώνει τον χρήστη για το λάθος. Αλλιώς τα στοιχεία του νέου ασθενή καταχωρούνται στη βάση δεδομένων και εμφανίζεται ένα παράθυρο διαλόγου που ενημερώνει το χρήστη ότι τα στοιχεία του νέου ασθενή, καταχωρηθήκαν με επιτυχία στη βάση δεδομένων.

Όταν γίνει η προσθήκη του νέου ασθενή στη βάση, το παράθυρο προσθήκης ασθενή κλείνει και ενεργοποιείται ξανά το κύριο παράθυρο, το οποίο όμως εμφανίζει τις πληροφορίες του νέου ασθενή. Έτσι ο χρήστης μπορεί αμέσως να ξεκινήσει να επεξεργάζεται τα δεδομένα που επιθυμεί, είτε αυτά αφορούν στην διόρθωση των στοιχείων του ασθενή που πρόσθεσε, είτε στην προσθήκη πληροφοριών πάνω στο γραφικό μοντέλο σχετικά με κάποιον τραυματισμό του ασθενή. Ειδικά αν, αν δεν επιθυμεί να αλλάξει κάτι, μπορεί να επιστρέψει σε λειτουργία παρουσίασης,

πατώντας το κουμπί αρχικής κατάστασης στη γραμμή μενού ή να αναζητήσει κάποιον άλλο ασθενή.

3.3.3 ΠΑΡΑΘΥΡΟ ΑΝΑΖΗΤΗΣΗΣ ΑΣΘΕΝΗ

Το παράθυρο αναζήτησης έχει φτιαχτεί με σκοπό την εύκολη εύρεση κάποιου ασθενή, που είναι καταχωρημένος στη βάση. Το παράθυρο εμφανίζεται όταν ο χρήστης επιλέξει την *Αναζήτηση Ασθενή (Find Patient)*, που βρίσκεται στο υπό-μενού του μενού *Αρχείο (File)*, στη γραμμή μενού του κύριου παραθύρου. Σε αυτό το παράθυρο, στην πρώτη γραμμή εμφανίζονται ο τίτλος του παραθύρου, '*Μηχανή αναζήτησης Ασθενών (Patients Search Engine)*' και το κουμπί τερματισμού του παραθύρου. Στη συνέχεια υπάρχει η γραμμή αναζήτησης, όπου ο χρήστης εισάγει τα στοιχεία του ασθενή που επιθυμεί να βρει. Στη γραμμή αυτή, μπορεί να εισάγει το όνομα, το επίθετο ή το όνομα και το επώνυμο μαζί, προκειμένου να βρει τον ασθενή που αναζητεί. Στα δεξιά της γραμμής αναζήτησης, υπάρχουν δύο κουμπιά. Το πρώτο είναι το *κουμπί αναζήτησης*, ενώ το δεύτερο διαγράφει τα δεδομένα της γραμμής αναζήτησης. Το *κουμπί αναζήτησης* αναπαρίσταται με ένα μεγεθυντικό φακό. Όταν η γραμμή αναζήτησης είναι κενή, τότε το κουμπί αυτό είναι απενεργοποιημένο. Μόλις ο χρήστης ξεκινήσει να εισάγει τα κριτήρια αναζήτησης που επιθυμεί, το κουμπί ενεργοποιείται. Όταν ο χρήστης πατήσει αυτό το κουμπί, ή πατήσει το πλήκτρο '*ENTER*', τότε το πρόγραμμα αναλαμβάνει να αναζητήσει τον ασθενή στον πίνακα *Patients* της βάσης δεδομένων και επιστρέφει τα αποτελέσματα, τα οποία εμφανίζονται υπό μορφή λίστας. Στη λίστα αποτελεσμάτων εμφανίζεται ο *κωδικός (ID)* του ασθενή, το ονοματεπώνυμο του, το όνομα πατρός και μητρός, η ημερομηνία γέννησης και το τηλέφωνο του.



Εικόνα 10 - Το παράθυρο αναζήτησης ασθενή

Τέλος, εμφανίζονται τα κουμπιά *Επιλογή (Select)* ασθενή και *Ακύρωση (Cancel)* αναζήτησης. Το κουμπί ακύρωσης αναζήτησης, τερματίζει το παράθυρο αναζήτησης ασθενή. Αν ο χρήστης επιλέξει έναν ασθενή και πατήσει το κουμπί *Επιλογή (Select)* τότε το παράθυρο αναζήτησης τερματίζεται και στο κύριο παράθυρο εμφανίζονται τα στοιχεία του ασθενή που επιλέχθηκε και το γραφικό μοντέλο με τα σημεία τραυματισμού, αν αυτά υπάρχουν. Επίσης τα σημεία στα οποία έχει τραυματιστεί στο παρελθόν ο ασθενής, εμφανίζονται και στο παράθυρο γρήγορης πρόσβασης, κάτω από τη μπάρα στοιχείων του ασθενή.

Ωστόσο αν ο χρήστης δεν επιλέξει κάποιον ασθενή από τη λίστα αποτελεσμάτων και πατήσει το κουμπί *Επιλογή (Select)*, τότε εμφανίζεται ένα παράθυρο διαλόγου, που τον ειδοποιεί να επιλέξει κάποιον ασθενή πρώτα. Επίσης για να επιλέξει κάποιον ασθενή ο χρήστης, μπορεί να κάνει διπλό κλικ πάνω στο ασθενή που επιθυμεί. Τότε αυτόματα, επιλέγεται ο ασθενής και τερματίζεται το παράθυρο αναζήτησης, προκειμένου να εμφανιστούν τα στοιχεία του ασθενή στο κύριο παράθυρο του προγράμματος.

Τέλος, το παράθυρο αναζήτησης ασθενή, προσφέρει τη δυνατότητα στο χρήστη να εμφανίσει στη λίστα αποτελεσμάτων όλους τους ασθενείς που υπάρχουν στη βάση δεδομένων. Αυτό επιτυγχάνεται δίνοντας τρεις αστερίσκους (***), ως κριτήριο αναζήτησης.

3.3.4 ΦΩΝΗΤΙΚΕΣ ΕΝΤΟΛΕΣ

Όπως έχει ήδη αναφερθεί, το πρόγραμμα παρέχει στο χρήστη, εκτός από το ποντίκι και το πληκτρολόγιο ως μέσα επικοινωνίας με το πρόγραμμα και την δυνατότητα αλληλεπίδρασης με φωνητικές εντολές. Όμως οι φωνητικές εντολές λειτουργούν μόνο όταν επιλεγμένη γλώσσα προβολής είναι τα αγγλικά. Αυτό συμβαίνει γιατί το πρόγραμμα χρησιμοποιεί τη μηχανή αναγνώρισης φωνής των Windows. Η Microsoft δεν έχει συμπεριλάβει την ελληνική γλώσσα στις γλώσσες που υποστηρίζει η μηχανή αναγνώρισης της και για αυτό το λόγο στο πρόγραμμα δεν υποστηρίζονται οι ελληνικές φωνητικές εντολές.

Στο λεξικό των φωνητικών εντολών του προγράμματος υπάρχουν οι παρακάτω λέξεις:

Exit Program ⇒ Τερματίζει το πρόγραμμα.

Return Home ⇒ Το πρόγραμμα επιστρέφει στην αρχική κατάσταση του.

New Patient ⇒ Εμφανίζει το παράθυρο προσθήκης νέου ασθενή.

Find Patient ⇒ Εμφανίζει το παράθυρο αναζήτησης ασθενή.

Hide Info Points ⇒ Αφαιρεί τα σημεία πληροφοριών από το πρόγραμμα, με σκοπό την καλύτερη προβολή του γραφικού μοντέλου.

Show Info Points ⇒ Εμφανίζει τα σημεία πληροφοριών.

Change Language To Greek ⇒ Αλλάζει την επιλεγμένη γλώσσα σε Ελληνικά.

Change Language To English ⇒ Αλλάζει την επιλεγμένη γλώσσα σε Αγγλικά.

Close Voice Commands ⇒ Απενεργοποιεί τη δυνατότητα αλληλεπίδρασης με το πρόγραμμα με τη χρήση φωνητικών εντολών.

Zoom ⇒ Εμφανίζεται μια πιο κοντινή εικόνα του σημείου που επιλέχθηκε να μεγεθυνθεί.

Unzoom ⇒ Εμφανίζεται μια πιο μακρινή εικόνα του σημείου που επιλέχθηκε να σμικρυνθεί.

Info Box ⇒ Εμφανίζεται το παράθυρο πληροφοριών του επιλεγμένου σημείου.

Edit Info Box ⇒ Ενεργοποιείται η επεξεργασία των δεδομένων του παραθύρου πληροφοριών.

Save Info Box ⇒ Αποθηκεύει τα δεδομένα του παραθύρου πληροφοριών.

Close Info Box ⇒ Τερματίζει το παράθυρο πληροφοριών.

Rotate Left ⇒ Περιστρέφει το μοντέλο αριστερά.

Rotate Right ⇒ Περιστρέφει το μοντέλο δεξιά.

Edit Personal Data ⇒ Ενεργοποιεί την επεξεργασία των προσωπικών δεδομένων του ασθενή και εμφανίζει το κουμπί επεξεργασίας στο παράθυρο πληροφοριών.

Save Personal Data ⇒ Αποθηκεύει τα προσωπικά δεδομένα του ασθενή και αποκρύπτει το κουμπί επεξεργασίας στο παράθυρο πληροφοριών.

Cancel Personal Data ⇒ Αναιρεί τις όποιες αλλαγές στα προσωπικά δεδομένα του ασθενή και αποκρύπτει το κουμπί επεξεργασίας στο παράθυρο πληροφοριών.

"Όνομα Συστήματος" + System ⇒ Εμφανίζει το σύστημα που επιλέγει ο χρήστης. πχ "Skeletal System".

Point + "Αριθμός Σημείου" ⇒ Εμφανίζει το μενού του σημείου που επιλέγει ο χρήστης. πχ "Point Two".

ΚΕΦΑΛΑΙΟ 4

ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΛΕΙΤΟΥΡΓΙΩΝ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΜΕ ΠΑΡΑΔΕΙΓΜΑΤΑ

Στο κεφάλαιο αυτό παρουσιάζονται οι λειτουργίες του προγράμματος μέσα από παραδείγματα. Αυτό γίνεται για την καλύτερη κατανόηση του τρόπου λειτουργίας του προγράμματος. Συγκεκριμένα θα επεξηγηθεί ο τρόπος προσθήκης νέου ασθενή στη βάση δεδομένων, πως γίνεται η αναζήτηση ενός ασθενή και πως μπορούμε στη συνέχεια να επεξεργαστούμε τα προσωπικά στοιχεία αυτού. Επίσης θα δούμε πως γίνεται η δημιουργία, η επεξεργασία και η διαγραφή σημείων τραυματισμού σε έναν ασθενή, αλλά και πως μπορούμε, να μετακινηθούμε εύκολα και γρήγορα σε κάποιο σημείο τραυματισμού με το παράθυρο γρήγορης πρόσβασης στα σημεία τραυματισμών.

4.1 ΠΡΟΣΘΗΚΗ ΝΕΟΥ ΑΣΘΕΝΗ

Το παράθυρο προσθήκης ασθενή επιτρέπει στον χρήστη να προσθέσει έναν καινούργιο ασθενή στη βάση δεδομένων. Το παράθυρο προσθήκης νέου ασθενή δεν είναι ορατό στο κύριο παράθυρο, αλλά πρέπει να το εμφανίσει ο χρήστης. Αυτό γίνεται μέσω της γραμμής μενού και της επιλογής *Αρχείο (File)*. Αν ο χρήστης μεταβεί στη γραμμή μενού και επιλέξει *Αρχείο (File)*, τότε εμφανίζεται το υπό-μενού στο οποίο περιέχεται η επιλογή *Προσθήκη Ασθενή (New Patient)*. Η επιλογή προσθήκης νέου ασθενή, παρέχεται και με φωνητικές εντολές. Ο χρήστης μπορεί να κάνει κλικ πάνω στην επιλογή αυτή, ή αρκεί να πει τη φράση «*New Patient*». Τότε εμφανίζεται το παράθυρο προσθήκης νέου ασθενή, όπως φαίνεται και στην παρακάτω εικόνα.

Εικόνα 11 - Το παράθυρο προσθήκης νέου ασθενή

Ας υποθέσουμε ότι θέλουμε να εισάγουμε έναν καινούργιο ασθενή. Βλέπουμε ότι το πρόγραμμα έχει αυτομάτως παράγει τον κωδικό του νέου ασθενή, ο οποίος είναι μοναδικός. Στη συνέχεια εισάγουμε το επίθετο του ασθενή, το όνομά του, το όνομα πατρός και μητρός και το τηλέφωνό του. Ακόμη υπάρχει το πεδίο Κατάσταση (*Status*). Η προεπιλεγμένη κατάσταση είναι Εν Ζωή (*Alive*), αλλά αν ο ασθενής εισήχθη για παράδειγμα στο νοσοκομείο σε κρίσιμη κατάσταση και απεβίωσε πριν την

καταχώρηση των στοιχείων του στη βάση, υπάρχει η αντίστοιχη επιλογή *Απεβίωσε (Deceased)*.

Επίσης ζητείται η ημερομηνία γέννησης του ασθενή. Για την αποφυγή λάθος διατυπώσεων το πρόγραμμα μας επιτρέπει να επιλέξουμε την ημερομηνία μέσα από έτοιμες λίστες. Η πρώτη λίστα περιέχει τις ημέρες, η δεύτερη τους μήνες και η τρίτη τα έτη. Μόλις επιλέξουμε και την ημερομηνία γέννησης, είμαστε έτοιμοι να εισάγουμε τα στοιχεία του καινούργιου ασθενή στη βάση δεδομένων. Για να το επιτύχουμε αυτό, το τελευταίο βήμα είναι να πατήσουμε το κουμπί *Προσθήκη (Add)*. Αν έχουμε ξεχάσει να συμπληρώσουμε κάποιο πεδίο, το πρόγραμμα μας ενημερώνει με σχετικό μήνυμα. Ειδάλλως, τα στοιχεία καταχωρούνται στη βάση δεδομένων, το παράθυρο προσθήκης νέου ασθενή τερματίζεται και στο κύριο παράθυρο εμφανίζεται το μήνυμα ότι ο νέος ασθενής προστέθηκε με επιτυχία. Παράλληλα επιλέγεται ο νέος ασθενής και τα στοιχεία του εμφανίζονται στη μπάρα στοιχείων ασθενή του κύριου παραθύρου.

Εάν ωστόσο έχουμε κάνει κάποιο ορθογραφικό λάθος στα στοιχεία του ασθενή μπορούμε να το διορθώσουμε πατώντας το κουμπί *Επεξεργασία (Edit)*, που υπάρχει στο τέλος της μπάρας. Στη συνέχεια διορθώνουμε τα λανθασμένα δεδομένα και πατάμε το κουμπί *Αποθήκευση (Save)*.

Εικόνα 12 - Το παράθυρο προσθήκης νέου ασθενή συμπληρωμένο

4.2 ΑΝΑΖΗΤΗΣΗ ΑΣΘΕΝΗ

Το παράθυρο αναζήτησης ασθενή επιτρέπει στον χρήστη να αναζητήσει κάποιον ασθενή στη βάση δεδομένων. Το παράθυρο αυτό, επίσης δεν είναι ορατό στο κύριο παράθυρο, αλλά πρέπει να το εμφανίσει ο χρήστης. Αυτό γίνεται μέσω της γραμμής μενού και της επιλογής **Αρχείο (File)**. Αν ο χρήστης μεταβεί στη γραμμή μενού και επιλέξει *Αρχείο (File)*, τότε εμφανίζεται το υπό-μενού στο οποίο περιέχεται η επιλογή *Αναζήτηση Ασθενή (Find Patient)*. Η επιλογή αναζήτησης ασθενή, παρέχεται και με φωνητικές εντολές. Ο χρήστης μπορεί να κάνει κλικ πάνω στην επιλογή αυτή, ή αρκεί να πει τη φράση «*Find Patient*». Τότε εμφανίζεται το παράθυρο εύρεσης ασθενή, όπως φαίνεται και στην παρακάτω εικόνα.



Εικόνα 13 - Το παράθυρο αναζήτησης ασθενή

Στη γραμμή αναζήτησης αναγράφεται η λέξη *Search...*, η οποία εξαφανίζεται αυτομάτως μόλις ο χρήστης κάνει κλικ στη γραμμή. Έτσι η γραμμή είναι έτοιμη να δεχθεί τα κριτήρια αναζήτησης. Μόλις ο χρήστης ξεκινήσει να εισάγει τα δεδομένα που επιθυμεί ενεργοποιείται το κουμπί αναζήτησης, το οποίο απεικονίζεται με ένα μεγεθυντικό φακό. Η αναζήτηση ενός ασθενή μπορεί να γίνει με πέντε τρόπους. Μπορούμε να

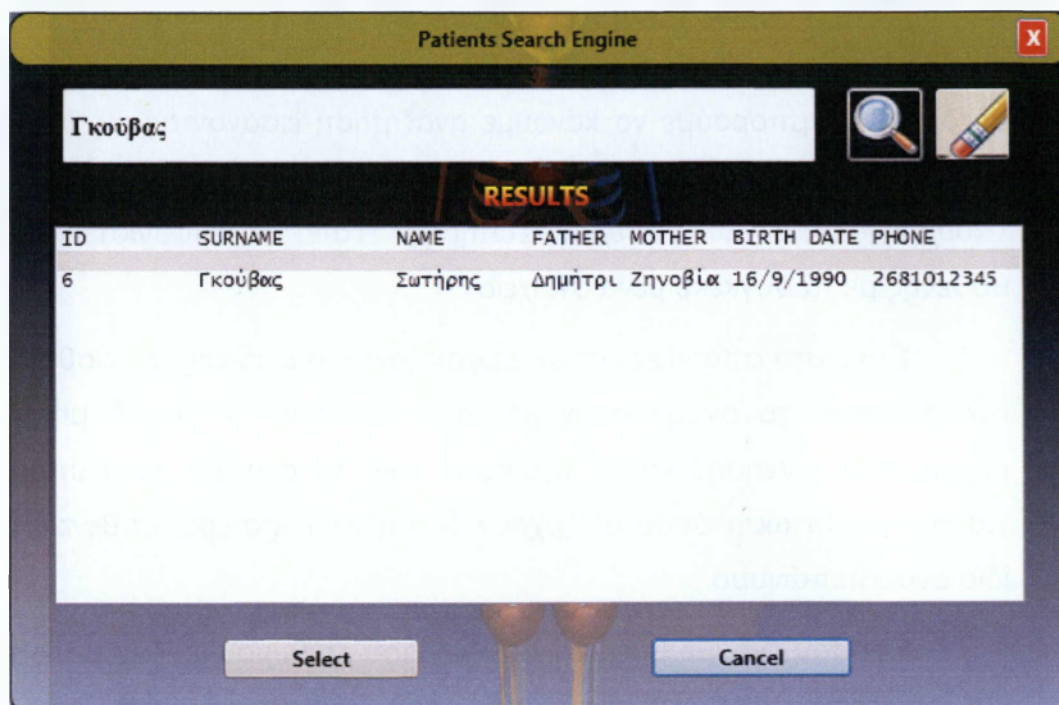
κάνουμε αναζήτηση με βάση το όνομα ή το επίθετο, ή μπορούμε να εισάγουμε το όνομα και το επίθετο μαζί ή και το αντίστροφο, δηλαδή πρώτα το επίθετο και μετά το όνομα. Αν θέλουμε να αναζητήσουμε, για παράδειγμα, τον ασθενή 'Σωτήρη Γκούβα' τότε μπορούμε να εισάγουμε το όνομα 'Σωτήρης' στη γραμμή αναζήτησης. Ωστόσο, μόλις πατήσουμε το κουμπί αναζήτησης ή το πλήκτρο *Enter*, στη λίστα αποτελεσμάτων θα εμφανιστούν όλοι οι ασθενείς, εφόσον υπάρχουν, με το όνομα 'Σωτήρης' και όχι μόνο αυτός που αναζητούμε. Επίσης μπορούμε να αναζητήσουμε τον ασθενή μας με βάση το επίθετό του. Αν γράψουμε στη γραμμή αναζήτησης 'Γκούβας', τότε θα εμφανιστούν οι ασθενείς με το επίθετο αυτό. Ακόμα μπορούμε να κάνουμε αναζήτηση εισάγοντας ως δεδομένα το όνομα και το επίθετο μαζί. Δηλαδή μπορούμε να εισάγουμε 'Σωτήρης Γκούβας', αλλά και 'Γκούβας Σωτήρης'. Τότε θα εμφανιστεί μόνο ο ασθενής με τα συγκεκριμένα στοιχεία.

Στη λίστα αποτελεσμάτων, εμφανίζονται ο κωδικός του ασθενή που αναζητήσαμε, το ονοματεπώνυμό του, το όνομα πατρός & μητρός, η ημερομηνία γέννησης και το τηλέφωνό του. Τα στοιχεία αυτά υπάρχουν για την περίπτωση όπου υπάρχουν δύο ή περισσότεροι ασθενείς με το ίδιο ονοματεπώνυμο.

Εφόσον έχουμε βρει τον ασθενή που επιθυμούμε, δεν μένει παρά να επιλέξουμε τη εγγραφή που του αντιστοιχεί από τη λίστα αποτελεσμάτων και να πατήσουμε το κουμπί *Επιλογή (Select)*. Μπορούμε ακόμη, για ευκολία να κάνουμε διπλό κλικ πάνω στα στοιχεία του. Αν ωστόσο πατήσουμε το κουμπί *Επιλογή (Select)*, χωρίς να έχουμε πρώτα επιλέξει τον ασθενή μας, τότε εμφανίζεται ένα παράθυρο διαλόγου που μας προτρέπει να επιλέξουμε έναν.

Τέλος, εφόσον έχουμε κάνει τα πάντα σωστά, μόλις πατήσουμε το κουμπί *Επιλογή (Select)* το παράθυρο αναζήτησης τερματίζεται και ενεργοποιείται το κύριο παράθυρο του προγράμματος. Στη μπάρα στοιχείων ασθενή εμφανίζονται τα προσωπικά στοιχεία του ασθενή που αναζητήσαμε, ενώ στο γραφικό μοντέλο εμφανίζονται τα σημεία

πληροφοριών, κίτρινα αντί για κόκκινα, όπου υπάρχει κάποιος τραυματισμός. Ακόμη τα σημεία στα οποία έχει υπάρξει κάποιος τραυματισμός, εμφανίζονται και στο παράθυρο γρήγορης πρόσβασης σημείων τραυματισμών του ασθενή. Ο χρήστης μπορεί τώρα να προβάλει το ιατρικό ιστορικό του ασθενή, να περιηγηθεί στα σημεία αυτά και να επεξεργαστεί τα σημεία και τα στοιχεία του ασθενή ή να προσθέσει κάποιον νέο τραυματισμό.

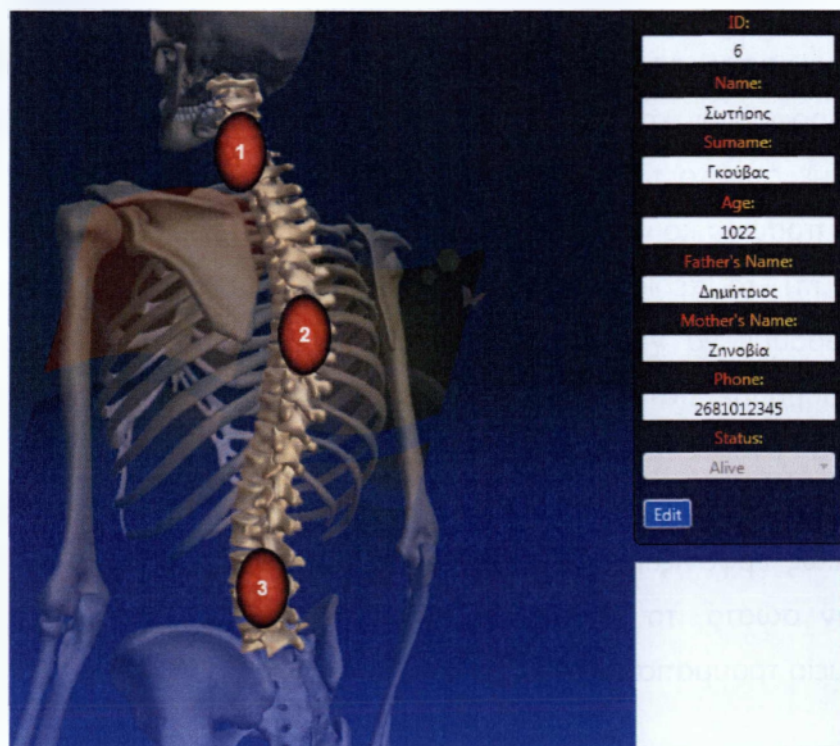


Εικόνα 14 - Υπόδειγμα εμφάνισης αποτελεσμάτων στο παράθυρο αναζήτησης ασθενή

4.3 ΠΡΟΣΘΗΚΗ / ΕΠΕΞΕΡΓΑΣΙΑ / ΔΙΑΓΡΑΦΗ ΤΩΝ ΣΗΜΕΙΩΝ ΤΡΑΥΜΑΤΙΣΜΩΝ ΤΟΥ ΑΣΘΕΝΗ

Όταν επιλέξουμε έναν ασθενή, τότε εμφανίζονται τα σημεία πληροφοριών στα οποία έχει υποστεί κάποιον τραυματισμό με κίτρινο χρώμα αντί για κόκκινο. Αυτά τα σημεία ονομάζονται σημεία τραυματισμών του ασθενή. Σε αυτή την ενότητα, θα δούμε πως μπορούμε να δημιουργήσουμε ένα τέτοιο σημείο, να το επεξεργαστούμε και να το διαγράψουμε.

Στην παρακάτω εικόνα φαίνεται η σπονδυλική στήλη του ασθενή 'Σωτήρη Γκούβα', στην οποία φαίνονται τα σημεία πληροφοριών. Τα σημεία πληροφοριών αναπαρίστανται με ελλείψεις, διατεταγμένες κατά μήκος της σπονδυλικής στήλης. Όπως βλέπουμε οι ελλείψεις είναι κόκκινες που υποδηλώνει ότι ο ασθενής δεν έχει υποστεί κάποιον τραυματισμό.

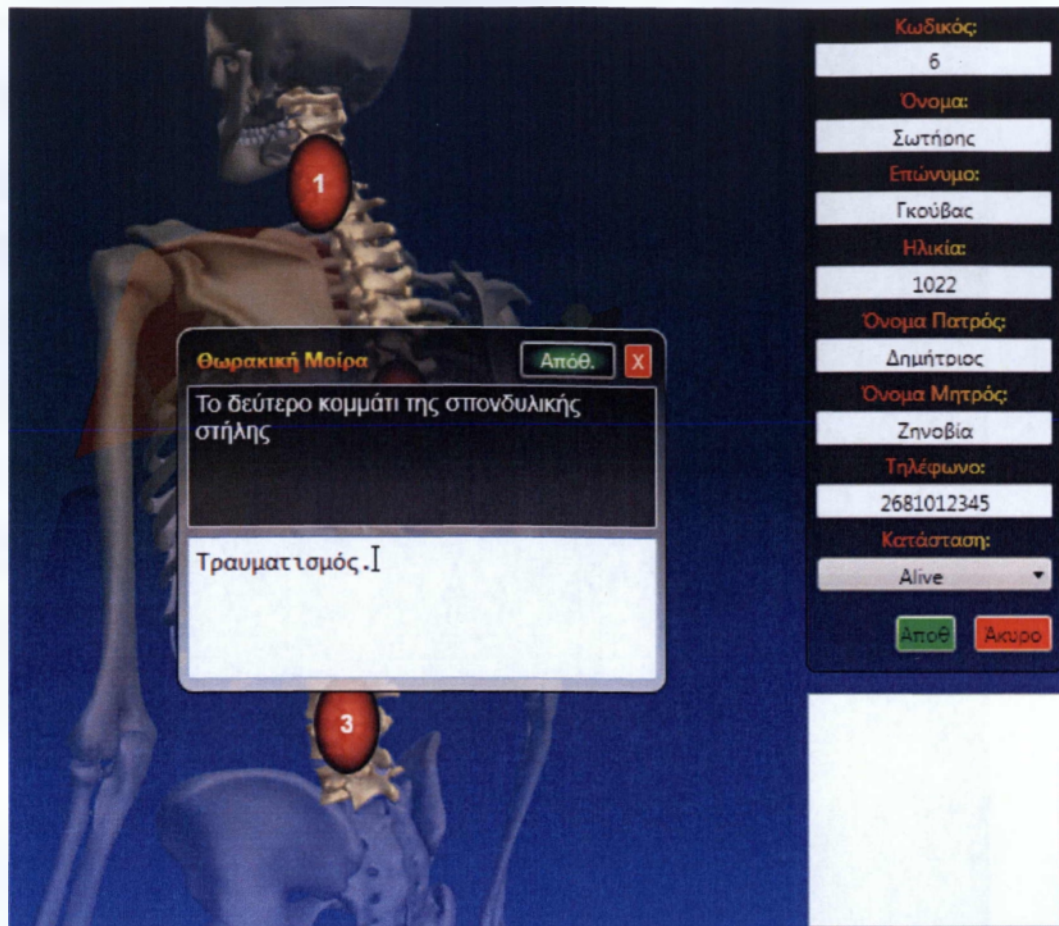


Εικόνα 15 - Στην εικόνα εμφανίζεται το μοντέλο της σπονδυλικής στήλης, όπου υπάρχουν κόκκινες ελλείψεις, που υποδηλώνουν ότι ο επιλεγμένος ασθενής δεν έχει τραυματιστεί.

Στη συνέχεια, θα δούμε πως μπορούμε να δημιουργήσουμε ένα σημείου τραυματισμού στον επιλεγμένο ασθενή. Αρχικά μετακινούμε τον κέρσorra του ποντικιού στο σημείο πληροφοριών, το οποίο επιθυμούμε να μετατρέψουμε σε σημείο τραυματισμού. Τότε εμφανίζεται το μενού πληροφοριών. Πατώντας το *κουμπί πληροφοριών* που συμβολίζεται με το σύμβολο 'ⓘ' εμφανίζεται το *παράθυρο πληροφοριών ασθενή*. Το παράθυρο αυτό περιέχει δύο πεδία. Στο πρώτο εμφανίζονται οι γενικές πληροφορίες για το σημείο που επιλέχτηκε, ενώ το δεύτερο εμφανίζει τις πληροφορίες τραυματισμού.

Όταν το πεδίο με τις πληροφορίες τραυματισμού είναι κενό, σημαίνει ότι ο ασθενής, στο σημείο αυτό δεν έχει υποστεί κάποιον τραυματισμό και έτσι το σημείο πληροφοριών εμφανίζεται με κόκκινο χρώμα. Αν ωστόσο πληκτρολογήσουμε κάτι σε αυτό το πεδίο, τότε το σημείο πληροφοριών θα αντικατασταθεί με μια κίτρινη έλλειψη. Αν δοκιμάσουμε, τώρα να πληκτρολογήσουμε κάτι, θα δούμε ότι το πρόγραμμα δεν μας το επιτρέπει.

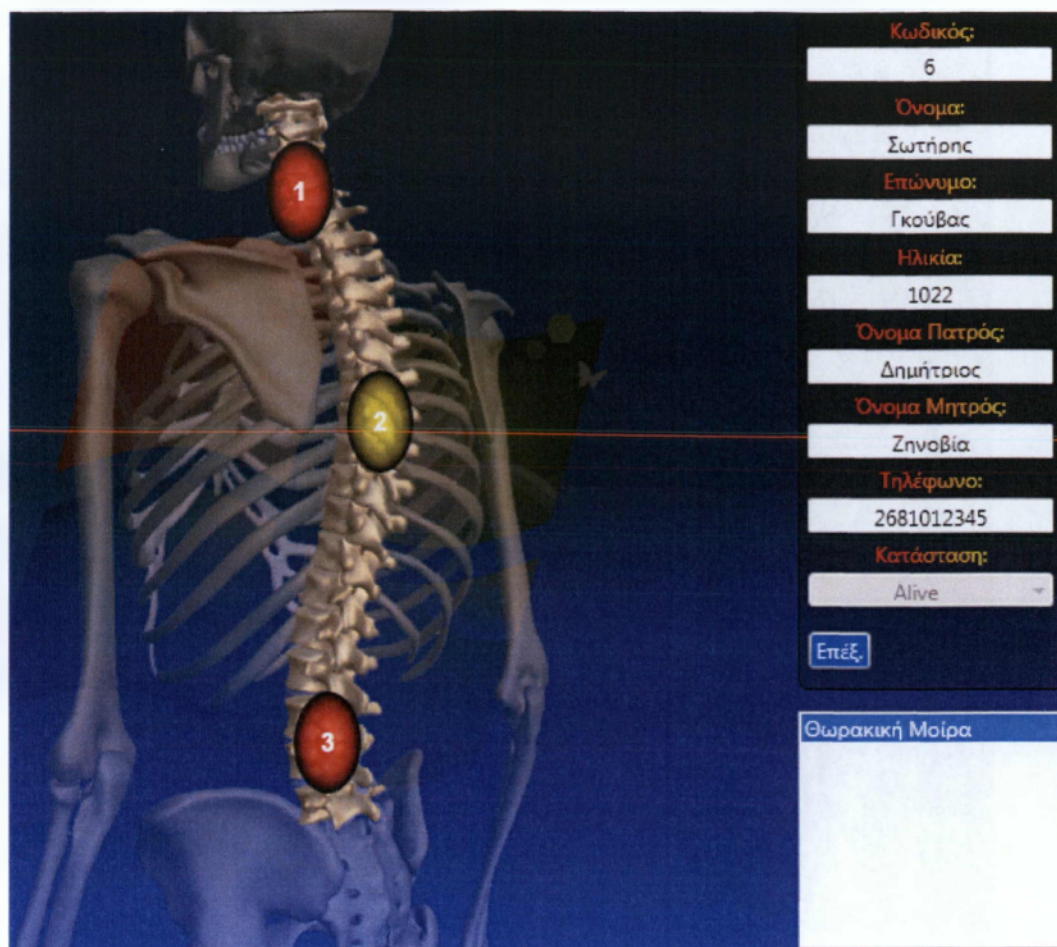
Για να μπορέσουμε να εισάγουμε τις πληροφορίες για τον τραυματισμό του ασθενή πρέπει να πάμε στην μπάρα προσωπικών πληροφοριών και να πατήσουμε το κουμπί *Επεξεργασία (Edit)*. Τότε θα δούμε ότι στο παράθυρο πληροφοριών εμφανίστηκε στη γραμμή τίτλου ένα πράσινο κουμπί με το όνομα *Επεξεργασία (Edit)*. Πατώντας σε αυτό το κουμπί, το πεδίο πληροφοριών τραυματισμού ξεκλειδώνεται και πλέον μπορούμε να γράψουμε σε αυτό. Μόλις εισάγουμε τα δεδομένα που επιθυμούμε, πατάμε το πράσινο κουμπί στο παράθυρο πληροφοριών που τώρα ονομάζεται *Αποθήκευση (Save)*. Αν τα δεδομένα εισήχθησαν επιτυχώς στη βάση δεδομένων, τότε εμφανίζεται το αντίστοιχο μήνυμα, αλλιώς εμφανίζεται ένα μήνυμα λάθους. Ταυτοχρόνως, αν τα δεδομένα ήταν σωστά, το κόκκινο σημείο πληροφοριών μετατρέπεται σε κίτρινο σημείο τραυματισμού.



Εικόνα 16 - Το παράθυρο πληροφοριών πριν αποθηκευτούν οι νέες πληροφορίες και το σημείο πληροφοριών μετατραπεί σε σημείο τραυματισμού.

Αν θέλαμε να επεξεργαστούμε ένα υπάρχον σημείο τραυματισμού, θα ακολουθούσαμε την ίδια διαδικασία, δηλαδή θα πατούσαμε στο κουμπί *Επεξεργασία (Edit)*, του παραθύρου πληροφοριών για να ξεκλειδώσει το πεδίο πληροφοριών και θα επεξεργαζόμασταν το περιεχόμενό του. Με τον ίδιο τρόπο αν διαγράφαμε τα δεδομένα τραυματισμού και πατούσαμε το κουμπί *Αποθήκευση (Save)*, θα διαγραφόταν το σημείο τραυματισμού και στη θέση του θα εμφανιζόταν ένα κόκκινο σημείο πληροφοριών.

Τέλος, αφού επεξεργαστούμε τα σημεία που θέλουμε πρέπει να πατήσουμε το κουμπί *Αποθήκευση (Save)* στη μπάρα στοιχείων ασθενή, για να αποκρύψουμε το κουμπί επεξεργασίας στο παράθυρο πληροφοριών και να αποτρέψουμε τις ανεπιθύμητες αλλαγές στο ιατρικό ιστορικό του ασθενή.

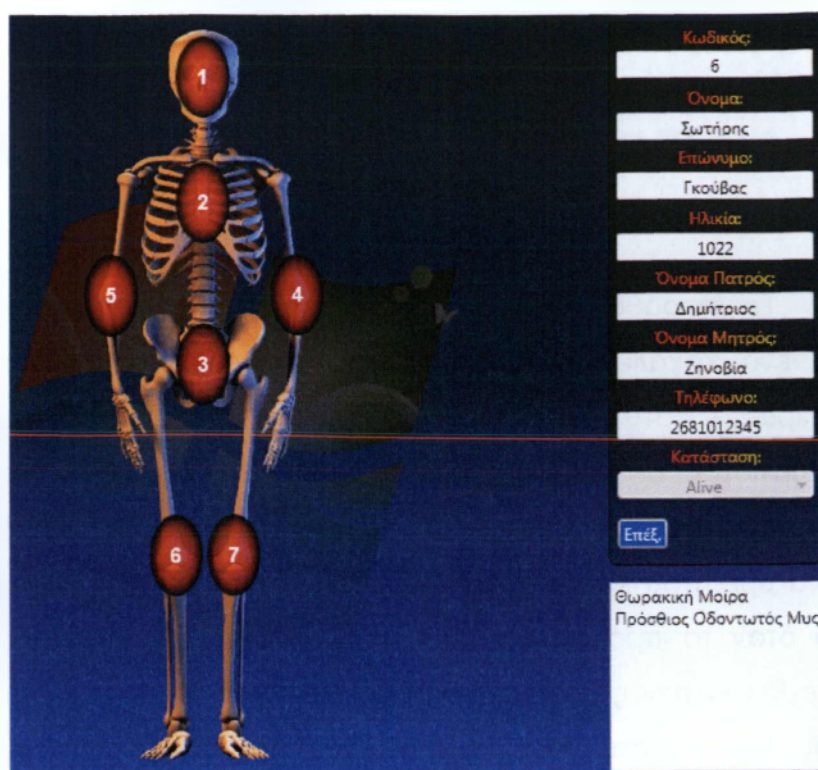


Εικόνα 17 - Στην εικόνα εμφανίζεται το γραφικό μοντέλο με τα σημεία πληροφοριών και το σημείο τραυματισμού που δημιουργήσαμε.

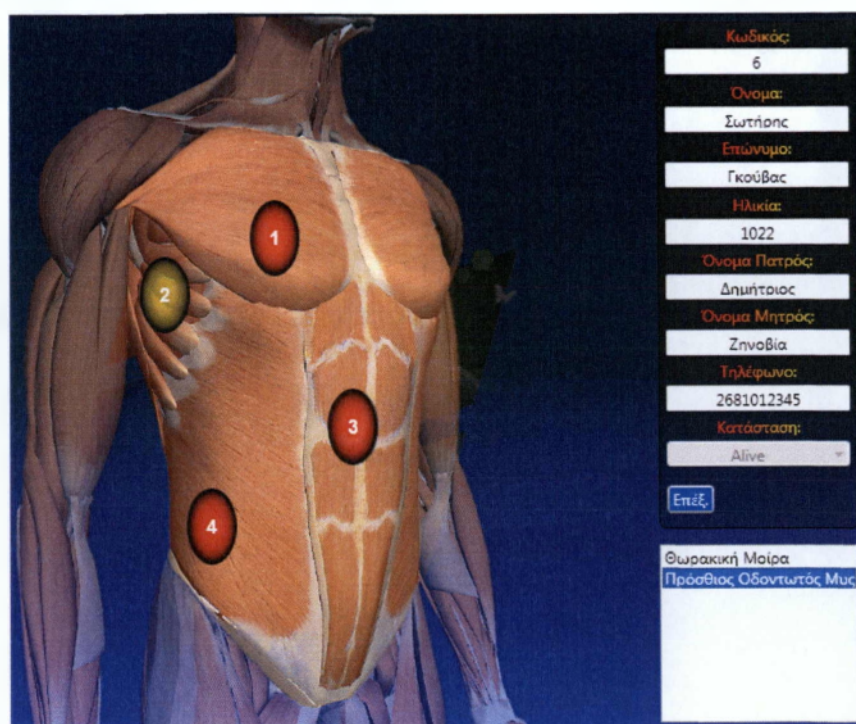
4.4 ΜΕΤΑΚΙΝΗΣΗ ΣΤΟΥΣ ΤΡΑΥΜΑΤΙΣΜΟΥΣ ΤΟΥ ΑΣΘΕΝΗ ΜΕ ΧΡΗΣΗ ΤΟΥ ΠΑΡΑΘΥΡΟΥ ΓΡΗΓΟΡΗΣ ΠΡΟΣΒΑΣΗΣ ΣΤΑ ΣΗΜΕΙΑ ΤΡΑΥΜΑΤΙΣΜΩΝ.

Το παράθυρο γρήγορης πρόσβασης στα σημεία τραυματισμών, είναι ένα εργαλείο που επιτρέπει στο χρήστη απλά και εύκολα να ενημερωθεί, καταρχάς για όλα τα σημεία στα οποία έχει υποστεί κάποιον τραυματισμό ο ασθενής αλλά επίσης και να δει όλα τα σημεία τραυματισμών χωρίς να πρέπει να ψάξει σε όλα τα επίπεδα-συστήματα για να βρει που και αν υπάρχουν τραυματισμοί. Το παράθυρο αυτό είναι κενό όταν το πρόγραμμα είναι σε λειτουργία παρουσίασης, αλλά μόλις επιλεγθεί κάποιος ασθενής με ιατρικό ιστορικό, τότε όλα τα σημεία στα οποία έχει υποστεί τραυματισμό, εμφανίζονται στο παράθυρο γρήγορης πρόσβασης.

Αν για παράδειγμα έχουμε επιλέξει έναν ασθενή τότε στο παράθυρο γρήγορης πρόσβασης εμφανίζονται όλα τα σημεία τραυματισμών, όπως φαίνεται και στην παρακάτω εικόνα. Αν λοιπόν θέλουμε να μετακινηθούμε σε κάποιο από τα σημεία, τότε απλά το επιλέγουμε με το ποντίκι και αυτομάτως μεταφερόμαστε στο επίπεδο-σύστημα όπου υπάρχει το συγκεκριμένο σημείο τραυματισμού. Έτσι δεν χρειάζεται να ψάξουμε σε κάθε επίπεδο-σύστημα για να βρούμε που είναι το σημείο τραυματισμού.



Εικόνα 18 - Στην εικόνα εμφανίζεται το παράθυρο γρήγορης πρόσβασης και περιέχει τα σημεία τραυματισμού του επιλεγμένου ασθενή.



Εικόνα 19 - Αν επιλέξουμε από το παράθυρο γρήγορης πρόσβασης να δούμε το σημείο τραυματισμού για τον πρόσθιο οδοντωτό μυ, αυτόματα εμφανίζεται το αντίστοιχο γραφικό μοντέλο στο οποίο περιέχεται το συγκεκριμένο σημείο τραυματισμού.

ΕΠΙΛΟΓΟΣ

Το πρόγραμμα παρουσίασης του ανθρώπινου σώματος που αναπτύχθηκε για τις ανάγκες της πρακτικής άσκησης δεν αποτελεί μια ολοκληρωμένη εφαρμογή που θα μπορούσε να χρησιμοποιηθεί για να ικανοποιήσει τις ανάγκες του αγοραστικού κοινού, αλλά χρήζει περαιτέρω ανάπτυξης και βελτίωσης. Στόχος της πτυχιακής εργασίας ήταν η ανάπτυξη ενός δυναμικού προγράμματος, το οποίο θα λειτουργεί χρησιμοποιώντας πληροφορίες από μια βάση δεδομένων. Στόχος ήταν επίσης η ενασχόληση με νέες τεχνολογίες, οι οποίες προσφέρουν στον προγραμματιστή τη δυνατότητα εύκολης και γρήγορης κατασκευής προγραμμάτων.

Το αποτέλεσμα της εργασίας ωστόσο αποδείχθηκε ότι μπορεί να έχει και άλλες χρήσεις εκτός από την παρουσίαση του ανθρώπινου σώματος. Καθώς το περιεχόμενο είναι δυναμικό, θα μπορούσε εύκολα, με πολύ λίγες αλλαγές στον κώδικά του και αντικατάσταση των δεδομένων της βάσης, να μετατραπεί σε πρόγραμμα παρουσίασης ενός άλλου αντικειμένου. Θα μπορούσε για παράδειγμα να μετατραπεί σε ένα πρόγραμμα για την παρουσίαση ηλεκτρονικών υπολογιστών. Για να γίνει αυτό αρκεί να δημιουργηθεί μια βάση δεδομένων με πληροφορίες για Η/Υ και να γίνουν κάποιες μετατροπές στον πηγαίο κώδικα του προγράμματος. Επίσης θα μπορούσε να χρησιμοποιηθεί από καταστήματα με Service Η/Υ, όπου θα κατέγραφαν τα προβλήματα των υπολογιστών των πελατών τους.

Η εφαρμογή περιλαμβάνει πολλά χαρακτηριστικά και παρέχει στον χρήστη αρκετές δυνατότητες. Ωστόσο, όπως κάθε εφαρμογή, επιδέχεται βελτιώσεων. Παρότι είναι δυναμική, στην παρούσα έκδοση οι τίτλοι και τα μενού, καθώς και οι φωνητικές εντολές για τα μενού είναι ενσωματωμένα στον κώδικα της εφαρμογής. Σε μελλοντικές εκδόσεις, τα χαρακτηριστικά αυτά θα μπορούσαν να υπάρχουν αποθηκευμένα για κάθε γλώσσα, στη βάση δεδομένων. Αυτό παρέχει την ευκολία προσθήκης και άλλων

γλωσσών στο πρόγραμμα αλλά και τη δυνατότητα μετατροπής του προγράμματος σε άλλο αντικείμενο (από παρουσίαση του ανθρώπινου σώματος σε παρουσίαση Η/Υ), παραμετροποιώντας μόνο τα δεδομένα της βάσης και όχι του πηγαίου κώδικα.

Επίσης το πρόγραμμα θα μπορούσε να ενσωματώσει μια λειτουργία επεξεργασίας, όπου ο χρήστης θα μπορεί να προσθέσει, να αφαιρέσει ή να επεξεργαστεί τα επίπεδα, τα σημεία και τις εικόνες μέσα από την εφαρμογή. Αυτό φυσικά προϋποθέτει την ύπαρξη μηχανισμών ασφαλείας, όπως για παράδειγμα ένα όνομα χρήστη και έναν κωδικό, ώστε να μην μπορεί ο καθένας να επεξεργαστεί τις πληροφορίες του προγράμματος. Ένα ακόμη χαρακτηριστικό που θα μπορούσε να προστεθεί σε μελλοντικές εκδόσεις είναι η υποστήριξη τρισδιάστατων μοντέλων, αντί των εικόνων. Αυτό θα προσφέρει στο χρήστη πιο ευχάριστη εμπειρία αλληλεπίδρασης με το γραφικό μοντέλο, καθώς θα μπορεί να το προβάλει από όποια οπτική γωνία επιθυμεί και να πάρει πληροφορίες για ακόμα περισσότερα όργανα.

Η ενασχόληση μου με αυτό το πρόγραμμα και χάρη στη βοήθεια του κύριου Τριτάκη, μου πρόσφερε πολύτιμη εμπειρία και γνώση σε τεχνολογίες αιχμής.

ΠΑΡΑΡΤΗΜΑ Α

Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Σε αυτό το παράρτημα παρουσιάζεται αναλυτικά η δομή της βάσης δεδομένων.

Τα δεδομένα της βάσης χρησιμοποιούνται:

- Από το πρόγραμμα για την δημιουργία των γραφικών και των στοιχείων ελέγχου (buttons κτλ).
- Από το χρήστη για την εμφάνιση ή επεξεργασία των πληροφοριών για τους υπάρχοντες ασθενείς ή την προσθήκη νέων ασθενών.

1 ΟΙ ΠΙΝΑΚΕΣ ΤΗΣ MYSQL

Η βάση δεδομένων περιέχει τους ακόλουθους πίνακες:

- Levels, όπου είναι αποθηκευμένα τα επίπεδα του ανθρώπινου σώματος (μυϊκό, σκελετικό κτλ).
- Patients, όπου υπάρχουν τα προσωπικά στοιχεία των ασθενών (όνομα, τηλέφωνο κτλ).
- Medical Records, όπου βρίσκεται το ιατρικό ιστορικό των ασθενών
- Points, όπου υπάρχουν καταχωρημένα τα σημεία που εμφανίζονται πάνω σε κάθε επίπεδο (πχ για το σκελετικό σύστημα τα σημεία είναι: κρανίο, λεκάνη, κόκαλα των χεριών κτλ).
- Photos, όπου υπάρχει καταχωρημένη η διαδρομή που είναι αποθηκευμένες η εικόνες, οι οποίες χρησιμοποιούνται από την εφαρμογή.

Παρακάτω γίνεται παρουσίαση των πινάκων αυτών και εξηγούνται τα πεδία που χρησιμοποιήθηκαν.

1.1 Πίνακας Levels

Ο πίνακας Levels περιέχει όλα τα δεδομένα που αφορούν τα διάφορα συστήματα (επίπεδα) του ανθρώπινου σώματος όπως είναι το σκελετικό, το μυϊκό, το αναπνευστικό και το πεπτικό. Τα δεδομένα που περιέχει ο πίνακας αυτός είναι:

ID : Στο πεδίο αυτό αποθηκεύεται ο κωδικός του κάθε επιπέδου. Ο αριθμός αυτός είναι μοναδικός για το κάθε επίπεδο και για αυτό χρησιμοποιείται σαν πρωτεύων κλειδί του πίνακα.

PhotoID : Το πεδίο PhotoID περιέχει έναν κωδικό που αντιστοιχεί στο αντίστοιχο ID του πίνακα Photos όπου κρατούνται πληροφορίες για τις

θέσεις στις οποίες βρίσκονται αποθηκευμένες οι φωτογραφίες των επιπέδων.

EnglishName : Το πεδίο αυτό περιέχει την αγγλική ονομασία του επιπέδου.

GreekName : Το πεδίο αυτό περιέχει την ελληνική ονομασία του επιπέδου.

ChildrenOfLevel : Το πρόγραμμα δίνει τη δυνατότητα στο χρήστη να κάνει zoom in και zoom out στα επίπεδα του ανθρώπινου σώματος. Το πεδίο **ChildrenOfLevel** : Χρησιμοποιείται όταν ο χρήστης επιλέγει να κάνει zoom out. Τότε το πρόγραμμα ελέγχει το πεδίο αυτό και ανάλογα με τον αριθμό που γράφει εμφανίζεται το επίπεδο με το αντίστοιχο ID.

Υπάρχουν ωστόσο και κάποια πεδία που έχουν τον αριθμό '0'. Οι εγγραφές που στο πεδίο 'ChildrenOfLevel' έχουν την τιμή '0' λαμβάνονται από το πρόγραμμα ως κύρια επίπεδα του ανθρώπινου σώματος. Κατά την εκκίνηση του προγράμματος γίνεται έλεγχος στη βάση και για τα πεδία 'ChildrenOfLevel', όπου βρεθεί η τιμή '0', το πρόγραμμα φροντίζει ώστε να εμφανιστεί το αντίστοιχο όνομα επιπέδου υπό τη μορφή radio button στην αριστερή πλευρά του παραθύρου. Επίσης όταν το πεδίο 'ChildrenOfLevel' έχει τιμή '0' αυτόματα απενεργοποιείται η επιλογή 'zoom out'.

1.2 Πίνακας Points

Ο πίνακας αυτός περιέχει πληροφορίες σχετικά με τα σημεία πληροφοριών που εμφανίζονται πάνω σε κάθε επίπεδο του ανθρώπινου σώματος. Ειδικότερα περιέχει τα εξής:

ID : Το πεδίο αυτό περιέχει τον κωδικό του σημείου πληροφοριών ο οποίος χρησιμοποιείται σαν πρωτεύων κλειδί, καθώς είναι μοναδικός.

EnglishName : Το πεδίο αυτό περιέχει την αγγλική ονομασία του σημείου πληροφοριών.

GreekName : Το πεδίο αυτό περιέχει την ελληνική ονομασία του σημείου πληροφοριών.

LevelID : Το πεδίο αυτό περιέχει τον κωδικό του επιπέδου, πάνω στο οποίο θα εμφανίζεται το σημείο πληροφοριών.

LeftMargin : Η τιμή που περιέχεται σε αυτό το σημείο υποδηλώνει την απόσταση που θα έχει το σημείο πληροφοριών από την αριστερή πλευρά του παραθύρου.

TopMargin : Η τιμή που περιέχεται σε αυτό το σημείο υποδηλώνει την απόσταση που θα έχει το σημείο πληροφοριών από την πάνω πλευρά του παραθύρου.

GenerallInfoEN : Το σημείο αυτό περιέχει τις γενικές πληροφορίες για το συγκεκριμένο σημείο πληροφοριών, γραμμένες στα αγγλικά. (πχ γενικές πληροφορίες για το ρινικό οστό).

GenerallInfoGR : Το σημείο αυτό περιέχει τις γενικές πληροφορίες για το συγκεκριμένο σημείο πληροφοριών, γραμμένες στα ελληνικά. (πχ γενικές πληροφορίες για τον δикέφαλο μυ).

ZoomInToLevelID : Το πεδίο αυτό περιέχει τον κωδικό του επιπέδου του ανθρώπινου σώματος που θα εμφανιστεί στο πρόγραμμα, όταν ο χρήστης επιλέξει να κάνει 'Zoom In' στο σημείο πληροφοριών.

RotatePlusToLevelID : Το πεδίο αυτό περιέχει τον κωδικό του επιπέδου του ανθρώπινου σώματος που θα εμφανιστεί στο πρόγραμμα, όταν ο χρήστης επιλέξει να κάνει δεξιά περιστροφή του μοντέλου.

RotateMinusToLevelID : Το πεδίο αυτό περιέχει τον κωδικό του επιπέδου του ανθρώπινου σώματος που θα εμφανιστεί στο πρόγραμμα, όταν ο χρήστης επιλέξει να κάνει αριστερή περιστροφή του μοντέλου.

1.3 Πίνακας Patients

Στον πίνακα Patients είναι καταχωρημένα τα προσωπικά στοιχεία των ασθενών. Τα πεδία που περιέχει είναι:

ID : Στο πεδίο αυτό αποθηκεύεται ο κωδικός του πελάτη ο οποίος είναι μοναδικός και για αυτό χρησιμοποιείται σαν πρωτεύων κλειδί. Ο κωδικός είναι ένας αύξων αριθμός ο οποίος δημιουργείται από το πρόγραμμα κάθε φορά που προστίθεται ένας νέος ασθενής.

Surname : Στο πεδίο αυτό καταχωρείται το επίθετο του ασθενή.

Name : Στο πεδίο αυτό καταχωρείται το όνομα του ασθενή.

Father_Name : Στο πεδίο αυτό καταχωρείται το όνομα πατρός του ασθενή.

Mother_Name : Στο πεδίο αυτό καταχωρείται το όνομα μητρός του ασθενή.

BirthDate : Στο πεδίο αυτό καταχωρείται η ημερομηνία γέννησης του ασθενή.

Phone : Στο πεδίο αυτό καταχωρείται το τηλέφωνο του ασθενή.

Status : Στο πεδίο αυτό καταχωρείται η κατάσταση του ασθενή, δηλαδή αν είναι εν ζωή ή όχι.

1.4 Πίνακας Medical Records

Ο πίνακας αυτός περιέχει το ιατρικό ιστορικό των ασθενών. Περιέχει τα πεδία:

ID : Το πεδίο αυτό περιέχει τον κωδικό. Ο κωδικός είναι μοναδικός και χρησιμοποιείται ως πρωτεύων κλειδί.

PatientID : Στο πεδίο αυτό καταχωρείται ο κωδικός του ασθενή.

PointID : Στο πεδίο αυτό καταχωρείται ο κωδικός του σημείου.

Info : Περιέχει τις πληροφορίες σχετικά με τον τραυματισμό ή την ασθένεια του ασθενή.

1.5 Πίνακας Photos

Στον πίνακα Photos αποθηκεύονται οι θέσεις που είναι αποθηκευμένες οι εικόνες που χρησιμοποιούνται στο πρόγραμμα. Ο πίνακας περιέχει τα πεδία:

ID : Το πεδίο αυτό περιέχει τον κωδικό. Ο κωδικός είναι μοναδικός και χρησιμοποιείται ως πρωτεύων κλειδί.

Name : Το πεδίο αυτό περιέχει ένα ενδεικτικό όνομα για την εικόνα.

Path : Στο πεδίο αυτό βρίσκεται η διαδρομή που είναι αποθηκευμένη η εικόνα.

2 ΣΧΕΣΕΙΣ ΜΕΤΑΞΥ ΤΩΝ ΠΙΝΑΚΩΝ

Στην παρακάτω εικόνα απεικονίζονται οι πίνακες και τα πεδία που αποτελούν τη βάση δεδομένων αλλά και οι σχέσεις μεταξύ τους. Σε αυτό εμφανίζονται οι σχέσεις μεταξύ των πινάκων. Για τη βάση δεδομένων χρησιμοποιήθηκαν οι σχέσεις ένα προς ένα, ένα προς πολλά και πολλά προς ένα.



Τα ζευγάρια τα οποία συσχετίζονται είναι:

Photo – Level : Θεωρούμε από τη μεριά του πίνακα Photo, ότι μία φωτογραφία μπορεί να αντιστοιχεί σε ένα επίπεδο του ανθρώπινου

σώματος και από τη μεριά του πίνακα Level ότι κάθε επίπεδο μπορεί να έχει μόνο μία φωτογραφία. Έτσι προκύπτει ότι η σχέση μεταξύ αυτών των πινάκων είναι ένα προς ένα.

Level – Point : Η σχέση μεταξύ αυτών των δύο πινάκων είναι ένα προς πολλά γιατί σε ένα επίπεδο του ανθρώπινου σώματος μπορεί να εμφανίζονται πολλά σημεία πληροφοριών. Ωστόσο ένα σημείο πληροφοριών δεν μπορεί να εμφανίζεται σε πάνω από ένα επίπεδα.

Point – MedicalRecord : Κάθε εγγραφή στον πίνακα Medical Record αντιστοιχεί σε ένα σημείο του πίνακα Point. Ωστόσο περισσότερες από μια εγγραφές μπορεί να αντιστοιχούν στο ίδιο σημείο υπό την προϋπόθεση ότι πρόκειται για διαφορετικό ασθενή. Έτσι η σχέση που προκύπτει είναι ένα προς πολλά.

MedicalRecord – Patient : Ένας ασθενής μπορεί να έχει πολλές εγγραφές στον πίνακα MedicalRecord, καθώς κάθε εγγραφή αναφέρεται σε διαφορετικό σημείο τραυματισμού ή ασθένειας. Όμως μια εγγραφή ιατρικού ιστορικού αντιστοιχεί σε έναν και μόνο ασθενή. Έτσι προκύπτει η σχέση πολλά προς ένα.

ΠΑΡΑΡΤΗΜΑ Β

Η εφαρμογή παρουσίασης του ανθρώπινου σώματος 'τρέχει' σε λειτουργικό σύστημα Windows και επιπλέον χρειάζεται:

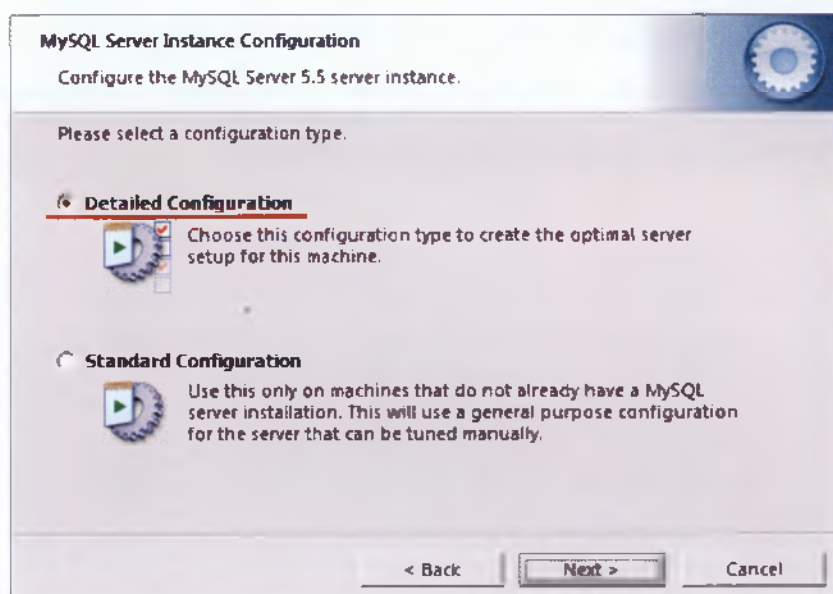
- [.NET Framework 4.0.](#)
- [MySQL Server.](#)
- [ADO.NET Driver for MySQL \(Connector/NET\).](#)
- Η βάση δεδομένων 'TheHumanBody'.
- Να έχει παραμετροποιηθεί η λειτουργία Speech Recognition των Windows

Επίσης απαιτείται μικρόφωνο για τη λειτουργία των φωνητικών εντολών.

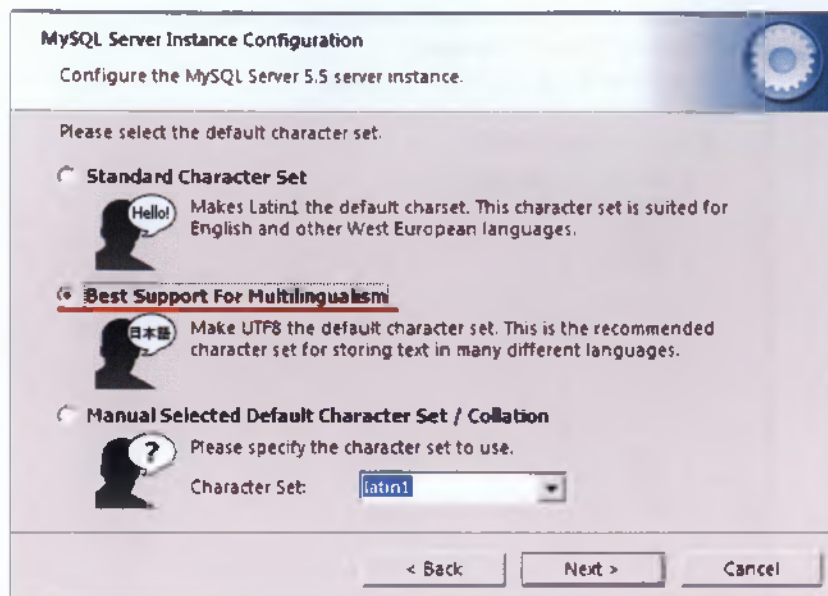
Όλα τα προγράμματα που χρειάζεται το πρόγραμμα για να λειτουργήσει υπάρχουν και στο CD στον φάκελο «Εκτελέσιμο Πρόγραμμα, Βάση & Απαραίτητα Εργαλεία → Απαραίτητα Εργαλεία».

1 ΕΓΚΑΤΑΣΤΑΣΗ MYSQL SERVER

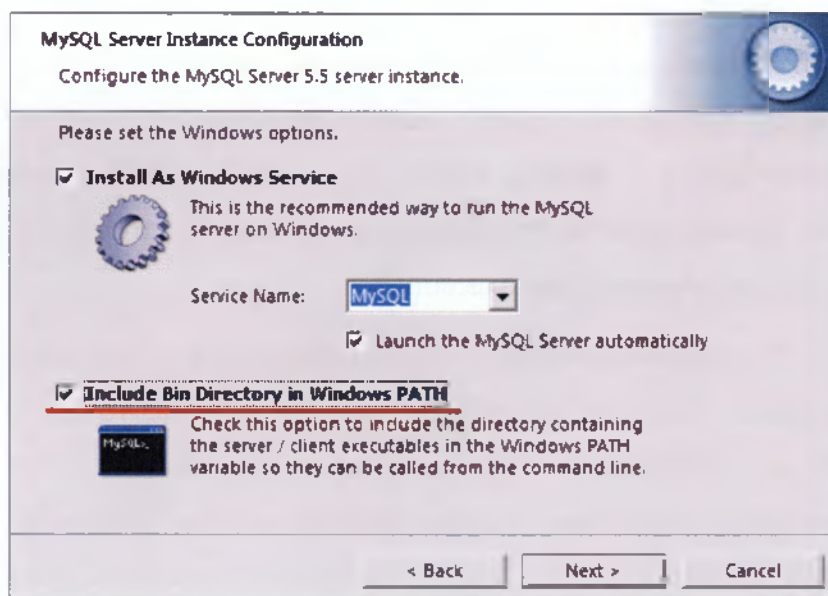
Για να λειτουργήσει το πρόγραμμα πρέπει να είναι εγκατεστημένος ο MySQL Server. Στο CD υπάρχει το αρχείο *'mysql-5.5.28-win32.msi'* το οποίο βρίσκεται στην τοποθεσία *'Εκτελέσιμο Πρόγραμμα, Βάση & Απαραίτητα Εργαλεία\Απαραίτητα Εργαλεία'*. Αν εκτελέσουμε αυτό το αρχείο θα εμφανιστεί το παράθυρο εγκατάστασης με όνομα *'MySQL Server 5.5 Setup'*. Αποδεχόμαστε τους όρους χρήσης και επιλέγουμε πλήρη εγκατάσταση. Όταν ολοκληρωθεί η εγκατάσταση και πατήσουμε το κουμπί *'Finish'*, εμφανίζεται το παράθυρο παραμετροποίησης του MySQL Server. Αφήνουμε τις προεπιλεγμένες τιμές, αλλά παραμετροποιούμε μόνο τις τιμές των παραθύρων που φαίνονται στις εικόνες πιο κάτω.



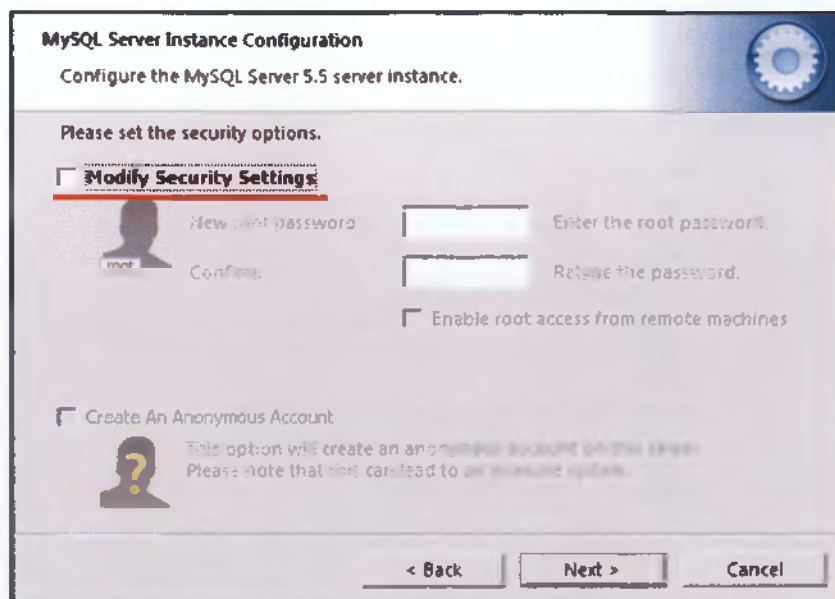
Εικόνα 20 - Επιλέγουμε λεπτομερή παραμετροποίηση



Εικόνα 21 - Επιλέγουμε το UTF8 ως προεπιλεγμένο σετ χαρακτήρων, ώστε στη βάση να μπορούμε να εισάγουμε ελληνικούς χαρακτήρες.



Εικόνα 22 - Φροντίζουμε να ενεργοποιήσουμε το δεύτερο πεδίο 'Include Bin Directory in Windows PATH'.

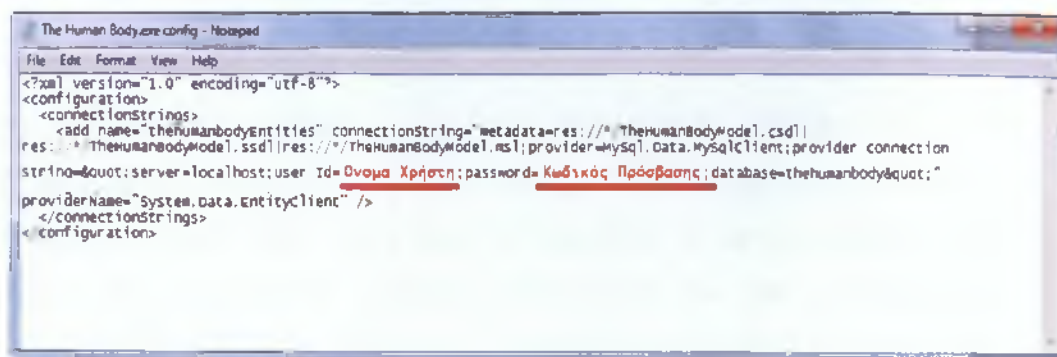


Εικόνα 23 - Απενεργοποιούμε την επιλογή τροποποίησης των ρυθμίσεων ασφαλείας.

Η απενεργοποίηση της τροποποίησης των ρυθμίσεων ασφαλείας έχει ως αποτέλεσμα την δημιουργία ενός λογαριασμού για τη σύνδεση με τη βάση δεδομένων, χωρίς κωδικό πρόσβασης. Ωστόσο αν υπήρχε εγκατεστημένος ο MySQL Server και υπήρχε κωδικός πρόσβασης, τότε για να λειτουργήσει το πρόγραμμα, ο κωδικός θα έπρεπε να δηλωθεί στο αρχείο 'The Human Body.exe.config'.

Το αρχείο αυτό περιέχει τις πληροφορίες για τη σύνδεση με τη βάση δεδομένων. Αυτές τις πληροφορίες τις χρησιμοποιεί το Entity Framework, ώστε να επιτευχθεί η επικοινωνία του προγράμματος με τη βάση. Στην περίπτωση λοιπόν που υπάρχει κάποιος κωδικός για την σύνδεση με τον MySQL Server, ο κωδικός πρέπει να δηλωθεί και στο αρχείο αυτό.

Ανοίγουμε το αρχείο 'The Human Body.exe.config' με την εφαρμογή 'Σημειωματάριο' και προσθέτουμε τα στοιχεία για τη σύνδεση με τη βάση δεδομένων, όπως φαίνεται και στην παρακάτω εικόνα.

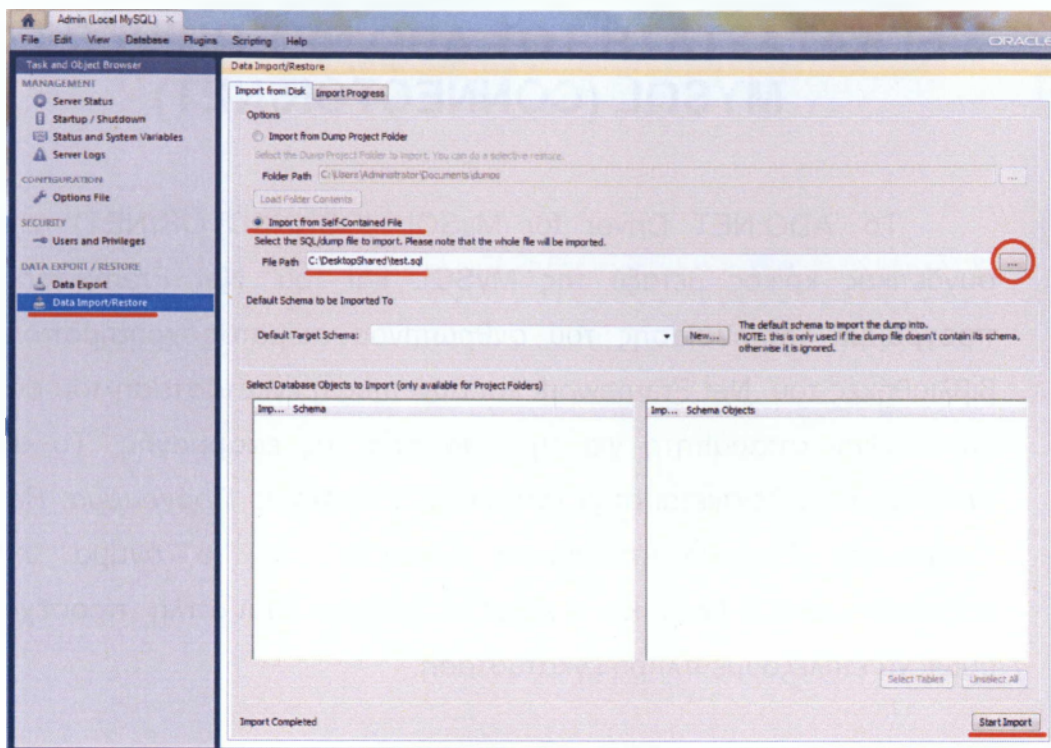


Εικόνα 24 - Υπόδειγμα για την εισαγωγή των στοιχείων σύνδεσης με τη βάση δεδομένων στο αρχείο 'The Human Body.exe.config'.

2 ΕΓΚΑΤΑΣΤΑΣΗ MYSQL WORKBENCH

Το MySQL Workbench είναι ένα γραφικό interface το οποίο επιτρέπει στον χρήστη να διαχειριστεί βάσεις δεδομένων MySQL με απλό και εύκολο τρόπο. Ο χρήστης μπορεί μέσα από γραφικά μενού να δημιουργήσει και να επεξεργαστεί βάσεις δεδομένων, να κρατήσει αντίγραφα ασφαλείας των βάσεων ή να εισάγει μια βάση δεδομένων από κάποιο αντίγραφο. Το MySQL Workbench θα χρησιμοποιηθεί για την εύκολη εισαγωγή του αντιγράφου της βάσης. Η εγκατάσταση του είναι απλή καθώς δεν χρειάζεται να κάνουμε κάποια ρύθμιση στον οδηγό εγκατάστασης. Το αρχείο εγκατάστασης βρίσκεται στην τοποθεσία *'Εκτελέσιμο Πρόγραμμα, Βάση & Απαραίτητα Εργαλεία\Απαραίτητα Εργαλεία'*, με το όνομα *'mysql-workbench-gpl-5.2.44-win32.exe'*.

Αφού τελειώσει η εγκατάσταση το πρόγραμμα εκκινεί αυτομάτως. Στον Server Administration κάνουμε διπλό κλικ στην επιλογή *'Local MySQL'* και ανοίγει ο *'MySQL Workbench Administrator'*. Στη μπάρα *'Task and Object Browser'* επιλέγουμε *'Data Import/Restore'* και στο παράθυρο που ανοίγει επιλέγουμε *'Import from Self-Contained File'* και εισάγουμε τη διαδρομή που είναι αποθηκευμένο το αντίγραφο της βάσης ή επιλέγουμε τη διαδρομή από το μενού που μας παρέχει το πρόγραμμα. Τέλος πατάμε το κουμπί *'Start Import'* και εισάγουμε τα στοιχεία σύνδεσης με τη βάση δεδομένων. Στο παράθυρο πρέπει να βλέπουμε ότι η εισαγωγή τελείωσε επιτυχώς.



Εικόνα 25 - Ο MySQL Administrator.

3 ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ ADO.NET DRIVER FOR MYSQL (CONNECTOR/NET)

Το 'ADO.NET Driver for MySQL (CONNECTOR/NET)' είναι ο συνδετικός κρίκος μεταξύ της MySQL και του .Net Framework. Το πρόγραμμα παρουσίασης του ανθρώπινου σώματος χρησιμοποιεί τις βιβλιοθήκες του .Net Framework και συνεπώς η εγκατάσταση του οδηγού αυτού είναι απαραίτητη για τη λειτουργία της εφαρμογής. Το αρχείο εγκατάστασης βρίσκεται στην τοποθεσία *Εκτελέσιμο Πρόγραμμα, Βάση & Απαραίτητα Εργαλεία\Απαραίτητα Εργαλεία*, με το όνομα *'mysql-connector-net-6.6.4.exe'* και η εγκατάστασή του είναι απλή, προσέχοντας όμως να επιλέξουμε πλήρη εγκατάσταση.

4 ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ

Η εγκατάσταση του προγράμματος γίνεται μέσα από έναν απλοποιημένο οδηγό εγκατάστασης. Ο οδηγός αυτός βρίσκεται στην τοποθεσία *‘Εκτελέσιμο Πρόγραμμα, Βάση & Απαραίτητα Εργαλεία\Εκτελέσιμο Πρόγραμμα’* του CD με όνομα *‘Setup.exe’*. Η εγκατάσταση είναι πολύ απλή, καθώς δεν χρειάζεται να γίνει κάποια ρύθμιση στις προεπιλεγμένες τιμές. Μετά το πέρας της εγκατάστασης, θα έχει δημιουργηθεί μία συντόμευση στην επιφάνεια εργασίας και μία στο μενού έναρξης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Andrew Troelsen (2010), *Pro C# 2010 and the .NET 4 Platform*, Apress.
2. Jeffrey Richter (2010), *CLR via C#, Third Edition*, Microsoft Press.
3. Adam Nathan (2010), *WPF 4 Unleashed*, Sams.
4. Julie C. Meloni (2009), *ΜΑΘΕΤΕ PHP, MYSQL ΚΑΙ APACHE Όλα Σε Ένα*,
Μ. Γκιούρδας.

Διαδικτυακές πηγές:

5. *.NET Framework Conceptual Overview*, Microsoft Corporation,
<[http://msdn.microsoft.com/en-us/library/zw4w595w\(v=vs.100\).aspx#feedback](http://msdn.microsoft.com/en-us/library/zw4w595w(v=vs.100).aspx#feedback) >
6. Rahul Rajat Singh (2012), *An Introduction to Entity Framework for Absolute Beginners*, <<http://www.codeproject.com/Articles/363040/An-Introduction-to-Entity-Framework-for-Absolute-B>>
7. Sacha Barber (2007), *LinQ Introduction*,
<<http://www.codeproject.com/Articles/18116/LINQ-Introduction-Part-1-Of-3>>
8. *System.Speech.Recognition Namespace*, Microsoft Corporation,
[http://msdn.microsoft.com/en-us/library/system.speech.recognition\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/system.speech.recognition(v=vs.100).aspx)
9. *Multitier architecture*, http://en.wikipedia.org/wiki/Multitier_architecture
10. *What Is 3-Tier(Multi-Tier) Architecture And Why Do You Need It*,
<http://blog.simcrest.com/what-is-3-tier-architecture-and-why-do-you-need-it/>

Πηγές πληροφοριών ανατομίας:

11. Φίλιππος Παπασιδέρης, Το Ανθρώπινο σώμα, Στοιχεία Ανατομίας και Φυσιολογίας.

Πηγές εικόνων που χρησιμοποιηθήκαν στην εφαρμογή:

12. http://www.plasticboy.co.uk/store/Human_Male_Anatomy_Complete_semi_textured.html

12. Φίλιππος Παπασιδέρης, Το Ανθρώπινο σώμα, Στοιχεία Ανατομίας και Φυσιολογίας.

13. <http://www.zygotebody.com>