Abstract

An alumni association is an association of graduates (alumni) or, more broadly, of former students. In the alumni of universities, colleges, schools, fraternities and sororities often form groups with alumni from the same organization. These associations often organize social events; publish newsletters or magazines raise funds for the organization. Many provide a variety of benefits and services that help alumni maintain connections to their educational institution and fellow graduates. Additionally, such groups often support new alumni, and provide a forum to form new friendships and business relationships with people of similar background.

Alumni associations are mainly organized around universities or departments of universities, but may also be organized among students that studied in a certain country. In the past, they were often considered to be the universities or schools old boy network. Today, alumni associations involve graduates of all age groups and demographics.

This project is aimed at developing a repository for the alumni of the college, which is of importance to a college. The Repository and Search Engine (RASE) is an Internet based application that can be accessed throughout the World. Anyone can access the Search Engine to know about any Alumni of that college but can’t able to add. Alumni can only update the database when they are in the college.

The project is envisaged to be completed in two phases. The initial phase will be the creation of a simple system that will be used to capture data from current final year students before the end of term. The second phase of development will extend the functionality of the system to allow past alumni to register.

Introduction

Purpose:

This system can be used as an application for the **Alumni Information Database** to manage the college information and student’s information. The system is an online application that can be accessed throughout the organization and outside customers as well with proper login provided, which will give better service to the customers.

Scope:

This system can be used as the Office of Alumni and College Relations seeks to protect the privacy of its alumni and friends, and thus, endeavors to safeguard the use of information in its custody. To that end, the Office of Alumni and College Relations provides constituent information to requestors only under the conditions.

Overview:

Overall description consists of background of the entire specific requirement. It also gives explanation about actor and function which is used. It gives explanation about architecture diagram and it also gives what we are assumed and dependencies. It also support specific requirement and also it support functional requirement, supplementary requirement other than actor which is used. It also gives index and appendices. It also gives explanation about any doubt and queries.

Once a student graduates from the institute, his/her professional life or career begins, with higher education playing an important role in establishing himself/herself in the profession. In respect of College, it has been our experience that from the very beginning, the alumni have maintained personal contacts with one another, rather than use the channel of Alumni Association.

The advancements in information technology have certainly helped in creating new resources such as alumni web pages, list servers etc., so as to permit greater interactions between the alumni.

System Analysis

Existing System:

The Existing system is a computerized system but which is maintained at individual databases i.e in excels sheets, it’s a time delay process. And maintaining all the records in Excel sheets is difficult. If they want any record they have to search all the records. It doesn’t provide multiple user accessibility and also doesn’t have different user privileges. So the system is not accessible for all the employees of the organization.

Limitations in Existing System

- The current system is not completely complete computerized and manual system in entering students and staff data and handling it.
- There is no centralized database maintenance
- There is no easy access to the particular students record
- The student cannot easily navigate through the database

Proposed System:

The Proposed system is a computerized system but which is maintained at Centralized databases i.e. in automated forms it’s a very fast process. And maintaining all the records in online systems database which makes it very easy to access and retrieve data from the database. If they want any record they can easily search all the records. It provides multiple user accessibility and also has different user privileges. So the system is accessible for all the employees of the organization.

Advantages over Existing System

- It is completely automated system in handling the college database
- This system provides centralized database maintenance
- This system provides easy access to the particular students account or his complete details
- This system provides student to easily navigate through the application for more information in a most secure manner.

MODULES

The System is proposed to have the following modules:

1.Administrator module

2.Event manager module

3.Alumni module

4.Student module

Administrator Module:

The administrator is responsible for maintaining information of students. When a student submits the registration form, administrator will complete the verification process and, if successful, the student details are added into the database. The administrator maintains the passwords of Event Manager and that of himself.

Event Manager Module:

This module maintains the information about various events that are conducted by various colleges and universities. Details of notifications are also maintained. The manager can add, delete, edit and view event details.

Alumni & Student Module:

The Alumni/Students can register themselves and after the approval from the administrator, they can logon into their account and can send mails, post queries, update their profiles and even search for other student details. one can view the event details and search for specific information. The module provides mail and query functionalities.

SYSTEM SPECIFICATION

HARDWARE SPECIFICATION

PROCESSOR	:	Intel Pentium IV 1.8 GHz
MOTHERBOARD	:	Intel 915GVSR chipset board
RAM	:	1 GB DDR2 RAM
HARD DISK DRIVE	:	160 GB
FLOPPY DRIVE	:	1.44 MB
DVD/CD DRIVE	:	Sony 52 x Dual layer drive
MONITOR	:	17” Color TFT Monitor
KEYBOARD	:	Multimedia Keyboard 108 Keys
MOUSE	:	Logitech Optical Mouse
CABINET	:	ATX iball.
HUB	:	Compex 16 lines.

BANDWIDTH	:	100 mbps.

SOFTWARE CONFIGURATION

FRONTEND	:	ASP.NET 2005
OPERATING SYSTEMS	:	Microsoft windows xp
DOCUMENTATION	:	Microsoft word 2003.
SCRIPTING LANGUAGE	:	Java Script

Software description

LANGUAGE DESCRIPTION

Overview of Visual Studio.Net 2005

What is .NET?

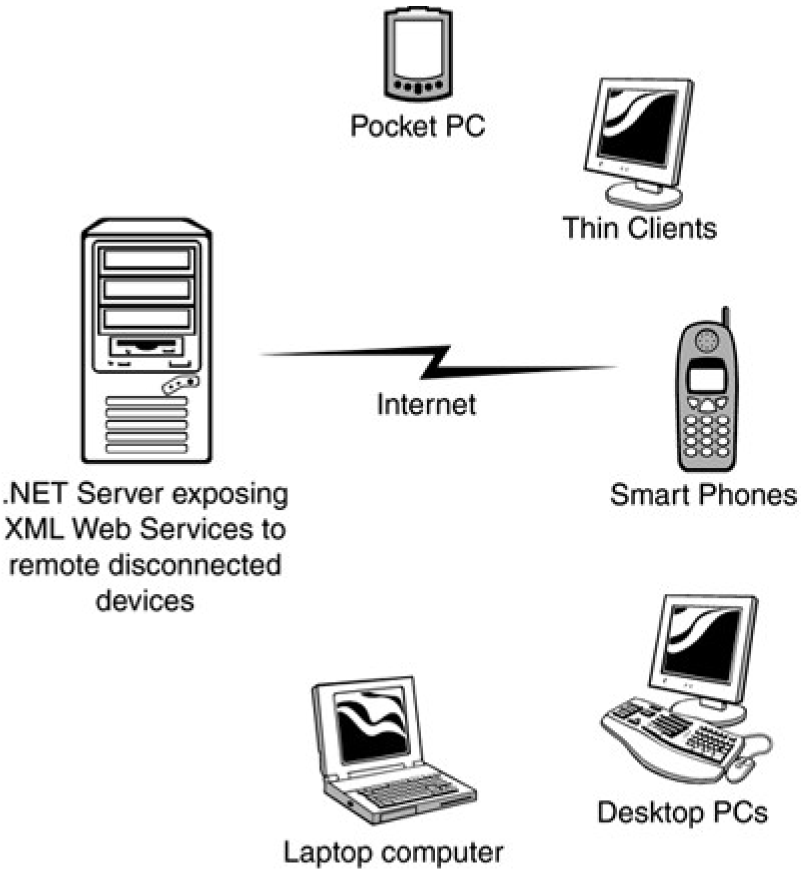
When .NET was announced in late 1999, Microsoft positioned the technology as a platform for building and consuming Extensible Markup Language (XML) Web services. XML Web services allow any type of application, be it a Windows- or browser-based application running on any type of computer system, to consume data from any type of server over the Internet.

The reason this idea is so great is the way in which the XML messages are transferred: over established standard protocols that exist today. Using protocols such as

SOAP, HTTP, and SMTP, XML Web services make it possible to expose data over the wire with little or no modifications to your existing code.

Figure presents a high-level overview of the .NET Framework and how XML Web services are positioned.

Figure 1.1. Stateless XML Web services model.

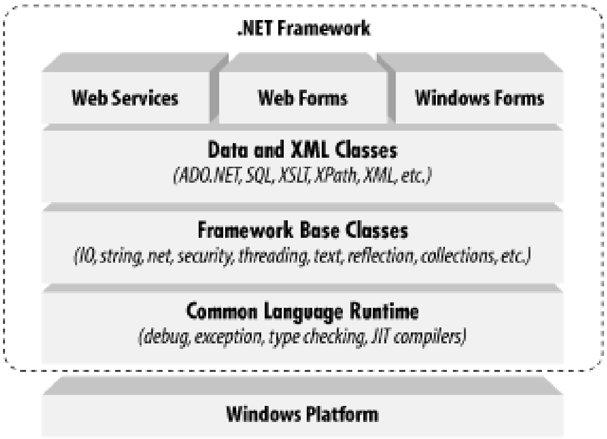

Pocket PC
Thin Clients
Internet
Smart Phones
.NET Server exposing XML Web Services to remote disconnected devices
Desktop PCs
Laptop computer

Since the initial announcement of the .NET Framework, it's taken on many new and different meanings to different people. To a developer, .NET means a great environment for creating robust distributed applications. To an IT manager, .NET means simpler deployment of applications to end users, tighter security, and simpler management. To a CTO or CIO, .NET means happier developers using state-of-the-art development technologies and a smaller bottom line.

To understand why all these statements are true, you need to get a grip on what the .NET Framework consists of, and how it's truly a revolutionary step forward for application architecture, development, and deployment.

NET Framework

Now that you are familiar with the major goals of the .NET Framework, let's briefly examine its architecture. As you can see in Figure 1-2, the .NET Framework sits on top of the operating system, which can be a few different flavors of Windows and consists of a number of components .NET is essentially a system application that runs on Windows.


.NET Framework
Web Services
Web Forms
Windows Forms
Data and XML Classes
(ADO.NET, SQL, XSLT, XPath, XML, etc.)
Framework Base Classes
(IO, string, net, security, threading, text, reflection, collections, etc.)
Common Language Runtime
(debug, exception, type checking, JIT compilers)
Windows Platform

Conceptually, the CLR and the JVM are similar in that they are both runtime infrastructures that abstract the underlying platform differences. However, while the JVM officially supports only the Java language, the CLR supports any language that can be represented in its Common Intermediate Language (CIL). The JVM executes bytecode, so it can, in principle, support many languages, too. Unlike Java's bytecode, though, CIL is never interpreted. Another conceptual difference between the two infrastructures is that Java code runs on any platform with a JVM, whereas .NET code runs only on platforms that support the CLR. In April, 2003, the International Organization for Standardization and the International Electro technical Committee (ISO/IEC) recognized a functional subset of the CLR, known as the Common Language Interface (CLI), as an international standard.

This development, initiated by Microsoft and developed by ECMA International, a European standards organization, opens the way for third parties to implement their own versions of the CLR on other platforms, such as Linux or Mac OS X. For information on third-party and open source projects working to implement the ISO/IEC CLI and C# specifications

The layer on top of the CLR is a set of framework base classes. This set of classes is similar to the set of classes found in STL, MFC, ATL, or Java. These classes support rudimentary input and output functionality, string manipulation, security management, network communications, thread management, text management, reflection functionality, collections functionality, as well as other functions.

On top of the framework base classes is a set of classes that extend the base classes to support data management and XML manipulation. These classes, called ADO.NET, support persistent data management—data that is stored on backend databases. Alongside the data classes, the .NET Framework supports a number of classes to let you manipulate XML data and perform XML searching and XML translations.

Classes in three different technologies (including web services, Web Forms, and Windows Forms) extend the framework base classes and the data and XML classes. Web services include a number of classes that support the development of lightweight distributed components, which work even in the face of firewalls and NAT software. These components support plug-and-play across the Internet, because web services employ standard HTTP and SOAP.

Web Forms, the key technology behind ASP.NET, include a number of classes that allow you to rapidly develop web Graphical User Interface (GUI) applications. If you're currently developing web applications with Visual Interdev, you can think of Web Forms as a facility that allows you to develop web GUIs using the same drag-and-drop approach as if you were developing the GUIs in Visual Basic. Simply drag-and-drop controls onto your Web Form, double-click on a control, and write the code to respond to the associated event.

Windows Forms support a set of classes that allow you to develop native Windows GUI applications. You can think of these classes collectively as a much better version of the MFC in C++ because they support easier and more powerful GUI development and provide a common, consistent interface that can be used in all languages.

The Common Language Runtime

At the heart of the .NET Framework is the common language runtime. The common language runtime is responsible for providing the execution environment that code written in a .NET language runs under. The common language runtime can be compared to the Visual Basic 6 runtime, except that the common language runtime is designed to handle all .NET languages, not just one, as the Visual Basic 6 runtime did for Visual Basic 6. The following list describes some of the benefits the common language runtime gives you:

- Automatic memory management

- Cross-language debugging

- Cross-language exception handling

- Full support for component versioning

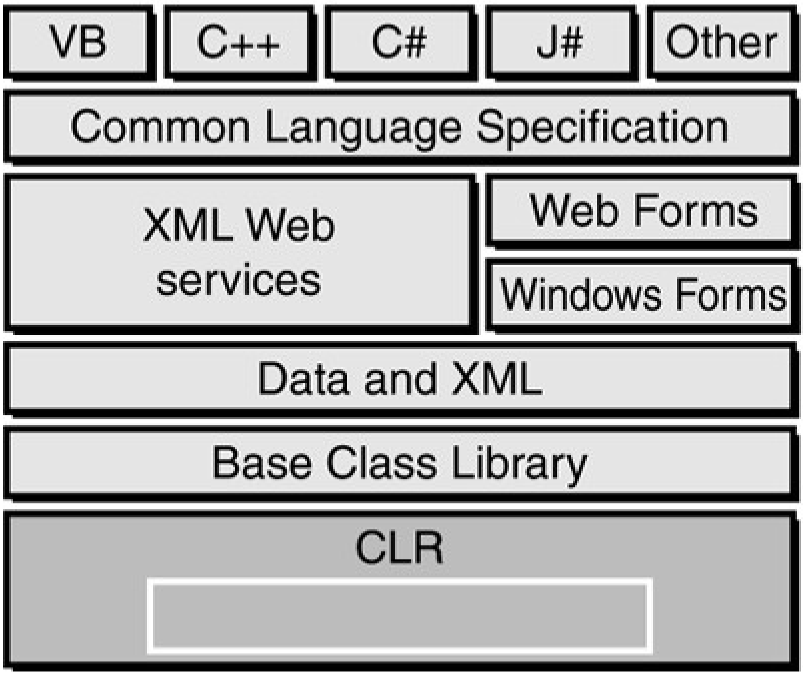
- Access to legacy COM components

- XCOPY deployment

- Robust security model

You might expect all those features, but this has never been possible using Microsoft development tools. Figure 1.3 shows where the common language runtime fits into the .NET Framework.

Figure 1.3. The common language runtime and the .NET Framework.



Note

Code written using a .NET language is known as managed code. Code that uses anything but the common language runtime is known as unmanaged code. The common language runtime provides a managed execution environment for .NET code, whereas the individual runtimes of non-.NET languages provide an unmanaged execution environment.

Inside the Common Language Runtime

The common language runtime enables code running in its execution environment to have features such as security, versioning, memory management and exception handling because of the way .NET code actually executes. When you compiled Visual Basic 6 forms applications, you had the ability to compile down to native node or p-code.

Visual Studio.net offers a rich set of development tools that hide a lot of the complexity inherent in working with the .NET Framework. This reduces time spent both learning the product and developing applications. When Visual Studio .NET is installed, the CLR and the .NET Framework classes are also installed.

These are command line applications that are built without a graphical user interface (GUI) and which can be executed from, and interact with, a DOS Window. In addition to their intrinsic value, they can be particularly useful for testing sections of code to ensure proper functionality.

Windows Forms

These are rich-client applications that are built around a GUI similar to desktop applications written using Visual FoxPro 7.0. New features for client applications developed in the .NET Framework include visual inheritance, code-free resizing, automatic control updates, and new controls.

Web Forms

These are browser-based applications that are built around a GUI with special controls. Web form applications are written using ASP.NET.

XML Web services

These are applications that define XML Web services that can be consumed by other XML Web services or applications either on a local network or exposed on the Internet. Because they are based on HTTP and XML, the information they transfer can be passed through firewalls.

Overview of ASP.NET

ASP.NET is used to create programmable Web pages as part of Web form applications. Not only can ASP.NET produce dynamic Web pages, but it can also tailor them to the browser being used by the user. Development in ASP.NET is much easier because it contains a wide range of object-oriented server-side controls ready for use in applications. ASP.NET actually supports four different types of control:

HTML server controls

These are programmable HTML elements that are exposed to the server. HTML server controls expose an object model that maps very closely to the HTML elements that they render.

Web server controls

These controls have more built-in features than HTML server controls. Web server controls include not only form-type controls, such as buttons and text boxes, but also special-purpose controls, such as a calendar. Web server controls are more abstract than HTML server controls in that their object model does not necessarily reflect HTML syntax.

Validation controls

These are controls that incorporate logic and that can be attached to an input control to test what the user enters. Some validation controls check for a required field, a specific value or pattern of characters, that a value falls within a specified range, and so on.

User controls

These are custom controls created as Web Forms pages. Web Forms user controls can be embedded in other Web Forms pages and provide an easy way to create menus, toolbars, and other reusable elements.

Web Controls used in ASP.NET and Web Forms

1. Label Control:

Displays text on the HTML page.

2. Textbox:

Gives the user an input area on an HTML form.

3. Button:

A normal button control used to respond to click events on the server. You are allowed to pass additional information by setting the CommandName and CommandArguments properties.

4. Link Button:

Like a button in that it posts back to a server, but the button looks like a hyperlink.

5. Image:

Very similar to the normal HTML control that displays an image within the page

6. Hyperlink:

A normal hyperlink control that responds to a click event.

7. DropDownList:

A normal dropdown list control like the HTML control, but can be data bound to a data source.

8. ListBox:

A normal ListBox control like the HTML control, but can be data bound to a data source.

9. DataGrid:

Like a <TABLE> on steroids. You bind a data source to this control and it displays all of the column information. You can also perform paging, sorting, and formatting very easily with this control.

10. AdRotator:

Allows you to specify a list of ads to display. Each time the user re-displays the page, the display rotates through the series of ads.

Field Validator controls

1. RequiredFieldValidator:

   Allows you to check a control on the form to see if it is filled in with anything. If

   it isn't, the ErrorMessage that you set will be displayed in this control.

2. CompareValidator:

   Allows you to check the contents of one control against the contents of another

   control on the form to see if they match. If they don't, the ErrorMessage that you set will be displayed into this control.

MS-SQL SERVER 2000:

Microsoft SQL Server 2000 includes powerful features to support international operations and environments. Extensive multilingual features make SQL Server 2000 a compelling database product and applications platform. This article provides a complete overview of how to use these features in a global context. This article is not limited to a list of features but also will explain how international/multilingual requirements can affect many aspects of a project.

ACCESSING DATA WITH ASP.NET:

ASP.NET includes data access tools that make it easier than ever for you to design sites that allow your users to interact with databases through Web pages.

The .NET Framework includes two data provider for accessing enterprise databases: the .NET Framework Data Provider for OLE DB and the .NET Framework Data Provider for SQL Server. This section focuses on accessing SQL Server (version 7.0 or later) databases using the .NET Framework Data Provider for SQL Server, but you can adapt the code examples to other databases with only minor changes.

To access SQL SERVER 2005 databases from ASP.NET

1. Create a database connection using the SqlConnection class.
2. Select a set of records from the database using the SqlDataAdapter class.
3. Fill a new DataSet using the SqlDataAdapter class.
4. If you are selecting data from a database for non-interactive display only, it is recommended that you use a read-only, forward-only SqlDataReader (or OleDbDataReader for non-SQL databases) for best performance. When using a SqlDataReader, select the records using a SqlCommand query and create a SqlDataReader that is returned from the SqlCommand object's ExecuteReader method.

   In some cases, such as when you want to sort or filter a set of data, you might also want to create a new DataView based on a DataSet for the desired table.

5. Bind a server control, such as a DataGrid, to the DataSet, SqlDataReader, or DataView.

SqlConnection:

A SqlConnection object represents a unique session to a SQL Server data source. In the case of a client/server database system, it is equivalent to a network connection to the server. SqlConnection is used in conjunction with SqlDataAdapter and SqlCommand to increase performance when connecting to a Microsoft SQL Server database. For all third-party SQL server products, as well as other OLE DB-supported data sources, use OleDbConnection.

SqlDataAdapter:

The SqlDataAdapter serves as a bridge between a DataSet and SQL Server for retrieving and saving data. The SqlDataAdapter provides this bridge by mapping Fill, which changes the data in the DataSet to match the data in the data source, and Update, which changes the data in the data source to match the data in the DataSet, using the appropriate Transact-SQL statements against the data source.

DataSet:

The DataSet, which is an in-memory cache of data retrieved from a data source, is a major component of the ADO.NET architecture. The DataSet consists of a collection of DataTable objects that you can relate to each other with DataRelation objects. You can also enforce data integrity in the DataSet by using the UniqueConstraint and ForeignKeyConstraint objects. For further details about working with DataSet objects, see Creating and Using Datasets.

SqlDataReader:

The SqlDataReader class is the SQL server version of the OleDbDataReader. Only one SqlDataReader per associated SqlCommand may be open at a time. & any attempt to open another will fail until the first one is closed. The ExecuteReader is one of the methods of SqlDataReader. The ExecuteReader is used to execute the statements.

SqlCommand:

The SqlCommand class represents an SQL Statementsor stored procedures for use in a database using SQL Server.The CommandText is used to gets/sets the SQL Statement for this command to execute. The CommandType is used to gets/sets the type of CommandText property.

The Connection Object

The Connection object represents the actual connection to the database. Visual Studio .NET 2003 supplies two types of Connection classes: the SqlConnection object, which is designed specifically to connect to SQL Server 7 or later, and the OleDbConnection object, which can provide connections to a wide range of database types.

Visual Studio .NET 2003 further provides a multipurpose ODBCConnection class, as well as an OracleConnection class optimized for connecting to Oracle databases. The Connection object contains all of the information required to open a channel to the database in the ConnectionString property. The Connection object also incorporates methods that facilitate data transactions.

The Command Object

The Command object is represented by two corresponding classes, SqlCommand and OleDbCommand. You can use Command objects to execute commands to a database across a data connection. Command objects can be used to execute stored procedures on the database and SQL commands, or return complete tables. Command objects provide three methods that are used to execute commands on the database:

- ExecuteNonQuery.
  - Executes commands that return no records, such as INSERT, UPDATE, or DELETE
- ExecuteScalar.
  - Returns a single value from a database query
- ExecuteReader.
  - Returns a result set by way of a DataReader object

The DataReader Object

The DataReader object provides a forward-only, read-only, connected stream recordset from a database. Unlike other components of a data provider, DataReader objects cannot be directly instantiated. Rather, the DataReader is returned as the result of a Command object’s ExecuteReader method. The SqlCommand.ExecuteReader method returns a SqlDataReader object, and the OleDbCommand.ExecuteReader method returns an OleDbDataReader object.

Likewise, the ODBC and Oracle Command.ExecuteReader methods return a DataReader specific to the ODBC and Oracle Data Providers respectively. The DataReader can supply rows of data directly to application logic when you do not need to keep the data cached in memory. Because only one row is in memory at a time, the DataReader provides the lowest overhead in terms of system performance, but it requires exclusive use of an open Connection object for the lifetime of the DataReader.

The DataAdapter Object

The DataAdapter is the class at the core of ADO.NET disconnected data access. It is essentially the middleman, facilitating all communication between the database and a DataSet. The DataAdapter fills a DataTable or DataSet with data from the database whenever the Fill method is called. After the memory-resident data has been manipulated, the DataAdapter can transmit changes to the database by calling the Update method. The DataAdapter provides four properties that represent database commands. The four properties are:

- SelectCommand.

  Contains the command text or object that selects the data from the database. This command is executed when the Fill method is called and fills a DataTable or a DataSet.

- InsertCommand.

  Contains the command text or object that inserts a row into a table.

- DeleteCommand.

  Contains the command text or object that deletes a row from a table.

- UpdateCommand.

Contains the command text or object that updates the values of a database.

When the Update method is called, changes in the DataSet are copied back to the database, and the appropriate InsertCommand, DeleteCommand, or UpdateCommand is executed.

Accessing Data

Visual Studio .NET has many built-in wizards and designers to help you shape your data-access architecture rapidly and efficiently. With minimal actual coding, you can implement robust data access for your application. However, the ADO.NET object model is fully available through code to implement customized features or to fine-tune your program. In this lesson, you will learn how to connect to a database with ADO.NET and retrieve data to your application. You will learn to use the visual designers provided by Visual Studio .NET and direct code access.

System Design

INPUT DESIGN

The Input Design is the main feature of the system. Input design determines the format and validations criteria for data entering the system. Inputs originate with end-users; human factors play a significant role in input design. The input design is designed to control the input, to avoid delay, errors in data, to avoid extra steps, to keep the process simple.

The following are the general principals, which are considered in designing inputs.

- Enter only variable data
- Do not input data that can be calculated
- List of values
- Sequence entry

Completeness Check

This checks whether the important fields have been keyed or not. For example the Document number, which is required field for any entry, if it is not entered by the user than message should be raised that the ‘Field must be entered’. Insisting the user to enter value for the field and by user from accessing the other fields without entering it.

The following are the forms:

- ✓ Supplier Details
- ✓ Purchase details
- ✓ Sales details
- ✓ Stock in
- ✓ Stock out

OUTPUT DESIGN

The output form the system is either by screen or by hard copies. Output design aims at communicating the results of the processing of the users. The reports are generated to suit the needs of the users. The reports have to be generated with appropriate levels.

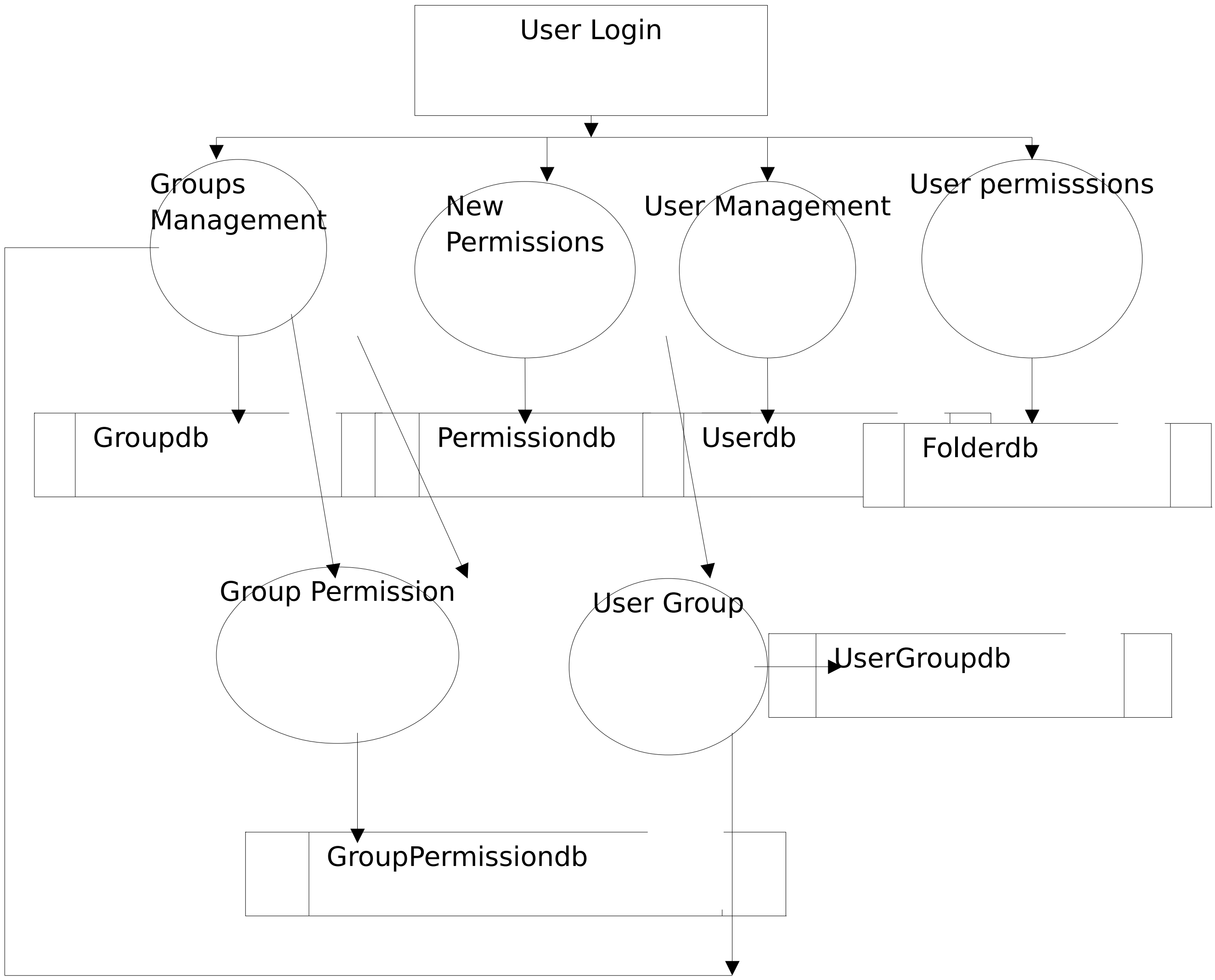
DATABASE DESIGN

The goal of database design is to generate a set of relations that allows storing information easily. The database is designed in relational model in which the data are organized into entities and relational between them.

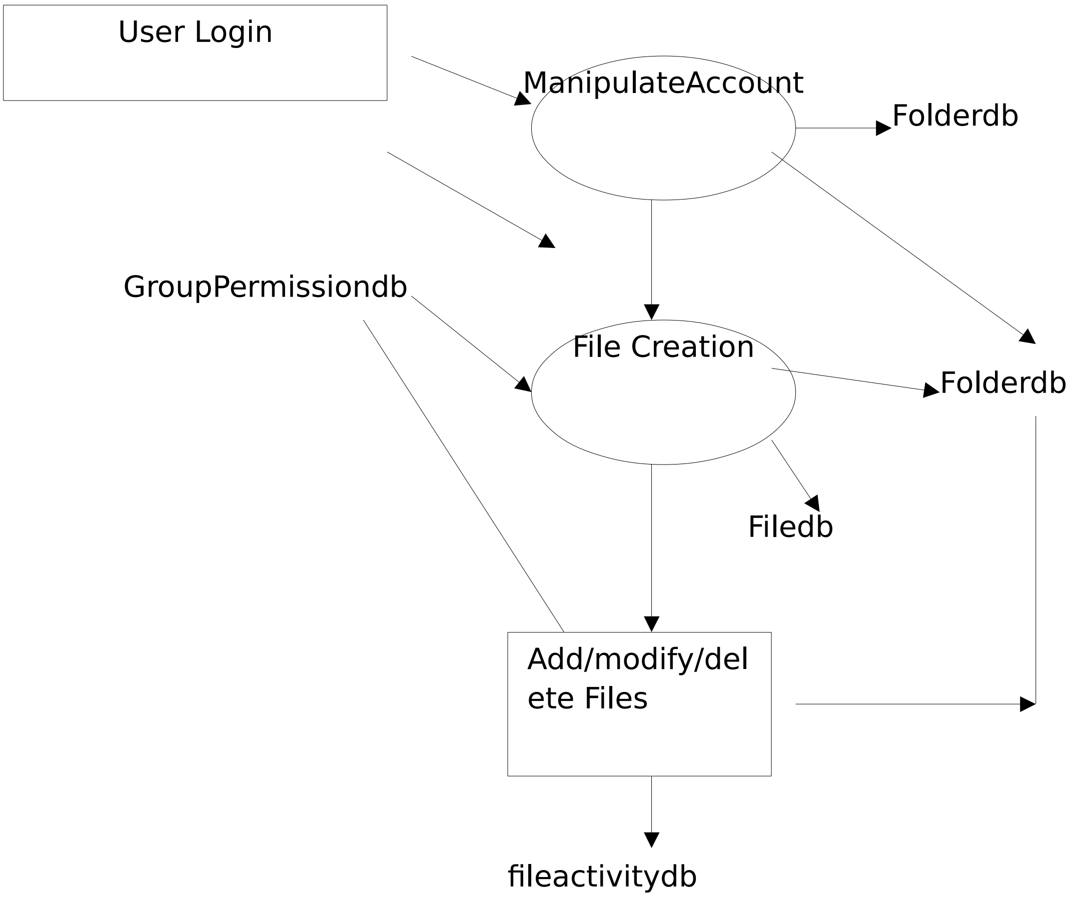
In our flow diagrams, we have given names to data flows, process and data stores. Although the names are descriptive of the data, they do not give details. So the following DFD gives the details of the fields used. A data dictionary has many advantages in improving analyst of user communication by establishing consistent definitions of terms elements and procedures.

Data Flow Diagram

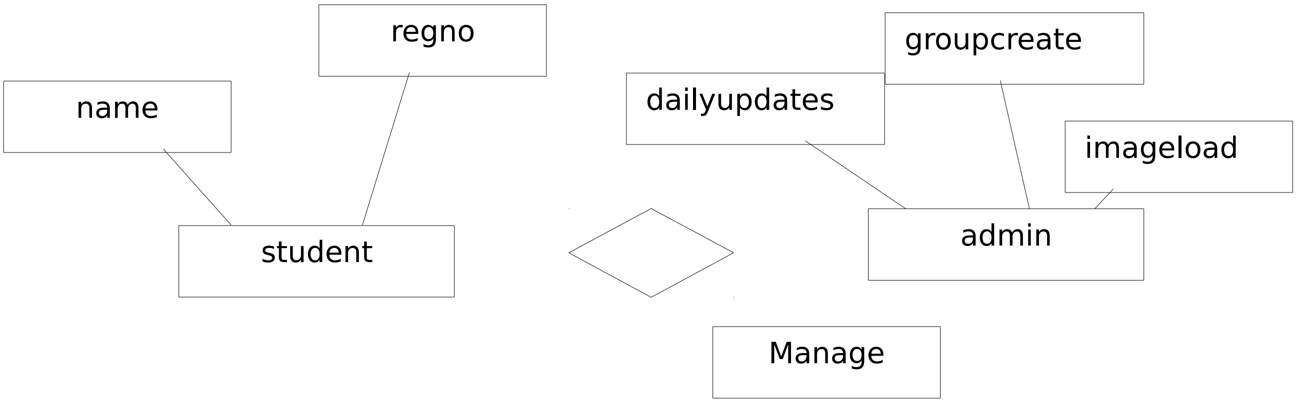
Admin Process



User Process


User Login
ManipulateAccount
Folderdb
GroupPermissiondb
File Creation
Folderdb
Filedb
Add/modify/delete Files
fileactivitydb

ER diagram


regno
name
student
groupcreate
dailyupdates
imageload
admin
Manage

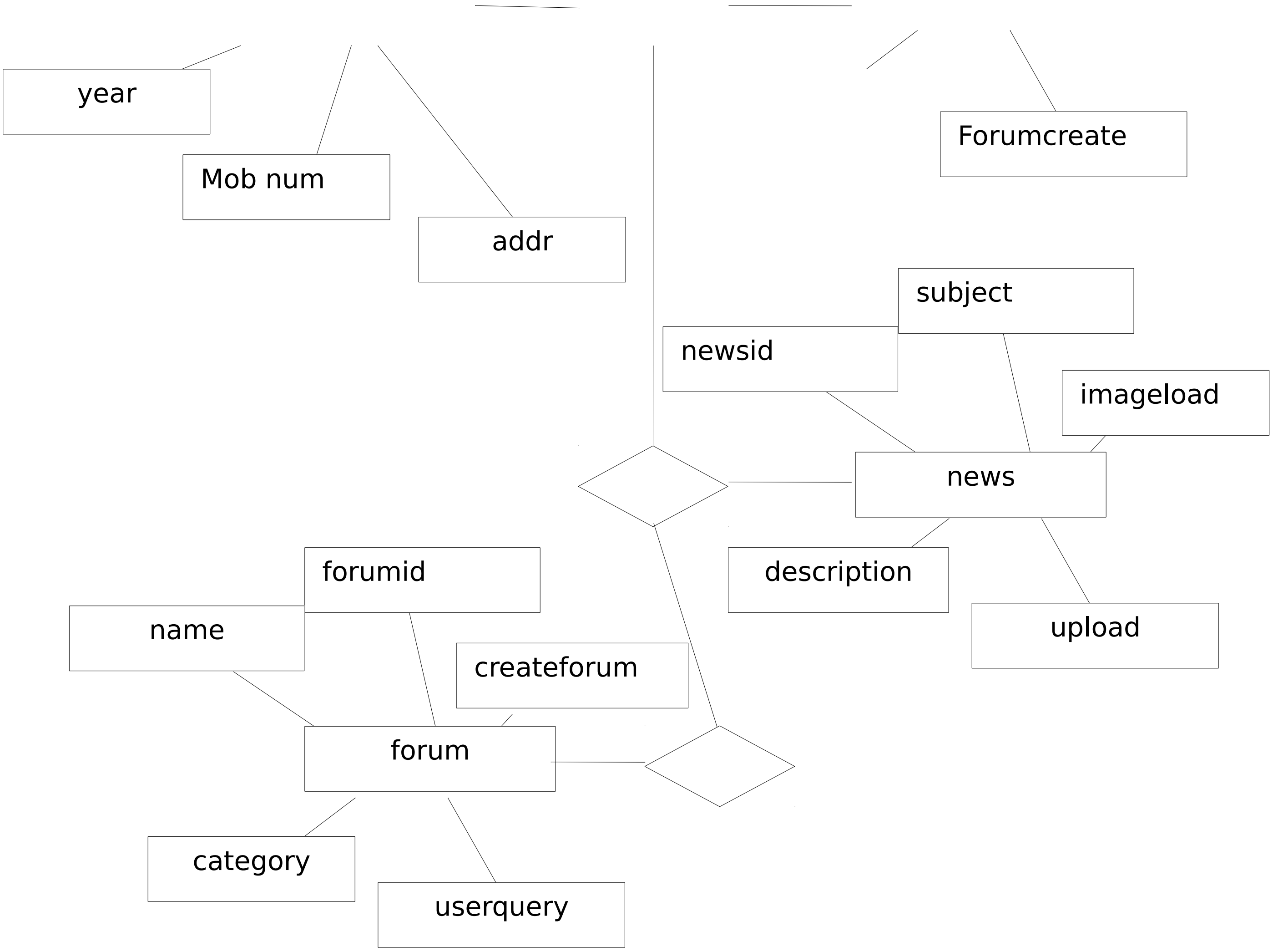


Table design

Id	int	Unchecked
Userid	varchar(50)	Checked
Fname	varchar(50)	Checked
Lname	varchar(50)	Checked
userpwd	varchar(50)	Checked
Address	varchar(150)	Checked
usercity	varchar(50)	Checked
userstate	varchar(50)	Checked
usercountry	varchar(50)	Checked
Aboutu	varchar(150)	Checked
Specialinu	varchar(150)	Checked
hobbies	varchar(150)	Checked
photname	varchar(50)	Checked
userphoto	image	Checked
useremail	varchar(50)	Checked
dateofcreation	varchar(50)	Checked
dateofmodification	varchar(50)	Checked
lastaccessdate	varchar(50)	Checked
lastaccesstime	varchar(50)	Checked
		Unchecked

User Table:

News Table:

newsid	int	Unchecked
newsheading	varchar(250)	Checked

newsdetail	varchar(500)	Checked
imageexists	varchar(50)	Checked
imagename	varchar(50)	Checked
newspicture	image	Checked
dateofcreation	varchar(50)	Checked
		Unchecked

Msgtable:

msgid	int	Unchecked
senderid	varchar(50)	Checked
receiverid	varchar(50)	Checked
postedmsg	varchar(350)	Checked
posteddate	varchar(50)	Checked
msgviewed	varchar(50)	Checked
		Unchecked

ImageTable:

imageid	int	Unchecked
imagename	varchar(50)	Checked
userphoto	image	Checked
userid	varchar(50)	Checked
posteddate	varchar(50)	Checked
		Unchecked

FriendList:

listid	int	Unchecked
userid	varchar(50)	Checked
friendid	varchar(50)	Checked
Rstatus	varchar(50)	Checked
dateofrequest	varchar(50)	Checked
dateofconform	varchar(50)	Checked
		Unchecked

Forumtable

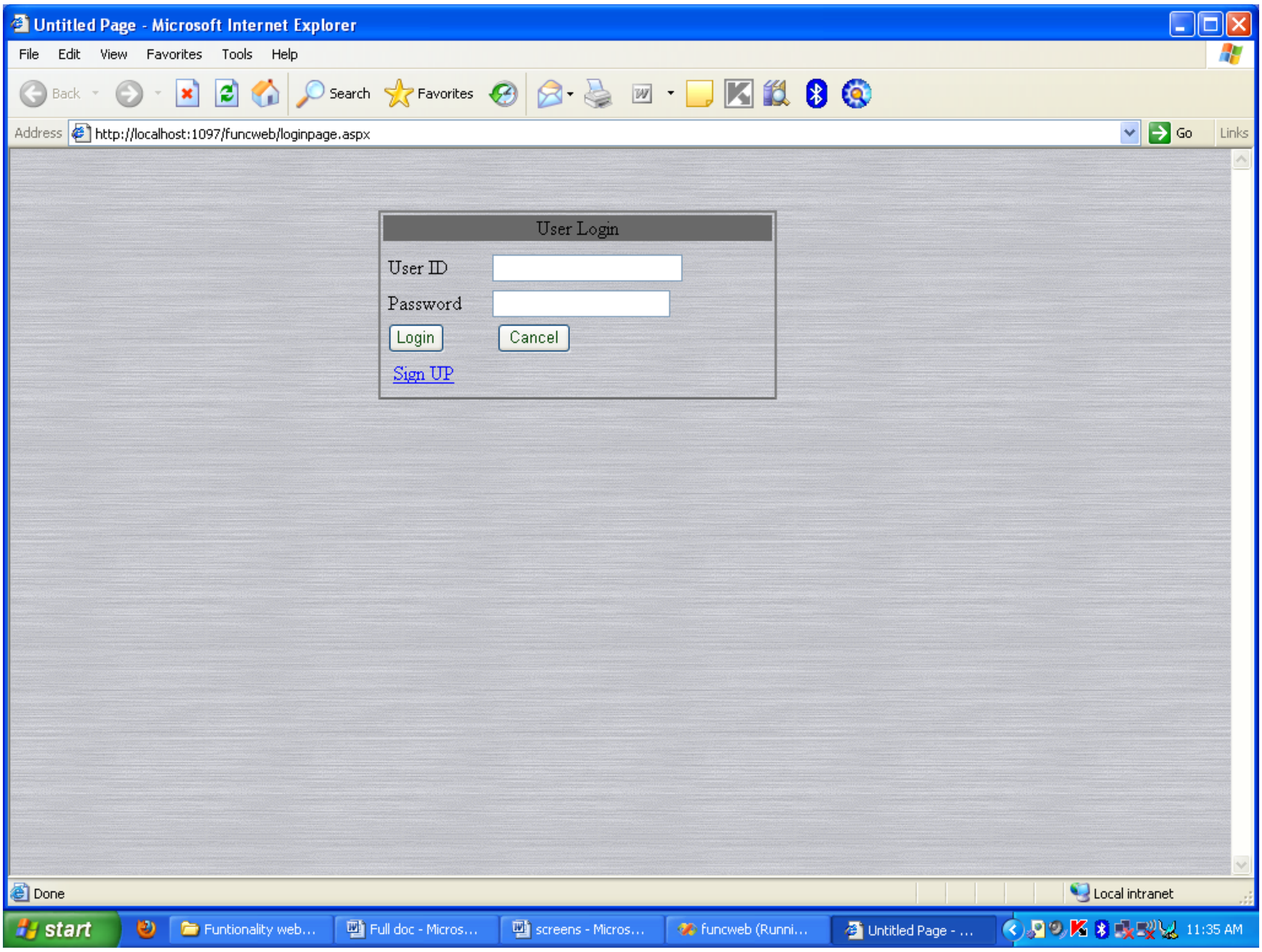
forumid	int	Unchecked
forumname	varchar(50)	Checked
topicid	varchar(50)	Checked
topicname	varchar(50)	Checked
dateofcreation	varchar(50)	Checked
		Unchecked

Forummsg

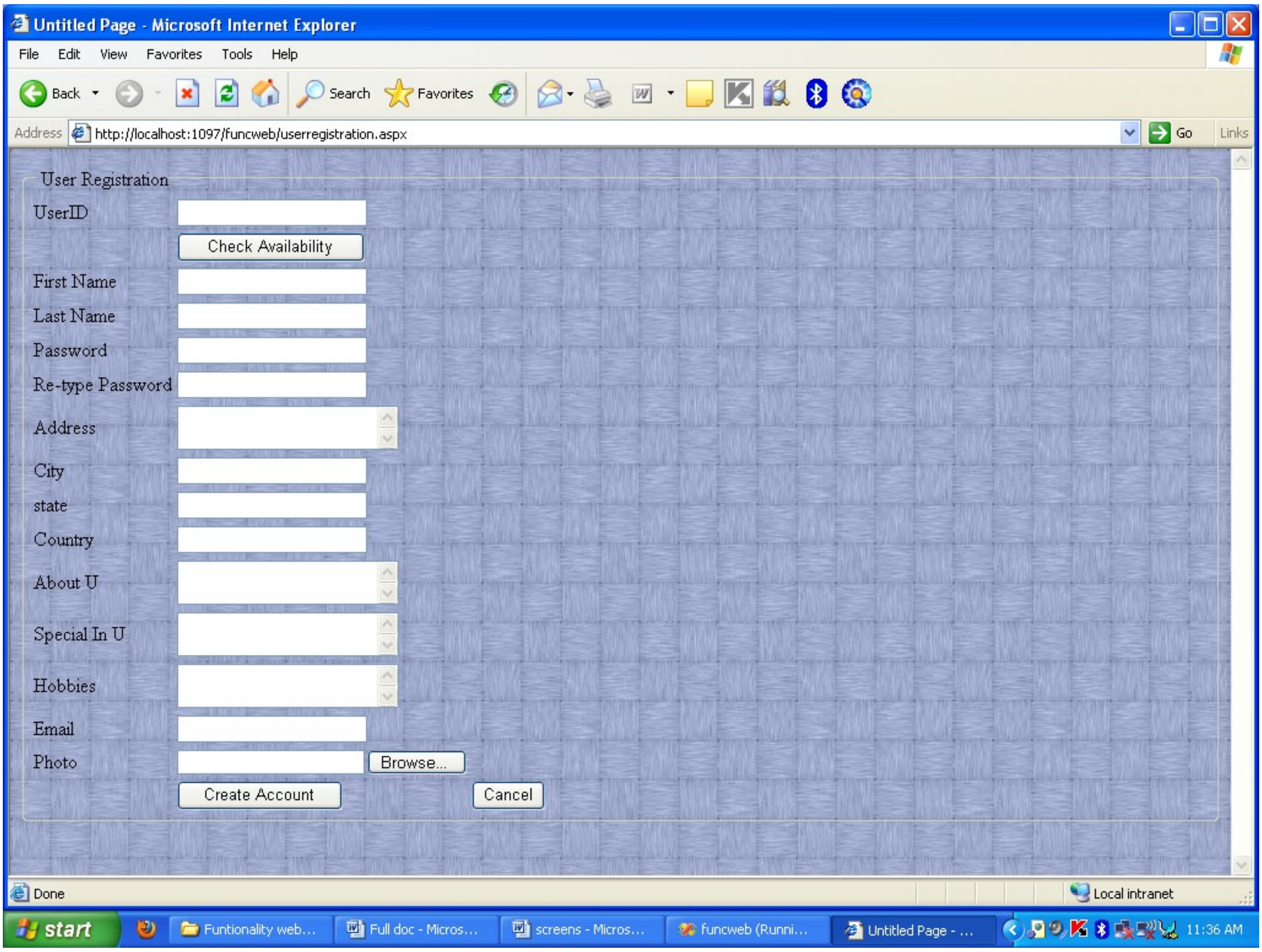
forummsgid	int	Unchecked

forumname	varchar(50)	Checked
topicname	varchar(50)	Checked
userid	varchar(50)	Checked
usermsg	varchar(500)	Checked
posteddate	varchar(50)	Checked
		Unchecked

Screen Shots



Untitled Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:1097/funcweb/loginpage.aspx
Go Links
User Login
User ID
Password
Login
Cancel
Sign UP
Done
Local intranet
start
Funtionality web...
Full doc - Micros...
screens - Micros...
funcweb (Runni...
Untitled Page - ...
11:35 AM



User Registration

UserID

First Name

Last Name

Password

Re-type Password

Address

City

state

Country

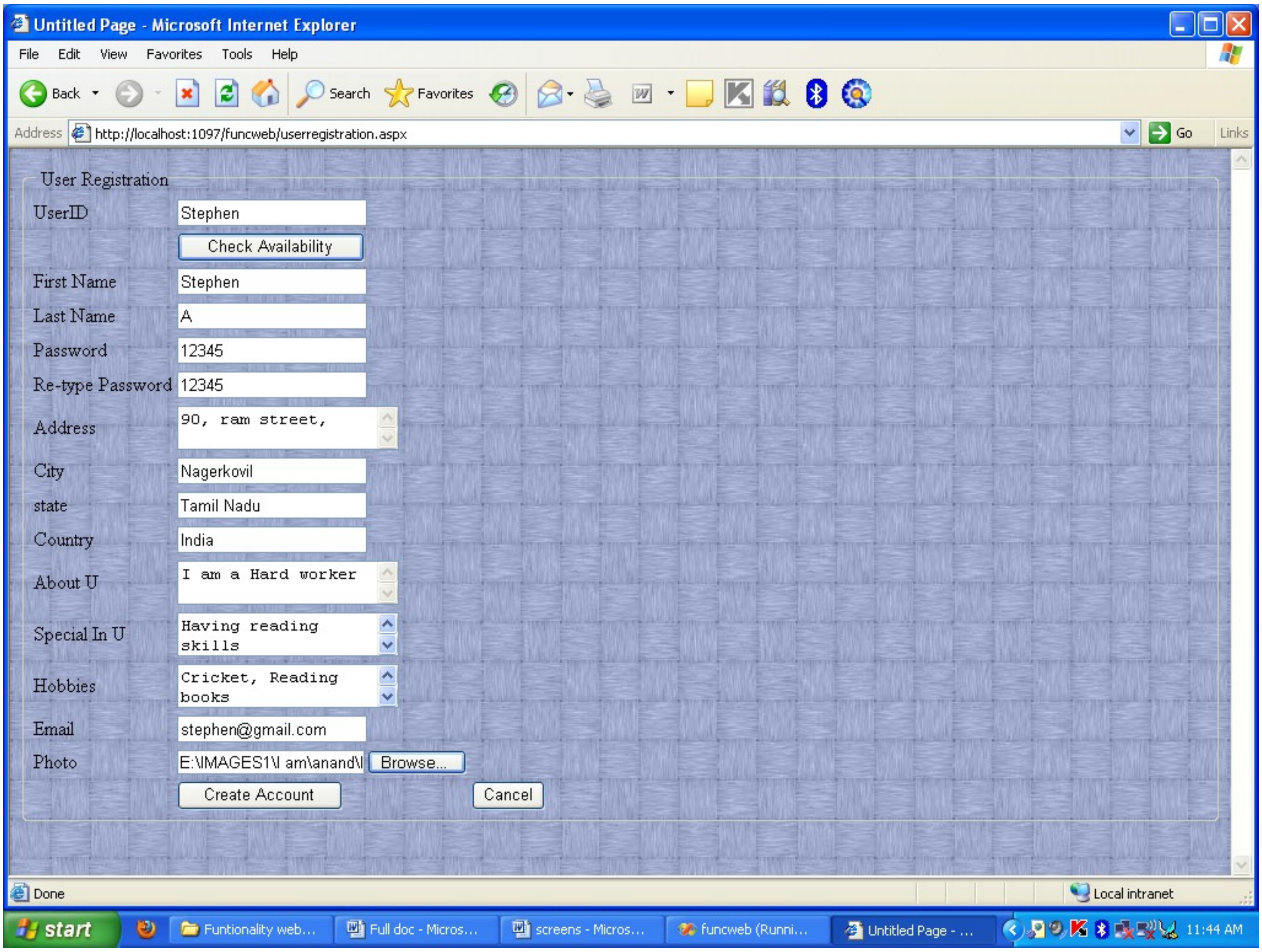
About U

Special In U

Hobbies

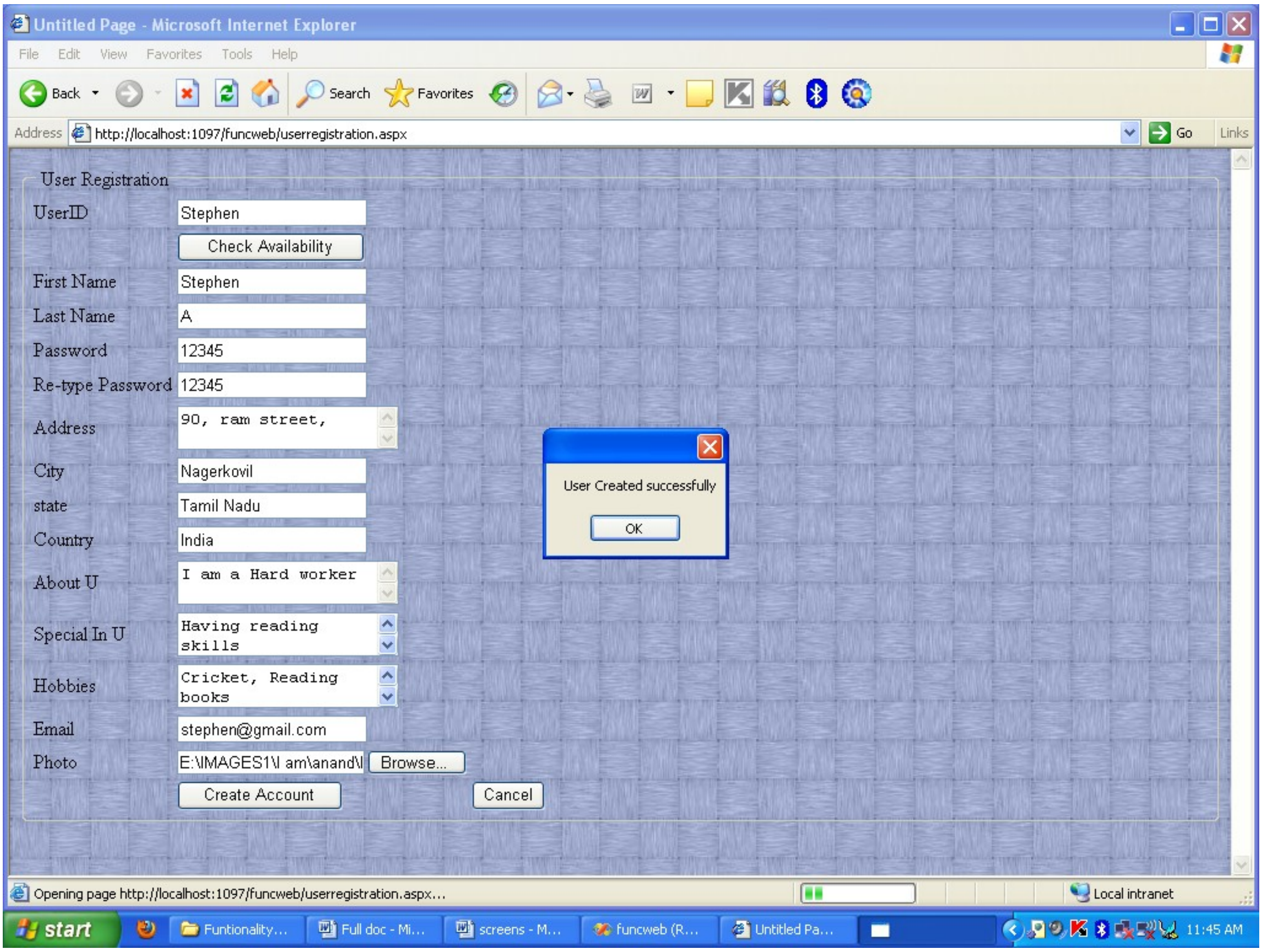
Email

Photo



User Registration

UserID	Stephen
First Name	Stephen
Last Name	A
Password	12345
Re-type Password	12345
Address	90, ram street,
City	Nagerkovil
state	Tamil Nadu
Country	India
About U	I am a Hard worker
Special In U	Having reading skills
Hobbies	Cricket, Reading books
Email	stephen@gmail.com
Photo	E:\IMAGES1\I am\anand\I



http://localhost:1097/funcweb/userregistration.aspx

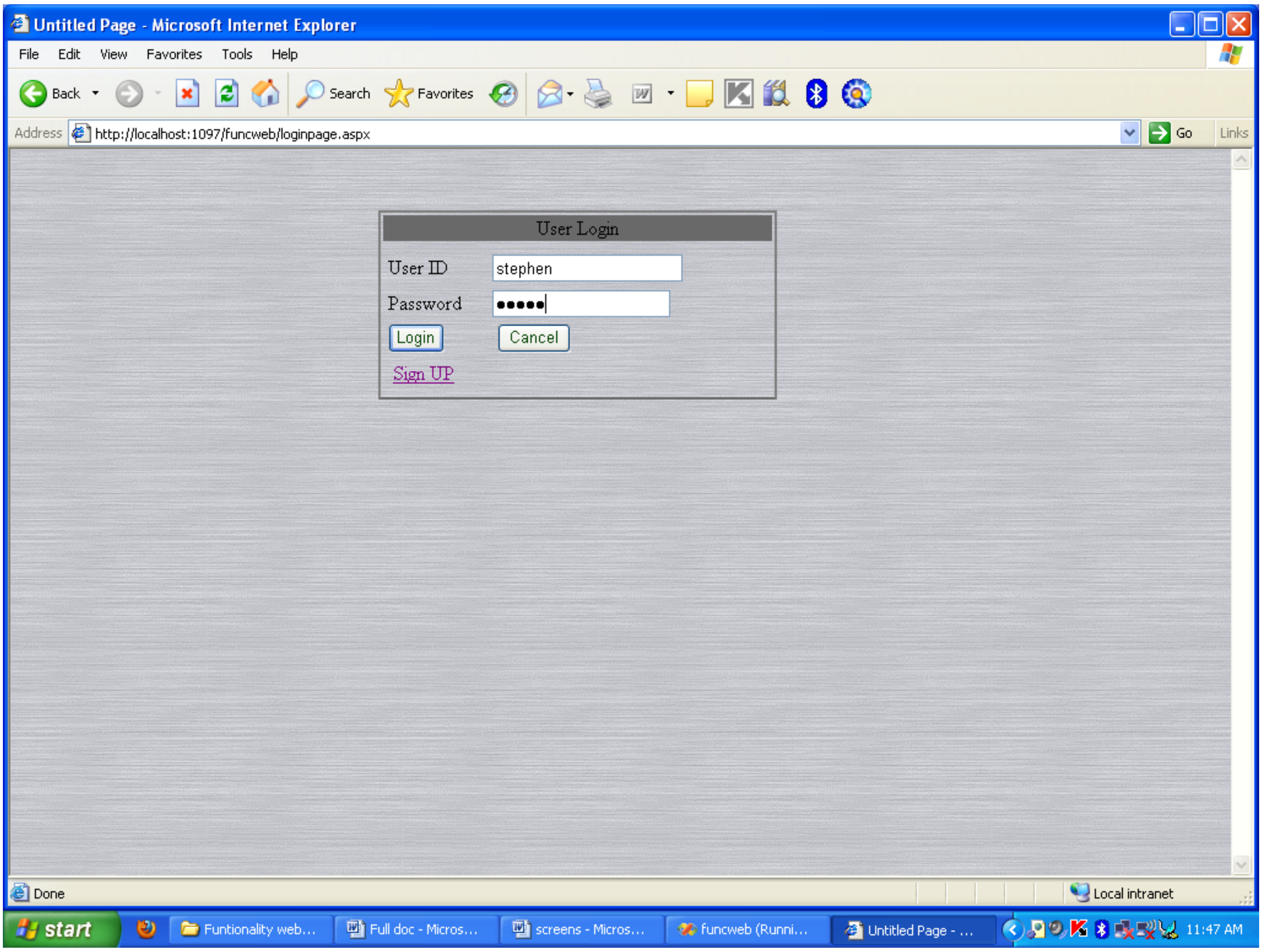
User Registration

UserID	Stephen
	Check Availability
First Name	Stephen
Last Name	A
Password	12345
Re-type Password	12345
Address	90, ram street,
City	Nagerkovil
state	Tamil Nadu
Country	India
About U	I am a Hard worker
Special In U	Having reading skills
Hobbies	Cricket, Reading books
Email	stephen@gmail.com
Photo	E:\IMAGES1\I am\anand\I Browse...
	Create Account Cancel

User Created successfully

OK

Opening page http://localhost:1097/funcweb/userregistration.aspx...

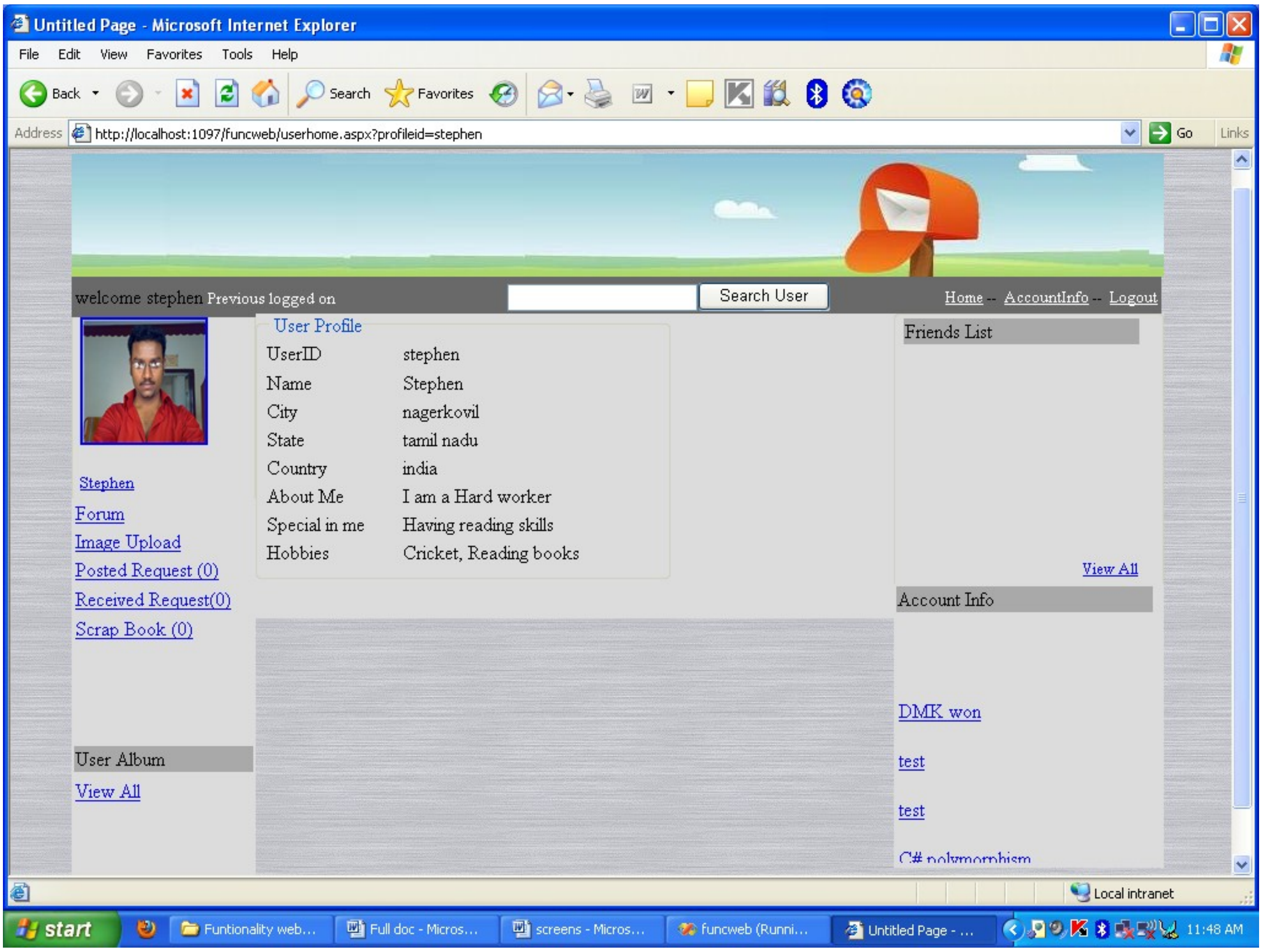


Untitled Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:1097/funcweb/loginpage.aspx
Go Links

User Login

User ID stephen

Password ●●●●●


Untitled Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
http://localhost:1097/funcweb/userhome.aspx?profileid=stephen
Go
Links
Search User
Home -- AccountInfo -- Logout
Local intranet
start
Funtionality web...
Full doc - Micros...
screens - Micros...
funcweb (Runni...
Untitled Page - ...
11:48 AM

welcome stephen Previous logged on

User Profile

UserID	stephen
Name	Stephen
City	nagerkovil
State	tamil nadu
Country	india
About Me	I am a Hard worker
Special in me	Having reading skills
Hobbies	Cricket, Reading books

Stephen

Forum

Image Upload

Posted Request (0)

Received Request(0)

Scrap Book (0)

User Album

View All

Friends List

View All

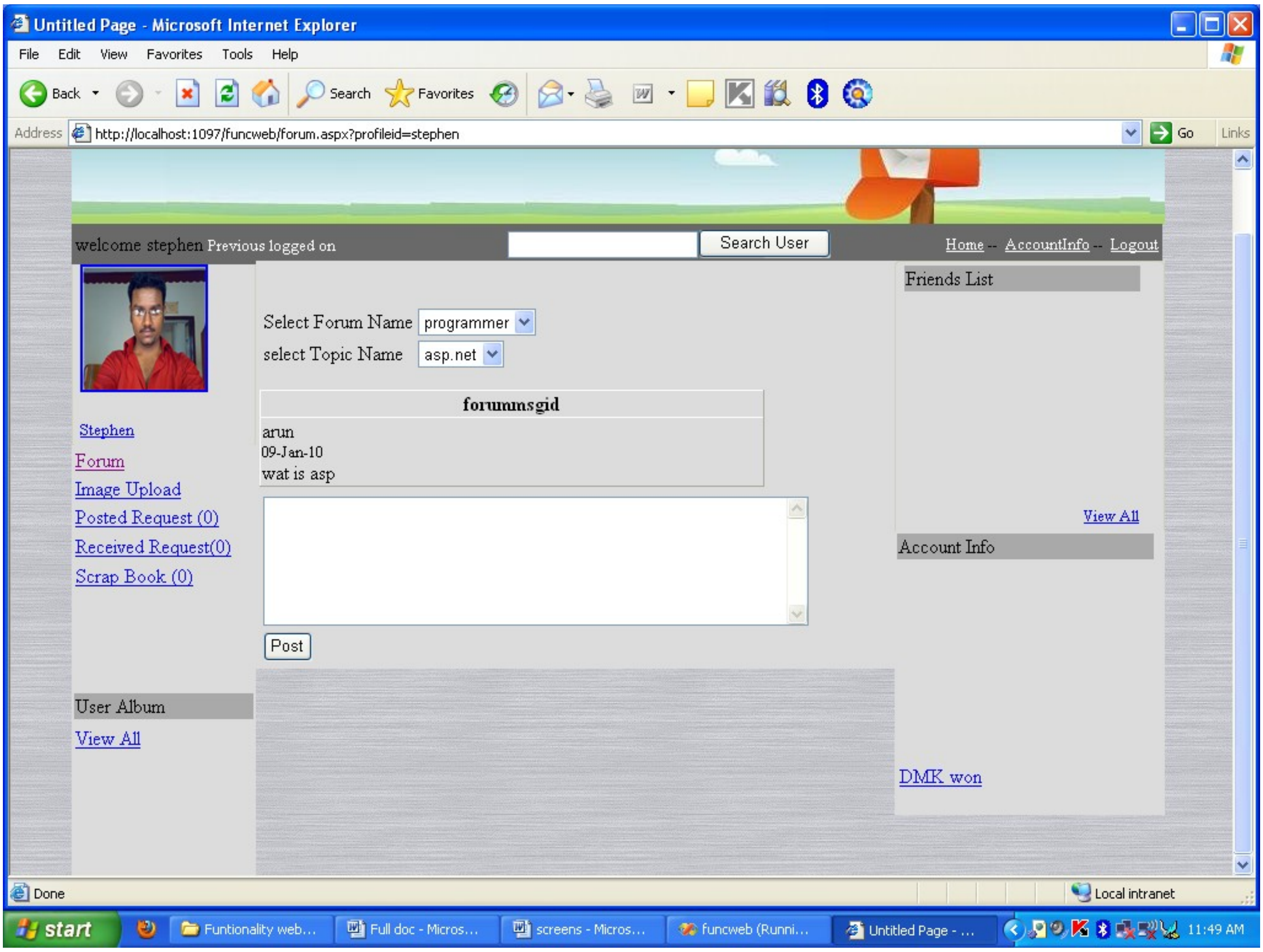
Account Info

DMK won

test

test

C# polymorphism



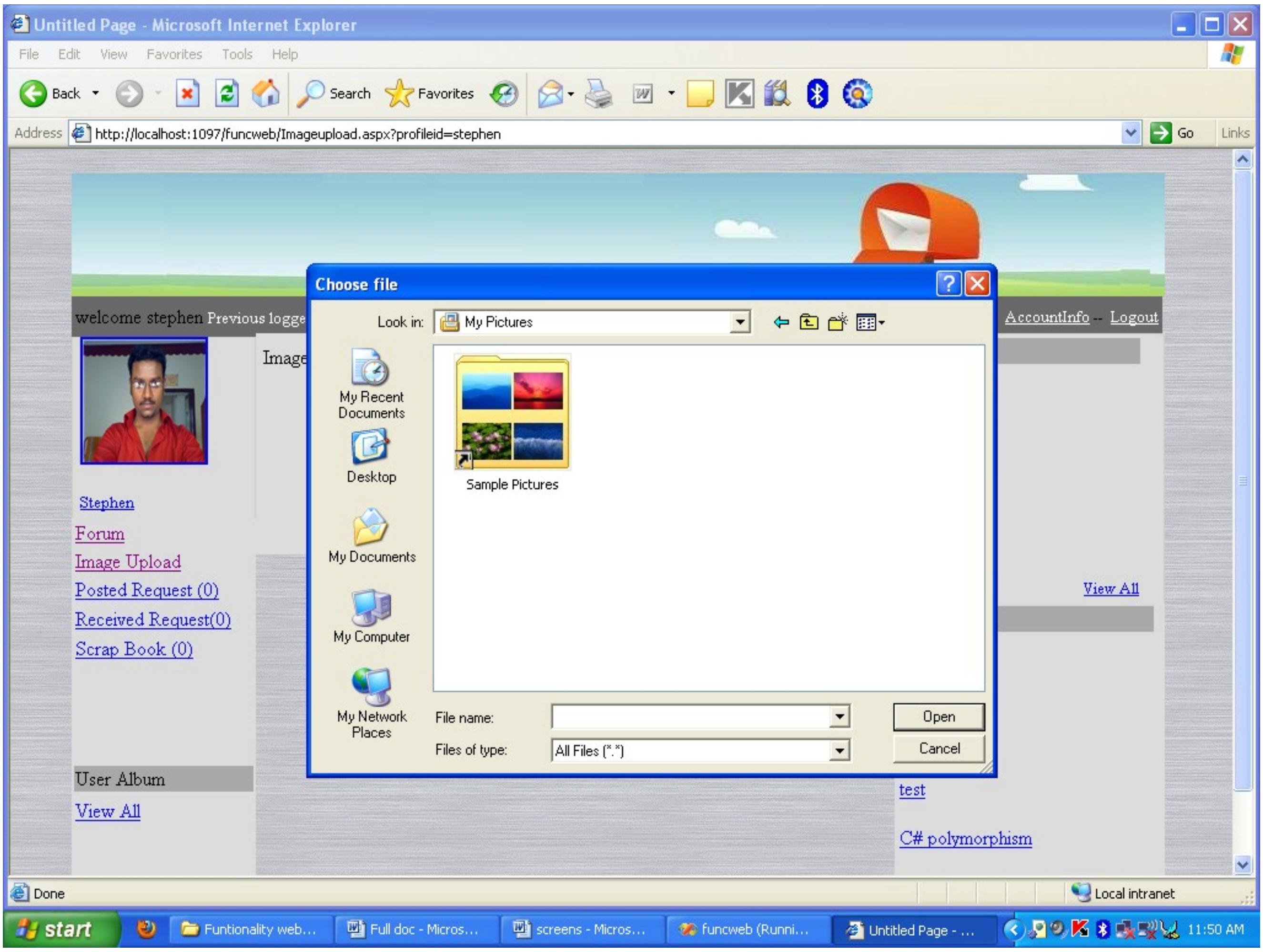
http://localhost:1097/funcweb/forum.aspx?profileid=stephen

welcome stephen Previous logged on

Select Forum Name programmer

select Topic Name asp.net

forummsgid
arun 09-Jan-10 wat is asp

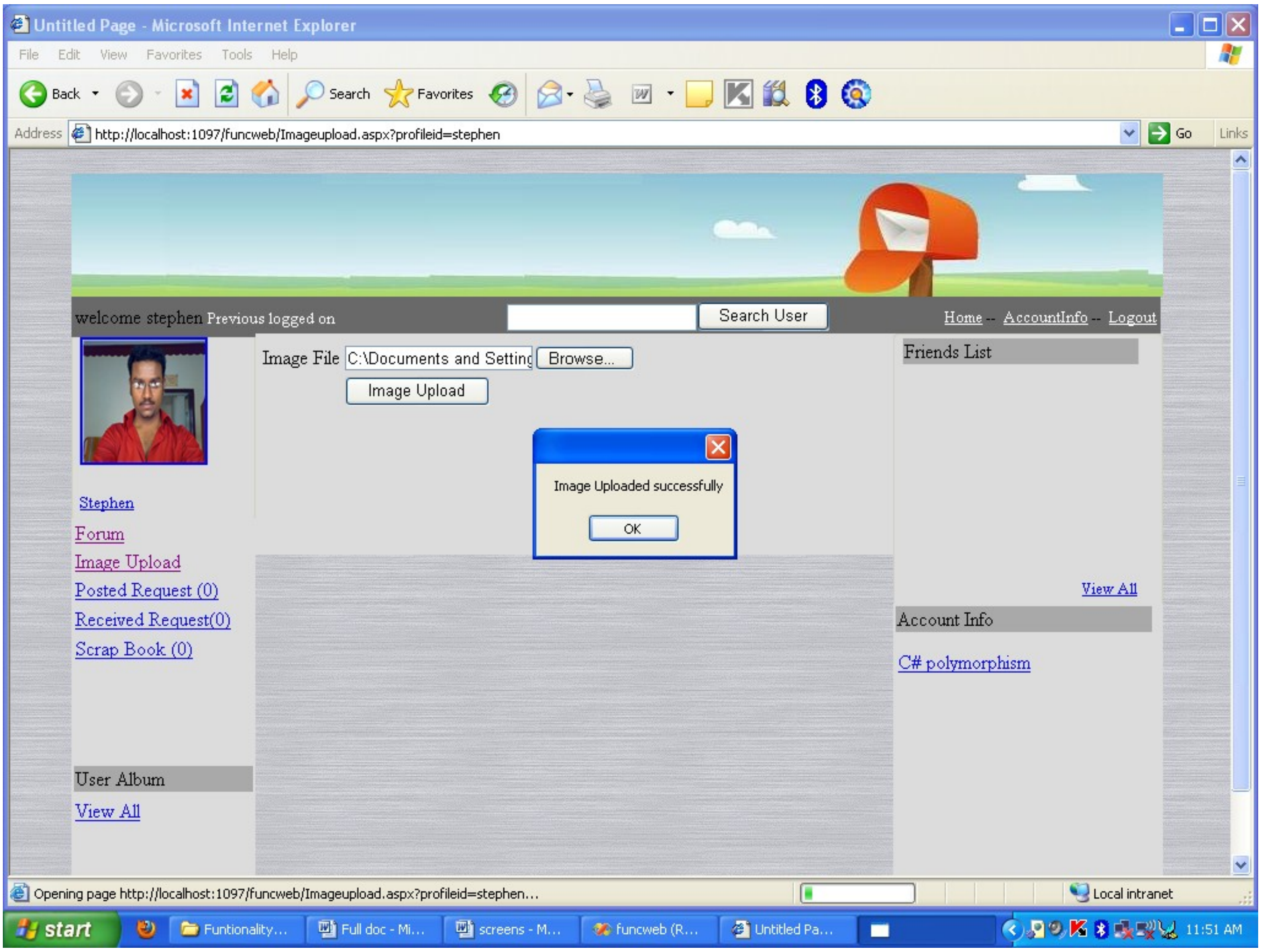


Address: http://localhost:1097/funcweb/Imageupload.aspx?profileid=stephen

Look in: My Pictures

File name:

Files of type: All Files (*.*)



Address: http://localhost:1097/funcweb/Imageupload.aspx?profileid=stephen

welcome stephen Previous logged on

Image File: C:\Documents and Settinç

Image Uploaded successfully

Opening page http://localhost:1097/funcweb/Imageupload.aspx?profileid=stephen...