

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ &
ΠΟΛΥΜΕΣΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Σύστημα Εσωτερικής Υποστήριξης Δανειστικής Βιβλιοθήκης»



Εισηγητής: Παχουλάκης Ιωάννης
Σπουδάστρια: Ιωαννίδη Λαρίσα

ΠΕΡΙΕΧΟΜΕΝΑ

1. Κεφάλαιο – Εισαγωγή.....	4
1.1 Αντικείμενο της πτυχιακής εργασίας.....	5
1.2 Απαιτήσεις του Συστήματος.....	6
2. Κεφάλαιο – Χρησιμοποιούμενη Τεχνολογία.....	7
2.1 Σύστημα Διαχείρισης Βάσεων Δεδομένων – Sql Server 2000.....	8
2.1.1 Βάσεις Δεδομένων.....	8
2.1.2 Microsoft Sql Server 2000.....	11
2.2 Περιβάλλον ανάπτυξης Microsoft .NET	13
2.2.1 Γλώσσα Προγραμματισμού Microsoft Visual Basic.NET.....	17
2.2.2 WebForms και VB.NET.....	19
3. Κεφάλαιο – Περιγραφή της Βάσης Δεδομένων.....	21
3.1 Πίνακες της Βάσης Δεδομένων.....	22
3.2 Διάγραμμα Βάσης Δεδομένων – Συσχετίσεις πινάκων.....	27
3.3 Αποθηκευμένες Διεργασίες – Stored Procedures.....	29
4. Κεφάλαιο – Ανάλυση διαδικασιών σε VB.NET.....	32
4.1 Επικύρωση για τις Web φόρμες.....	33
4.1.1 RequiredFieldValidator.....	34
4.1.2 RegularExpressionValidator.....	34
4.2 Ασφάλεια.....	35
4.2.1 Ασφάλεια του IIS.....	35
4.2.2 Το αρχείο web. config.....	35
4.3 Sessions.....	36
4.4 Σύνδεση με Βάση Δεδομένων.....	37
5. Κεφάλαιο – Υλοποίησης της εφαρμογής μας.....	39
5.1 Η φόρμα logon.aspx.....	40
5.2 Η φόρμα IntroForm.aspx.....	41
5.3 Η φόρμα StudentForm.aspx.....	42
5.4 Η φόρμα LendForm.aspx.....	44

5.5 Η φόρμα BookForm.aspx.....	48
5.6 Η φόρμα Info.aspx.....	53
6. Κεφάλαιο – Βιβλιογραφία – Πηγές.....	54

1

Εισαγωγή

Κεφάλαιο 1 – Εισαγωγή

Το αντικείμενο της πτυχιακής εργασίας

Το αντικείμενο της παρούσας πτυχιακής εργασίας είναι ο σχεδιασμός και η υλοποίηση μιας εφαρμογής υποστήριξης μιας δανειστικής βιβλιοθήκης. Συγκεκριμένα η εφαρμογή έχει αναπτυχθεί σε τεχνολογία VB.NET, με βάση δεδομένων σε SQL Server και θα επιτρέπει τις παρακάτω διαδικασίες στο front desk της βιβλιοθήκης.

Πιο αναλυτικά η εφαρμογή μας περιλαμβάνει:

- Την εισαγωγή νέο - αποκτηθέντων συγγραμμάτων (βιβλίων, σημειώσεων, κ.τ.λ.) στο σύστημα καθώς και την αφαίρεση από αυτό.
- Την εύρεση / ανάκτηση καρτέλας συγγράμματος (συγγραφέας, τίτλος, χρόνος έκδοσης, κατηγορία, κ.τ.λ.)
- Την υποστήριξη δανεισμού σε εγγεγραμμένους χρήστες.
- Την υποστήριξη δανειστικού ιστορικού για συγγράμματα.
- Κλείδωμα δυνατότητας δανεισμού πέραν εγκεκριμένου χρονικού ορίου.
- Εκτύπωση κατάστασης δανεισμού για χρήστες ως μερικό δικαιολογητικό για αποφοίτηση (μηδέν χρωστούμενα βιβλία).

Απαιτήσεις Συστήματος

Οι απαιτούμενες τεχνικές προδιαγραφές της πλατφόρμας εγκατάστασης είναι οι ακόλουθες:

➤ Για την πλατφόρμα .NET Framework απαιτείται ένα από τα παρακάτω λειτουργικά συστήματα:

- Microsoft Windows 2000 Professional with Service Pack 2.0
- Microsoft Windows 2000 Server with Service Pack 2.0
- Microsoft Windows 2000 Advanced Server with Service Pack 2.0
- Microsoft Windows 2000 Datacenter Server with Service Pack 2.0
- Microsoft Windows XP Professional

➤ Για την πλατφόρμα του SQL Server 2000 ένα από τα παρακάτω:

- Microsoft Windows NT Server 4.0
- Microsoft Windows NT Server Enterprise Edition 4.0
- Microsoft Windows 2000 Server
- Microsoft Windows 2000 Advanced Server
- Microsoft Windows 2000 Data Center Server
- Microsoft Windows Server 2003
- Microsoft Windows XP Professional

➤ Κοινές προδιαγραφές:

- Μνήμη: 128MB τουλάχιστον (προτεινόμενη 512MB)
- Επεξεργαστής: Pentium 133 MHz ή γρηγορότερος
- Web server: Microsoft Internet Information Services (IIS) 5.0+

2

Χρησιμοποιούμενη Τεχνολογία

Κεφάλαιο 2 – Χρησιμοποιούμενη Τεχνολογία

2.1 Σύστημα Διαχείρισης Βάσης Δεδομένων- Sql Server 2000

2.1.1. Βάσεις Δεδομένων

Βάση δεδομένων (database) είναι μια συλλογή από σχετιζόμενα δεδομένα. Όταν λέμε δεδομένα εννοούμε πληροφορίες που μπορούμε να καταγράψουμε και που έχουν κάποια υπονοούμενη σημασία. Για παράδειγμα δεδομένα είναι οι αριθμοί τηλεφώνου, οι διευθύνσεις και τα άλλα προσωπικά στοιχεία των υπαλλήλων μιας εταιρίας. Τα δεδομένα αυτά μπορεί να είναι καταγραμμένα είτε μέσα σε ταξινομημένους φακέλους της εταιρίας, είτε σε έναν υπολογιστή της εταιρίας. Και στις δύο περιπτώσεις έχουμε συλλογή από σχετιζόμενα δεδομένα, οπότε μπορούμε να λέμε ότι πρόκειται για βάση δεδομένων.

Η συνήθης χρήση του όρου βάση δεδομένων είναι αρκετά πιο περιορισμένη. Η βάση δεδομένων πρέπει να είναι μια λογικά συνεκτική συλλογή δεδομένων που έχει κάποια εγγενή σημασία. Μια τυχαία διευθέτηση δεδομένων δεν είναι σωστό να αναφέρεται ως βάση δεδομένων. Επίσης η βάση δεδομένων σχεδιάζεται, χτίζεται και γεμίζει με δεδομένα για κάποιο σκοπό. Με άλλα λόγια κύριο χαρακτηριστικό της βάσης δεδομένων είναι ότι αλληλεπιδρά ενεργά με τους χρήστες της και διαμορφώνει δυναμικά το περιεχόμενό της. Η αλληλεπίδραση αυτή στον χώρο της πληροφορικής γίνεται μέσω συστημάτων διαχείρισης βάσεων δεδομένων.

Σύστημα διαχείρισης βάσεων δεδομένων είναι μια συλλογή από προγράμματα που επιτρέπουν στους χρήστες να δημιουργήσουν και να συντηρήσουν μια βάση δεδομένων. Το σύστημα διαχείρισης μιας βάσης δεδομένων αναλαμβάνει τις διαδικασίες ορισμού, κατασκευής και χειρισμού των βάσεων δεδομένων.

Για παράδειγμα, έστω ότι θέλουμε να καταγράψουμε στον υπολογιστή τα προσωπικά στοιχεία των υπαλλήλων της εταιρίας μας. Με ένα σύστημα διαχείρισης βάσεων δεδομένων θα κατασκευάζαμε τους εξής πίνακες:

Πίνακας Υπάλληλοι

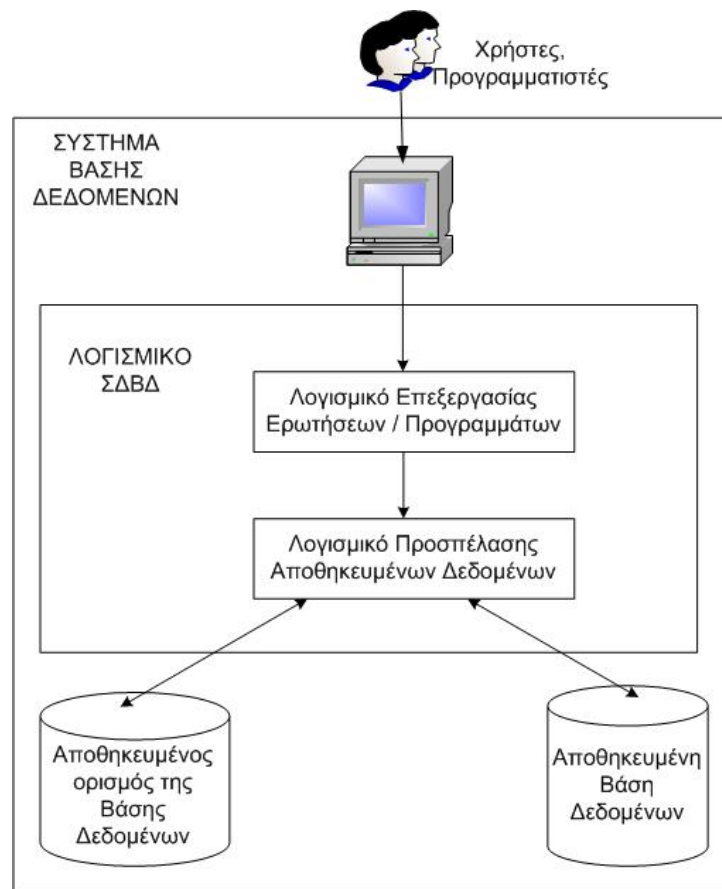
<i>A.A.T</i>	<i>Επώνυμο</i>	<i>Όνομα</i>	<i>Ημ.Γέννησης</i>	<i>Διεύθυνση</i>

Στην συνέχεια η γραμματεία της εταιρίας συμπληρώνει τα στοιχεία των υπαλλήλων, κάνει δηλαδή εισαγωγή δεδομένων στην βάση δεδομένων.

<i>A.A.T</i>	<i>Επώνυμο</i>	<i>Όνομα</i>	<i>Ημ.Γέννησης</i>	<i>Διεύθυνση</i>
Σ058640	Πατατούδης	Μπάμπης	12/04/1934	Φιλιπώτου 40
T032456	Αντωνάκοπουλος	Ζήσης	12/05/1954	Νικοπολίτου 13
Λ123452	Ελευθερόπουλος	Νίκος	03/01/1920	Δικαιάρχου 180

Αφού εισαχθούν τα στοιχεία, μπορεί να γίνει οποιαδήποτε αλλαγή με μεγάλη ευκολία και αξιοπιστία. Δεδομένου ότι χρησιμοποιούμε ένα καλό σύστημα διαχείρισης βάσης δεδομένων, τα δεδομένα της βάσης δεδομένων μπορούν να κοινοποιηθούν ελεγχόμενα και με ασφάλεια στους χρήστες. Ανάλογα με την εξουσία που έχει κάθε χρήστης μπορεί να ανακτήσει, να τροποποιήσει ή να διαγράψει δεδομένα.

Στο σχήμα που ακολουθεί φαίνεται ένα απλουστευμένο περιβάλλον συστήματος βάσης δεδομένων.



Εικόνα-Σύστημα Διαχείρισης Βάσης Δεδομένων

Η πιο γνωστή γλώσσα που χρησιμοποιείται σε Συστήματα Διαχείρισης Βάσεων Δεδομένων (Σ.Δ.Β.Δ) είναι η SQL, της οποίας το όνομα προέρχεται από το Structured Query Language (Δομημένη Γλώσσα Ερωτήσεων).

Αρχικά η SQL ονομαζόταν SEQUEL (από το Structured English Query Language, Γλώσσα Ερωτήσεων Δομημένων Αγγλικών) και σχεδιάστηκε και υλοποιήθηκε στο ερευνητικό τμήμα της IBM ως διεπαφή για ένα πειραματικό σχεσιακό σύστημα βάσεων δεδομένων που ονομαζόταν SYSTEM R. Σήμερα η γλώσσα SQL είναι η γλώσσα του συστήματος DB2 της IBM και του εμπορικού σχεσιακού Σ.Δ.Β.Δ SQL/DS και αποτέλεσε, μαζί με την QUEL, μια από τις πρώτες υψηλού επιπέδου γλώσσες βάσεων δεδομένων. Παραλλαγές της SQL έχουν υλοποιηθεί από τους κατασκευαστές των περισσότερων εμπορικών Σ.Δ.Β.Δ. Μια κοινή προσπάθεια του ANSI (American National Standards Institute, Αμερικανικό Εθνικό Ινστιτούτο Προτύπων) και του ISO (International Standards Organization, Διεθνής Οργανισμός Προτύπων)

οδήγησε σε μια τυποποιημένη έκδοση της SQL (ANSI 1986), που λέγεται SQL1. Ένα αναθεωρημένο και πολύ πιο εκτεταμένο πρότυπο έχει επίσης αναπτυχθεί με το όνομα SQL2 (αναφέρεται και ως SQL-92). Υπό εκκώλυση βρίσκεται η SQL3, που θα αποτελεί μια περαιτέρω επέκταση της SQL με αντικειμενοστραφή στοιχεία και άλλα σύγχρονα χαρακτηριστικά των βάσεων δεδομένων.

Η SQL είναι μια πλήρης γλώσσα βάσεων δεδομένων. Διαθέτει εντολές για ορισμό δεδομένων, ερωτήσεις και ενημερώσεις. Επομένως είναι ταυτόχρονα Γλώσσα Ορισμού Δεδομένων αλλά και Γλώσσα Χειρισμού Δεδομένων. Επιπλέον, περιλαμβάνει ευκολίες για τον ορισμό όψεων στη βάση δεδομένων, για τη δημιουργία και διαγραφή ευρετηρίων πάνω σε αρχεία που παριστάνουν σχέσεις (αν και αυτό το χαρακτηριστικό απαλείφθηκε από την SQL2), και για την εμφύτευση εντολών της SQL σε γενικής χρήσης γλώσσες προγραμματισμού όπως η C ή η Pascal.

2.1.2 Microsoft Sql Server 2000

Το Microsoft Sql Server 2000, αναμφίβολα, αποτελεί ένα απ' τα δημοφιλέστερα συστήματα διαχείρισης βάσεων δεδομένων. Οι περισσότεροι πιστεύουν ότι αποτελεί αποκλειστικά μια σχεσιακή βάση δεδομένων. Στην πραγματικότητα ο SQL Server 2000 αποτελεί μια ολόκληρη οικογένεια από ξεχωριστά, εντούτοις εσωτερικά συνεργαζόμενα μεταξύ τους, εργαλεία. Τα κυριότερα από αυτά τα εργαλεία είναι η Μηχανή Σχεσιακής Βάσης Δεδομένων (Relational Database Engine), οι Υπηρεσίες Μετασχηματισμού Δεδομένων (Data Transformation Services), οι Υπηρεσίες Ανάλυσης Δεδομένων (Analysis Services), Υπηρεσίες Μεταδεδομένων (Meta Data Services), η δημιουργία αντιτύπων (Replication), και οι ερωταποκρίσεις σε ελεύθερη γλώσσα (English Query).

Η μηχανή σχεσιακής βάσης δεδομένων (Relational Database Engine) αποτελεί την καρδιά του Microsoft SQL Server. Η μηχανή αυτή υποστηρίζει πλήρως το SQL νέο πρότυπο, το SQL3. Το πρότυπο SQL3 περιλαμβάνει αρκετές επεκτάσεις όπως υποστήριξη σκανδαλισμών (triggers), περιορισμών (constraints), ασφάλειας, αποθηκευμένων διαδικασιών (stored procedures), νέες εντολές ή επεκτάσεις τους. Το πρότυπο προσπαθεί να υποστηρίξει κάποιες

αντικειμενοστραφείς επεκτάσεις που γεφυρώνουν σχεσιακές βάσεις με αντικειμενοστραφή προγραμματισμό.

Το SQL Server 2000 ακολουθεί την αρχιτεκτονική του μοντέλου πελάτη/ εξυπηρετητή. Ο SQL Server 2000 εξυπηρετητής (server) αποτελείται από την βάση δεδομένων – το σύνολο των αποθηκευμένων δεδομένων, συμπεριλαμβανομένων και του ημερολογίου και των αρχείων ελέγχου, τις διάφορες διαδικασίες - συμπεριλαμβανομένων των διαδικασιών του SQL Server και των διαδικασιών των χρηστών συνολικά. Σε ένα σύστημα μπορεί να εγκαταστήσουμε παραπάνω από έναν εξυπηρετητή και η διάκριση μεταξύ τους γίνεται δίνοντας σε κάθε έναν ένα μοναδικό όνομα του στιγμιότυπου του (instance name). Ο πελάτης μπορεί να συνδεθεί σε οποιοδήποτε στιγμιότυπο εξυπηρετητή είναι προσβάσιμο μέσω δικτύου.

Η μηχανή βάσης δεδομένων του Microsoft SQL Server 2003 είναι εξαιρετικά ευέλικτη. Μπορεί να ρυθμιστεί να τρέχει με τον βέλτιστο τρόπο σε έναν απομονωμένο υπολογιστή που έχει λειτουργικό σύστημα Windows 2000 Professional, ή να ρυθμιστεί να τρέχει ως μια συστάδα εξυπηρετητών που τρέχουν το Windows 2000 Datacenter Server.

Η μηχανή βάσης δεδομένων του SQL Server έχει την δυνατότητα ακόμα και να αυτορυθμίζεται δυναμικά. Με άλλα λόγια, μπορεί να ζητάει ή να απελευθερώνει πόρους συστήματος ανάλογα με τις εκάστοτε ανάγκες. Για παράδειγμα, αν στο σύστημα έχουν συνδεθεί ταυτόχρονα 50 χρήστες και ξαφνικά συνδεθούν άλλοι 50 χρήστες, ο SQL Server αυτόματα θα ζητήσει παραπάνω πόρους συστήματος. Ένα άλλο παράδειγμα είναι, όταν τα αρχεία καταγραφής των δοσοληψιών γεμίσουν, ο SQL Server αυτόματα θα αναθέσει και άλλο χώρο δίσκου για αυτά, έτσι ώστε η βάση δεδομένων να συνεχίσει την δραστηριότητα της.

Το Microsoft SQL Server 2000 δεν περιλαμβάνει μόνο μια καλή μηχανή βάσης δεδομένων αλλά και αρκετές εφαρμογές που έχουν ένα εξαιρετικά πλούσιο περιβάλλον χρήσης. Ανάμεσα τους, το Enterprise Manager, που μας επιτρέπει να πραγματοποιήσουμε λειτουργίες διαχείρισης στις βάσεις δεδομένων και το Query Analyzer, ένα χρήσιμο εργαλείο για τους προγραμματιστές.

Κύρια πλεονεκτήματα του Microsoft SQL Server είναι η ταχύτητα, η αντοχή σε μεγάλους όγκους δεδομένων, η εξαιρετική αξιοπιστία, η ασφάλεια δεδομένων και η ευελιξία. Τα

χαρακτηριστικά αυτά είναι απαραίτητα για ένα απαιτητικό σύστημα , όπως είναι ένα σύστημα διαχείρισης εγγράφων.

2.2 Περιβάλλον ανάπτυξης Microsoft .Net

Οι γλώσσες προγραμματισμού δημιουργήθηκαν αμέσως μετά την εφεύρεση του Η-Υ, με σκοπό να διευκολυνθεί η ανάπτυξη των απαιτούμενων λογισμικών συστημάτων για την εκμετάλλευση αυτής της θαυμαστής μηχανής. Βέβαια, κάθε Η-Υ διαθέτει από κατασκευής του μια γλώσσα προγραμματισμού που ονομάζεται γλώσσα μηχανής. Η γλώσσα μηχανής είναι μια γλώσσα προγραμματισμού πολύ χαμηλού επιπέδου. Κάθε εντολή της αντιστοιχεί ακριβώς σε μία λειτουργία που θα εκτελέσει ο Η-Υ και, άρα, για την αυτοματοποίηση μιας πολύπλοκης εργασίας απαιτούνται πολλές εντολές γλώσσας μηχανής. Ως εκ τούτου, η γλώσσα μηχανής δεν προσφέρεται για την ανάπτυξη λογισμικών συστημάτων. Ήταν λοιπόν επιτακτική η ανάγκη δημιουργίας άλλων γλωσσών, υψηλότερου επιπέδου.

Πράγματι ξεκίνησε μια μεγάλη προσπάθεια για τη δημιουργία γλωσσών προγραμματισμού υψηλού επιπέδου και είναι αξιοθαύμαστο ότι σε σύντομο διάστημα ετοιμάστηκαν, αρκετές που βέβαια στην αρχή ήταν πρωτόγονες αλλά με την πάροδο του χρόνου βελτιώθηκαν σε διαδοχικές εκδόσεις και η κατάσταση σήμερα είναι να έχουμε πάρα πολλές γλώσσες προγραμματισμού ικανοποιητικής ποιότητας.

Ένα ισχυρό κριτήριο της ποιότητας μιας γλώσσας προγραμματισμού είναι ο βαθμός συμβατότητας της με τα διάφορα λειτουργικά συστήματα που υπάρχουν στην αγορά, την τρέχουσα εποχή. Χαρακτηριστικό παράδειγμα αποτελεί η γλώσσα C , που γρήγορα δημιουργήθηκαν μεταγλωττιστές της για κάθε λειτουργικό σύστημα που υπάρχει, γεγονός που συνέβαλε στην ισχύ και δημοφιλότητα της. Ένα ακόμα παράδειγμα είναι αυτό της γλώσσας προγραμματισμού Java, της εταιρίας Sun. Η Java έχει την ικανότητα να τρέχει σε οποιαδήποτε λειτουργικό σύστημα , αρκεί να είναι εγκατεστημένη στο σύστημα η πλατφόρμα διερμηνείας της. Η συμβατότητα με άλλα λειτουργικά συστήματα ήταν ένα απ' τα πιο δυνατά επιχειρήματα των επικριτών της πανίσχυρης πολυεθνικής εταιρίας Microsoft, κατά των εργαλείων ανάπτυξης λογισμικού της. Η Microsoft είχε δημιουργήσει αρκετά δημοφιλή πακέτα ανάπτυξης λογισμικού που κύρια πλεονεκτήματα τους ήταν η εύκολη δημιουργία γραφικού περιβάλλοντος χρήστη (Visual Basic) και η χρήση ισχυρών συναρτήσεων που εκμεταλλεύονται αποδοτικά τις

κλήσεις συστήματος του λειτουργικού συστήματος Windows (Visual C++) . Ωστόσο η Microsoft μπορούσε να παίζει μόνο μέσα στην έδρα της. Τα λειτουργικά συστήματα Windows. Το τοπίο άλλαξε δραματικά με τον ερχομό του περιβάλλοντος Microsoft .NET.

Το περιβάλλον ανάπτυξης Microsoft .NET ενσωματώνει αρκετές καινοτομίες τόσο στον τρόπο ανάπτυξης όσο και στην εγκατάσταση των εφαρμογών. Ο λόγος για αυτό εντοπίζεται στην ίδια την αρχιτεκτονική των εκτελέσιμων αρχείων.

Στην καρδιά του .NET βρίσκεται το CLR (Common Language Runtime). Είναι η κοινή λειτουργική μονάδα που ευθύνεται για τη διαχείριση της μεταγλώττισης και στη συνέχεια της εκτέλεσης του κώδικα. Η Microsoft αποφάσισε να αλλάξει τη μορφή των εκτελέσιμων αρχείων και κατ' επέκταση τον τρόπο εκτέλεσης των εφαρμογών.

Μέχρι σήμερα, η μεταγλώττιση μιας εφαρμογής από τον αντίστοιχο compiler δημιουργούσε εκτελέσιμα αρχεία (EXE ή DLL). Αυτά περιελάμβαναν απευθείας κώδικα μηχανής για την πλατφόρμα x86 και φυσικά μπορούσαν να τρέξουν μόνο σε αυτού του είδους τα συστήματα.

Στο περιβάλλον .NET η μεταγλώττιση από έναν compiler, όπως είναι οι Visual Basic, C++ και C# στο Visual Studio, δημιουργεί έναν ενδιάμεσο κώδικα, που η Microsoft ονομάζει MSIL (Microsoft Intermediate Language). Αυτός είναι ανεξάρτητος από τον επεξεργαστή και βρίσκεται σε υψηλότερο επίπεδο από τον κώδικα μηχανής. Περιλαμβάνει μάλιστα και μεταδεδομένα (στοιχεία με πληροφορίες για τη χρήση τους) που χρησιμοποιεί το CLR για να φορτώσει συγκεκριμένες λειτουργίες, να τοποθετήσει αντικείμενα στην μνήμη, να ελέγξει την ασφάλεια και να πραγματοποιήσει μια σειρά από πρόσθετες λειτουργίες. Δεν πρόκειται ασφαλώς για απλό κώδικα μηχανής, καθώς το MSIL έχει σκοπό να κάνει πιο εύκολη τη μετάβαση από τα 32μπιτα Windows στη 64μπιτη πλατφόρμα των Windows 2000 και 2003. Φυσικά, οι εντολές του MSIL δεν μπορούν να εκτελεστούν ως έχουν από τους σημερινούς επεξεργαστές, καθώς πρέπει πρώτα να μεταφραστούν σε γλώσσα μηχανής, στοιχείο που αναλαμβάνει το CLR.

Για τη δημιουργία του MSIL η Microsoft συνεργάστηκε με αρκετούς σχεδιαστές και συγγραφείς γλωσσών και μεταγλωττιστών τόσο σε εταιρικό όσο και σε ακαδημαϊκό επίπεδο. Ίσως ορισμένοι, στα ενδότερα της Microsoft, να ελπίζουν ότι το MSIL θα σημάνει και το τέλος

της Java, δίνοντας ένα δυνατό χτύπημα στον αντίπαλο της, την Sun. Ωστόσο η ύπαρξη του MSIL είναι ένας από τους λόγους για τους οποίους δεν είναι δυνατή η εκτέλεση ακόμα και του πιο μικρού προγράμματος που έχει μεταγλωττιστεί με το Visual Studio .NET σε ένα σύστημα όπου δεν έχει εγκατασταθεί η πλατφόρμα του .NET Framework.

Τα πλεονεκτήματα αυτού του τρόπου εκτέλεσης των εφαρμογών είναι πολλά. Πρώτα από όλα προσφέρεται ένας ενιαίος τρόπος για τη χρήση των υπηρεσιών του λειτουργικού συστήματος, σε αντίθεση με την σημερινή κατάσταση, κατά την οποία μερικές υπηρεσίες βρίσκονται σε DLL και άλλες σε αρχεία τύπου COM. Αυτό σημαίνει επίσης ότι παύει η δυσχρηστία με τα DLL, δηλαδή το φαινόμενο κατά το οποίο η εγκατάσταση μιας εφαρμογής αντικαθιστά κάποιο DLL μιας άλλης εφαρμογής με απρόβλεπτα, όπως πάντα, αποτελέσματα. Στην πλατφόρμα .NET υπάρχει πια ο διαχωρισμός των στοιχείων της κάθε εφαρμογής, έτσι ώστε όταν αυτή εκτελείται, να φορτώνονται συγχρόνως και τα στοιχεία με τα οποία αναπτύχθηκε και ελέγχθηκε. Η δυνατότητα της εύκολης μεταφοράς μεταξύ διαφορετικών επεξεργαστών και λειτουργικών συστημάτων και η ευελιξία στον προγραμματισμό με τη χρήση πολλαπλών γλωσσών συμπληρώνουν με μια πρώτη ματιά τα πιο αξιόλογα πλεονεκτήματα της πλατφόρμας .NET.

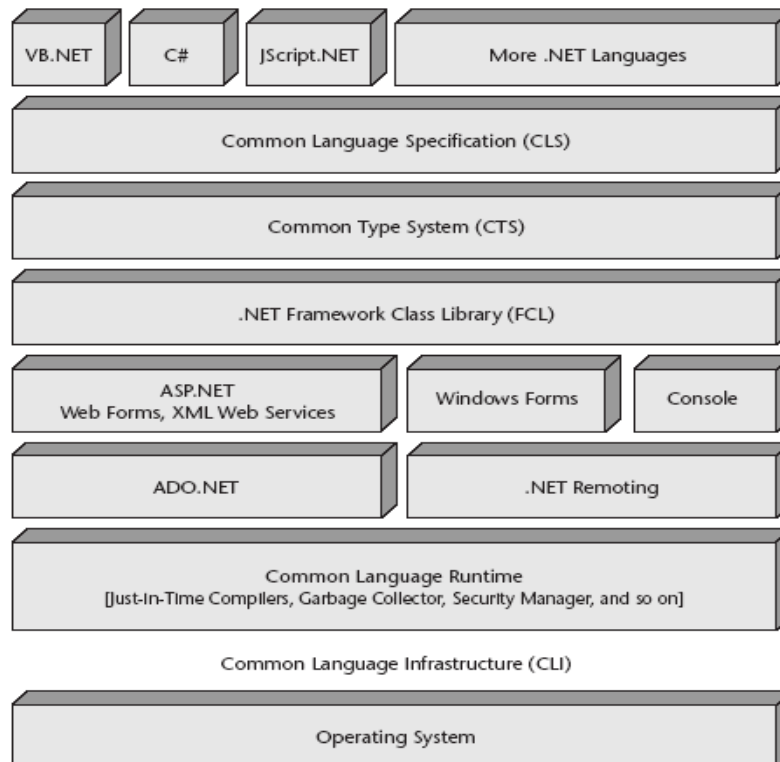
Σήμερα υπάρχουν αρκετές εκδόσεις των Windows, τα Windows 95, Windows 98, Windows 98 SE, Windows Millennium, Windows CE, Windows NT 4.0, Windows 2000 και σύντομα πρόκειται να κυκλοφορήσουν και τα 64μπιτα Windows 2000. Τα περισσότερα από αυτά τρέχουν σε επεξεργαστές x86. Η εξαίρεση αυτού του κανόνα έρχεται με τα Windows CE και τα 64μπιτα Windows. Υπάρχουν φυσικά και αρκετές νέες συσκευές στα άμεσα πλάνα των κατασκευαστών που αναμένονται να έχουν κάποιο, διαφορετικό ίσως, τρόπο επικοινωνίας με τον χρήστη. Θεωρητικά, μια εφαρμογή .NET μπορεί να εκτελείται σε λειτουργικά συστήματα διαφορετικά από τα Windows, αρκεί να υπάρχει ένα CLR για κάθε διαφορετική συσκευή, λύνοντας έτσι τα χέρια του προγραμματιστή.

Ένα ακόμα πλεονέκτημα είναι ότι το Visual Studio .NET επιτρέπει την ενσωμάτωση περισσότερων γλωσσών προγραμματισμού. Μπορεί δηλαδή να δημιουργήσει κανείς μια κλάση σε C++ που να προέρχεται από μια άλλη κλάση σε Visual Basic ή το αντίστροφο.

Η εγκατάσταση μιας εφαρμογής αποτελεί, σε όλες ανεξαιρέτως εκδόσεις των Windows, ένα από τα πιο δύστροπα σημεία. Συνήθως, ο κατασκευαστής συνοδεύει την εφαρμογή με

αρκετά πρόσθετα αρχεία, εκτελέσιμα DLL και OCX, εγγραφές στο μητρώο και συντομεύσεις. Τα Windows 2000 έφεραν την εφαρμογή ανάπτυξης προγραμμάτων εγκατάστασης, το Windows Installer που βελτίωσε κάπως την κατάσταση. Στην πλατφόρμα .NET η Microsoft αποσύνδεσε επιτέλους τα στοιχεία των εφαρμογών από το μητρώο του συστήματος. Στην πράξη, το .NET υπόσχεται να κάνει την εγκατάσταση τόσο απλή όσο μια αντιγραφή των αρχείων από τον ένα κατάλογο στον άλλο και την απεγκατάσταση τόσο εύκολη όσο τη διαγραφή αυτών των αρχείων.

Figure 1: An overview of the .NET architecture



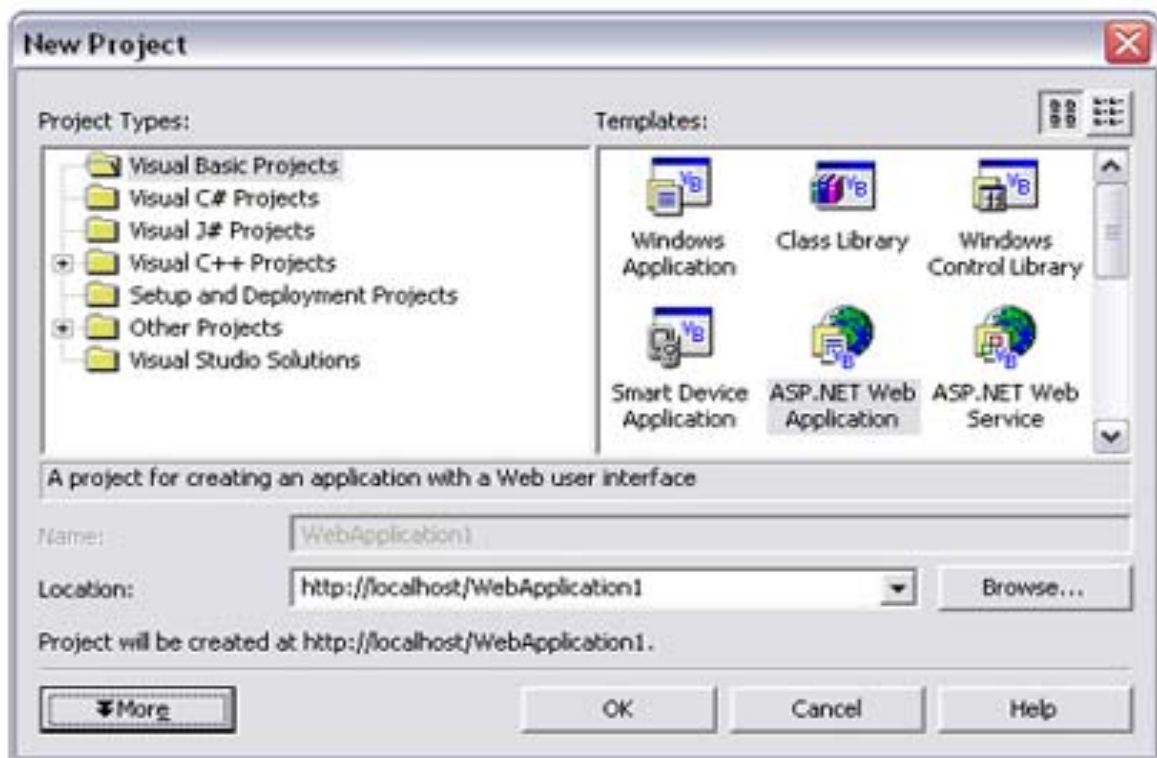
2.1.1.1 Αρχιτεκτονική .NET

2.2.1 Γλώσσα Προγραμματισμού Microsoft Visual Basic .NET

Η Visual Basic ακολούθησε εντυπωσιακή εξέλιξη την τελευταία δεκαετία, από την έκδοση 1.0 στην σημερινή έκδοση .NET, ή 7.0, όπως είναι η εσωτερική αρίθμηση του προϊόντος. Οι νέες εντολές, οι βελτιώσεις στον τρόπο ανίχνευσης των λαθών αλλά και όσα εργαλεία προστέθηκαν μέχρι τη σημερινή έκδοση, δεν άλλαξαν τον αντικειμενικό σκοπό, που είναι η εύκολη και γρήγορη ανάπτυξη εφαρμογών.

Έτσι, η Visual Basic, μέσα από το Visual Studio, παραμένει ένα εύκολο και ιδιαίτερα γρήγορο εργαλείο για την ανάπτυξη αυτόνομων εφαρμογών αλλά και υπηρεσιών στο Web. Επιλέγοντας μια νέα λύση (Solution), μέσα από το περιβάλλον της Visual Basic, μπορείτε να δημιουργήσετε μια αυτόνομη εφαρμογή (Windows Application), μια εφαρμογή στο Internet (Web Application) ή μια υπηρεσία στο Web (Web Service). Και στις τρεις περιπτώσεις όλα γίνονται εύκολα, με το σύστημα να κάνει για εσάς την περισσότερη δουλειά.

Ωστόσο, η νέα έκδοση Visual Basic.NET φέρνει αρκετές αλλαγές στην ίδια τη γλώσσα, στο περιβάλλον προγραμματισμού και κατ' επέκταση στη μεταφορά μιας υπάρχουσας εφαρμογής. Οι βελτιώσεις στη γλώσσα περιλαμβάνουν νέες εντολές και σχήματα που φέρνουν πιο κοντά τη VB στις υπόλοιπες αντικειμενοστρεφείς γλώσσες, όπως η C++ και η Java. Από τις πιο σημαντικές βελτιώσεις σε αυτό τον τομέα είναι η δυνατότητα της κληρονομικότητας (inheritance), της δημιουργίας κλάσεων (class) αλλά και της ενσωμάτωσης πιο αυστηρών κανόνων για την αντίδραση σε λάθη (error trapping). Ο προγραμματιστής έχει στη διάθεσή του την εντολή "inherits", που επιτρέπει τη μεταφορά όλων των ιδιοτήτων ενός αντικειμένου ή μιας κλάσης. Υπάρχει επίσης και το function overloading, που προσφέρει τη δυνατότητα να χρησιμοποιούμε τις ίδιες λειτουργίες με διαφορετικές παραμέτρους. Επιπλέον, κανείς είναι σε θέση να δημιουργήσει ένα νέο νήμα (thread) με την εντολή New Thread. Με αυτό τον τρόπο η εκτέλεση μιας εφαρμογής μπορεί να μην είναι σύγχρονη, να μην χρειάζεται δηλαδή να περιμένει μια εντολή το τέλος της προηγούμενης.



Σχήμα 2.2.1 Το Visual Studio απλοποιεί τις διαδικασίες χρησιμοποιώντας έτοιμα πρότυπα εφαρμογών.

Οι εντολές On Error και Resume Next έχουν αντικατασταθεί με πιο σταθερές και συνεπείς λύσεις, όπως οι εντολές Try, Catch και Final, επιτρέποντας τη συγγραφή πιο καθαρού κώδικα. Υπάρχει επίσης η δυνατότητα της αρχικοποίησης μιας μεταβλητής κατά τη δήλωσή της (Dim Y as Integer = 1). Θα βρείτε αλλαγές και σε πολλά ακόμα σημεία που αφορούν σε μεμονωμένα στοιχεία ελέγχου, όπως το μήνυμα MsgBox. Επιπλέον, μια αλλαγή που πιθανόν θα δημιουργήσει προβλήματα με τον κώδικα της προηγούμενης έκδοσης, είναι ο τρόπος αντιμετώπισης των παραμέτρων. Στην έκδοση 6 οι παράμετροι σε μια διαδικασία (function DoSomething(X As Integer)) περνούσαν ως ByRef, που σημαίνει ότι οι τιμές τους μπορούσαν να αλλαχτούν μέσα στη συγκεκριμένη διαδικασία.

Στην έκδοση .NET οι παράμετροι περνούν ως ByVal, που σημαίνει ότι πρέπει να αλλαχτεί ο κώδικας (function DoSomething (ByRef X As Integer)) για να περάσει η σωστή τιμή στον κορμό του προγράμματος. Μια αλλαγή που προβλέπεται επίσης να δημιουργήσει αρκετά προβλήματα είναι τα χαρακτηριστικά των αντικειμένων σε μια φόρμα.

Όπως θα έχετε καταλάβει, οι αλλαγές είναι πολλές και η μεταφορά από την έκδοση 6 απαιτεί την πληκτρολόγηση αρκετού κώδικα. Για αυτόν το λόγο η Microsoft ενσωμάτωσε ένα βοηθητικό εργαλείο (Upgrade Tool), που αναλαμβάνει την αναβάθμιση της εφαρμογής στο νέο περιβάλλον, αλλάζοντας αυτόματα τις εντολές στο νέο φορμά. Αυτό το εργαλείο ενεργοποιείται αυτόματα αν ανοίξετε κάποια παλιότερη εφαρμογή μέσα από το Visual Studio .NET. Το αποτέλεσμα είναι να δημιουργηθεί ένα νέο έργο, αφήνοντας τα αρχικά αρχεία της έκδοσης 6.0 όπως έχουν. Δύσκολα όμως θα αποφύγετε την επέμβαση στον κώδικα, αφού τις περισσότερες φορές η αυτόματη αναβάθμιση θα δημιουργήσει αρκετές ασυμβατότητες.

Ένα ακόμα ενδιαφέρον χαρακτηριστικό στις φόρμες της Visual Basic .NET είναι η μεγαλύτερη ακρίβεια. Χαρακτηριστικά όπως το Menu Designer, Control Anchoring, Control Docking και μια σειρά από νέα στοιχεία ελέγχου επιτρέπουν να αναπτύξει κανείς πιο καλαίσθητες φόρμες. Σε αυτό έρχεται να προστεθεί και η υποστήριξη του GDI+, του διαδόχου του GDI (Graphic Device Interface) που συναντούσαμε στις προηγούμενες εκδόσεις των Windows, το οποίο προσφέρει πιο πλούσια χαρακτηριστικά. Έτσι, μπορεί κανείς εύκολα να σχεδιάσει οθόνες που εμφανίζονται σε διαφανή και πολυεπίπεδα παράθυρα, χαρακτηριστικό που επιτρέπουν τα Windows 2000 και τα Windows XP. Άλλο ένα ενδιαφέρον στοιχείο της Visual Basic .NET είναι η πλήρης υποστήριξη του Unicode, ακόμα και στις φόρμες.

Μαζί με το Visual Studio έρχεται και μια έκδοση του Crystal Reports, της γνωστής εφαρμογής για εύκολη και γρήγορη κατασκευή εκτυπώσεων, με τις ανάλογες επεκτάσεις για την υποστήριξη εκτυπώσεων σε εφαρμογές για το Internet.

2.2.2. Web Forms και VB .NET

Η Visual Basic ακολούθησε εντυπωσιακή εξέλιξη την τελευταία δεκαετία, από την έκδοση 1.0 στην σημερινή έκδοση .NET, ή 7.0, όπως είναι η εσωτερική αρίθμηση του προϊόντος. Οι νέες εντολές, οι βελτιώσεις στον τρόπο ανίχνευσης των λαθών αλλά και όσα εργαλεία προστέθηκαν μέχρι τη σημερινή έκδοση, δεν άλλαξαν τον αντικειμενικό σκοπό, που είναι η εύκολη και γρήγορη ανάπτυξη εφαρμογών.

Τα Web Form (φόρμες για το Web) αποτελούν μια καινούρια μέθοδο, μια νέα προσέγγιση στον τρόπο δημιουργίας εφαρμογών για το Internet. Πρακτικά γεφυρώνει το κενό μεταξύ της ανάπτυξης αυτόνομων εφαρμογών για τα Windows και για το Internet, καθώς χρησιμοποιεί τα

ίδια εργαλεία, κενές φόρμες, μια εργαλειοθήκη και κώδικα σε Visual Basic (ή οποιαδήποτε άλλη γλώσσα). Απαραίτητη προϋπόθεση είναι η ύπαρξη του αντίστοιχου διακομιστή της πλατφόρμας .NET (Windows 2000 και το .NET Framework).

Μια φόρμα Web Form αποτελείται από δύο μέρη: το αρχείο HTML που αντιπροσωπεύει την οπτική απεικόνιση της σελίδας, έτσι όπως θα τη δει ο τελικός χρήστης στο PC, και ένα αρχείο με κώδικα σε VB που χειρίζεται τα γεγονότα, τα αντικείμενα και τα αποτελέσματα στη συγκεκριμένη σελίδα.

Η δημιουργία μιας απλής εφαρμογής, στην οποία ο τελικός χρήστης θα μπορεί να εισαγάγει το όνομα και τον κωδικό πρόσβασης, είναι υπόθεση λίγων λεπτών. Αρκεί να "τραβήξει" κανείς δύο αντικείμενα τύπου Label για τις περιγραφές, δύο άλλα αντικείμενα TextBox και ένα κουμπί εντολών. Το σκεπτικό που ακολουθούν οι φόρμες για το Web είναι το ίδιο όπως και στον κλασικό προγραμματισμό. Με διπλό κλικ, για παράδειγμα, πάνω στο κουμπί εντολών μπορείτε να γράψετε τον κώδικα που θέλετε και να αλλάξετε οποιοδήποτε από τα χαρακτηριστικά των αντικειμένων που βρίσκονται πάνω στη φόρμα. Αν χρησιμοποιήσετε ελληνικούς χαρακτήρες στις επιλογές, τα μενού και τα πεδία κειμένου, καλό είναι να επιλέξετε, με δεξί κλικ πάνω στη φόρμα, το ελληνικό σετ χαρακτήρων.

Ο διακομιστής ανταλλάσσει στοιχεία και δεδομένα σε μορφή XML μέσα από τις φόρμες Web Form και εμφανίζει τα αποτελέσματα σε μορφή HTML 3.2. Αυτό σημαίνει ότι τα αποτελέσματα είναι δυνατόν να εμφανιστούν μέσα από οποιοδήποτε πρόγραμμα αναζήτησης. Τα αρχεία HTML έχουν τη μορφή .aspx, η οποία είναι η νέα έκδοση των σελίδων .asp, που επιτρέπει την πολύ καλύτερη διαχείριση της VB .NET μέσα στην ίδια την ιστοσελίδα. Θα δείτε ότι μια φόρμα με το όνομα WebForm, για παράδειγμα, αποτελείται από δύο αρχεία, το WebForm.aspx και το WebForm.vb με κώδικα HTML και VB αντίστοιχα.

3

Περιγραφή της Β. Δ.

Κεφάλαιο 3 – Περιγραφή της Βάσης Δεδομένων

3.1 Πίνακες της Βάσης Δεδομένων

Η βάση δεδομένων αποτελείται από τους παρακάτω πίνακες:

❖ Πίνακας tblStudent :

	Column Name	Data Type	Length	Allow Nulls	
?	STUDENT_ID	int	4		
	AM	int	4		
	DEPARTMENT_ID	tinyint	1		
	FIRSTNAME	char	15		
	LASTNAME	char	30		
	ENTRY_YEAR	int	4		
	ENTRY_SEMESTER	tinyint	1		
	ADDRESS	char	20		
	AREA	char	20		
	TELEPHONE	bigint	8	✓	
	EMAIL	char	40	✓	
	ISLOCKED	tinyint	1		
	LIMIT	tinyint	1		

Σχήμα 3.1.1 Πίνακας tblStudent

Στο παραπάνω πίνακα (Σχήμα 3.1.1) γίνεται η εισαγωγή των στοιχείων του σπουδαστή. Ο πίνακας περιλαμβάνει τον κωδικό του σπουδαστή (STUDENT_ID), τον Αριθμό Μητρώου (AM), τον κωδικό του Τμήματος στο οποίο ανήκει (DEPARTMENT_ID), το όνομα (FIRSTNAME), το επίθετο (LASTNAME), την χρονιά εισαγωγής του σπουδαστή στο Τμήμα (ENTRY_YEAR), το εξάμηνο εισαγωγής του (ENTRY_SEMESTER), την διεύθυνση (ADDRESS), τον νομό (AREA), το τηλέφωνο (TELEPHONE), το e-mail (EMAIL), ένα πεδίο με το όνομα (ISLOCKED) στο οποίο φαίνεται αν ο λογαριασμός του σπουδαστή είναι κλειδωμένος ή όχι και το όριο δανεισμού (LIMIT). Παρατηρούμε πως πρωτεύων κλειδί είναι το πεδίο STUDENT_ID το οποίο παίρνει αυτόματα ένα αύξοντα αριθμό από την βάση ενώ επιτρέπεται σε δύο πεδία του πίνακα (TELEPHONE, EMAIL) να εισάγουμε NULL.

❖ Πίνακας tblItem

	Column Name	Data Type	Length	Allow Nulls
▶	ITEM_ID	int	4	
	TYPE_ID	tinyint	1	
	CATEGORY_ID	char	5	
	ISLENDABLE	bit	1	
	TAG_ID	varchar	50	
	TITLE	nvarchar	50	
	AUTHOR	nvarchar	50	
	ISBN	nvarchar	50	
	EDITION	tinyint	1	
	PUBLISHER	nvarchar	50	
	PUBL_DATE	datetime	8	
	INFO	nvarchar	50	✓

Σχήμα 3.1.2 Πίνακας tblItem

Ο παραπάνω πίνακας (Σχήμα 3.1.2) περιέχει πληροφορίες σχετικά με τα στοιχεία της βιβλιοθήκης, δηλαδή τα βιβλία ή τα εκπαιδευτικά DVD . Ο πίνακας tblItem περιλαμβάνει τον κωδικό του στοιχείου (ITEM_ID) , το είδος του στοιχείου (TYPE_ID) , τον κωδικό της κατηγορίας του (CATEGORY_ID) , την ιδιότητα του ως δανειστικό (ISLENDABLE), τον κωδικό ταξινόμησης του στοιχείου μέσα στην βιβλιοθήκη(TAG_ID), τον τίτλο του εκάστοτε στοιχείου (TITLE), τον συγγραφέα (AUTHOR), το κωδικό ISBN (ISBN), την έκδοση (EDITION), τον εκδότη (PUBLISHER), την ημερομηνία έκδοσης (PUBL_DATE) και το πεδίο με τις πληροφορίες (INFO) στο οποίο ο administrator, μπορεί να κρατήσει σημειώσεις για το συγκεκριμένο στοιχείου, χρήσιμες για εκείνον. Τέλος παρατηρούμε πως πρωτεύων κλειδί είναι το πεδίο ITEM_ID το οποίο παίρνει αυτόματα ένα αύξοντα αριθμό από την βάση ενώ επιτρέπεται η εισαγωγή του NULL στο πεδίο INFO.

❖ Πίνακας tblstudentitem

	Column Name	Data Type	Length	Allow Nulls
▶	STUDENT_ID	int	4	
	ITEM_ID	int	4	
	START_DATE	datetime	8	
	RETURN_DATE	datetime	8	✓

Σχήμα 3.1.3 Πίνακας tblstudentitem

Ο παραπάνω πίνακας (Σχήμα 3.1.3) περιέχει πληροφορίες όσον αφορά τις διαδικασίες του δανεισμού των στοιχείων. Περιλαμβάνει τέσσερα πεδία , τον κωδικό του σπουδαστή (STUDENT_ID), τον κωδικό του στοιχείου το οποίο θα δανειστεί ο σπουδαστής (ITEM_ID), την ημερομηνία δανεισμού (START_DATE) και την ημερομηνία επιστροφής (RETURN_DATE). Παρατηρούμε επίσης πως το πεδίο RETURN_DATE μπορεί να πάρει την τιμή NULL.

- Πίνακας tbldepartment

	Column Name	Data Type	Length	Allow Nulls
▶	DEPARTMENT_ID	tinyint	1	
	SCHOOL_ID	tinyint	1	
	NAME	text	16	

Σχήμα 3.1.4 Πίνακας tblDepartment

Ο παραπάνω πίνακας (Σχήμα 3.1.4) περιέχει πληροφορίες όσον αφορά το Τμήμα στο οποίο ανήκει ο σπουδαστής καθώς και την Σχολή στην οποία ανήκει το Τμήμα. Περιλαμβάνει τρία πεδία , το ID του Τμήματος στο οποίο ανήκει ο σπουδαστής (DEPARTMENT_ID), το ID της Σχολής (SCHOOL_ID) και το όνομα του Τμήματος (NAME). Το πρωτεύον κλειδί του πίνακα είναι το DEPARTMENT_ID.

❖ Πίνακας tblschool

	Column Name	Data Type	Length	Allow Nulls
?	SCHOOL_ID	tinyint	1	
	NAME	char	30	

Σχήμα 3.1.5 Πίνακας tblschool

Ο παραπάνω πίνακας (Σχήμα 3.1.5) περιέχει πληροφορίες όσον αφορά την Σχολή όπου υπάρχουν δύο πεδία, ο κωδικός της Σχολής (SCHOOL_ID) και το όνομά της (NAME). Παρατηρούμε ότι σε αυτόν τον πίνακα το πρωτεύον κλειδί είναι το SCHOOL_ID.

❖ Πίνακας tblsemester

	Column Name	Data Type	Length	Allow Nulls
?	SEMESTER_ID	tinyint	1	
	NAME	char	1	

Σχήμα 3.1.6 Πίνακας tblsemester

Ο παραπάνω πίνακας (Σχήμα 3.1.6) περιέχει πληροφορίες που αφορούν τα είδη των εξαμήνων σε ένα Τμήμα. Περιλαμβάνει δύο πεδία, τον κωδικό του εξαμήνου (SEMESTER_ID), το οποίο είναι και το πρωτεύον κλειδί και το όνομα του εξαμήνου, δηλαδή Χειμερινό ή Εαρινό (NAME).

• Πίνακας tbltype

	Column Name	Data Type	Length	Allow Nulls
?	TYPE_ID	tinyint	1	
	NAME	char	15	

Σχήμα 3.1.7 Πίνακας tbltype

Ο παραπάνω πίνακας (Σχήμα 3.1.7) αφορά τα αντικείμενα δανεισμού μιας βιβλιοθήκης και συγκεκριμένα το είδος τους. Περιέχει τον κωδικό του είδους των στοιχείων της βιβλιοθήκης (TYPE_ID) και το όνομα το οποίο είναι είτε Βιβλίο είτε DVD (NAME). Το πρωτεύον κλειδί αυτού του πίνακα είναι το TYPE_ID.

❖ Πίνακας tblcategory

	Column Name	Data Type	Length	Allow Nulls
🔑	CATEGORY_ID	char	5	
	NAME	text	16	

Σχήμα 3.1.8 Πίνακας tblcategory

Ο παραπάνω πίνακας (Σχήμα 3.1.8) αφορά τις κατηγορίες στις οποίες ανήκει το κάθε στοιχείο της βιβλιοθήκης. Περιέχει τον κωδικό της κατηγορίας (CATEGORY_ID) και το όνομα της (NAME). Το πρωτεύον κλειδί του πίνακα αυτού είναι το CATEGORY_ID.

- Πίνακας tblUser

	Column Name	Data Type	Length	Allow Nulls
🔑	USER_ID	tinyint	1	
	FirstName	varchar	50	
	LastName	varchar	50	
	Email	varchar	50	
	Password	varchar	50	
	Role	varchar	10	

Σχήμα 3.1.9 Πίνακας tblUser

Ο παραπάνω πίνακας (Σχήμα 3.1.9) αφορά κυρίως το στοιχείο ελέγχου αυτού του προγράμματος. Περιέχει το πεδίο με τον κωδικό του χρήστη (USER_ID), το όνομα του (FIRSTNAME), το επώνυμο (LASTNAME), το email του (EMAIL), το προσωπικό κωδικό του (PASSWORD) και την ιδιότητά του (ROLE). Το πρωτεύον κλειδί του πίνακα είναι το πεδίο USER_ID.

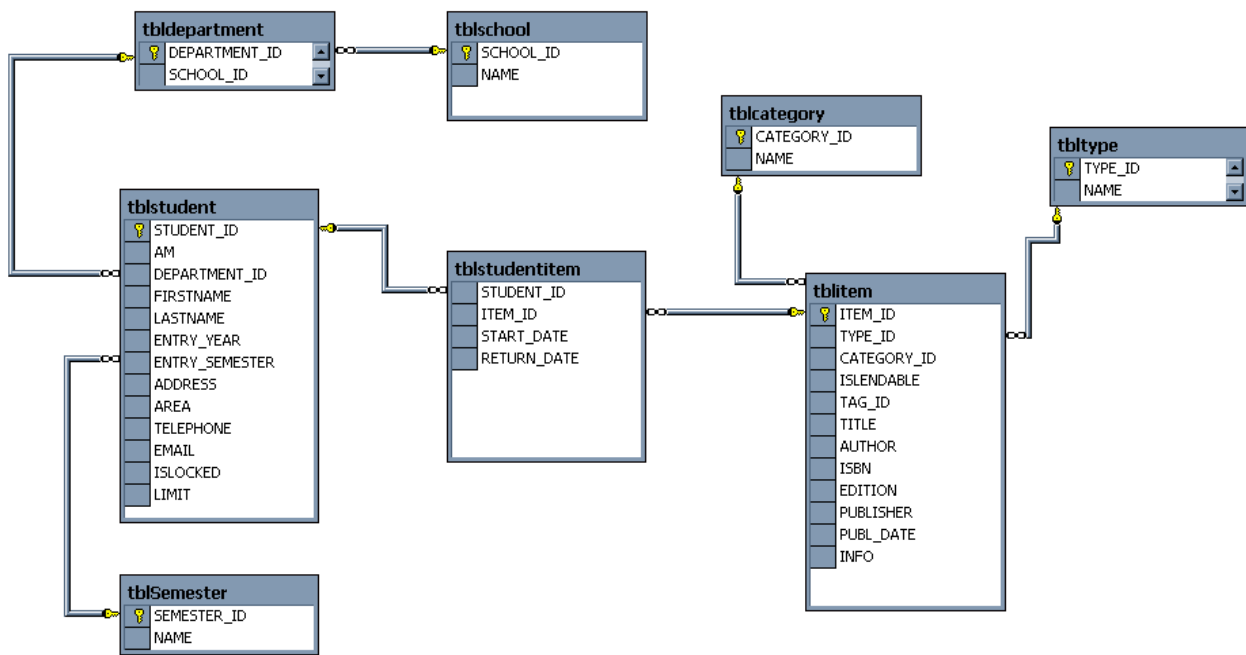
3.2 Διάγραμμα βάσης δεδομένων – Συσχετίσεις πινάκων

Το διάγραμμα της βάσης δεδομένων είναι μια γραφική αναπαράσταση των πινάκων και των μεταξύ τους σχέσεων. Η σχεδίασή του είναι ένας καλός τρόπος για να έχουμε μια εποπτική εικόνα της δομής και των συσχετίσεων της βάσης δεδομένων. Εκτός αυτού είναι ένας βολικός τρόπος τεκμηρίωσης της βάσης, επειδή εκεί απεικονίζονται αυτόματα όποιες αλλαγές επιχειρούνται.

Στο Σχήμα 3.2.1 φαίνεται το διάγραμμα της βάσης δεδομένων. Μπορούμε να παρατηρήσουμε τις σχέσεις μεταξύ των πινάκων. Ας ξεκινήσουμε από τους κύριους πίνακες που βρίσκονται στο κέντρο. Ο πίνακας `tblstudent` συνδέεται με τον πίνακα `tblstudentitem` στο πεδίο `STUDENT_ID` με σχέση «ένα προς πολλά» δηλαδή ένας σπουδαστής μπορεί να δανειστεί πολλά αντικείμενα της βιβλιοθήκης. Ο πίνακας `tblstudentitem` συνδέεται με τον πίνακα `tblitem` στο πεδίο `ITEM_ID` με σχέση «ένα προς πολλά» δηλαδή ένα αντικείμενο μπορεί να περάσει από τις διαδικασίες δανεισμού παραπάνω από μία φορά. Ο πίνακας `tblitem` συνδέεται με τον πίνακα `tbltype` στο πεδίο `TYPE_ID` με σχέση «ένα προς πολλά» δηλαδή μπορούμε να έχουμε πολλά αντικείμενα της βιβλιοθήκης τα οποία να ανήκουν στο ίδιο είδος. Ο πίνακας `tblitem` επίσης συνδέεται με τον πίνακα `tblcategory` στο πεδίο `CATEGORY_ID` με σχέση «ένα προς πολλά» δηλαδή σε μια κατηγορία μπορούν να ανήκουν πολλά αντικείμενα της βιβλιοθήκης.

Ξαναγυρίζοντας στον πίνακα `tblstudent` παρατηρούμε ότι συνδέεται με δύο πίνακες, τους `tblsemester` και `tbldepartment`. Στον πίνακα `tblsemester` συνδέεται στο πεδίο `SEMESTER_ID`, η σχέση είναι «σχέση ένα προς πολλά» δηλαδή πολλοί φοιτητές μπορούν να ανήκουν στο ίδιο εξάμηνο. Επίσης με τον πίνακα `tbldepartment`, όπου συνδέεται στο πεδίο `DEPARTMENT_ID` με σχέση «ένα προς πολλά», βλέπουμε ότι πολλοί φοιτητές αντίστοιχα μπορούν να ανήκουν στο ίδιο Τμήμα.

Τέλος ο πίνακας `tbldepartment` συνδέεται με τον πίνακα `tblschool` στο πεδίο `SCHOOL_ID` με σχέση «ένα προς πολλά» δηλαδή μια Σχολή μπορεί να έχει παραπάνω από ένα Τμήματα.



Σχήμα 3.2.1 Διάγραμμα βάσης δεδομένων

3.3 Αποθηκευμένες Διαδικασίες - *Stored Procedures*

Οι αποθηκευμένες διαδικασίες ή αλλιώς stored procedures που χρησιμοποιήθηκαν στην εφαρμογή, είναι διαδικασίες οι οποίες αποθηκεύονται μέσα στον SQL SERVER και οποιαδήποτε στιγμή χρειαστεί να εκτελεστεί κάποια απ' αυτές, καλείται από την εφαρμογή και εκτελείται.

Ενδεικτικά μερικά από τα stored procedure που χρησιμοποιήθηκαν είναι:

- **CREATE PROCEDURE** dbo.item_ins
(

 @TYPE_ID tinyint,
 @CATEGORY_ID char(5),
 @ISLENDABLE bit,
 @TAG_ID varchar(50),
 @TITLE nvarchar(50),
 @AUTHOR nvarchar(50),
 @ISBN nvarchar(50),
 @EDITION tinyint,
 @PUBLISHER nvarchar(50),
 @PUBL_DATE datetime,
 @INFO nvarchar(50)

)
AS
SET NOCOUNT OFF;

INSERT INTO tblitem(TYPE_ID,CATEGORY_ID, ISLENDABLE,
TAG_ID, TITLE,AUTHOR, ISBN, EDITION, PUBLISHER, PUBL_DATE,
INFO) **VALUES** (@TYPE_ID,@CATEGORY_ID, @ISLENDABLE,
@TAG_ID, @TITLE, @AUTHOR, @ISBN, @EDITION, @PUBLISHER,
@PUBL_DATE, @INFO);

SELECT ITEM_ID,TYPE_ID,CATEGORY_ID, ISLENDABLE, TAG_ID,
TITLE, AUTHOR, ISBN, EDITION,PUBLISHER, PUBL_DATE,INFO
FROM tblitem **WHERE** (TAG_ID= @TAG_ID)
GO

- **CREATE PROCEDURE** dbo.student_del
 (
 @STUDENT_ID int
)
AS
SET NOCOUNT OFF;

DELETE FROM tblstudent **WHERE** (STUDENT_ID=@STUDENT_ID)
GO

- **CREATE PROCEDURE** dbo.lend_upd
 (
 @STUDENT_ID int,
 @ITEM_ID int,
 @START_DATE datetime,
 @RETURN_DATE datetime
)
AS
SET NOCOUNT OFF;

UPDATE tblstudentitem **SET** STUDENT_ID = @STUDENT_ID, ITEM_ID
 = @ITEM_ID, START_DATE=@START_DATE, RETURN_DATE =
 @RETURN_DATE **WHERE** (STUDENT_ID = @STUDENT_ID AND
 ITEM_ID=@ITEM_ID);

SELECT STUDENT_ID, ITEM_ID, START_DATE, RETURN_DATE
FROM tblstudentitem **WHERE** (STUDENT_ID = @STUDENT_ID AND
 ITEM_ID=@ITEM_ID)
GO

Τα stored procedure προσφέρουν :

- **ΕΥΚΟΛΙΑ** : Όταν ένα stored procedure τροποποιηθεί στον SQL, αν το όνομά του και οι μεταβλητές, που χρειάζεται, παραμείνουν ίδια δεν είναι αναγκαίο να επέμβει κάποιος στον κώδικα VB.NET
- **TAXYTHTA ΣΤΗΝ ΕΦΑΡΜΟΓΗ** : Τα stored procedure εκτελούνται γρήγορα διότι δεν χρειάζεται να περάσουν οι παράμετροι στις υπορουτίνες κάθε φορά που καλούνται. Αυτό γίνεται μόνο την πρώτη φορά που καλούνται τα stored procedure.

- **ΚΑΛΥΤΕΡΗ ΧΡΗΣΗ ΤΩΝ ΠΟΡΩΝ :** Τα stored procedure εκτελούνται στον SQL, δεν επιβαρύνουν το σύστημα του χρήστη και εκμεταλλεύονται καλύτερα το δίκτυο, αφού ο πελάτης δεν στέλνει μέσω δικτύου μεγάλα ερωτήματα, αλλά μόνο το όνομα του stored procedure με τις απαραίτητες παραμέτρους. Ο server λαμβάνει το μήνυμα του πελάτη, εκτελεί το stored procedure και το αποτέλεσμα αποστέλλεται στον πελάτη. Τέλος τα stored procedure εκμεταλλεύονται την ταχύτητα του SQL server και γενικά το υλικό (Hardware) του.
- **ΑΞΙΟΠΙΣΤΙΑ - ΣΤΑΘΕΡΟΤΗΤΑ - ΑΝΕΞΑΡΤΗΣΙΑ :** Όλοι οι χρήστες της Βάσης Δεδομένων μπορούν να χρησιμοποιούν τα ίδια stored procedure χωρίς να παίρνουν μεγάλο ρίσκο λάθους και είναι ευκολότερη η διόρθωση των διεργασιών αυτών.

4

Ανάλυση των Διαδικασιών VB.NET

Κεφάλαιο 4 – Ανάλυση Διαδικασιών σε VB.NET

4.1 Επικύρωση για τις Web φόρμες

Στο VB.NET, μπορείτε να προσθέσετε στοιχεία ελέγχου επικύρωσης σε όλα σχεδόν τα στοιχεία ελέγχου δεδομένων εισόδου της σελίδας, που είναι είτε HTML στοιχεία ελέγχου διακομιστή, είτε στοιχεία ελέγχου διακομιστή μιας Web φόρμας. Μπορούν να ορισθούν περισσότερα από ένα είδη επικύρωσης σε ένα συγκεκριμένο στοιχείο ελέγχου εισόδου, κάνοντας δυνατή την επικύρωση με διάφορα κριτήρια. Όταν εκτελείται η σελίδα με τα στοιχεία ελέγχου επικύρωσης, οι τιμές των στοιχείων ελέγχου εισόδου που επηρεάζονται από τα στοιχεία ελέγχου επικύρωσης υπόκεινται σε επεξεργασία σε σχέση με την αντίστοιχη λογική που τα διέπει, όπως αναφέρεται στα στοιχεία ελέγχου επικύρωσης. Μετά τους υπολογισμούς, οι ιδιότητες των στοιχείων ελέγχου επικύρωσης παίρνουν την τιμή True ή False, ανάλογα με το αποτέλεσμα. Μόλις γίνει η επεξεργασία όλων των στοιχείων ελέγχου επικύρωσης, η ίδια η σελίδα ορίζει μια ιδιότητα με βάση τις τιμές των ιδιοτήτων επικύρωσης των στοιχείων ελέγχου. Αν κάποιο από τα στοιχεία ελέγχου δεν περάσει την επικύρωση, η ιδιότητα της σελίδας δίνει αυτόματα την τιμή False στην ιδιότητα επικύρωσης. Όταν δημιουργείτε τον κώδικα επικύρωσης, η ιδιότητα της σελίδας θα είναι μια σημαντική ιδιότητα που πρέπει να ελέγχετε για να είστε σίγουροι ότι όλα είναι σωστά. Αφού ελεγχθεί η ιδιότητα της σελίδας και δείτε ότι η επικύρωση απέτυχε, η σελίδα συνήθως επιστρέφει στο χρήστη και θα εμφανιστεί κάποιο μήνυμα σφάλματος, αν έχει ορισθεί ένα τέτοιο μήνυμα στα στοιχεία ελέγχου που απέτυχαν να περάσουν την επικύρωση.

Η VB.NET έχει έξι είδη στοιχείων ελέγχου επικύρωσης: το RequiredFieldValidator, RegularExpressionValidator, CompareValidator, RangeValidator, CustomValidator και το ValidationSummary. Όλα αυτά τα στοιχεία ελέγχου έχουν ένα κοινό σύνολο από ιδιότητες και μεθόδους, οι περισσότερες από τις οποίες, κληρονομούνται από την κλάση BaseValidator και από την κλάση WebControl που βρίσκονται στο χώρο ονομάτων System.UI.WebControls και την κλάση Control που μπορεί να βρεθεί στο χώρο ονομάτων System.Web.UI. Μια εξαίρεση σε αυτόν τον κανόνα είναι η ιδιότητα Text, η οποία κληρονομείται από την κλάση Label.

4.1.1 RequiredFieldValidator

Το στοιχείο ελέγχου RequiredFieldValidator ελέγχει αν ο χρήστης έδωσε δεδομένα ή αν επέλεξε μια τιμή στο στοιχείο ελέγχου εισόδου. Μόλις γίνει η αντιστοίχιση με ένα στοιχείο ελέγχου εισόδου, αυτό το στοιχείο ελέγχου επικύρωσης διασφαλίζει ότι εισάγεται μια τιμή στο πεδίο του στοιχείου ελέγχου. Μαζί με τις κοινές ιδιότητες επικύρωσης των στοιχείων ελέγχου, το RequiredFieldValidator έχει μια μοναδική ιδιότητα: την ιδιότητα InitialValue. Αυτή καθορίζει μια αρχική τιμή για το στοιχείο ελέγχου επικύρωσης. Αν η τιμή στο στοιχείο ελέγχου είναι ίση με την τιμή της ιδιότητας InitialValue, η επικύρωση αποτυγχάνει.

4.1.2 RegularExpressionValidator

Το στοιχείο ελέγχου RegularExpressionValidator ελέγχει αν συμπίπτει ένα στοιχείο με ένα μοτίβο που καθορίζεται από μια κανονική παράσταση (regular expression). Μια *κανονική παράσταση* είναι ένα μοτίβο κειμένου, το οποίο αποτελείται από συνηθισμένους και ειδικούς χαρακτήρες, που είναι γνωστοί (*μετα-χαρακτήρες* ή *metacharacter*). Το μοτίβο χρησιμεύει ως πρότυπο, καθώς περιγράφει μία ή περισσότερες συμβολοσειρές τις οποίες πρέπει να βρει όταν κάνει αναζήτηση σε ένα κείμενο.

Αυτό το είδος της επικύρωσης σας επιτρέπει να ψάχνετε για προβλεπόμενες ακολουθίες χαρακτήρων, όπως εκείνες που χρησιμοποιούνται για τις διευθύνσεις ηλεκτρονικού ταχυδρομείου, για αριθμούς τηλεφώνου κτλ. Υπάρχει μια μοναδική ιδιότητα που σχετίζεται με τα στοιχεία ελέγχου RegularExpressionValidator: η ιδιότητα ValidationExpression. Προσδιορίζει την κανονική παράσταση που ορίζει τα κριτήρια επικύρωσης.

4.2 Ασφάλεια

Η αρχιτεκτονική ασφαλείας του VB.NET αποτελείται από δύο μέρη. Το πρώτο στοιχείο είναι η ενσωμάτωσή της στο IIS. Το δεύτερο στοιχείο βρίσκεται στο βασικό κατάλογο της εφαρμογής VB.NET, σε ένα αρχείο που ονομάζεται Web.config. Αυτό το αρχείο περιέχει διάφορες ρυθμίσεις διαμόρφωσης, όπως ρυθμίσεις ασφαλείας, για την εφαρμογή μας.

4.2.1 Ασφάλεια του IIS (Internet Information Server)

Το πρώτο βήμα για την ασφάλεια της εφαρμογής μας ξεκινά με το Web διακομιστή οποίος φιλοξενεί την τοποθεσία μας. Στις περισσότερες περιπτώσεις, αυτός ο Web διακομιστής είναι ο Internet Information Server 5.0+, ο οποίος εγκαθίσταται αυτόματα με τα Windows 2000+ Server και μπορεί να προστεθεί με μη αυτόματο τρόπο στα Windows 2000+ Professional από τα Windows Components. Το IIS μας δίνει αρκετές μεθόδους για να ασφαλίσουμε τη Web τοποθεσία μας. Αυτές οι επιλογές βρίσκονται στην καρτέλα Directory Security (ασφάλεια καταλόγου) στο παράθυρο Properties (ιδιότητες) της τοποθεσίας μας.

4.2.2 Το αρχείο Web.config

Το web.config που είναι αποθηκευμένο σε ένα xml αρχείο στο κατάλογο ρίζα περιγράφει διάφορες ρυθμίσεις διαμόρφωσης. Υπάρχει ένας κόμβος σε αυτό το αρχείο που ονομάζεται <security> και περιέχει πληροφορίες διαμόρφωσης ασφαλείας για την VB.NET εφαρμογή μας. Ακόμα σε αυτό το αρχείο υπάρχουν και διάφορες άλλες ιδιότητες που ανάλογα το πως θα ορισθούν επιτρέπουν ή απαγορεύουν σε διάφορους χρήστες την πρόσβαση στον πόρο που ζητούν.

Στο αρχείο Web.config εισάγαμε το connection string (απαραίτητο για την σύνδεση με τη βάση) έτσι ώστε να χρησιμοποιείται από όλες τις φόρμες με ευκολία και να μην φαίνεται στον HTML κώδικα. Ο κώδικας που προσθέτουμε στο Web.config για να ορίσουμε το connection string είναι ο παρακάτω:

```
<appSettings>
  <add key="CONNECTION_STRING" value="workstation id=SUZETTE-
19EDBAA;packet size=4096;integrated security=SSPI;data source=SUZETTE-
19EDBAA;persist security info=False;initial catalog=eLibrary" />
  <add key="PAGE_EXPIRED_STRING" value="Page has expired. Please log in
again" />
</appSettings>
```

4.3 Sessions

Ένα από τα γνωστά μειονεκτήματα του πρωτοκόλλου http είναι ότι δεν διατηρείται η σύνδεση μεταξύ browser και server κατά όλη την διάρκεια της επικοινωνίας αλλά διακόπτεται κάθε φορά που ολοκληρώνεται η αποστολή δεδομένων από την μια μεριά στην άλλη. Έτσι δεν είναι δυνατό να διατηρηθούν και να μεταβιβαστούν πληροφορίες μεταξύ δύο διαφορετικών sites.

Μια λύση για το παραπάνω πρόβλημα αποτελούν τα cookies. Τα cookies έχουν όμως το μειονέκτημα ότι αποθηκεύονται στον σκληρό δίσκο του client και έτσι δεν προτιμούνται για διατήρηση μεγάλου όγκου πληροφοριών. Η vb.net προτείνει μία διαφορετική λύση χρησιμοποιώντας τα αντικείμενα Session.

Έτσι σε δεδομένα που έχουν αποθηκευτεί με το αντικείμενο Session έχουν προσπέλαση μόνο εκείνοι οι clients που το δημιούργησαν. Τα δεδομένα αυτά καταστρέφονται όταν ο χρήστης τερματίσει την σύνδεση του με τον server.

4.4 Σύνδεση με Βάση Δεδομένων

Όπως προαναφέραμε απαραίτητη είναι η δημιουργία ενός connection string για την σύνδεση με τη βάση. Παρακάτω παραθέτουμε τις εντολές που χρησιμοποιήσαμε για την σύνδεση με την βάση μας.

```
Dim con As New SqlConnection
Dim cmd As New SqlCommand
```

```
con.ConnectionString =
System.Configuration.ConfigurationSettings.AppSettings("CONNECTION_STRING")
cmd.CommandType = CommandType.StoredProcedure
cmd.CommandText = "....."
cmd.Connection = con
With cmd
    .Parameters.Add("@.....", studentID.Trim)
    .....
End With

Try
    con.Open()
    cmd.ExecuteNonQuery()
Catch ex As Exception
    Session("MESSAGE") = ex.Message.ToString
    Session("PAGE") = "StudentForm.aspx"
    Response.Redirect("info.aspx")
Finally
    cmd.Dispose()
    con.Close()
End Try
```

Στον παραπάνω κώδικα δηλώνουμε ένα SqlConnection του οποίου το connection string το διαβάζουμε από το αρχείο web.config.

Στο επόμενο παράδειγμα βλέπουμε τον τρόπο με τον οποίο εκτελούμε ένα stored procedure, του δίνουμε τις επιθυμητές παραμέτρους και με την βοήθεια ενός datareader ο οποίος διαβάζει μέσα στην βάση και ταυτόχρονα ψάχνει να βρει εάν υπάρχει ο σπουδαστής στη βάση δεδομένων μας.

```
Private Function StudentExists() As Boolean
    Dim con As New SqlClient.SqlConnection
    Dim cmd As New SqlClient.SqlCommand
    Dim dr As SqlClient.SqlDataReader

    con.ConnectionString =
System.Configuration.ConfigurationSettings.AppSettings("CONNECTION_STRING")
    cmd.CommandType = CommandType.StoredProcedure
    cmd.CommandText = "student_sel"
    cmd.Connection = con
    With cmd
        .Parameters.Add("@STUDENT_ID", GetStudentID())
    End With
    Try
        con.Open()
        dr = cmd.ExecuteReader(CommandBehavior.SequentialAccess)
        If dr.Read Then
            StudentExists = True
        Else
            StudentExists = False
        End If
    Catch ex As Exception
        Session("MESSAGE") = ex.Message.ToString
        Session("PAGE") = "StudentForm.aspx"
        Response.Redirect("info.aspx")
    Finally
        dr.Close()
        cmd.Dispose()
        con.Close()
    End Try
End Function
```

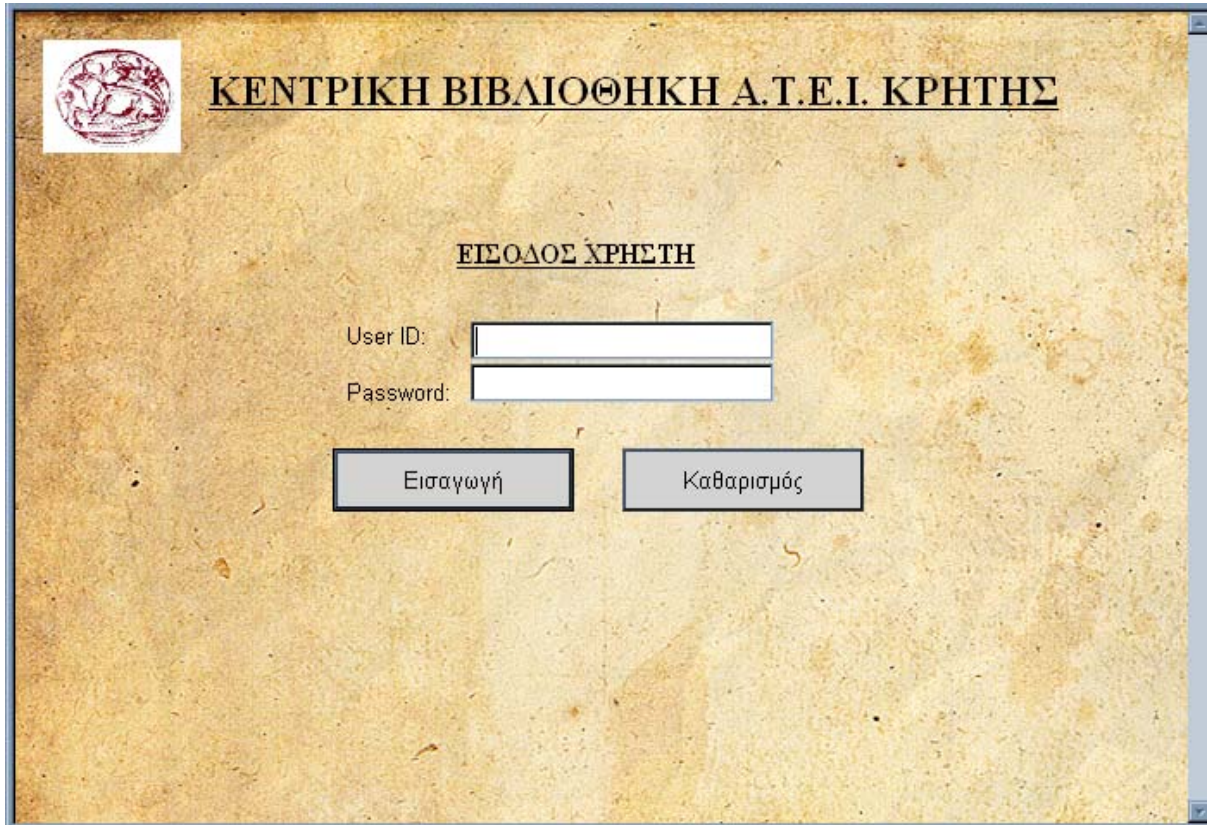
Στον παραπάνω κώδικα πρέπει να παρατηρήσουμε την μέθοδο ExecuteReader() η οποία στέλνει το commandtext (cmd) στο connection (con) και δημιουργεί ένα SqlDataReader. Το αντικείμενο SqlDataReader χρησιμοποιείται για να ελέγξει εάν μέσα στην βάση υπάρχει σπουδαστής με το ίδιο STUDENT_ID.

5

Υλοποίηση

Κεφάλαιο 5 – Υλοποίηση της εφαρμογής

5.1 Η φόρμα *logon.aspx*



The image shows a web browser window displaying a login page. In the top left corner, there is a circular logo with a red and white design. To the right of the logo, the text "ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ" is displayed in a bold, black, serif font. Below this, the text "ΕΙΣΟΔΟΣ ΧΡΗΣΤΗ" is centered. Underneath, there are two input fields: "User ID:" followed by a white text box, and "Password:" followed by a white text box. At the bottom, there are two grey buttons with black text: "Εισαγωγή" on the left and "Καθαρισμός" on the right. The background of the page is a textured, light brown color.

Σχήμα 5.1 *logon.aspx*

Η παραπάνω φόρμα (Σχήμα 5.1) είναι η πρώτη φόρμα της εφαρμογής μας. Η φόρμα αυτή αποτελεί ένα στοιχείο ελέγχου όπου ο χρήστης καλείται εισάγει τον ΚΩΔΙΚΟ ΧΡΗΣΤΗ (στην προκειμένη περίπτωση είναι το e-mail του) και το PASSWORD του ώστε να μπορέσει να περιηγηθεί στα υπόλοιπα τμήματα της εφαρμογής μας.

5.2 Η φόρμα *IntroForm.aspx*



ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ

ΕΠΙΛΟΓΕΣ

Σπουδαστές

Βιβλία / DVD

Αποσύνδεση

Σχήμα 5.2 *IntroForm.aspx*

Η παραπάνω φόρμα (Σχήμα 5.2) είναι η δεύτερη φόρμα της εφαρμογής μας αφού ο χρήστης περάσει με επιτυχία τον έλεγχο που αναφέραμε παραπάνω. Σε αυτήν ο χρήστης, που στην προκειμένη περίπτωση είναι ο υπάλληλος της βιβλιοθήκης, έχει την δυνατότητα να επιλέξει ανάμεσα σε 3 ενέργειες. Η πρώτη «ΣΠΟΥΔΑΣΤΕΣ» μας μεταφέρει στην φόρμα των σπουδαστών που θα δούμε παρακάτω, η δεύτερη «ΒΙΒΛΙΑ / DVD» στην φόρμα με τα αντικείμενα δανεισμού της βιβλιοθήκης και η τρίτη «ΑΠΟΣΥΝΔΕΣΗ» μας επιστρέφει στην φόρμα του Σχήματος 5.1, η οποία είναι και η αρχική μας.

5.3 Η φόρμα StudentForm.aspx

ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ

ΦΟΡΜΑ ΣΠΟΥΔΑΣΤΗ

Νέα Εισαγωγή Αποθήκευση Έξοδος

Κωδ. Σπουδαστή: Εύρεση Αρ. Μητρώου:

Όνομα: Επίθετο:

Τμήμα: Σχολή:

Έτος Εισαγωγής: Εξάμηνο:

Διεύθυνση: Περιοχή:

E-mail: Τηλέφωνο:

Κλειδωμένος Λογαριασμός: Όριο Δανεισμού:

Δανεισμός

Σχήμα 5.3.1 StudentForm.aspx

Η παραπάνω φόρμα (Σχήμα 5.3a) είναι η φόρμα που αντιστοιχεί στην επιλογή «ΣΠΟΥΔΑΣΤΕΣ» της φόρμας του Σχήματος 5.2. Σε αυτήν την φόρμα υπάρχουν πολλές ενέργειες τις οποίες θα αναλύσουμε παρακάτω.

Αρχικά, η εισαγωγή νέων σπουδαστών με σκοπό να καταχωρηθούν στην Β.Δ. της βιβλιοθήκης και να αποκτήσουν τα ανάλογα δικαιώματα, γίνεται απλά με την συμπλήρωση των πεδίων και την αποθήκευσή τους με το αντίστοιχο κουμπί.

Επίσης στην ίδια φόρμα μπορεί να γίνει μια εύκολη εύρεση του σπουδαστή, συμπληρώνοντας το πεδίο με τον Κωδικό Σπουδαστή, όπου παρουσιάζονται τα στοιχεία του στα κατάλληλα πεδία και μπορεί να γίνει μετατροπή αυτών, ξανά με την επιλογή της αποθήκευσης.

Με την επιλογή «ΕΥΡΕΣΗ» εκτός από την αυτόματη συμπλήρωση των πεδίων, γίνεται και η παρουσίαση της κατάστασης δανεισμού του εκάστοτε σπουδαστή. Αν ο σπουδαστής έχει καθυστερήσει ένα συγκεκριμένο αντικείμενο της βιβλιοθήκης τότε ο πίνακας της κατάστασης δανεισμού γίνεται κόκκινος, ενώ δεν επιτρέπεται ο δανεισμός παρά μόνο η επιστροφή των δανεισμένων αντικειμένων.

Επίσης, στη φόρμα (Σχήμα 5.3b) γίνεται επιλογή του αντικειμένου το οποίο δανείστηκε ο σπουδαστής. Με την ενέργεια αυτή παρουσιάζονται αυτόματα δύο ακόμα επιλογές, η «ΕΠΙΣΤΡΟΦΗ» και η «ΠΑΡΑΤΑΣΗ» του αντικειμένου. Στην πρώτη επιλογή γίνεται επιστροφή του στοιχείου ενώ στην δεύτερη γίνεται επαναδανεισμός για ένα χρονικό διάστημα, το οποίο επιτρέπει η εκάστοτε βιβλιοθήκη.

ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ

ΦΟΡΜΑ ΣΠΟΥΔΑΣΤΗ

Κωδ. Σπουδαστή: 15002810
 Αρ. Μητρώου: 1000
 Όνομα: Στέλιος
 Τμήμα: Λογιστικής
 Έτος Εισαγωγής: 2007
 Διεύθυνση: Χαλκοκονδύλη 5
 E-mail: "/>

Κλειδωμένος Λογαριασμός:
 Όριο Δανεισμού: 4

ΚΩΔΙΚΟΣ	ΤΙΤΛΟΣ	ΗΜΕΡ. ΔΑΝΕΙΣΜΟΥ
198115	Ο ΑΛΧΗΜΙΣΤΗΣ	25/10/2007 12:00:00 πμ

Σχήμα 5.3.2 StudentForm.aspx

5.4 Η φόρμα *LendForm.aspx*

The screenshot shows a web form titled "ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ" (Central Library of A.T.E.I. Crete). The form is titled "ΔΑΝΕΙΣΜΟΣ" (Loan) and contains the following fields and buttons:

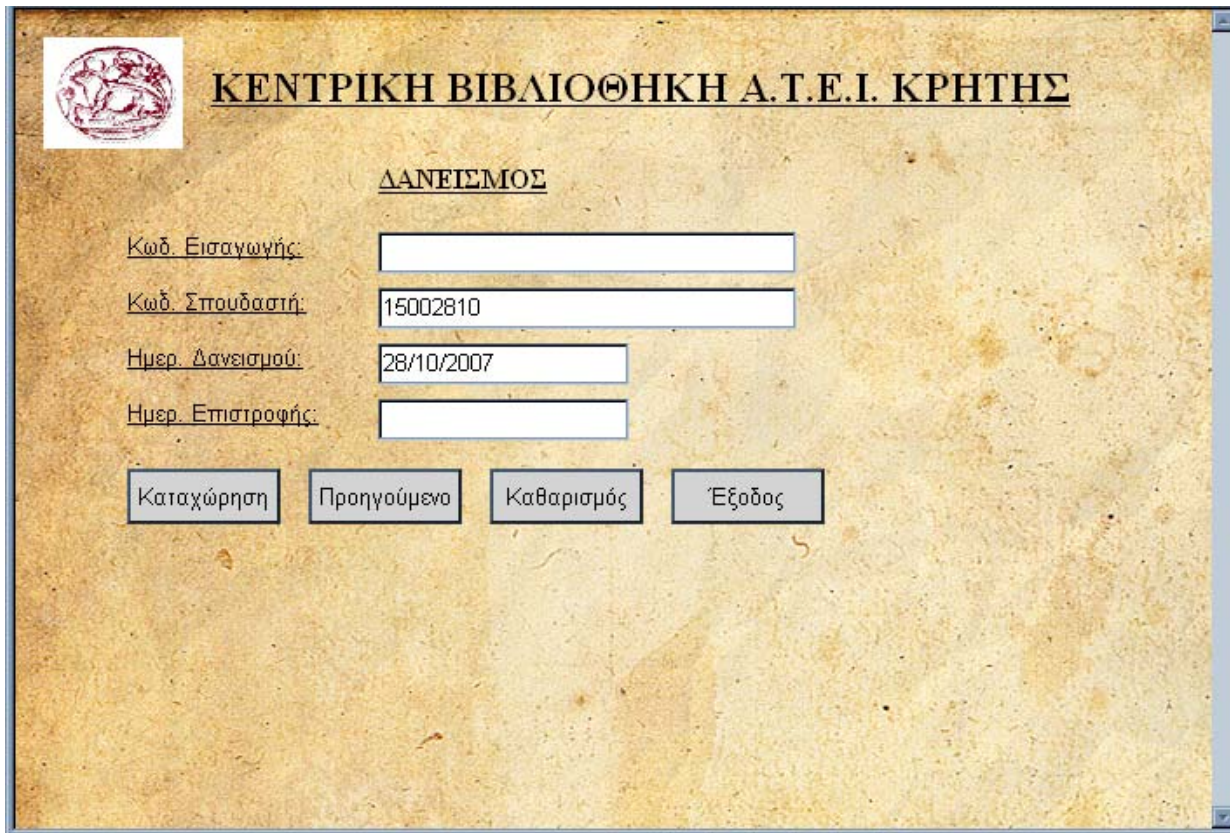
- Κωδ. Εισαγωγής: [Text input field]
- Κωδ. Σπουδαστή: [Text input field]
- Ημερ. Δανεισμού: [Text input field]
- Ημερ. Επιστροφής: [Text input field]
- Buttons: Καταχώρηση, Προηγούμενο, Καθαρισμός, Έξοδος

Σχήμα 5.4.1 *LendForm.aspx*

Η παραπάνω φόρμα (Σχήμα 5.4) είναι η τέταρτη φόρμα της web εφαρμογής και σε αυτήν εμφανίζεται η ενέργεια του δανεισμού, της επιστροφής και της παράτασης των αντικειμένων της βιβλιοθήκης. Συμπληρώνοντας τα κατάλληλα πεδία ο σπουδαστής μπορεί να δανειστεί, να επιστρέψει και να παρατείνει τον δανεισμό ενός αντικείμενου της βιβλιοθήκης.

Στην παραπάνω φόρμα τα πεδία είναι κενά, όμως κατά την επιλογή του αντικείμενου στον πίνακα δανεισμού του εκάστοτε σπουδαστή, και φυσικά την επιλογή κάποιας από τις ενέργειες που αναφέραμε παραπάνω δηλ. δανεισμός, επιστροφή, παράταση, κάποια από τα πεδία συμπληρώνονται αυτόματα. Ορισμένα παραδείγματα φαίνονται στα Σχήματα 5.4b, 5.4c, 5.4d.

ΔΑΝΕΙΣΜΟΣ



The screenshot shows a web form titled "ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ" (Central Library of A.T.E.I. Crete) with a logo on the left. The form is titled "ΔΑΝΕΙΣΜΟΣ" (Borrowing) and contains the following fields and buttons:

Κωδ. Εισαγωγής:	<input type="text"/>
Κωδ. Σπουδαστή:	<input type="text" value="15002810"/>
Ημερ. Δανεισμού:	<input type="text" value="28/10/2007"/>
Ημερ. Επιστροφής:	<input type="text"/>


Below the form are four buttons: "Καταχώρηση", "Προηγούμενο", "Καθαρισμός", and "Έξοδος".

Σχήμα 5.4.2 LendForm.aspx

Παρατηρούμε ότι αυτόματα συμπληρώνονται τα πεδία με τον Κωδικό Σπουδαστή και την Ημερομηνία Δανεισμού.

ΕΠΙΣΤΡΟΦΗ



 **ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ**

ΕΠΙΣΤΡΟΦΗ

Κωδ. Εισαγωγής:

Κωδ. Σπουδαστή:

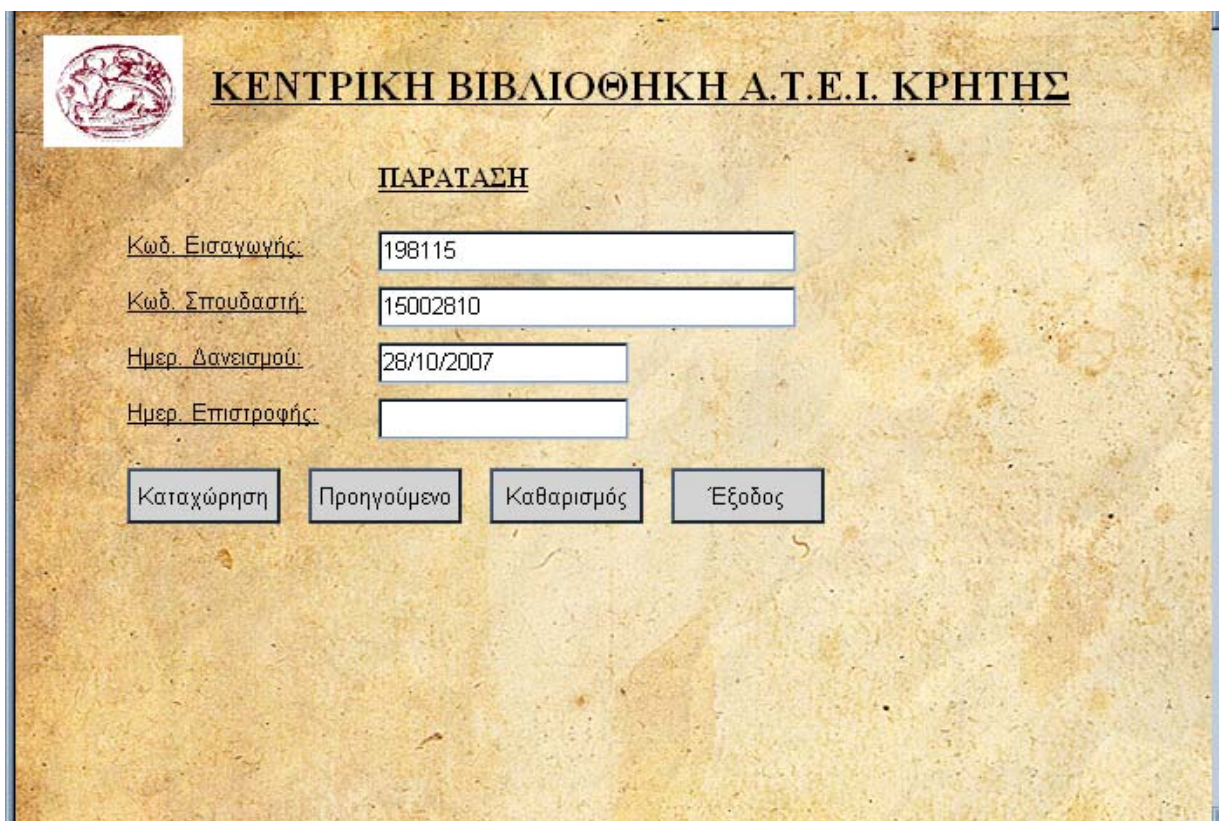
Ημερ. Δανεισμού:


Ημερ. Επιστροφής:

Σχήμα 5.4.3 LendForm.aspx

Σε αυτή την εκδοχή, όπου γίνεται η επιστροφή του εκάστοτε αντικειμένου όλα τα πεδία είναι συμπληρωμένα.

ΠΑΡΑΤΑΣΗ



 **ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ**

ΠΑΡΑΤΑΣΗ

Κωδ. Εισαγωγής:

Κωδ. Σπουδαστή:

Ημερ. Δανεισμού:

Ημερ. Επιστροφής:

Σχήμα 5.4.4 LendForm.aspx

Τέλος, σε αυτή την περίπτωση παρατηρούμε πως μόνο τα τρία πρώτα πεδία είναι συμπληρωμένα, με τον Κωδικό Εισαγωγής, τον Κωδικό Σπουδαστή και την Ημερομηνία Δανεισμού.

5.5 Η φόρμα BookForm.aspx

The screenshot shows a web form titled "ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ" (Central Library of Crete). Below the title is the subtitle "ΦΟΡΜΑ ΒΙΒΛΙΩΝ/DVD" (Book/DVD Form). At the top, there are three buttons: "Νέα Εισαγωγή" (New Entry), "Αποθήκευση" (Save), and "Έξοδος" (Exit). The form contains several input fields and dropdown menus:

- Τύπος Εισαγωγής :** A dropdown menu with "Βιβλίο" selected.
- Κωδ. Εισαγωγής :** A text input field.
- Κατηγορία :** A dropdown menu with "Οικονομική Ιστορία" selected.
- Τίτλος :** A text input field.
- Συγγραφέας :** A text input field.
- ISBN :** A text input field.
- Πληροφορίες :** A large text area for additional information.
- Δανειστικό :** A dropdown menu with "Όχι" selected.
- Εύρεση :** A button next to the "Κωδ. Ταξινόμησης" field.
- Κωδ. Ταξινόμησης :** A text input field.
- Έκδοση :** A text input field.
- Εκδότης :** A text input field.
- Ημερ. Έκδοσης :** A text input field.

Σχήμα 5.5.1 BookForm.aspx

Η παραπάνω φόρμα (Σχήμα 5.5) είναι η πέμπτη φόρμα της εφαρμογής μας, η οποία προκύπτει από την επιλογή «Βιβλία / DVD» του Σχήματος 5.2. Σε αυτήν ο χρήστης της εφαρμογής μας, μπορεί να εισάγει νέα βιβλία ή DVD, απλά συμπληρώνοντας τα πεδία και επιλέγοντας την αποθήκευσή τους.

Όπως και στην φόρμα του Σχήματος 5.3b συμπληρώνοντας το πεδίο με τον Κωδικό Εισαγωγής και επιλέγοντας την «ΕΥΡΕΣΗ »μπορούμε να εντοπίσουμε το συγκεκριμένο αντικείμενο και να τροποποιήσουμε τα δεδομένα του. Επίσης με αυτή την επιλογή φαίνεται αν το συγκεκριμένο αντικείμενο είναι δανεισμένο ή όχι.

ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ

ΦΟΡΜΑ ΒΙΒΛΙΩΝ/DVD

Νέα Εισαγωγή Αποθήκευση Έξοδος Ιστορικό

Τύπος Εισαγωγής :

Κωδ. Εισαγωγής : Κωδ. Ταξινόμησης :

Κατηγορία : Εκδοση :

Τίτλος : Εκδότης :

Συγγραφέας : Ημερ. Έκδοσης :

ISBN :

Πληροφορίες :

Δανειστικό :

Κατάσταση : **ΔΑΝΕΙΣΜΕΝΟ**

Σχήμα 5.5.2 BookForm.aspx

Επίσης στην παραπάνω φόρμα έχουμε την δυνατότητα να δούμε το ιστορικό του κάθε βιβλίου /DVD, η οποία επιλογή μας μεταφέρει σε μια άλλη φόρμα την History.aspx στην οποία μπορούμε να δούμε την κίνηση του κάθε στοιχείου και την τρέχουσα κατάσταση δανεισμού.

ΚΩΔΙΚΟΣ ΒΙΒΛΙΟΥ/DVD	ΚΩΔΙΚΟΣ ΦΟΙΤΗΤΗ	ΗΜΕΡ. ΔΑΝΕΙΣΜΟΥ	ΗΜΕΡ. ΕΠΕΤΡΟΦΗΣ
1210757	15000994	25/10/2007 12:00:00 πμ	

Σχήμα 5.5.3 History.aspx

Αξίζει να αναφέρουμε πως και την BookForm αλλά και στην StudentForm (εν μέρει), ο χρήστης έχει την δυνατότητα να διαγράψει όποια εισαγωγή επιθυμεί, η οποία όμως συμβαίνει κάτω από συγκεκριμένες προϋποθέσεις.

Στην φόρμα BookForm η διαγραφή ενός στοιχείου είναι δυνατή μόνο όταν το βιβλίο έχει επιστραφεί ενώ στην φόρμα StudentForm η επιλογή της διαγραφής γίνεται με την «ΕΚΤΥΠΩΣΗ ΒΕΒΑΙΩΣΗΣ ΔΑΝΕΙΣΜΟΥ» η οποία εκτυπώνεται όταν ο σπουδαστής ολοκληρώσει τις σπουδές του και έχει επιστρέψει όλα τα στοιχεία. Κατά την ενέργεια αυτή ο λογαριασμός αυτόματα κλειδώνεται και δεν επιτρέπεται πια ο δανεισμός αλλά δεν διαγράφεται.

Παρακάτω βλέπουμε την φόρμα Report όπου αυτόματα συμπληρώνονται τα στοιχεία του σπουδαστή.



ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ

Ηράκλειο Κρήτης, 31/10/2007

Βεβαιώνεται ότι ο/η φοιτητής/τρια

Δημοσθένης Παπαδημητρίου με αριθμό μητρώου(Α.Μ.) 1800 , του Τμήματος Εφαρμοσμένης Πληροφορικής και Πολυμέσων, δεν έχει καμία οφειλή προς την Βιβλιοθήκη του Α.Τ.Ε.Ι. Κρήτης και έχει επιστρέψει την προσωπική κάρτα δανεισμού.

Ο Υπεύθυνος της Βιβλιοθήκης

[Προηγούμενο](#)

Σχήμα 5.5.4 Report.aspx

Ενώ στην παρακάτω φόρμα βλέπουμε ότι ο λογαριασμός του σπουδαστή είναι κλειδωμένος, το όριο δανεισμού έχει μηδενιστεί και εμφανίζεται ένα μήνυμα όπου ειδοποιεί τον χρήστη ότι η ενέργεια του δανεισμού δεν μπορεί να πραγματοποιηθεί.

ΚΕΝΤΡΙΚΗ ΒΙΒΛΙΟΘΗΚΗ Α.Τ.Ε.Ι. ΚΡΗΤΗΣ

ΦΟΡΜΑ ΣΠΟΥΔΑΣΤΗ

Νέα Εισαγωγή Αποθήκευση Έξοδος Εκτύπωση Βεβαίωσης Δανεισμού

Κωδ. Σπουδαστή: 15002814 Εύρεση Αρ. Μητρώου: 1800

Όνομα: Δημοσθένης Επίθετο: Παπαδημητρίου

Τμήμα: Εφαρμοσμένης Πληροφορικής και Σχολή: ΣΤΕΦ

Ετος Εισαγωγής: 2002 Εξάμηνο: Χειμερινό

Διεύθυνση: Στεφάνου 44 Περιοχή: Ηράκλειο Κρήτης

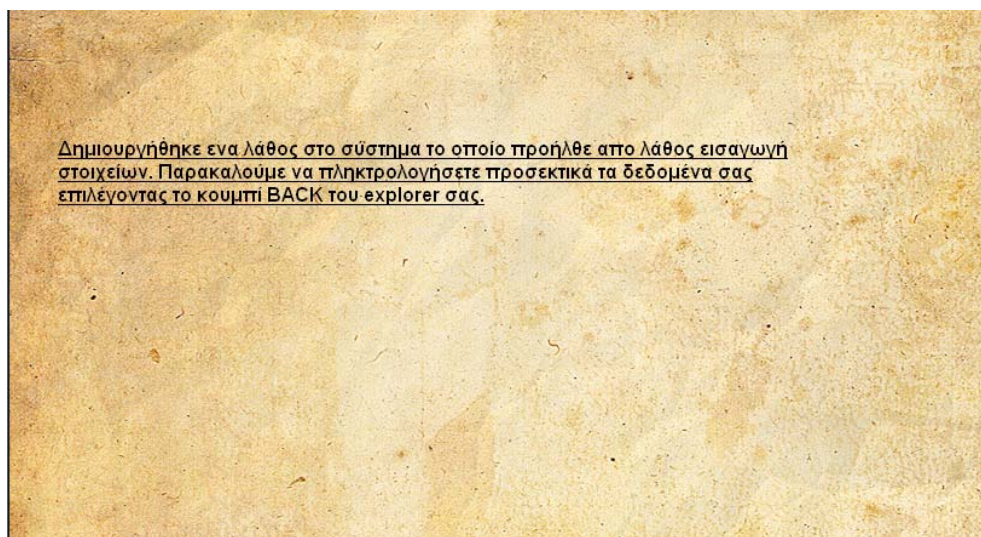
E-mail: dim@yahoo.gr Τηλέφωνο: 2810233333

Κλειδωμένος Λογαριασμός: Όριο Δανεισμού:

Δεν μπορείτε να πραγματοποιήσετε άλλο την ενέργεια του δανεισμού.

Σχήμα 5.5.5 StudentForm.aspx

5.6 Η φόρμα Info.aspx



Σχήμα 5.6 Info.aspx

Η φόρμα Info.aspx (Σχήμα 5.6) είναι η φόρμα στην οποία, εμφανίζονται τα μηνύματα λάθους, σε περίπτωση που ο χρήστης κάνει έναν λάθος χειρισμό ή υπάρχει κάποιο πρόβλημα στον κώδικα της εφαρμογής, το οποίο θα πρέπει να επιλύσει ο administrator της βιβλιοθήκης.

6

Βιβλιογραφία

Κεφάλαιο 6 - Βιβλιογραφία

- Visual Basic .Net “Εισαγωγή στο .NET και τα Web Services”, δεύτερη έκδοση , Deitel & Nieto , Μ. Γκιούρδας , 2003
- Πλήρες Εγχειρίδιο της Visual Basic .NET , Πετρούτσος , Μ. Γκιούρδας , 2005
- SQL Server 2000 Web Application Developer's Guide , Craig Utley, Berkley 2001.
- Rebecca M. Riordan, *Προγραμματισμός στο SQL Server 2000 Βήμα Βήμο* (μτφρ. Τάκης Αλβας), Αθήνα 2001.
- Beginning VB.NET, Gary,Cornell, Matthew,MacDonald, Apress, 2004.
- VB.NET Applied, Bradley,Wist, Bob,Flanders, Pearson Professional Education, 2003.
- Professional VB.NET, Jonathan,Crossland, Bill,Evjén, Billy,Hollis, John Wiley and Sons Ltd, 2004.
- SQL Server 2000, Dusan,Petkovic, McGraw-Hill Education – Europe, 2000
- Οδηγός του SQL Server 2000, PETKOVIC, Μ. Γκιούρδας , 2001
- SQL Server 2000, Jeffrey,Shapiro, McGraw-Hill Education – Europe, 2001
- Learn Microsoft Sql Server 2000, Jose A.,Ramalho, Wordware Publishing Inc.,U.S., 2001
- www.google.gr
- www.thecodeproject.com
- www.coder.gr
- www.msdn.microsoft.com
- www.vbdotnetheaven.com
- <http://www.devarticles.com/c/b/VB.Net/>
- <http://www.programmersheaven.com/2/VB-NET-School>

