RESEARCH ON MICROSOFT DOT NET

Submitted By:

M.Usman Siddiqui

AND

Touseef Hafeez Abbasi.

Supervised By:

Dr.Ghulam Muhammad.

Submitted In Partial Fulfillment of

POST GRADUATE DIPLOMA

IN

COMPUTER SCIENCE S.

Computer Center

QUAID-I-AZAM UNIVERSITY

ISLAMABAD

2002.

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

Computer Center
Quaid-I-Azam University
Islamabad.

FINAL APPROVAL

This is to certify that we have studied the Project Report title " **RESEARCH ON DOT NET FRAME WORK"** prepared and submitted by **Mr.USMAN SIIDDIQUE** AND **TOUSEEF HAFEEZ ABBASI**, students of Computer Center, Quaid-I-Azam University (QAU), Islamabad, as a final Project of Post Graduate Diploma in Computer Sciences (PGD). We verify that this report is based on his personal efforts. It bears a sufficient standard to be warranted thus it is acceptable by Quaid-I-Azam University, Islamabad for the Post Graduate Diploma In Computer Science (PGD).

Examamination's Committee

1. **External Examiner** ============== Signature

2. **Internal Supervisor** ============== Signature
Dr.Ghulam Muhammad
Quaid-I-Azam, University
Islamabad.

3. **DIRECTOR** ============== Signature
Dr.Ghulam Muhammad
Quaid-I-Azam, University
Islamabad.

DEDICATED

TO OUR PARENTS.

AND

OUR FRIENDS.

WHO EVER PRAY FOR OUR SUCCESS.

Acknowledgements

"All and every Kind of Praise to be for Almighty **Allah**, Who guides us from darkness to light and help us in difficulties."

First Of all, we are grateful to Almighty **Allah**, His Kindness and Compassion that he made us able to complete this Project.

We are very grateful to my external supervisor ________________________________for providing guidance, support and knowledge to understand existing system and its Problems and also to provide everything We need.

We pay special thanks to **Mr. Muhammad Haroon Siddiqui** for his extra and special help in the Project. Infect the idea of the project was of him. We are thankful to him for providing me a lot of encouragement, some special techniques and extra help.

We can not forget respected **Sir Munawar Tiwana**, who made it possible for us to do our project in an environment like QAU Labs, which is included in top most institutes in Pakistan.

We pay special thanks to my teacher and supervisor **Mr. Dr.Ghulam Muhammad** for providing us guidance, books and specially the encouragement and moral support. We are thankful to him for providing us knowledge about Website design patterns, his innovative ideas and special care.

A bundle of thanks to **Mr. Nabeel Khan** for his valuable help in the project without which we could not complete the project .He worked with us in every difficult step. We cannot forget his sincere efforts.

We are infect so much thankful to **Sir Javed Hussain** who guided us in every step of our academic career as well as in this project and to make me ready for difficult tasks in future.

Special thanks to all of our Class Fellows, **Mr. Kashif Abbas** and his friend **Mr.Naeem Abbas** who helped and co-operated with me.

We are thankful to all of my **friends the whole PGD class** Specially, Mr **Aman uhlla Butt, Mr Atif Sleemi ,Farid Khan , Mr. Jhanghir Butt and Gazanfer** for giving us ideal friendship and source of encouragement during the entire project.

PROJECT BRIEF

Project TITLE.	RESEARCH ON DOT NET.
UNDER TAKEN BY.	M.Usman Siddiqui and Touseef Hafeez Abbasi.
SUPERVISED BY	Dr. Ghulam Muhammad Director of computer center Computer Center Quaid-I-Azam University Islamabad.
STARTING DATE	24th JUNE 2002
COMPLETION DATE	________________ 2002
DESIGNING TOOLS	➢ Adobe Photo Shop. ➢ Flash. ➢ Microsoft Visualstudio.NET
DATA BASE.	MICROSOFT ACCESS
LANGUAGES USED	ASP.NET VISUAL BASIC. NET
OPERATING SYSTEMS	WINDOW 2000 professional
SYSTEM USED	PENTIUM-III
APPLICATION REQUIREMENTS	➢ MS WIN 2000 Professional or Sever ➢ IIS 5.0 or later ➢ Visual Studio.NET

PREFACE

This report is concerned with 'RESEARH ON DOT NET. It consists of Eight chapters.

First chapter: Provides introduction to the project.

Second chapter: Explains what is web and how work.

Third chapter: is about Active server pages.

Fourth chapter: about the Microsoft .NET.

Fifth chapter: about Active Server Pages .Net.

Sixth chapter: about the Visual Basic.

Seventh chapter: Describes Visual Basic .Net.

Eighth chapter: Eighth chapter is about the application we built with **Microsoft.NET** as a proof that we have worked in the **Microsoft.NET** environment and **Microsoft Visual Studio.NET** have been made.

TABLE OF CONTENTS:

CHAPTER NO. 1

Pages No.

INTRODUCTION:

1.1 Project brief. 1.
1.2 About Project 1.
1.3 Objectives of project. 2.
1.4 Strategy. 3.
1.5 SUMMERY. 4.

CHAPTER NO. 2

WEB:

2.1 What is Web. 5.
2.2 History of the Internet. 5.
2.3 Internet connection. 6.
2.4 News groups. 7.
2.5 The World Wide Web. 7
2.6 What is World Wide Web. 8.
2.7 History of www. 9.
2.8 Why people make web pages. 10.
2.9 Other approaches. 10.
2.10 Internet tools for information. 11.
2.10.1 File transfer protocol. 11.
2.10.2 Gopher. 12
2.10.3 The point about gopher. 13.
2.10.4 Hypertext transfers protocol. 13.
2.11.5 Example of HTTP . 14.

2.11 Working of the Internet. 15.
2.11.1 IP addresses. 15.
2.11.2 TCP /IP. 16.
2.11.3 D.S.N 16

2.12 Universal resource locator. 17.
2.13 Dial up connection. 17
2.18 Summery 18.

CHAPTER NO .3

ACTIVE SERVER PAGES.

3.1 INTRODUCTION. 19.
3.2 WHAT IS ASP. 24.
3.3 How ASP WORKS. 25.
3.4 OBJECTS OF ASP. 25.
3.5.1 Application objects. 26.
3.5.2 Server objects. 26.
3.5.3 Request objects. 27.
3.5.4 Response objects. 27.
3.5.5 Server objects. 27.
3.5.6 Session objects. 28.

3.6 Draw backs of ASP 28.
3.7 Summery. 29.

CHAPTER No. 4

MICROSOFT. NET:

4.1 INTRODUCTION. 30.
4.2 WHAT IS .NET. 30.
4.3 OBJECTIVES OF .NET. 31.
4.4 THE .NET FRAMEWORK. 31.
4.5 COMPONENTS OF .NET. 32.
4.5.1 Common language runtime. 33.
4.5.2 .NET classes. 34.
4.5.3 ADO.NET and DATA. 34.
4.6 NEED FOR .NET. 35.
4.7 BENEFITS of .NET 35.
4.7.1. Simple development. 35.
4.7.2. Based on web. 36.
4.7.3. Designed using programming model. 36.
4.7.4. Easy fro developer using. 36.
4.7.5 Extension of classes. 37.
4.7.6 Easy to maintain and run. 37.
4.6 The .NET platform. 37.
4.8.1. The visual studio .net. 38.
4.8.2. .NET framework. 38.
4.8.3 . NET enterprise server. 39.

CHAPTER NO. 5

ACTIVE SERVER PAGES.NET

5.1 Introduction. 40.
5.2 What is ASP.NET. 40.
5.3 DIFFERENCE BETWEEN ASP and ASP.NET. 41.
5.4 Object in ASP. 42.
5.5 ASP.NET server control. 42.
5.6 HTML server control. 43.
5.7 Web server control. 43.
5.8 Validation control. 44.
5.9 User control. 44.
5.10 Web server control. 44.
5.11 HTML server control. 45.
5.12 Validation control. 46.
5.13 Custom user control. 48.
5.14 Properties in ASP.NET server control. 49.
5.14.1 custom server control. 49.
5.15 Adding control in a web form. 50.
5.15.1 objective. 50.
5.15.2 scenario. 50.
5.15.3 Procedure. 51.
5.16 To build and run the project. 52.
5.17 Custom server control. 52.
5.18 Web forms. 53.
5.19 Application logic. 53.
5.20 Web Request Processing cycle. 55.
5.21 Web page processing cycle. 55.
5.22 Page load event. 56.
5.23 Event handling. 56.
5.24 Page unloads. 57.

CHAPTER No. 6

VISUAL BASIC.

6.1 INTRODUCTION. 58.
6.2 What is Visual basic. 58.
6.3 Working with form. 59.
6.4 Option in the form. 59.
6.4.1 Working with form properties. 59.
6.4.2 Form event. 60.
6.4.3 Command button. 61.
6.4.4 Properties of command button. 61.
6.5 Text box. 62.
6.6 Check box. 62.
6.7 Code module. 63.
6.8 Class module. 64.
6.9 Menus. 64.

6.10 Tool bars. 65.

Chapter No. 7

VISUAL BASIC. NET.

7.1 Introduction 66.
7.2 What is visual basic. Net. 66.
7.3 Revealing the code. 67.
7.4 Menus. 68.
7.5 Tool bars. 68.
7.6 Different between visual basic and visual basic. Net. 69.
7.7 Object oriented programming. 70.
7.7.1 inheritance. 70.
7.7.2 Over loading. 70.
7.7.3 Overriding 70.
7.7.4 Interface. 71.
7.7.5 Delegates. 71.
7.7.6 Shared members. 71.
7.8 Structured exception handling. 72.
7.10. Reducing programming error. 73.
7.10.1 Garbage Collection. 73.
7.10.2 Constructors And Deconstructors. 74.
7.10.3 Variable initialization. 74.
7.10.4 Syntax checking. 74.
7.10.5 Type checking. 75.
7.10.6 Type conversion. 75.
7.10.7 Name spaces. 75.

Chapter No 8.

VISUAL STUDIO.NET

8.1 Visual studio. Net 77
8.2 Next Generation Internet 77
8.3 Design Goals 78
8.4 RAD for the next generation internet 79
8.4.1 RAD support for creating forms pages 80
8.4.2 RAD support for creating xml web services 80
8.4.3 RAD support for creating Access xml web services 80
8.4.4 Enhanced RAD Support for creating your own component 80
8.5 XML Web Services Support 81
8.6 XML Data Access with ADO.Net 82
8.7 Web Development environment 83
8.8 Web Forms 84
8.9 Windows forms 85
8.10 Practice Building a New application 87
8.10.1 Objectives 87
8.10.2 Scenario 87

8.10.3 Procedure 87
8.11 Mobile Computing 92
8.11.1 Mobile Internet tools Kit 93
8.11.2 Mobile internet tools 93
8.11.3 Mobile internet control Runtime 94
8.12 Dot Net Component Frame work 94
8.13 Visual Studio.Net Productivity Features 95
8.13.1 Integrated Development Environment 96
8.13.2 Start Page 96
8.13.3 Dynamic Help 97
8.13.4 Enhanced Intelligence 98
8.13.5 Developer Productivity 98
8.13.6 Interface Objects 99
8.14 Enterprise Templates 101
8.14.1 Shared Extensible Tools 102
8.15 Visual Studio Core Languages Enhancement 103
8.15.1 Visual Basic.net 103
8.15.2 Visual C++ 103
8.15.3 Visual C # 104
8.16 Cross Language Cross Machine Debugging 104
8.16.1 Cross language debugging 104
8.16.2 Cross process debugging 104
8.17 Extensibility & partners 105
8.17.1 Customize the environment 106
8.17.2 Entire IDE is extensible 106
8.17.3 Visual Studio Integration Program 106
8.18 Summary 107

Chapter 9

Site Map 105

Application Pages

9.1 Figure 9.1 106
9.2 Figure 9.2 107
9.3 Figure 9.3 108
9.4 Figure 9.4 109
9.5 Figure 9.5 110
9.6 Figure 9.6 111
9.7 Figure 9.7 115
9.8 Figure 9.8 118
9.9 Figure 9.9 119
9.10 Figure 9.10 125
9.11 Figure 9.11 128
9.12 Figure 9.12 130
9.13 Figure 9.13 135
9.14 Figure 9.14 138
9.15 Figure 9.15 139
9.16 Figure 9.16 140
9.17 Figure 9.17 141
9.18 Figure 9.18 142
9.19 Database 143

APPENDIX A

CHAPTERS

CHAPTER 1

INTRODUCTION

1.1 PROJECT BRIEF:

Today there is a rapid change in the world. All the things have been changed. There are many things involved in the changing, but the main changing in the world is information changing. This change has effected our life. Our life totally has been changed due to information technology. For example in the PIA people maintain the accounts manually and against this have a great error, but after the information technology all the error have reduced and PIA system is much improved.

1.2 ABOUT PROJECT:

Our project is about the .net framework. This is the latest technology, which has great effect on the information technology. This is very wide technology and it has many benefits.

Our project is basically a case study which we study that how this technology is introduced in the world and how it is beneficial for us and how it differ from other technologies such as asp, visual basic and java script. The author suggestions that this is the best technology from the others because we introduce we generate something its own himself. After the few months the .net framework is to be used in the programming languages.

The .net framework is a programming language and other languages are scripting languages such as asp, visual basic and the java script. The benefit of the programming language is that the object is made and after that it called again and again, but in the scripting language the tag is to be written again and again and it is a time consuming method because at this time people work with few minutes and output is to be get very soon.

1.3 OBJECTIVES:

- To describe .net technologies in detail.
- To get complete knowledge about .NET technologies.
- To explore the area using .NET application.
- To compare other Internet technologies with .NET technologies.
- To make a simple application using .NET technology.
- To test or debug the application in different environments.

1.4 Strategy:

- We have studied the different books of different authors and all the authors have different views about the NET technology. This is the latest technology in which describe that this technology is very beneficial in every field of computer. This technology is to be used in future and every company conducts their business on it..

- After the studies of the books we know that how this technology is to be used in the computer field. And also know that which method is to be used to develop this technology. And also knows that which type of technology is be the best. And after this we know that .NET technology is the best in the future

- Using the .NET technology we know that how to be the explore this technology using to make a different application. Which criteria is to be used to explore this technology and understandable easily in the programming field.

- After the deeply study of. NET we easily compare the .NET technology to the other technologies like asp.javascripta and visual basic. How this technology is solve the many problems faced by the programmer.

- After to study the many books we make an application in which we know that how the .NET framework works and how it runs and which tool is to be needed to run this application. For this purpose we make a web site and use the entire tool in the website, which is to be needed for this application. After this we know that .NET technology is better and its output is very fast then other technologies.

- We also know that when the application is be made and after this making it is to be run but before the running it must be compile or debug and if any error it can debug and after it the application is running.

1.5 Summery:

In this chapter we have just introduced the project. We tell that which method is to be used in this project and which type of tool is to be used for running this project.

CHAPTER 2

WHAT IS WEB

2.1 What is web:

The Internet is an open worldwide communications network, linking countless thousands of computer networks, through a mixture of private and public telephone lines. Government agencies, universities, commercial and voluntary organizations individually run its component networks. No single organization can own or control the Internet society and sets standard for its use.

2.2 History of the Internet:

The Internet is a long distance network developed by the us name as advanced research projects agency (**Arpanet**) in the late 1960's.the fast, high-volume telephone links proved reliable and the network was extended over the 10 years to connect 200 computers in military and research establishments throughout the U.S and overseas. It demonstrated early that internetworking was both practical and highly useful.

In the mid of 1980's joined the research part of Arpanet to form the Internet. The important thing is that Internet was not set up to as a commercial venture. There are till appropriate use rules that restrict the Internet for profit.

Today most of the universities are linked directly or individually, many businesses have joined. some to take advantage of the cheap and efficient international communications, some to advertise their wires or provide services, and others to again access to mass of on line data. And get their message across the world

These organizations bring many millions of people to the Internet, but there are also millions of individuals who link in their home computers through one or other of the many service providers. In the U.K members are 100000 and 1000 each new users joining each month.

Internet Net connections:

World total	151. Million.
Africa	0.92 million
Asia	26. Million.
Europe	32.28. Million
Middle east	0.78. Million
Canada and U.S.A.	87. Million
South Africa	4.5. Million.

Except a fourfold increase of online population by 2005. There are about 1.5 Million websites, consisting of over 320 Million pages.

40% of U.K net users log in at home on five U.Kpeople used net in 2001 one in three UK children have used the net.

43% of British schools online

26% adult Canadians online (6.3 access it on a weekly basis.

39% of Canadians have used the net

23% of Americans homes on line by end of 1998.

12% of Americans accessed the stars report.

2.3 News groups:

These are a development of mail list, and can be accessed by an e –mail connection. There are thousands of newsgroups, covering by amazing range of interests, activities and obsessions. Some newsgroups are very active, with hundreds of new articles of new every day.

Some groups clearly have members with to much free time and free access to the internet, in theirs, the articles are typically brief but relevant and interesting

.

2.4 The World Wide Web:

This is newest and of many, the most exciting aspect of the Internet. it consists of several million pages of information stored on host computers throughout the world. These pages contains text, graphics, videoclips.sounds and the most importantly hyperlinks to the other pages, which may be in the same computer or on another machine the other side of the world. To access the World Wide Web, we an interacting connection with our service provider and a web browser a program that can interpret the links and display web pages. Like gopher software web browser can pass graphics, sounds and other formatted files to viewers for display.

2.5 What Is The World Wide Web:

Burners lee, in a 1995 talk, which we can find on the web at http://www.w3.org/pub/www/talks/general.html. Described the web as a "distributed heterogeneous collaborative multimedia information system"

How does the web fit these criteria: consider these points:.

The web is distributed because information is available on computers all over the world. The web is heterogeneous because its information is stored on a variety of computers and networking systems.

The web is collaborative because any web user has the capacity to add information to this system.

The web is multimedia because it isn't limited to text: it can include graphics, sound, animation and video.

The home page of the World Wide Web consortium (wc3) is a good place gets more information on the web.

2.6. History of www:

World wide web was born at cern (European laboratory foe practical physics) in 1989 as a way to send graphical files world wide, but its popularity skyrocketed in 19930with the introduction of mosaic, developed by national center for super computing application at the university of Illinois. Currently there are more than 151 Million web users , more than 60000 commercial web sites, thousand of educational and government sites bringing the total to a staggering 100,000 www sites.

In sep/oct , since introduced graphical www in Pakistan then the popularity of www in Pakistan has skyrocketed and number of organizations have jumped in to capture the market. The isps offering www access have three areas of concentration that is Karachi, Lahore and rawalpindi/ Islamabad. Web access not graphics only, but can also be done using a text-based browser offered by ptcl

Netscape and the Internet browser are now largely superseding the first two-web browsers mosaic, and cello. The web performs a very similar to gopher, in providing links between scattered information, but it does so in a more flexible and a more user friendly way. It has links through gopher space, so that anything available there is a available through a web browser. Ftp file transfer and may news groups can also be accessed through the web. Finding information on the web is not difficult. There are several directories, which provide structured entries onto the mass of pages. The most popular of these at present is Yahoo we can also track down specific topics.

2.7 Why people make web pages or sites:

People set up pages for many reasons as a public service , as an academic exercise or resource, as a public service , their products or for themselves, or simply means as a means of sharing their interests with others. As a result some web pages are excellent source of information in their own right, some are treasure troves of links to other valuable pages; an some are pure trivia.

2.8 Other approaches:

For most people, e-mail, newsgroups, ftp, gopher and the World Wide Web provide the access to the Internet that they need, they are the ones that we will concentrate on. There are other approaches, however that we may want to explore, as we become more experienced in using the Internet. Telnet allows us to log in to a remote computer and use as if our machine was directly attached to it , so we can search its directions and runs programs. As most of these runs the UNIX operating system, we would be familiar with the essentials of UNIX before attempting to telnet. The area information search tool, will trawl through indexed text files searching for those relating to defined topics.

2.9 Internet tools for information:

Following tool are used in the Internet which provide information

- FILE TRANSFAR PROTOCOL.
- GOPHER.
- HYPER TEXT TRANFAR PROTOCOL.

Explanation of the following protocol.

2.9.1 File transfer protocol (FTP):

Through the Internet, there is a standard way of accessing directories on remote computers and transferring files to and from them. This is ftp file transfer protocol. It consists of set of user commands and underlying routines to manage the safe transmission of files. We can do FTP by logging in to a remote computer and giving the commands directly, but most users now manage FTP through a window interface such as Wes FTP. Which takes care of all the command for him. It also possible and sometimes more convenient to do FTP by e-mail.

Sending a request detailing the file we want and where it is to an server does it. This is a computer that runs a special program to deal with such requests.

Sending a request detailing the file we want and where it is an FTP server do it. This is a computer that runs a special program to deal with such requests. That raises the question of how we find where are files are the first place, and the solution there an Archie this is a program run on a certain computers
This is always the first place to check for files, as obtaining them from there is generally quicker and easier than searching for them. FTP transfers the files between the two computers or one computer and the server.
FTP it enable research to access programs and large data files. The file name is the last item of the list.

FTP://ftp.temple.edu/pub/info funstuff/smily. Text.

This is the address of the file smiley. text, which will tell us we want to know and a lot that we don't about smiley and abbreviation that are used in e-mail. It is in the fun stuff directory, inside the info directory in the us web pages.

2.9.2 Gopher:

Gopher is not an Internet service in this sense that it is not built in the TCP/IP protocol, but developed to search data and information on the Internet by offering a text based menu interface. The name Gopher has an interesting history.

It was developed at the university of MINNESTO, who's mascot is Gopher. Some say that as the software tunnels over/under the Internet looking for information, just as a mole like animal Gopher digs tunnels, therefore, it should be called a Gopher. And last but not the least as Americans say" GO FOR IT" WHEN SAID QUICKLY, SOUNDS MUCH LIKE gopher, try it.

Using Gopher does not require a live Internet connection, and can be accessed using PDN. To avail this facility use PAKNET, TO CONNECT TO THE SERVER OF ITU (international telecommunication union) by giving its DTE ADDRESS 0228468111112 and login with use name Gopher

Gopher is a package of complentary programs that organize data and provide access to it. If we running a gopher program and log in to a host computer that acts as a server. The great mass of information on these linked computers is sometimes referred to as a Gopher space. The mane at usually well structured, with clear indications of what each item lead to. We may already have some programs that can be used fir viewing; others can be obtained from the data banks of the Internet. A good Gopher program, such as a HOGOPHE will have the ability to display in a variety of formats, and links to other viewers.

2.9.3 The points about Gopher system:

The emphasis is on academic material we can reach all of Gopher space through our web browser, so dedicated Gopher software is not needed; there is a relatively little new development in the Gopher system, as it has been largely overtaken by the world wide web. It is connect to web page to locate a file for FTP transfer.

Gopher://gopher.ocf.berkely.edu:70/700/gopher/gopher-www

The number after the site name identify the port that we have to connect to on the emote computer, and the nature of this should become clear.

2.9.4 Hypertext transfer protocol: (HTTP).

This protocol is used to design to carry information from one place to another. When the user tells to fetch the web page, a message is sent from the browser to the web server.

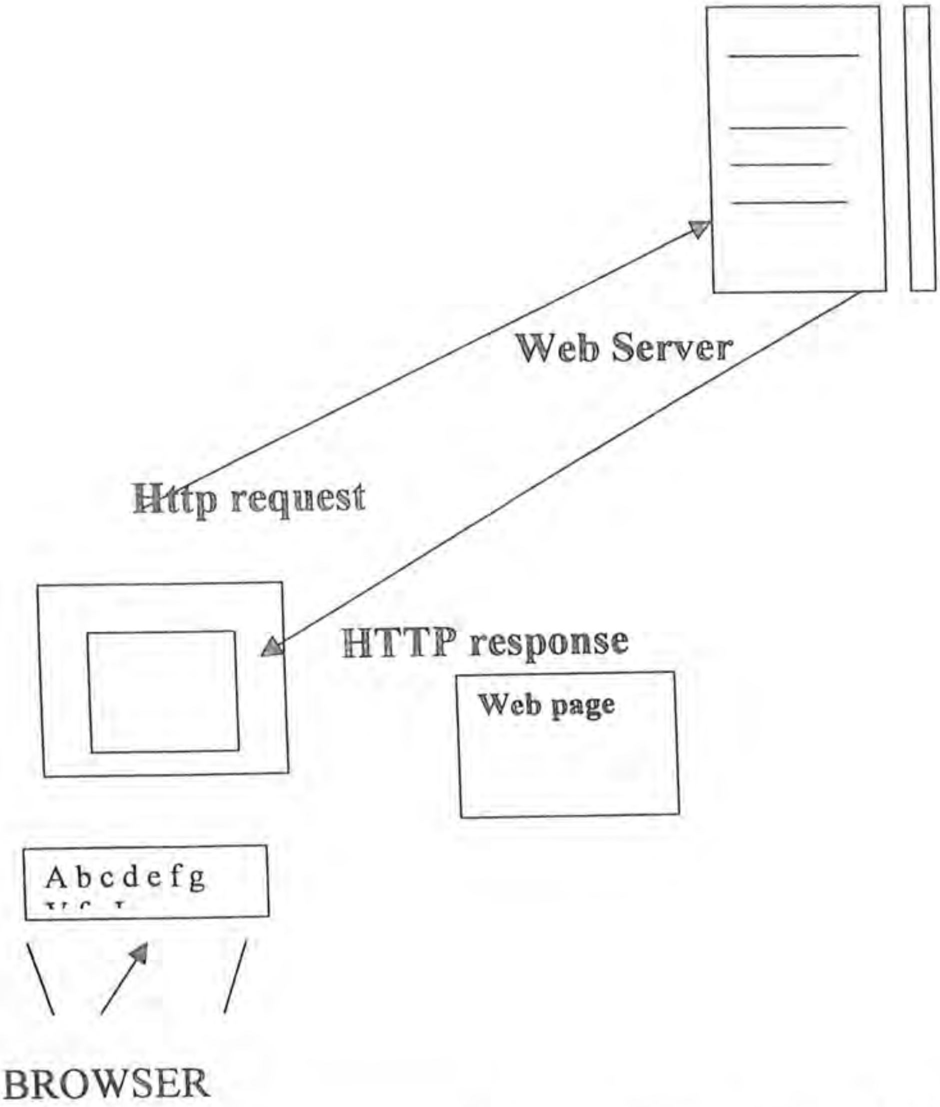
The message is sent using hypertext protocol (HTTP) this protocol is used by the World Wide Web in the transfer from one machine to another we see a URL prefixed with http:// we know that the Internet is used is http.

The message passed from the browser to the web server asking for a particular web page is known as an HTTP request when the web server receives this request. It checks its store to find the appropriate page . if the web server find the page it build the html, in an HTTP response, and sends this back across the network to the browser. If the web server cannot find the requested page, it issues a response that features an appropriate error message and dispatches this page to the browser.http is known a sat less protocol. This is because doesn't known the http request that has been made, is part of an on going correspondence

When a request to the web page it is sent to the server this request contains more than just the desired URL. There is a lot of information that as part of the request. This is also true of the response the server sends the extra information back to the browser. distinguish between different requests however and cannot assign different priorities, so it would be able to tell whether a request is from the users or an various infected machine, it will treat all request equally with the same status. Transfer web pages from a web server to a web client (browser).

2.9.5 Example of the HTTP:

Following is the diagram, which shows how the ASP works.



This example shows that browser send the request to the web server this is the request through the http and after the http response

2.10 Working of the Internet:

- Ip addresses.
- Tcp/ip
- DSN.

2.10.1 Ip Addresses:

Every computer site that is linked to the Internet has its own unique ip addresses. This is made up of two or more names, separated by dots and identify the nature of the organization, the organization itself and, may be, a particular or network. In the jargons these are referred to as domain.

Internet protocol addresses, which allows data packets to be delivered as a specific computer.

The ip addresses consists of four numbers separated by dots. for example 191.168.0.1.

2.10.2 TCP/IP:

Transmission control protocol.

It works is to send the data from one computer to the destination computer. This is a way, which receive information from the packet and send the packet to the destination point.

Generally information is to be sent in the form of packets and every packet has its separated sent. This have only receive the packet which have no error and than it is responsible to send the packet at a require destination. If any packet has an error it is rejected and the packet is to be send back the packet to the source for detecting the error.

.

2.10.3 Domain Name System (DSN):

- A system that gives the different names to the ip addresses.
- It translates the ip addresses into server names.
- Www_server/hostname.
- Top-level domain names.
- Two letter country codes.
- Domain name category, country code or tope level domain.
- Net- network sites and including isps.

2.11 Universal Resource Locator: (URL).

With all the millions of the files , gopher items and web pages that are accessible over the internet, a standardized and way of identifying them is essantial.URLs provide this. There are different styles of URLS for different approaches to the Internet, though they all follow the same pattern.

When you use a web browser, you identify the resources through the URL. A URL specifies both the location (domain name) and the protocol to access.(FTP ,HTTP,TFTP).

TYPE ://host_computer_addresses/directory and filename.

Gopher:// - gopher URL.

2.12 DIAL UP Connection:

- More dial up connections are used to a phone line and the modem connection.
- It also used the serial port or point-to-point is to be used for the connection.
- The data is to be sent are receiving through a modem.

- There are many types of speed of modem but the highest speed of modem is 56kbps

2.13. Summery:

In this chapter we know they what is web and how the Internet access the data and how it give a response, which is to be, send. For many users e-mail is the most useful Internet facility, as it enables them to keep in touch with friends throughout the world, at the cast of the phone call.

- News groups tell that which people can share ideas and the knowledge.

The only text files are transferred by e-mail and in news groups these facilities require only the simplest of connections to the Internet and are offered by providers.

- Archie will find files for us.
- FTP allows us to access files from host computers as long we know their names and the locations.
- The World Wide Web likewise access to masses of information but organized in more easy way.
- Gopherspace, ftp and the other Internet facilities can also be reached through the web.
- We also know that every site has its own ip addresses.
- Every file and the web pages have a uniform resource locator, which tells us where it is and how to reach it.

CHAPTER 3

ACTIVE SEVER PAGES: (ASP)

3.1 INTRODUCTION:

Active server pages was officially introduced to the world by Microsoft n July 16,1996. The author of this, which introduced by the asp is codenamed Denali. The Microsoft introduced the latest version in March 1997. Microsoft has updated this version in 1998. With release of window 2000 and also updated the version of Internet information services 5.0.Asp just like visual basic, java script.

Active server pages are basically HTML tags that contain VBScript code, which is executed on the server. That's why they are called severed site scripts. The results of the VBScripts (if any) are transmitted to the client. The HTML code is transmitted as it is. A server script can produce any output, but only HTML documents can be rendered on the client. Science VBScript's native commands. A server that supports ASP :IIS a personal Web server.

With ASP, one can simply write the code in the HTML page itself. The HTML tags and the code run side by side. Anyone can write the code in a simple scripting language that is easy to learn and easy to use and then can save the page to the website and it will be ready to go. No compiling and no complex interfacing is required in it.

The code inside ASP is mixed in with standard HTML and is NEVER seen by the browser. ASP pages run in ALL browsers UNLESS the person making the page uses HTML or browser commands outside of the ASP portions.

Now the question arises that why I used ASP in addition to the HTML. The answer is simple that ASP basically allows us to dynamically generate the HTML that the browser sees. Maintenance of an ASP site is also much simpler because of SSI (Server-Side Includes). This allows us to have a Header () function in one file that every page on the site includes. If I want to change the page header I just change that one file. Another advantage of using ASP is that it dramatically speeds up the pages too since I'm sending less data to the browser. When converting the site from FrontPage to ASP I saw some pages shrink from 40kb to 3kb, which translates to a speed improvement on a typical modem from 12 seconds to less than 1!

3.1.1 More About Active Server Pages

Microsoft® Active Server Pages (ASP) is a server-side scripting environment that you can use to create interactive Web pages and build powerful Web applications. When the server receives a request for an ASP file, it processes server-side scripts contained in the file to build the Web page that is sent to the browser. In addition to server-side scripts, ASP files can contain HTML (including related client-side scripts) as well as calls to COM components that perform a variety of tasks, such as connecting to a database or processing business logic.

3.1.2. Active Server Pages

Microsoft® Active Server Pages (ASP) is a server-side scripting environment that you can use to create and run dynamic, interactive Web server applications. With ASP, you can combine HTML pages, script commands, and COM components to create interactive Web pages or powerful Web-based applications, which are easy to develop and modify.

3.1.3 For the HTML Author

If you are an HTML author, you will find that server-side scripts written in ASP are an easy way to begin creating more complex, real-world Web applications. If you have ever wanted to store HTML form information in a database, personalize Web sites according to visitor preferences, or use different HTML features based on the browser, you will find that ASP provides a compelling solution. For example, previously, to process user input on the Web server you would have had to learn a language such as Perl or C to build a conventional Common Gateway Interface (CGI) application. With ASP, however, you can collect HTML form information and pass it to a database using simple server-side scripts embedded directly in your HTML documents. If you are already familiar with scripting languages such as Microsoft VBScript or Microsoft® JScript® (JScript is the Microsoft implementation of the ECMA 262 language specification), you will have little trouble learning ASP.

3.1.4 For the Experienced Web Scripted

Since ASP is designed to be language-neutral, if you are skilled at a scripting language such as VBScript, JScript, or PERL, you already know how to use Active Server Pages. What more, in your ASP pages you can use any scripting language for which you have installed a COM compliant scripting engine. ASP comes with VBScript and JScript scripting engines, but you can also install scripting engines for PERL, REXX, and Python, which are available through third-party vendors.

3.1.5 For the Web Developer and Programmer

If you develop back-end Web applications in a programming language, such as Visual Basic, C++, or Java, you will find ASP a flexible way to quickly create Web applications. Besides adding scripts to create an engaging HTML interface for your application, you can build your own COM components. You can encapsulate your application's business logic into reusable modules that you can call from a script, from another component, or from another program.

3.1.5 The Active Server Pages Model

A server-side script begins to run when a browser requests an .asp file from your Web server. Your Web server then calls ASP, which processes the requested file from top to bottom, executes any script commands, and sends a Web page to the browser.

Because your scripts run on the server rather than on the client, your Web server does all the work involved in generating the HTML pages sent to browsers. Server-side scripts cannot be readily copied because only the result of the script is returned to the browser. Users cannot view the script commands that created the page they are viewing.

3.1.6 Building ASP Pages

ASP provides a powerful and extensible framework for creating server-side scripts with any COM compliant scripting or programming language. This section is intended to teach the fundamentals of using a scripting language to create an .asp file. You will learn how to accomplish a wide range of basic programming tasks, from creating a loop to manipulating a database and processing transactions. Whether you are a beginning or experienced scripted, you can envision the topics in this section as development goals, that is, as demonstrations intended to encourage you by suggesting more sophisticated ways in which you can utilize ASP. This can lead to applications that perform better and are more maintainable.

3.1.7 Creating an ASP Page

An Active Server Pages (ASP) file is a text file with the extension .asp that contains any combination of the following:

Text

HTML tags

Server-side scripts

A quick way to create an .asp file is to rename your HTML files by replacing the existing .htm or .html file name extension with an .asp extension. If your file does not contain any ASP functionality, then the server dispenses with the ASP script processing and efficiently sends the file to the client. As a Web developer, this affords you tremendous flexibility because you can assign your files .asp extensions, even if you do not plan on adding ASP functionality until later.

To publish an .asp file on the Web, save the new file in a virtual directory on your Web site (be sure that the directory has Script or Execute permission enabled). Next, request the file with your browser by typing in the file's URL. (Remember, ASP pages must be served, so you cannot request an .asp file by typing in its physical path.) After the file loads in your browser, you will notice that the server has returned an HTML page. This may seem strange at first, but remember that the server parses and executes all ASP server-side scripts prior to sending the file. The user will always receive standard HTML. You can use any text editor to create .asp files. As you progress, you may find it more productive to use an editor with enhanced support for ASP, such as Microsoft® Visual InterDev™. (For more information, visit the Microsoft Visual InterDev Web site at http://msdn.microsoft.com/vinterdev/.)

3.1.8 Adding Server-Side Script Commands

A server-side script is a series of instructions used to sequentially issue commands to the Web server. (If you have developed Web sites previously, then you are probably familiar with client-side scripts, which run on the Web browser.) In .asp files, scripts are differentiated from text and HTML by delimiters. A delimiter is a character or sequence of characters that marks the beginning or end of a unit. In the case of HTML, these delimiters are the less than (<) and greater than (>) symbols, which enclose HTML tags. ASP uses the delimiters <% and %> to enclose script commands. Within the delimiters, you can include any command that is valid for the scripting language you are using. The following example shows a simple HTML page that contains a script command

3.2 What are active server pages:

Asp is a powerful tool for crating server-based technology for creating dynamic web pages. Asp is a scripting language which to create a dynamic web pages. Asp is a module that we attach on the web server that is process and it change into html before sending the server.

3.3 How works the ASP:

Asp is the tool for crating a web page. Asp is written in notepad and sends the information on the server.

- First we write the instruction on the text auditor.
- Than it will save at the extension .asp.

- The instruction is passed from the browser to the server.
- The server locates the file.
- The server will generate the HTML code.
- The server sends the code to the browser.
- The server process the HTML code and display the output.

3.4 OBJECTS OF ASP:

There are only six objects in asp and we work in these objects.

- The application object.
- The asp error object
- The request object
- The response object
- The server object.
- The session object

3.4.1 The application objects:

The application object is to be used when the asp .dll is loaded first request for an asp. it is used for storing the variables and the object on the page.

<Object>

Application. Contents.

3.4.2 The ASP Error Object:

This object is new in asp. This object is used to arrange for detailed information about the last error that is used in asp.

Asp error code

Asp error description

Asp error source.

3.4.3 The Request Object:

The request object is used that the script all the information that the client provide when requesting a page.

REQUEST. WRITE.

3.4.4 The Response Object:

The object is used to access the response that is being sent back to the client. It provides a sequence that is creating the returned page.

RESPONSE.WRITE

3.4.5 The Server Object:

This object is providing a method and the properties that is used for scripting with ASP.

SERVER.OBJECT.

3.4.6 The Session Object:

This object is crated for each visitors when the the first request in asp page from the site until the default timeout period. It stores the information and the variables just visitors open the page during the lifetime.

SESSION.STATICOBJECTS

3.5 Draw Backs of ASP:

- Asp is only scripting language.
- Asp is based on the six objects. All the work is done under these objects.
- There is no facility to make there own objects.
- There are no classes in ASP.
- Any web server is used to run the ASP..
- Restricted working environment.
- The code changes in ASP the whole project will be disturbed because of un object oriented nature.
- ASP has not multi compiler facilities.
- ASP is less compatible to programming language like ASP.NET OR VB.NET.

3.6 Summery:

In this chapter we introduce the ASP and the definition of ASP. And also how the ASP works and defined the ASP objects and the drawbacks of the ASP.

CHAPTER 4

MICROSOFT .NET

4.1 INTRODUCTION:

. Net is a new technology in this era, which is introduced by the Microsoft Company. The purpose of this is to introduce the programming language which is too much easy and that is too much facilitating for the user. The programming of this technology is too much short and its output is fast. In .net programming every time the page is viewed. . Net solve the many problems, which is faced by the programmer.

4.2 What is .Net:

. Net is an independent programming language. In .net programming you can use the different programming languages such as C++, VB and C SHARP. The .net framework supports the many runtime programs but also support the anther programs. The application is run on the machine and its code is accessed on the web.

4.3 Objectives of. NET:

Following are the objectives provide by the .net framework, which is provide a facilities to the user.

- . Net provide an object-oriented environment for the user.
- It is responsible for executing the Microsoft intermediate language code on the machine.
- Classes are defined which can easily built on the application.
- . Net exposes itself on the web, IIS is used to manage the Code that can be compiled into the full .Net program.
- It supports both the web application and the window application.
- The .Net code is used to reduce the software and the version problems.
- The .Net code is beneficial for eliminating the scripting environment.

4.4 _The .NET Framework:

The .is a language component library and provide the execution environment. The .Net framework builds service-oriented environment applications and also needs the current Internet business. The application that is together information from a wide variety of sources languages in use. That is the structure of the .net framework.

WEB SERVICES. WINDOW UI.

COMMON LANGUAGE RUNTIME.

. NET FRAMEWORK CLASS LIBRARY.

ADO.NET: DATA AND XML

4.5 Components of the .Net framework:

There are three major components of the .Net.

- Common language runtime.
- Classes.
- Ado .net: data and xml.

4.5.1 Common language runtime:

. Net is a common language runtime plate form. Common language runtime is the foundation of the. NET.comman language run time manages the code of the .NET.

Common language runtime means that we can use the different programming languages such as vb and C++ in the programmed

If we use the different programming languages in the program and at the compilation time we shell not use the different compiler for the debugging of the program. Because the .NET complier compile the each program at one's time. It means the all the program like C++, VB.NET is to be compiled by the .NET compiler at one time

Suppose we can write the program in C++ and the objects to be made in the VB.NET we can easily call the objects which is in the VB.NET in the C++ program.

4.5.2 . NET CLASSES:

. Net classes is another major component of the .net framework... NET class library cover multiple functions in a program. There are many codes hidden in the class library. This code we can perform to include in the program to simplify all the tasks.

If we want to display the picture in a program, for this purpose we perform a many operations, the class library provide a number of functions that are group together under the namespace. Namespace are used by the .NET together the classes in functionality in a similar group. For example all the jobs are directly involved with the producing of cement (owner, employee, manager) are classified with the owner. Like that jobs involving the packing and the transferring it can be classified as being with in the production department.

We can also import these classes in the asp.net programs by simply add the directive at the top of the file. For example if use the classes in the program that name is abbasi.drawing. Abbasi.drawing defined the classes we just only write that

<% Import namespace = abbasi.drawing%>

This means that abbasi.drawing class is to be import in the page. There are many types of classes in the .NET framework these classes help to simply access the data from other sources. If more classes are access in the application it is very slow at the time of work. . NET class browser is a great tool in the .NET classes. This browser gives the list the main framework classes defined on what is to machine run this form. if class browser application is available as part of the quick start along with the .NET framework SDK if these are installed we able to run this application it locally

http://localhost/class browser/awa.aspx.

4.5.3 Ado .Net: data and xml:

Ado .net is an abbreviation of active x data object technology. Ado .Net is related with the xml, that allowing transferring datasets between the various components.

Asp .net build on the programming classes of the .NET framework, they providing the web application model in the form of a set of controls that simple to build web applications. Developers have access to a rich set of asp.net web controls that encapsulate common HTTP, user interface and drop down manues.these control actually run on the web server and simply project user interfaces html to a browser. On the server the control exposes brings to the object-oriented programming developer. By using XML web

services, developers can simply write their business logic and the ASO.NET is responsible for delivering that service through the soap protocol.

4.6 Need for .Net:

There are many problems for the programmer in a programming. For this purposes it feels that .NET technology is introduced which solve the problems of the programmer.

- . NET is designed to well work on the Internet.
- . NET need to support the XML web service.
- In the .NET when the XML web service is used data easily access from other applications.
- Any change is possible on the application.
- It is very reliable in case of upgrading the application.

4.7 Benefits of .NET:

The .net framework is designed to achieve the following benefits.

4.7.1 Simple development:

The .NET framework makes development is more improve and solving the problems. At this time. NET framework support the many languages such as C++, VISUAL BASIC, JSCRIPT etc. it depends upon the developer which kind of language is used in the application. We can use the VISUAL BASIC .NET classes from with in the visual basic 6. These components develop in different languages are interoperable such as cross language inheritance. These components debuggers work with all supported languages.

4.7.2 Based on web.

The .NET FRAMWORK fully supported the internet technology including hyper text markup language(HTTP) and extension markup language(XML), simple object access protocol(SOAP) FOR TRANSFORMATIONS. The .NET framework favors loosely connected XML web service.

4.7.3 Designed using programming model:

The full functionality of the .NET class is available from any .NET language.

4.7.4 Easy for developers using:

In the .net framework code is managed into classes with in heraricaI name space. The .NET framework provides a commonly type system referred to unified system that is used by the .NET compatible language. in the unified all the languages elements are objects. There is no variant type is to be used. String data and string type are Unicode.

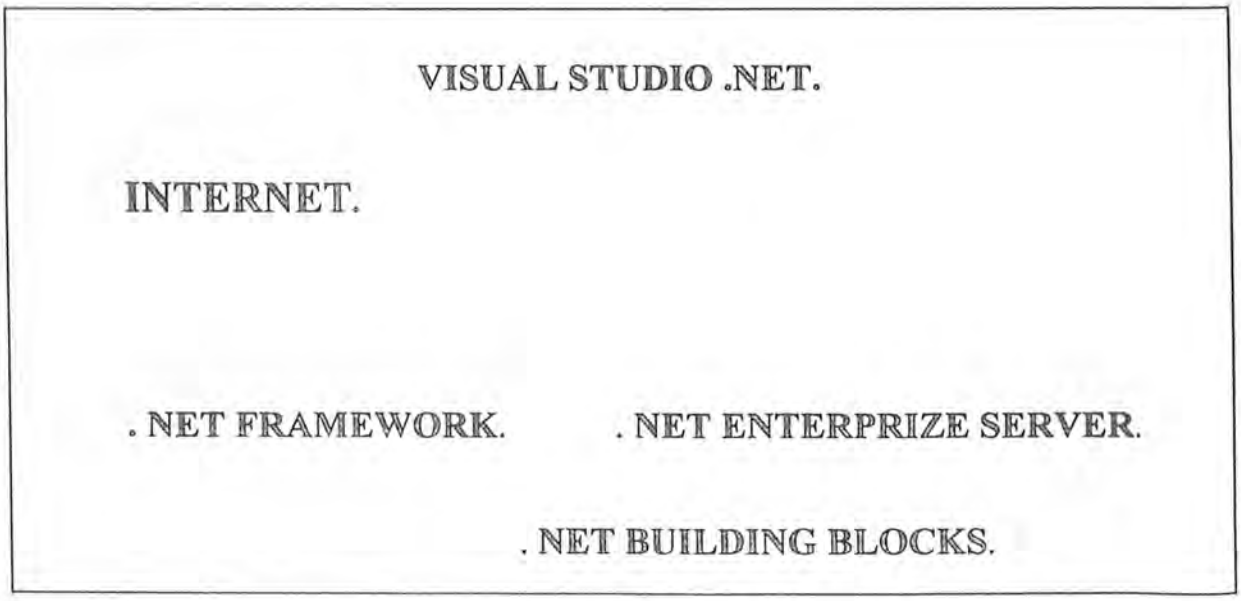
4.7.5 Extension of classes:

The classes are in the .NET framework is not hidden from the developer. You can access and also extend the classes through inheritance. You can also implement multiple language n heritance.

4.7.6 Easy to run and maintain:

The .NET framework is very simple to run and maintain. The application developed is robust, secure and self-describing. Multiple components are also run side by side by without affecting each other.

4.8 THE .NET PLATE FORM:



Microsoft .Net consists of the following core technology

- THE .NET FRAMEWORK.
- THE .NET ENTERPRISE SERVERS.
- BUILDING BLOCK SERVERS.

4.8.1 The Visual Studio .Net:

Microsoft visual studio .net provides a high-level development environment for building application s on the .net framework. it provides key enabling technologies to simplify the creation, deployment , and the highly available web application and the xml web service

4.8.2 .NET FRAMEWORK:

The .net framework is a language, which provide the library component and the execution environment for building the web application.

4.8.3 .NET ENTERPRISE SERVER:

THE .NET enterprise servers are Microsoft's comprehensive family for server application for building, deploying, and managed the web-based solution. Following are the various servers.

- Microsoft window 2000 server.
- Microsoft application server
- Microsoft commerce server
- Microsoft exchange server.
- Microsoft Internet security and acceleration server.

CHAPTER 5

ASP.NET

5.1 INTRODUCTION:

Asp.net is a new and powerful technology, which is introduced by the Microsoft. It is the combination of two major technologies ASP and .NET.

ASP is an old version in which it has provided a fast and affective way of creating dynamic web pages. And the .NET is a new technology in which all programming development is conducted in the future and the companies carry out there business. It becomes a complete with a wide range of predefined control ready to us to use in the project saving the time and making you more productive.ASP.NET allows to store information about the site in a database of self describing XML file. one of the feature is that we can create an application so we write them in VB.NET ,JSCRIPT or even the combination of them all a you shell it choose the best language for the job.

5.2 WHAT IS ASP.NET:

ASP.NET is a new and powerful server based technology for creating a dynamic web page. ASP.NET is comprised on the .NET framework. ASP.NET is a full fledged programming language to create a dynamic web page supported by the .NET framework.

5.3 DIFFERENCE BETWEEN ASP AND ASP.NET:

SRNO	ASP	ASP.NET
1.	ASP is a scripting language	ASP.NET is a programming language.
2.	ASP runs at any browser.	ASP.NET runs at the internet information services. (IIS).
3.	ASP has no concept of object oriented programming tool.	ASP.NET is a object oriented programming language.
4.	ASP not supports the web page.	ASP.NET support the web problems.
5.	In ASP only one language is used at a time.	ASP.NET more than one language is used at a time. Like vb.net.cCsharp.
6.	The environment in ASP is restricted.	The environment in ASP.NET is very expanded.
7.	There is no choice to make a new object	A new object is created in asp.net his own.
8.	There is no concept of classes.	A number of classes is defined in **ASP.NET.**
9.	ASP used a different compiler for every language	ASP.NET is only one compiler that is .NET compiler.
10.	If an changing in the project the whole project will be changed	If any change there is only effected by the changing line.

5.4 OBJECTS IN ASP.NET:

IN ASP.NET the same object is to be used which is to be used in ASP.

The response object
The request. object.
The session object.
The error object
The server object.
The application object.

These objects are the same. But the main feature is that we make also new object itself in the application.

5.5 ASP.NET Server Controls:

ASP.NET server controls are components that run on the server and encapsulate user-interface and other related functionality. They are used in ASP.NET pages and code-behind classes. They can detect browser capabilities and provide appropriate output for them.

Server controls are declared within an. **aspx** file using custom tags that contain a run at="server" attribute value. The run at="server" attribute enables server-side events and view state maintenance for controls. If you do not set the run at attribute to "server", the control works as a plain HTML control. ASP.NET server controls must be closed using standard XML syntax; that is, closing a tag with /> or using a separate closing tag, **</tag name>**. In addition, they must be inside **a <form run at="server"> < /form>.**

The server controls provide properties, methods, and events for use in your applications. The controls are programmable in your server code, allowing you to interact with those controls and their data. For instance, to access a text box you could use the following code:

My Name = textMyName. Text.

Many of the controls support data binding so you can quickly link data to a control.

When you create Web Forms, you can use the following types of controls:

5.6 HTML server controls:

HTML server controls expose an object model that maps very closely to the HTML elements they render

5.7 Web 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 the object model does not necessarily reflect HTML syntax.

5.8 Validation controls.

You attach a validation control to an input control to test what the user enters for that input control. Validation controls are provided to allow you to check for a required field, to test against a specific value or pattern of characters, and so on.

5.9 User controls:

Controls that you create as Web Forms pages. You can embed Web Forms User controls in other Web Forms pages, which is an easy way to create menus, toolbars, and other reusable elements.

The following figure illustrates a simple page built with server controls. It uses text boxes, buttons, images, and calendar controls. In this example, the calendar is rendered with the Calendar Web server control, which completes all necessary HTML or server coding for you.

5.10 Web Server Controls:

Web server controls include traditional user-entry controls as well as special-purpose controls such as a calendar. All Web server controls can be bound to a data source. The names and descriptions of some of the Web server controls are given in the table below.

Control	Description
Label	Displays text that users can't directly edit.
TextBox	Displays text entered at design time that can be edited by users at runtime or changed programmatically.
Drop Down List	Allows users to select one option from a list or enter text.
List Box	Displays a list of choices. This type of list allows multiple selections.
Image	Displays an image.
Ad Rotator	Displays a sequence (pre-defined or random) of images.
Checkbox List	Creates a grouping of check boxes.
Radio Button	Displays a single button that users click to enable or disable the control.

5.11 HTML Server Controls:

HTML server controls map directly to HTML elements. The controls listed in this section are pre-defined controls. However, any HTML element can be made into a control that supports a generic set properties and methods. All HTML server controls can be bound to a data source. The names and descriptions of some of the Web server controls are given in the table below.

Control	Description
Html Form	Defines an HTML form. The values of controls within the form are posted to the server when the form is submitted.
Html Input Text	Displays text entered at design time that can be edited by users at run time, or changed programmatically. This control can also be used to create password boxes that display their values as asterisks (*).
Html Text Area	Displays a large amount of text.
HtmlAnchor	Creates Web navigation.
Html Button	Performs a task you assign to it. This control can contain any type of HTML code, making it very flexible.

5.12 Validation Controls:

Validation controls offer a way to check user input in Web or
HTML server controls.

The following table lists the types of validation controls you can use in Web Forms

Type of validation	Control to use	Description
Required entry	Required Field Valuator	Ensures that the user does not skip a required entry.
Comparison to a value (including database fields)	Compare Validated	Compares a user's entry against a constant value or a property value of another control using a comparison operator (less than, equal to, greater than, and so on). To compare with a database field, you specify an expression that gets the user data to compare with the database field.
Range checking	RangeValidator	Verifies that a user's entry is between specified lower and upper boundaries. You can check ranges within pairs of numbers, alphabetic characters, and dates. Boundaries can be expressed as constants or as values derived from another control.
Pattern matching	Regular Expression Validated	Checks that the entry matches a pattern defined by a regular expression. This type of validation allows you to check for predictable sequences of characters, such as those in social security numbers, e-mail addresses, telephone numbers, postal codes, and so on.
User-defined	Custom Validated	Checks the user's entry using validation logic that you code yourself. This type of validation

		allows you to check for values derived at run time.

5.13 Custom User Controls:

As a developer, you need easy ways to create custom user controls in Web applications. For instance, you can create a header control that contains the entire header for your applications, so that those applications can reuse the header, and you have to maintain only one copy.

Custom user controls offer you an easy way to partition and reuse common UI functionality across your ASP.NET Web applications. Like a Web Forms page, you can author these controls with any text editor or develop them using code-behind classes. Also, like a Web Forms page, custom controls are compiled upon first request and stored in server memory, which reduces the response time for subsequent requests. Unlike pages, however, custom controls cannot be requested independently. They must be included in a Web Forms page. They do not have to contain any code; however, they can contain any HTML code except a BODY or HTML tag.

Custom user controls also provide greater flexibility than server-side includes (SSIs), because they can use the ASP.NET object-model support. Rather than simply including the functionality provided by another file, you can program against properties you declare in the control, just like any other ASP.NET server control.

Custom user controls can contain other controls, images, or any valid HTML code. For instance, in a header control you can place a property that allows you to set the header for the application

5.14 Properties in ASP.NET Server Controls:

5.14.1 Custom Server Controls:

Properties are similar to public variables, but they have access or methods that wrap their data. Because properties allow data hiding, you should expose properties instead of public fields from your controls. The accessory functions can perform additional program logic, in addition to setting or retrieving a property.

The following list describes some of the commonly accessed properties of a control.

ID

A user-supplied identifier for a control.

Controls-

The collection of child controls.

Page

-What contains the control.

Parent

-The control to which a Controls collection belongs. (Control A is a Control B's parent if B is an element of A. Controls).

View State

-A data structure that is sent to the client and back and is generally used for persisting form data across round trips. View State is of type State Bag, which stores data as name/value pairs.

Unique-

The unique identifier assigned to a control by the ASP.NET page

framework.

Visible-Determines whether a control is visible.

5.15 Adding Controls in a Web Form:

5.15.1 Objective

To identify what is a Web form and to add simple controls to the form.

5.15.2 Scenario

In this practice, you will create a new project in Visual Studio .NET, which will be an ASP.NET Web application. You will see how a Web form is created. Then, you will add various types of controls on this Web form. For each control you add, you will identify the code that is automatically added to the HTML page.

5.15.3 Procedure

Notice that a new Web Form, WebForm1.aspx opens in the Design

Drag an Image control onto the page. Position and size it so it looks the way you want.

Click the HTML tab at the bottom of the page to see the HTML code. You will see the following code:

<Asp: Image id="Image1" style="Z-INDEX: 101; LEFT: 323px; POSITION: absolute; TOP: 173px" run at="server"></asp: Image>

In Windows Explorer, navigate to <Drive letter>:/Practice/Module05, and select all .gif files. Drag the files from Windows Explorer and drop them into the Solution Explorer.

Click the Design tab.

Select the Image control in the Design page. Then, in the Properties window, click the Ellipsis (...) for the Imager attribute, select the dotnet.gif file, and then click OK.

Next, you will add a Hyperlink control. From the Web Forms tab in the Toolbox, drag a Hyperlink control to the Design page.

Click the HTML tab. You will see the following code:

<Asp: Hyperlink id="HyperLink1" style="Z-INDEX: 102; LEFT: 417px; POSITION: absolute; TOP: 178px" run at="server">Hyperlink</asp: Hyperlink>

Click the Design tab.

Select the Hyperlink control in the Design page. Then, in the Properties window, change the Text attribute to Microsoft.NET

In the Properties window, add a Web address to the NavigateUrl attribute. Either use the Ellipsis (...) and specify the path as http://www.microsoft.net or simply type the following in the field next to NavigateUrl attribute:

NavigateUrl=http://www.microsoft.net

Click the HTML tab. Notice that the complete tag for the hyperlink looks as follows:

```
<Asp: Hyperlink id="HyperLink1" style="Z-INDEX: 102; LEFT: 417px; POSITION:
absolute; TOP: 178px" run at="server" NavigateUrl="http://www.microsoft.net
">Microsoft. NET</asp: Hyperlink>
```

Save the file.

5.16 To build and run the project

On the Build menu, click Build.

The result is displayed in the Output window.

To verify that your Web application will run in a browser, click the Design tab, right-click anywhere in the Design page, and then choose View In Browser.

5.17 CUSTOMER User Controls:

As a developer, you need easy ways to create custom user controls in Web applications. For instance, you can create a header control that contains the entire header for your applications, so that those applications can reuse the header, and you have to maintain only one copy.

Custom user controls offer you an easy way to partition and reuse common UI functionality across your ASP.NET Web applications. Like a Web Forms page, you can author these controls with any text editor or develop them using code-behind classes. Also, like a Web Forms page, custom controls are compiled upon first request and stored in server memory, which reduces the response time for subsequent requests.

5.18 Web Forms:

Web Forms are an ASP.NET technology that you use to create dynamic Web pages. They can present information, using any markup language, to the user in any browser and use code on the server to implement application logic.

Web Forms: :

- □ Can run on any browser and automatically render the correct, browser-compliant HTML for features such as styles, layout, and so on. Alternatively, you can design your Web form to run on a specific browser such as Microsoft Internet Explorer 5.0 and take advantage of the features of a rich browser client.
- □ Can be programmed in any Common Language Specification (CLS) language, including Visual Basic, C#, and JScript .NET.
- □ Are built on the Common Language Runtime (CLR), so they provide all the benefits of a managed execution environment, type safety, inheritance, and dynamic compilation for improved performance.
- □ Support WYSIWYG editing tools and powerful RAD development tools, such as Microsoft Visual Studio .NET, for designing and programming your forms.
- □ Support a rich set of server controls that allows developers to cleanly encapsulate page logic into reusable components and declaratively handle page events.
- □ Allow for separation between code and content on a page, eliminating the "spaghetti-code" often found in ASP pages.
- □ Provide a set of view state management features that preserve the view of a page between requests.
- □ Are extensible with user-created and third party controls.

Components of Web Forms

- ❑ Web Forms divide the Web applications user interface into two parts:
- ❑ Visual components
- ❑ Application logic

- ❑ The visual components:

- ❑ Web Forms consist of a file containing markup and elements specific to Web Forms. This file is referred to as the page. The page works as a container for the text and controls you want to display. Using any HTML editor and Web Forms Server Controls, you can lay out the form as you want. The page is a file named with the extension.aspx.

5.19 Application logic:

Web Forms consists of code that you create to interact with the form. You can choose that the programming logic reside in the. aspx file or in a separate file (referred to as a "code-behind" file), written in Visual Basic, C#, or any other CLR-supported language. When you run the form, the code-behind class file runs and dynamically produce the output for your page.

5.20 Web Forms Request Processing Cycle

The Web Forms request processing cycle is described as follows:

- When a browser client requests. aspx resources, the ASP.NET runtime parses and compiles the requested .aspx file into a .NET Framework class. This class is used to dy manically process incoming requests.
- Although you create Web Forms from separate components, they form a unit. When the Web Forms unit is compiled, ASP.NET parses the page and its code, generates a new class dynamically, and then compiles the new class. The dynamically generated class is derived from the ASP.NET Page class, but is extended with controls, your code, and the static HTML text in the .aspx file.
- This new Page class becomes a single assembly that is executed on the server whenever the Web Forms page is requested. At run time, the Page class processes incoming requests and responds by dynamically creating HTML and streaming it back to the browser. If the page contains Web controls, the derived Page class creates instances of the controls at runtime and likewise renders their HTML text to the stream.

5.21 Web Form Page Processing Cycle:

During Web Forms processing, the Web Forms page goes through a number of distinct stages. At each stage, the Web Forms processor calls a corresponding page-processing method and runs the code for that method. These page-processing methods provide you with entry points-hooks-that you can use to work with the contents of the form.

The following illustrates the common stages of page processing.

5.22 Page load Event

- □ The Page load event is raised when the page and control view state is restored. The following tasks can be performed in this stage:
- □ Use the IsPostBack property of the page to check if this is the first time the page is being processed.
- □ Perform data binding the first time the page is processed, or reevaluate data-binding expressions on subsequent round trips.
- □ Read and update control properties.
- □ Restore state saved from a previous client request during the Save stage.

5.23 Event handling

- □ Event handling is raised if the page is called in response to a form event.
- □ You can include the following application-specific processing within this stage

- □ Handle the specific event.
- □ If the page contains validation controls, check the Is Valid property for the page or individual validation controls.
- □ Manually save the state of global page variables that you are maintaining yourself and of controls you dynamically added to the page.

5.24 Page unload

- This event is raised when the page has finished rendering and is ready to be discarded. In this stage, perform final cleanup tasks such as the following:
- Closing files.
- Closing database connections.
- Discarding objects.

Note:

It is important that resources be explicitly closed. Otherwise, they will remain open until the next garbage collection occurs. On a heavily loaded server, too many open resources can exhaust memory.

CHAPTER 6

VISUAL BASIC

6.1 INTRODUCTION:

Microsoft visual basic is the newest version. It allows us to quickly and easily develops windows application for the PC without an expert in c++ or other programming languages. Visual basic provides a graphical environment in which we visually design the forms and controls that become the building blocks of our application.

Visual basic supports many useful tools that design to develop an application. Visual basic especially designed to utilized the internet it comes with several controls that allow you to create web page application. These work just like standing alone visual basic application but they are access through the micro soft Internet explorer using this application we can revise the existing application and distribute them through the Internet

6.2 What is visual basic:

Visual basic give a graphical environment in which we design the forms and controls, which the application is to be based on. The visual basic tools help us to develop the application in a short period of time. Visual basic is used on the Internet in which it is easy for the computer user to jump for creating application with visual basic.

6.3 Working with form:

The most basic object that we will be working in a visual basic is the form object, which is the visual foundation of our application, is basically a window that we can add different aliment to in order to create a complete application. Every application is based on the some type of form.

- To create a new **form opens the visual basic.**
- Select the file manus.
- New project.
- Window will be open.
- Then select the visual basic form and work on it.

6.4 Options in the form:

6.4.1 Working with form Properties:

The properties describe the characteristics of an object. This can be used to manipulate the identify the object. Every visual basic has at least one property, but the most have many more the following properties used in the form.

Active control property:
Back color
Border style.
Max button.
Control box.
Font size etc.

There are some example of the properties which is used in the form but there are many properties used in the form. We can also help about the properties if are not familiar that which type of work is to be perform by the property.

6.4.2 Working with Forms event

Events are triggered by the messages when we click the button window will generate the message that describe the action which we perform there are many events in the form such as

Activate.

Click.

Double click.

Drag drop

Mouse move.

Load.

These events are very important which is used in the form.

I F any information about the event click the button f2 and it will give the every information about the event which type of work is to be perform this event.

6.4.3 Command button:

A command control button is one of the most common controls found in the application. We can use a command button to simple response from the user to perform special functions on forms. some example of the command button.

6.4.4 Properties of command button:

List of the properties which is used in the command control

Cancel.

Default.

Enabled.

Drag icon.

Tab index.

When we use in a form one button can have its default property set to a true at a time.

6.5 Text box:

Visual basic project involves at least one text box control。Text boxes are commonly used for accepting the user input or entering the data. some example of the text boxes.

Properties of text box:

List of the properties, which are used in the text box .

Multi line.

Password char.

Max length

Font italic

Data changed

6.6 Check boxes:

A check box control is just like the option button. Check boxes are valid as single control. A single option is probably counter intuitive. Check boxes is in three forms such as **on**, **off** or **grayed.**

Check box properties:

List the properties of the check boxes.

Alignment.
Caption.
Enabled.
Tab index.
Tool tip text

Using labels. A label control display the text
Option buttons. It allows the user to select the button in one.
Frame control. In this we can add the frame.
Combo box. It combines the text box and a list box.
Image objects. it work when we make a picture on the form.

Timers. it works when its time is to be come.

CHAPTER 7

VISUALBASIC.NET

7.1 Introduction:

Visual basic .net are a dialect of the basic computer language, which was not developing to help the computer but it also help the people, which they make a program. Visual basic .net replaces many of the machine language with words and symbols more easily. Visual basic .net enables the programmers to visually assemble a program ' window form parts in a toolbox. we can easily write the program in visual basic .net. The visual basic is a logical and easy to understand language. With a few practice we can make the useful programs and games in the visual basic. Net. Visual basic .net are interested and simple language in which we can easily make the application without any wastage of time. Visual basic comprise of two components such as Vb and .net.

7.2 What is visual basic .Net

There are several significant changes in Visual Basic .NET. For example, Visual Basic .NET allows you to write applications that can perform multiple tasks independently using a process known as multi-threading. Multi-threading allows you to write applications that are more responsive to user input, because you can cause complicated tasks to run on threads that are separate from your user interface.

Other changes have been made to Visual Basic to ensure it behaves as other languages that support the .NET platform. For example, arrays are now zero-based in Visual Basic- as they are in C#, and C++. Visual Basic .NET now supports object-oriented programming features such as inheritance. Visual Basic .NET also supports structured exception handling. In addition, Visual Basic .NET provides several ways of reducing programming errors

Visual basic .net is a new and power full programming language.

The visual basic.net provides us a wast environment for making any application. Every thing, which is to be needed which require us for making an application is already make. We just click the button and writ the text in a button. The compilation process is too much easy rather than other languages. We can use more then one languages in a program. And at the time of compilation there is no need of different compiler because the .NET complier have an ability to compile the every language such as ASP.NET , VB.NET AND CSHARP.

7.3 REVEALING THE CODE

When we start a new project the visual basic .net does not display the code in the window form. To bring the code in the window double click on the window form. A one more windows will appear on the window form. The visual
basic .net is guess that what we want to do.

7.4 Menus:

Menus an important part in the visual basic .net main window. The manus bar holds the series of menus, each of which contains commands that we need in a visual basic. Net. If we want to add a new project in a window we select the project menus add f windows form command.

Many o the menus command called hot keys that we can use to select the command directly from the keyboard. Press alt + shift+A listed after the new item command. In this we have no need to open the all menus.

7.5 Toolbars:

The toolbar provides access to many of the most commonly needed visual basic .NET commands. Most of the buttons on the toolbar are just quick ways to select a command from the menus. Every toolbar have s separate function in an application.e.g if we want to save the application click the button in the tool bar save icon then the visual basic.net save the project. We can access many commands from the toolbar. some commands in the toolbar is called the hotkeys that we select a command directly from the key board. All command in the toolbar has the same icon next to its con used on the toolbar. When we see that the icon like this next to a command we know that the command is available on the toolbar.

7.6. Difference between vb and vb .net.

SRNO	VISUAL BASIC	VISUAL BASIC .NET
1	IT have no concept of object oriented programming	Visual basic.net provide the facility of object-oriented programming.
2.	The data type is predefined	The data type is predefined and also makes self-created.
3.	It use the different compiler for compiling the application	In visual basic.net only one compiler is used for compiling the application that is **.NET COMPILER.**
4.	The capacity of storing of information in Data type is very low.	The capacity of storing of information in data type is very extended.
5.	Only one language is used for making the application	More than one language is used in the application which application is belong to .net family.

7.7 Object Oriented Programming:

7.7.1 Inheritance.

Visual Basic .NET supports inheritance by allowing you to define classes that serve as the basis for derived classes. Derived classes inherit-and cans extend-the properties, methods, and events of the base class. Derived classes can also override inherited methods with new implementations. All classes created with Visual Basic .NET are inheritable by default. Because the Windows forms you design are really classes, you can use visual inheritance to define new forms based on existing forms.

7.7.2 Overloading.

Visual Basic .NET now supports overloading. Overloading is the ability to define properties, methods, or procedures that have the same name but use different parameters (including different data types). Overloaded procedures allow you to provide as many implementations as necessary to handle different kinds of data, while appearing to be a single, versatile procedure.

7.7.3 Overriding.

Visual Basic .NET now supports overriding. The Overrides keyword allows derived objects to override characteristics inherited from base objects. Overridden methods have the same arguments but different implementations than the methods inherited from the base class. The new implementation of a method can call the original implementation in the parent class by specifying My Base before the method name.

7.7.4 Interfaces

Visual Basic .NET also supports interfaces. Interfaces describe the properties and methods of classes, but unlike classes, interfaces do not provide implementations. In Visual Basic .NET, the Interface keyword allows you to declare interfaces, while the Implements keyword lets you write code that puts into practice items described in the interface.

7.7.5 Delegates.

Visual Basic .NET provides built-in support for delegates. Delegates are objects that can call the methods of objects on your behalf. For this reason, delegates are sometimes described as type-safe, object-oriented function pointers. You can use delegates in procedures to specify an event-handler method to run when an event occurs. Delegates are also used with free threading.

7.7.6 Shared Members.

Shared members are also called instance members (they are known as static members in other languages). You can have instance members in Visual Basic .NET. Shared members are properties, procedures, and fields that are shared by all instances of the class. Sharing a single instance of a data member or function among all instances of a class can be useful to programmers using your objects.

7.8 Structured Exception Handling

Implementing comprehensive error handling in Visual Basic has always been a challenge. A consistent error-handling scheme meant a lot of duplicated code. Error handling using On Error GoTo statements sometimes slowed down the development and maintenance of large applications. Handling several errors with various combinations of Resume and next quickly resulted in illegible code, and, when execution paths weren't completely planned, it often led to bugs. Here is an example of code that used the On Error GoTo statement:

Visual Basic now supports structured exception handling, using an enhanced version of the Try...Catch...Finally syntax supported by other languages, such as C++. Structured exception handling combines a modern control structure, similar to Visual Basic's Select Case or while statements, that include exceptions, protected blocks of code, and filters. Structured exception handling makes it easy to create and maintain programs with robust, comprehensive error handlers.

The Try statement manages code between Try and first the Catch statement. The Catch statement filters errors, which are normally derived from a common System. Exception. The Finally statement always executes and can be used for clean up.

Note:

Visual Basic .NET still supports the On Error Go To statement. However, using this statement will result in significantly slower execution.

7.9 Reducing Programming Errors

7.9.1 Garbage Collection

When using the .NET Framework, you no longer need to worry about memory management-garbage collection takes care of it for you. The CLR checks for objects that are no longer used at regular intervals, and garbage collects them. The CLR does not use reference counting to govern object lifetime. Instead, a background garbage collection task traces object references and identifies objects that can no longer be reached by running code.

With garbage collection, you no longer have to worry about circular references. If a group of objects contains references to each other, but all references outside the group have been set to nothing, garbage collection will automatically reclaim the memory. You do not have to worry about memory leakages.

An additional advantage of traced garbage collection is that allocation of new objects in memory is extremely fast. Performance is further improved by the elimination of the COM AddRef and Release mechanisms, and objects require less memory.

7.9.2 Constructors and Destructors

Constructors are procedures that control initialization of new instances of a class. Conversely, destructors are methods that are used to free system resources when a class leaves scope or is set to Nothing. Used together, constructors and destructors support the creation of robust and predictable class libraries.

The Sub New and Sub Destruct procedures in Visual Basic .NET replace the Class Initialize and Class Terminate methods used in previous versions of Visual Basic to initialize and destroy objects.

7.9.3 Variable Initialization

Variables can now be initialized during declaration. This results in simplified, compact code.

7.9.4 Syntax Checking.

Visual Studio .NET, the development environment for Visual Basic .NET, provides excellent support for writing programs. It uses IntelliSense® extensively to help programmers write valid code. IntelliSense displays syntax tips for statements as you type them, and it provides code completion on keywords. For example, when you type goto and a space, IntelliSense displays a list of the defined labels.

7.9.5 Type Checking

Type checking, performed at compile time, catches statements using an improper data type before they can cause subtle execution errors at runtime.

7.9.6 Type Conversion

The process of changing a value from one type to another type is called type conversion. Type conversions can be widening or narrowing. Widening conversions never fail, and they can be accomplished with no loss of information. Narrowing conversions might fail, and they always involve some kind of information loss.

Visual Basic .NET expects explicit type conversions. For example, you can no longer put a Long value into an Int type variable. Strict type checking helps prevent accidental data loss. It also prevents implicit conversions that might fail at runtime. However, you can disable strict type checking by disabling Option Strict on a per project basis.

7.9.7 Namespaces.

Namespaces prevent naming conflicts by organizing classes, interfaces, and methods into hierarchies.

CHAPTER 8

VISUALSTUDIO .NET

8.1 Visual Studio .NET:

Microsoft® Visual Studio® .NET is a suite of multi-language programming tools. It provides a complete development environment for building on the Microsoft .NET platform.

Using Visual Studio .NET, you can build Web applications and XML Web services that render in any browser and on virtually any device. By using the language of your choice, you can leverage your existing investments in skills and systems. The result is increased productivity, end-to-end Web development, and a shorter time to delivery.

After completing this module, you will be able to:

- Define the design goals of Microsoft Visual Studio .NET.
- Describe the Visual Studio .NET tools.
- Identify and describe the components of the .NET development environment.
- Describe the productivity features of Visual Studio .NET.

8.2 Next Generation Internet

The new features of Visual Studio .NET make it a complete development environment for building applications for the Microsoft .NET Framework, Microsoft's next generation Web application platform. It provides key enabling technologies to simplify the creation and deployment of secure, reliable, scalable, highly available XML Web services while using existing developer skills.

In addition, the .NET Framework provides features to help Web developers use XML Web services as if they were local objects in the developers' preferred development language to simplify service and application development. The result is faster time to market, improved developer productivity, and ultimately, higher quality software.

8.3 Design Goals

Visual Studio .NET increases developer productivity and helps businesses address the demands of a rapidly changing and competitive marketplace. The fundamental idea behind Visual Studio .NET is to simplify the development of powerful and reliable enterprise Web solutions by offering end-to-end Web development capabilities and scalable and reusable server-side components

The design goals for Visual Studio .NET are:

- Maximize developer productivity

For developers to maximize productivity, Visual Studio .NET provides tools that can adapt to an everchanging set of individual, team, and application requirements. It provides a rich model for customizing, automating, and extending the integrated development environment (IDE). The developers get appropriate assistance through features such as dynamic Help, enhanced IntelliSense®, and automated routine development tasks.

- Simplify server-based development

One of the most complex tasks in developing distributed applications is building server-side components that implement the business logic of an application. Visual Studio .NET makes building server-based components easier by leveraging Rapid Application Development (RAD) principles and applying them to component development

- Deliver powerful design tools

With Visual Studio, you can leverage a familiar programming approach for a broad range of user interfaces, including browser, mobile devices, and rich

Windows clients. Shared HTML, XML, and stylesheet editors make it easy to develop Web applications from any Visual Studio language, including the new C#.

Web Forms is part of the new Microsoft .NET Framework and leverages many new technologies, including a common application framework, managed execution environment, integrated security, and object-oriented design principles. In addition, Windows Forms offers full support for quickly and easily connecting to XML Web services and building rich, data-aware applications based on the ADO.NET data model.

8.4 RAD for the Next Generation Internet

Rapid Application Development (RAD) tools provide the ability to develop and deploy applications quickly by automating much of the development process and eliminating repetitive tasks.

Developers using Visual Studio RAD for the server tools can gain time to market with fast access to XML Web services. Developers can drag and drop reusable server components to a design canvas, set properties for those objects and then double-click to write code. This drag-and-drop metaphor for programming is already familiar to millions of Visual Studio programmers who will be able to immediately leverage their skills in building scalable enterprise and Web-based applications.

8.4.1 RAD Support for Creating Web Forms Pages

Visual Studio includes a forms designer for creating HTML and ASP.NET-based forms. The designer allows you to drag HTML elements, controls, data classes, and other objects onto your form, use the **Properties** window to work with them, and use RAD techniques for adding code to them.

8.4.2 RAD Support for Creating XML Web Services

For creating XML Web services, Visual Studio includes a project template, in which you can expose methods for public access on the Web. You can create XML Web services using Visual Basic, C#, or Visual C++.

8.4.3 RAD Support for Accessing XML Web Services

Visual Studio contains several features that make it easy for you to locate existing XML Web services and make use of them in your applications. These include Server Explorer, which displays available XML Web services on a particular server, and the Add Web Reference dialog box, which can be used to navigate services and add the necessary proxies to your project to access them.

8.4.4 Enhanced RAD Support for Creating Your Own Component

The RAD support makes authoring controls and components as easy as creating forms. Use inheritance to start from an existing component, and add other components to build in the functionality you need. Inheritance enables you to create a class that's identical to some existing class, with a few minor specializations.

8.5 XML Web Services Support:

The use of XML, an open standard managed by the World Wide Web Consortium (W3C), removes barriers to data sharing and software integration. XML makes it easy to exchange data, and .NET software gives users the ability to work with the data once it's received.

Support for XML Web services is deeply integrated into Visual Studio .NET. You can easily expose any method using an XML Web Service and consume XML Web services as if they were COM objects. XML Web services can be running on any platform and any hardware, so a customer can expose an IBM mainframe application as an XML Web service and enable their developers to code against it using all of the key productivity features in Visual Studio .NET, such as IntelliSense.

In Visual Studio .NET, you can easily expose any function, written in any language, as an XML Web service. There is no need to learn XML and SOAP to take advantage of XML Web services. When you compile your business objects, Visual Studio .NET will automatically generate an XML file that describes the function, and when it is called, the function will automatically send and receive XML packets.

After the Web Service has been built, both the compiled code and the XML file describing the public methods of the service are published to the Web server. The XML Web service can now be invoked using HTTP, and XML will automatically be used to pass data to and from the service.

In Visual Studio .NET, you can drag any exposed XML Web service directly into your application. Doing so enables Visual Studio to treat the XML Web service as a class. Calling the XML Web service is as simple as creating a new instance of the XML Web service class and then calling its exposed methods.

8.6 XML Data Access With ADO.NET:

ADO.NET is the successor to ADO. It is a W3C standard-based programming model for creating distributed, data-sharing applications. ADO.NET is an application programming interface (API) to access data and information.

In ADO, XML is nothing more than a mere input and output format. However, in ADO.NET, XML is the data format that gives you the means of manipulating, reorganizing, sharing, and transferring your data. ADO.NET utilizes XML as the universal transmission format. This guarantees interoperability. Any pair of software components can share ADO.NET data, as long as they agree to use the same XML schema for the format of the transmitted data.

ADO.NET **DataSets** are the evolution of ADO **RecordSets**. A **DataSet** is simply a disconnected, in-memory view of the database. It can contain any number of tables, each of which corresponds to a database table or view. A table, which corresponds to a **DataTable** object, is simply a collection of rows and columns. A **DataSet** reads and writes data and schema as XML documents. Both data and schema are transportable through HTTP and can be used on any platform that understands XML.

An ADO.NET **DataSet** is a sort of super RecordSet that provides:

- A better and richer programming interface to host data.
- An object model that offers a more generalized view of the data without any intrinsic binding with the actual data source.

- An XML-based standard console for input and output.

ADO.NET provides platform interoperability and scalable data access. Because XML is the format for transmitting data, any application that can read the XML format can process data. ADO.NET **DataSets** allow you to save content to XML and rebuild it from an external XML file.

ADO.NET also encapsulates the OLE DB API and allows applications to consume data from OLE DB data sources. This includes data stored in many different formats, not only SQL databases.

8.7 Web Development Environment:

When designing applications that involve a user interface, you have two choices: Windows Forms and Web Forms. Both technologies can provide a rich user interface and advanced application functionality to solve business problems. However, you can make your choice depending upon the business requirement.

For example, if you are creating an e-commerce Web site that will be accessible to the public over the Internet, you would develop the application using Web Forms. On the other hand, for building a processing-intensive, highly responsive application that would take advantage of the full functionality of the client machine, you would use Windows Forms.

8.8 Web Forms:

Web Forms are used to create applications in which the primary user interface is a browser. This includes applications intended to be available publicly via the World Wide Web, such as e-commerce applications.

Modeled after Visual Basic's Forms, Web Forms allow developers building form-based desktop applications to rapidly develop cross-platform, cross-browser, programmable Web applications using the very same techniques already used in Visual Basic.

Web Forms make building Web applications as easy as building Visual Basic forms based applications. A standard Web Forms page consists of an HTML file containing the visual representation of the page and a source file with event-handling code.

Developers visually design their Web forms applications and then implement the business logic with Visual Basic, C++, or C# - completely separated from the UI.

The source is compiled into executable code, providing fast runtime performance. The code compiles and executes on the server for maximum performance and scalability. The performance of Web Forms is thus greater than previously achieved with interpreted code and ASP . Additionally, Web Forms are more maintainable because they cleanly separate user interface (the HTML file) from code (a class file).

Developers can also take advantage of higher end features in newer browsers such as Internet Explorer™ 5 or Netscape™ 5, or can reduce functionality to support wireless devices using Wireless Application Protocol (WAP).

Because Web development deeply permeates Visual Studio .NET, the functionality originally found in Visual InterDev® is now a core part of the environment itself and is accessible from the various language products. Regardless of the language chosen for development, there is now just one environment to learn, configure, and use.

8.9 Windows Forms:

Windows Forms library is part of the Microsoft .NET Framework and leverages many new technologies, including a common application framework, managed execution environment, integrated security, and object-oriented design principles. In addition, Windows Forms offers full support for quickly and easily connecting to Web Services and building rich, data-aware applications based on the ADO.NET data model. With the new shared development environment in Visual Studio .NET, developers will be able to create Windows Forms applications using any of the languages supporting the .NET platform, including Microsoft Visual Basic and C#.

Windows Forms are used to develop applications that rely on the power of the desktop computer for processing and high-performance content display. These include classic Win32 desktop applications, such as drawing or graphics applications, data-entry systems, point-of-sale systems, and games.

Some Windows Forms applications might be entirely self-contained and perform all application processing on the user's computer. Others might be part of a larger system and use the desktop computer primarily for processing user input. For example, a point-of-sale system often requires a responsive, sophisticated user interface that is created on the desktop computer but is linked to other components that perform back-end processing.

Because a Windows-based application that uses Windows Forms is built around a Windows framework, it has access to system resources on the client computer, including local files, the Windows Registry, the printer, and so on. This level of access

can be restricted to eliminate any security risks or potential problems that arise from unwanted access. Additionally, Windows Forms can take advantage of the .NET graphics device interface (GDI+) classes to create graphically rich applications, which are often a required for data-mining and game applications.

With a Windows Forms application, there is no need to deploy an application to the end user's desktop. Instead, a user can call the application simply by typing a URL in a browser. The application will download to the client machine, run in a secure execution environment, and remove itself upon completion.

8.10 Practice: Building a New Application:

8.10.1 Objective

To show the creation of a simple Web directory and then build a new application in Visual Studio .NET.

8.10.2 Scenario

Visual Studio .NET represents a complete development environment in the .NET Framework for building applications on the .NET platform. Using Visual Studio .NET, developers can build applications for Web, Windows, or portable devices.

The Microsoft .NET Framework transforms application development into XML Web services development with a fully managed, protected, and feature-rich application execution environment, simplified development and deployment, and seamless integration with a wide variety of languages.

8.10.3 Procedure

To create a new application directory in IIS

1. To open Internet Services Manager, right-click the **My Computer** icon on your desktop, and click **Manage** in the drop-down menu.

2. In the **Computer Management** window, expand **Services and Applications** by double-clicking it.

3. Expand **Internet Information Services** by double-clicking it.

4. Expand the node for your server.

5. Right-click the **Default Web Site** node, point to New, and then click **Virtual Directory**.

6. The **Virtual Directory Wizard** appears. Click **Next**.

7. Type **Module03** for the application in the **Alias** box, and then click **Next**.

8. Type <*Drive Letter*>:**\folder\Module\** in the **Directory** box, and then click **Next**.

9. To accept the selected permissions, click **Next**.

10. Click **Finish**. The Virtual Directory is set up.

To create a page and add HTML using the HTML editor

Basic page editing tools are available with the Web Forms Designer. You can use the editor to add static HTML and text to your .aspx page.

One of the advantages of using the Visual Studio .NET editor to build new projects is that all of the infrastructure tasks are completed for you. When you create a new project, the editor creates a virtual directory in IIS and makes sure the alias and physical directories are correctly matched. Basic permissions are set, and the application is generally ready to run.

To create a new Web Forms page and add HTML and text:

1. Open Microsoft Visual Studio .NET.

2. On the **File** menu, point to New, and then click **Project**.

3. In the **New Project** dialog box, select **Visual Basic Projects** in the **Project Types** pane.

4. Select **ASP.NET Web Application** in the **Templates** pane.

5. Type **WebApplication1** in the **Name** box, and click **OK**.

> **Note** The Web server must have IIS version 5 or later and the .NET Framework installed on it.

6. Notice that a new Web Forms page, **WebForm1,** is created. Also, the **WebForm1.aspx** page is displayed in **Solution Explorer**.

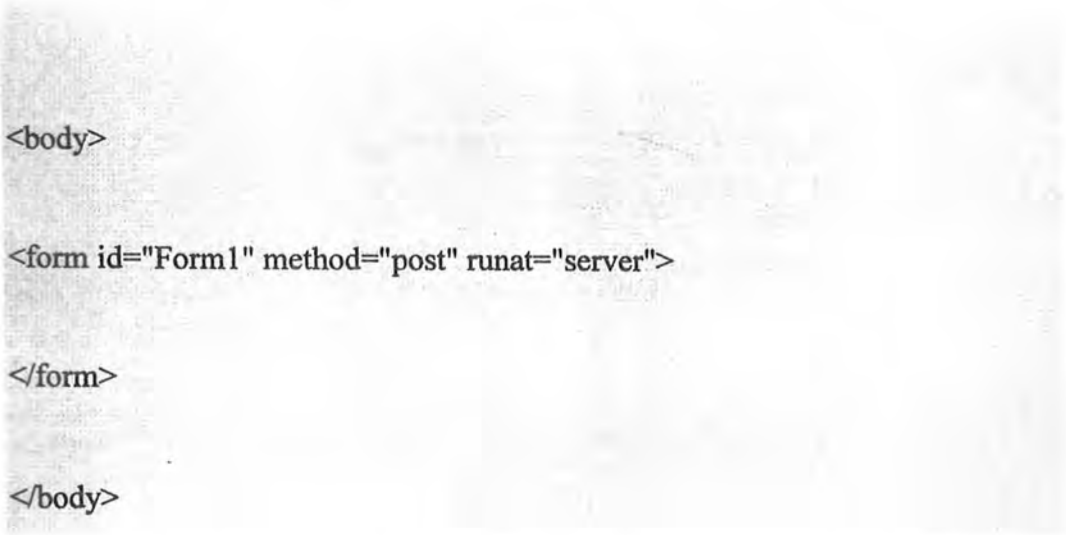
7. The page you are working on is in grid layout mode. All new Web Forms pages added to a project are created in this mode. In grid layout, objects are arranged using absolute (*x* and *y*) coordinates.

8. In the GridLayout mode, the .aspx page receives an MS_POSITIONING attribute which is set to GridLayout within your HTML body markup. Click the **HTML** tab to see the following markup:

```
9. <body MS_POSITIONING="GridLayout">

10. <form id="WebForm1" method="post" runat="server">

11. </form>

12. </body>
```

13. If you want to arrange objects from top to bottom, as in a word processing document, you need to use flow layout. To select flow layout, in the **Properties** window, change the value of the **pageLayout** attribute to **FlowLayout.**

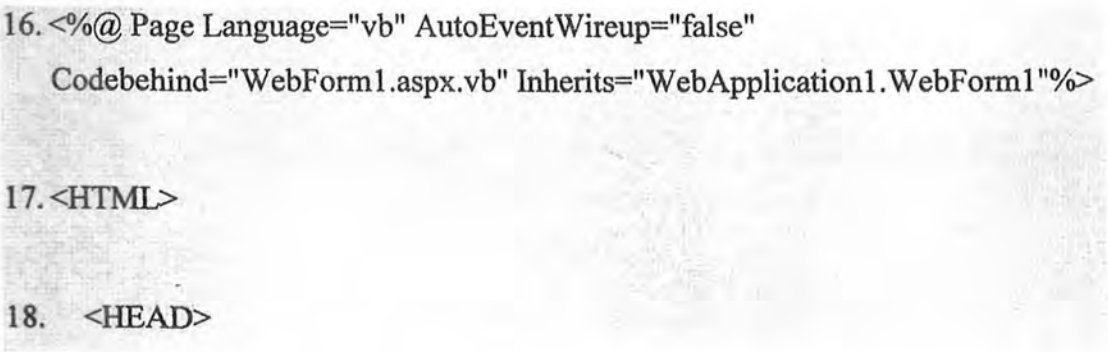
    This instructs the designer to remove the MS_POSITIONING attribute from your HTML body markup. This markup should now look like this:



```
<body>

<form id="Form1" method="post" runat="server">

</form>

</body>
```

14. Next, click the **Design** tab and type the following text directly on the page: This is a simple sentence. Here is another simple sentence.

15. Click the HTML tag to see the code for the text you added in step 8. The result should look like this:



```
16. <%@ Page Language="vb" AutoEventWireup="false"
    Codebehind="WebForm1.aspx.vb" Inherits="WebApplication1.WebForm1"%>

17. <HTML>

18.   <HEAD>
```

```
    <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
    <meta name="GENERATOR" content="Microsoft Visual Studio .NET 7.0">
    <meta name="CODE_LANGUAGE" content="Visual Basic 7.0">
    <meta name=vs_defaultClientScript content="JavaScript">
    <meta name=vs_targetSchema content="Internet Explorer 5.0">
</HEAD>
<BODY>
  <FORM id="Form1" method="post" runat="server">
    <P>
      This is a simple sentence.  Here is another simple sentence.
    </P>
  </FORM>
</BODY>
</HTML>
```

Next, switch to Design view and add a logo to the page. On the **File** menu, click **Add Existing Item**.

33. Browse to *<Drive Letter>*:\folder\Module, select file **northwindlogo.gif,** and click **Open**.

34 Drag **northwindlogo.gif** from **Solution Explorer,** and drop it onto the form.

35 From the **Web Forms** tab in the Toolbox, drag and drop the **TextBox** control onto the form.

37 Switch to the HTML view and check the code.

You can highlight any text, select a new element type such as Heading 1 from the Block Format drop-down list, and use the buttons in the Formatting toolbar to set the color, font, font size, or other characteristics of the text. Check the HTML code after doing these changes.

8.11 Mobile Computing:

Today, Internet access is fast becoming a standard feature on practically all mobile computing appliances.

As the diversity of client devices and network technology increases, enterprises require application servers that can easily scale and adapt across different networks, devices, and applications. The new Microsoft™ Mobile Information™ Server is a

scalable and reliable mobile application server that gives users secure, real-time access to information from mobile devices.

Microsoft Mobile Information Server - one of the Microsoft .NET Enterprise Servers - extends the reach of new and existing applications by providing an infrastructure to customize interactions based on the device used and the user's location and personal preferences.

Mobile Information Server allows carriers and enterprises to deploy customizable mobile applications and empowers users to control their communications by deciding how, where, when, and with what device they will access their information and applications.

8.11.1 Mobile Internet Toolkit

The Microsoft® Mobile Internet Toolkit™ is integrated into Visual Studio .NET and gives you the same code development, debugging, and code management environment that you get for building desktop Web applications. If the Mobile Internet Toolkit is not installed on your machine, you can download it from the Microsoft Developer Network (MSDN) Web site using the link on the **What's New** tab on the Visual Studio .NET **Start** page.

8.11.2 Mobile Internet Designer

Mobile Internet Designer™ is an extension to the Microsoft Visual Studio .NET Integrated Development Environment (IDE). The designer provides a comprehensive, flexible, and powerful environment to create mobile Web applications for wireless devices. This development environment provides the standard Microsoft Visual Studio design tools, such as drag and drop capabilities, forms creation, code windows (HTML view and Code view), Microsoft IntelliSense, dynamic Help, and debugging tools.

The designer allows you to interactively take mobile Web forms controls from the Toolbox, switch to the HTML view to customize the page, and select Code view to customize the code. You can write code in any language that supports the .NET Framework, giving you more flexibility and providing a powerful tool for designing code and targets diverse environments.

8.11.3 Mobile Internet Controls Runtime

Whether you develop wireless applications or device adapters in Visual Studio .NET or the .NET Framework, the Mobile Internet Controls Runtime delivers the richness of the mobile control development tools. Using this technology, you can build applications that run on the server, yet render in a broad variety of mobile Web browsers. The Mobile Internet Controls Runtime extends the programming model of the .NET Framework Software Development Kit (SDK) to mobile devices.

8.12 The .NET Compact Framework:

The .NET Compact Framework will make it easy for developers on smart devices to write applications that access XML Web Services, including building block services like Hailstorm. The .NET Compact Framework is a lightweight, factored, and portable version of the full .NET Framework. It is suitable for virtually any smart device, including mobile phones, PDAs, and set-top boxes.

Just as with the desktop .NET Framework, the .NET Compact Framework is the environment in which managed applications run. The Compact Framework includes the key parts of the desktop .NET Framework architecture such as the common language runtime, managed code execution, and programming development framework.

The .NET Compact Framework provides a small, manageable set of class libraries that are a subset of the classes in the full .NET Framework and are architected for minimal use of expensive system resources such as RAM.

8.13 Visual Studio Productivity Features.

Microsoft Visual Studio .NET enables developers to increase productivity, allowing them to customize the look and feel of the development environment, enhance and extend its functionality, automate repetitive tasks, and integrate Visual Studio .NET with other applications.

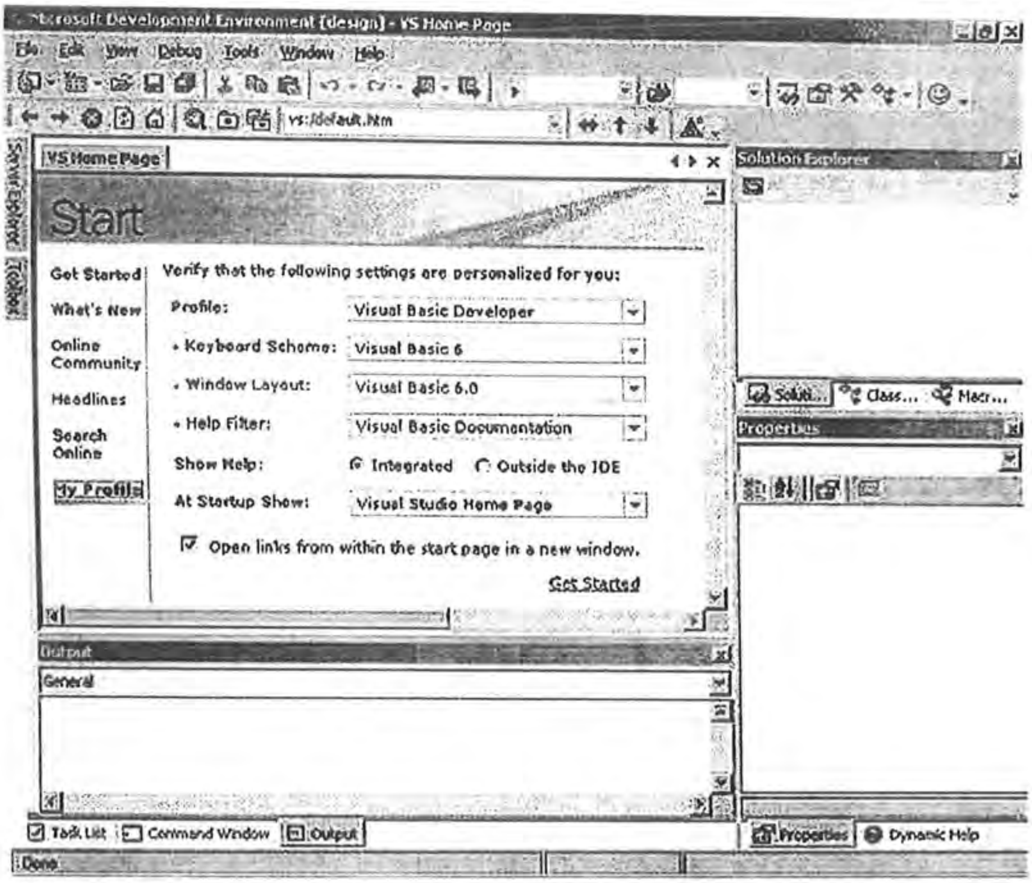
8.13.1 Integrated Development Environment

Microsoft Visual Studio .NET sports a single shared integrated development environment (IDE) for all the languages within it. It was designed to help developers build their solutions faster with less clutter and with all of the tools easily accessible in any of the languages in the Visual Studio® development system. The Visual Studio .NET IDE has a host of features that bring developers information when they need it and how they want it.

A tightly integrated and unified visual environment simplifies the process of developing Web applications and lessens the learning curve. Shared HTML, XML, and stylesheet editors make it easy to develop Web applications from any Visual Studio language, including C#.

8.13.2 Start Page

Each time a developer launches Visual Studio, the Start Page displays (see figure). The default Web browser home page for the IDE, it provides a central location for setting preferred options, reading product news, accessing discussions with fellow developers, and obtaining other information to get up and running within the Visual Studio .NET environment.



Verify that the following settings are personalized for you:

Profile: Visual Basic Developer

• Keyboard Scheme: Visual Basic 6

• Window Layout: Visual Basic 6.0

• Help Filter: Visual Basic Documentation

Show Help: ● Integrated ○ Outside the IDE

At Startup Show: Visual Studio Home Page

☑ Open links from within the start page in a new window.

Get Started

In addition to providing instant access to articles, events, and Help topics from MSDN® Online, the Start Page allows you to access existing and new projects with the click of a button.

The Start Page also enables you to quickly customize the look and feel of the IDE based on their development experience.

8.13.3 Dynamic Help

The Dynamic Help window provides one-click access to pertinent Help regardless of the task a developer is attempting to complete. By tracking the selections a developer makes, the placement of the cursor, and the items in focus within the IDE, Dynamic

Help filters through topics available on MSDN Online and provides pointers to relevant information specific to the development task at hand.

8.13.4 Enhanced IntelliSense

To make the writing of Web pages easier and less prone to errors, IntelliSense technology has been enhanced to handle not only compiled languages but also HTML and Extensible Markup Language (XML). This enables Web developers to get immediate information on available tags, properties, and even values within the code editor. By using IntelliSense within an application, developers get all the benefits of automatic statement completion and syntax notification as they write their code.

8.13.5 Developer Productivity:

To enable customization of nearly every aspect of the environment, Visual Studio .NET exposes a comprehensive object model. Consisting of almost 200 objects, this model provides direct access to the Visual Studio code editor, project hierarchy, code model, debugger, menus and commands, the build process, and the various tool windows that comprise Visual Studio .NET, including the task list and toolbox.

8.13.6 Interface Objects

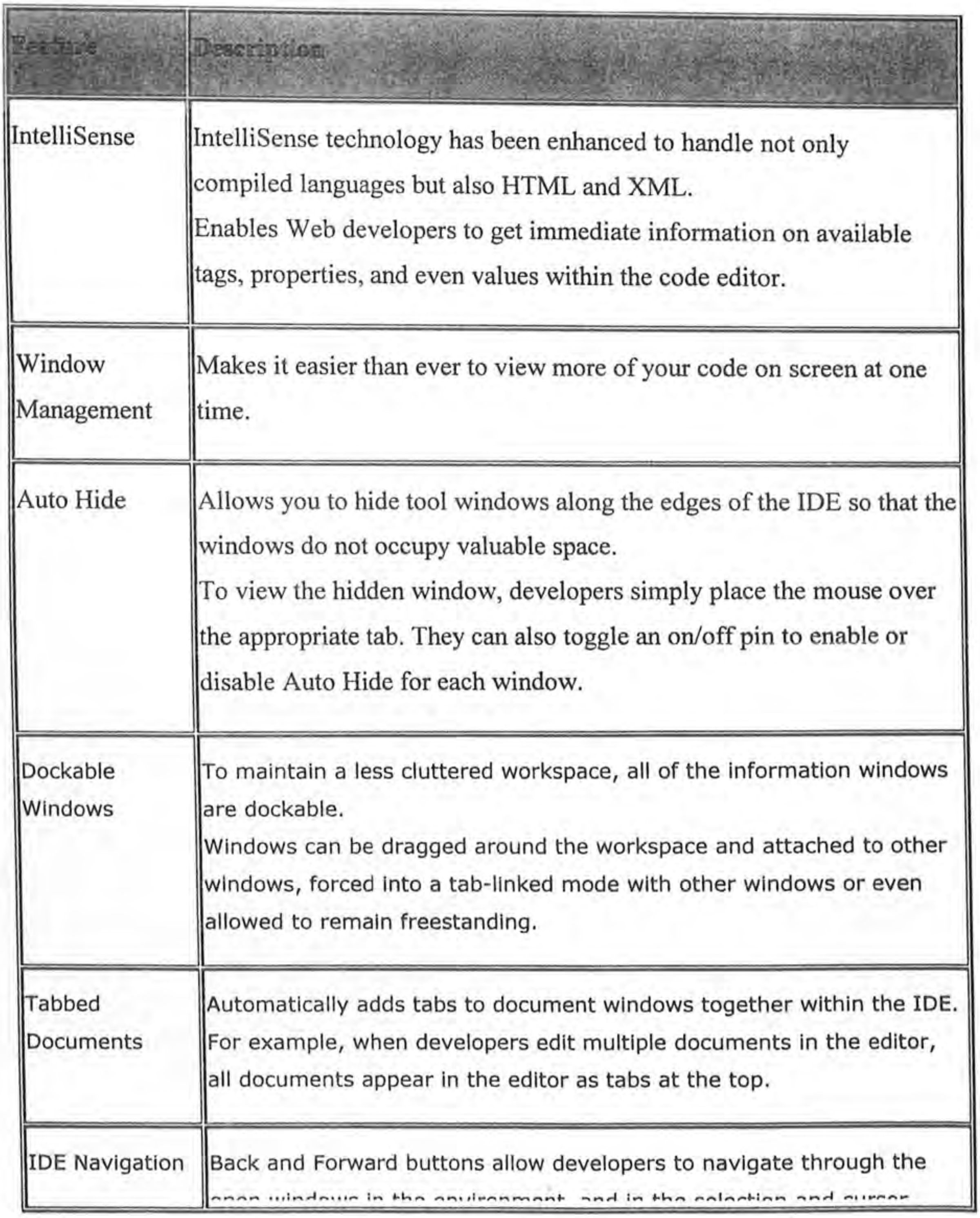
The following table describes the objects of the interface.

Object	Description

Enhanced Toolbox	Displays a variety of items for use in Visual Studio projects. The items change depending upon which developer is using. Items displayed can include Web and Windows-based form controls, ActiveX® controls, XML Web services, HTML elements and objects, and items from the Windows Clipboard.
Server Explorer	Is a new server-development console for Visual Studio .NET. Is a shared tool window that helps developers access and manipulate resources on any computer for which they have permission.
Task List	Provides code annotation capability by allowing developers to mark their code with specialized comments. These comments are then parsed and displayed in a tabular format within the Task List. Serves as a central location to ascertain the status of compile and build errors and warnings. Double-clicking the task jumps directly to the code containing the comment. Clearing the check mark removes the comment altogether. Developers can filter the task items to see only the items they are interested in.
Command Window	Enables developers to quickly execute Visual Studio commands directly in the Visual Studio .NET environment. Within the window, developers have keyboard access to all commands that may be issued within the IDE. Enables developers to directly interact with the IDE, bypassing the menu system, executing commands that don't appear in the menu, and avoiding dialog boxes by using command parameters, switches, and arguments.

Features of the Interface

The following table explains the features of the interface:



Feature	Description
IntelliSense	IntelliSense technology has been enhanced to handle not only compiled languages but also HTML and XML. Enables Web developers to get immediate information on available tags, properties, and even values within the code editor.
Window Management	Makes it easier than ever to view more of your code on screen at one time.
Auto Hide	Allows you to hide tool windows along the edges of the IDE so that the windows do not occupy valuable space. To view the hidden window, developers simply place the mouse over the appropriate tab. They can also toggle an on/off pin to enable or disable Auto Hide for each window.
Dockable Windows	To maintain a less cluttered workspace, all of the information windows are dockable. Windows can be dragged around the workspace and attached to other windows, forced into a tab-linked mode with other windows or even allowed to remain freestanding.
Tabbed Documents	Automatically adds tabs to document windows together within the IDE. For example, when developers edit multiple documents in the editor, all documents appear in the editor as tabs at the top.
IDE Navigation	Back and Forward buttons allow developers to navigate through the open windows in the environment, and in the selection and cursor

	open windows in the environment, and in the selection and cursor history within files. Both the Back and Forward buttons have a drop-down list that displays the navigation history.
Favorites	Developers can access their Web browser Favorites and add links to the Favorites list from within Visual Studio.
Multimonitor Support	Provides support for multiple monitors so that developers can have more windows open at the same time without sacrificing screen space.
Macros	Enable developers to automate repetitive processes easily and to customize the Visual Studio .NET IDE without creating add-ins. The Visual Studio .NET Macros IDE is a rich environment based on Visual Studio technology for authoring, debugging, and running macros.

8.14 Enterprise Templates

Microsoft provides a broad range of products and technologies that can be used to deploy infrastructure and build applications for running an enterprise. With the complexity of the applications being built, one of the key challenges facing information technology organization is the dilemma of where to begin.

Microsoft has used this feedback to create specific versions of Visual Studio .NET that address the unique requirements of enterprise architects and enterprise developers. Developers require more than an empty project as a starting point. The opposite of this is also a common theme. It's likely that several technologies can be used to solve any given problem, and it is often unclear which is appropriate. These two issues can be characterized as the blank slate (no starting point) and the full slate (too many alternatives).

Microsoft enterprise frameworks and templates provide a powerful way to solve both the blank and the full slate problems. Enterprise frameworks and templates enable

companies to leverage their most experienced people to define development guidance and policy that can be easily used by developers for building applications. Instead of being limited to documents, this guidance can take the form of architectural blueprints, reusable components, and policies or instructions for building applications, all delivered within the Visual Studio .NET environment.

Enterprise templates contain two key components:

1. Initial project structure, or template project - The template project is the solution for the blank slate problem. An architect can now create an application starting point that includes reusable components, projects, and solutions, which developers can use to construct an application.

2. Policy that is associated with the project - The policy component associated with a template project provides the solution to the full slate problem. An architect can specify which technologies should be used as well as which technologies should not be used so that developers can choose from a narrower list of appropriate options at any given time during the development of an application.

8.14.1 Shared Extensible Tools

By using their language of choice, developers can leverage their existing investments in skills and systems. The result is increased productivity, end-to-end Web development, and a shorter time to delivery. The key benefits include the following:

- Visual Studio core language enhancements
- Cross-language, cross machine debugging
- Extensibility and partners

8.15 Visual Studio Core Language Enhancements

The core language enhancements in Visual Studio .NET are:

8.15.1 Visual Basic:

Visual Basic® has been modernized with object-oriented features, structured exception handling, and support for multiple threading models.

8.15.2 Visual C++:

Visual C++® has been enhanced to include support for the XML Web services platform using Managed Extensions. Introduction of attribute-based programming has not only made the model flexible but also enhanced the productivity of the C++ developer.

8.15.3 Visual C#:

Visual C#® (pronounced "see-sharp") is a new, modern, object-oriented language derived from C and C++. It was designed specifically for the .NET Framework and provides a great set of productivity-enhancing features for the C++ developer.

8.16 Cross-Language, Cross-Machine Debugging

Visual Studio .NET contains an enhanced integrated debugger that shortens the development cycle by giving developers an easy way to run, track, and fix errors in their code. Developers can set conditional breakpoints that offer the fastest way to track down programming errors by stopping application execution only when a specified condition is met.

8.16.1 Cross-Language Debugging

The Visual Studio .NET debugger supports debugging of applications written in multiple languages. Cross-language debugging allows developers to step seamlessly between Visual Basic, Visual C++, Visual C#, Managed Extensions for C++, HTML, and Jscript®. Cross-language call stacks make it easy to debug components written in multiple languages.

8.16.2 Cross-Process Debugging

Visual Studio .NET also offers a complete range of cross-process debugging. Because today's applications are increasingly distributed solutions, developers need a way to step remotely from client calls into server calls. In Visual Studio .NET, cross-process debugging allows developers to step instantly from any client-side call to any server-side call. Cross-process debugging works in Web-based solutions such as HTML-hosted applications and in straightforward Windows-based applications.

In addition, the Visual Studio .NET debugger has the ability to attach to a program that is running outside Visual Studio. Developers can use this capability to debug programs not created in Visual Studio, debug multiple programs simultaneously, or debug applications running on a remote computer.

8.17 Extensibility and Partners

The Visual Studio .NET automation model provides a powerful mechanism for customizing, integrating, and automating the development environment. Using the automation model, developers can access a rich object model of nearly 200 objects within the IDE. Developers can leverage this object model through a number of provided interfaces, including add-ins and macros. In addition, developers can use third-party add-ins, or they may take advantage of Visual Studio Integration Partners (VSIP) to extend the functionality of Visual Studio .NET.

8.17.1 Customize the Environment

Whether Visual Studio .NET automation is used to customize the look and feel of the Visual Studio .NET IDE, extend its functionality, automate repetitive tasks, or integrate with other applications, it enables developers to maximize their productivity and develop applications in an environment tailored to their requirements.

For those who want to automate the Visual Studio. NET IDE without writing code, the Visual Studio .NET Macro Recorder provides one-click recording and playback of macro procedures.

8.17.2 Entire IDE is Extensible

For the majority of developers, add-ins, wizards, and macros will provide the rich functionality needed to tailor Visual Studio .NET to their needs. Some users, however, need to go beyond the considerable capabilities of the Visual Studio .NET automation model and achieve deeper integration into the IDE. For example, an organization may want to incorporate a new .NET programming language, such as Perl, Python, or COBOL, into the IDE. This may create a need for any of the following:

- A new project type
- A customized editor
- Advanced debugging

8.17.3 Visual Studio Integration Program (VSIP)

The Visual Studio Integration Program (VSIP) enables partners to integrate their product with the Visual Studio .NET environment by providing access to more interfaces and Help integration. Using VSIP, partners are able to provide greater functionality in their users' design environment by leveraging the Visual Studio .NET IDE.

8.18 Summary

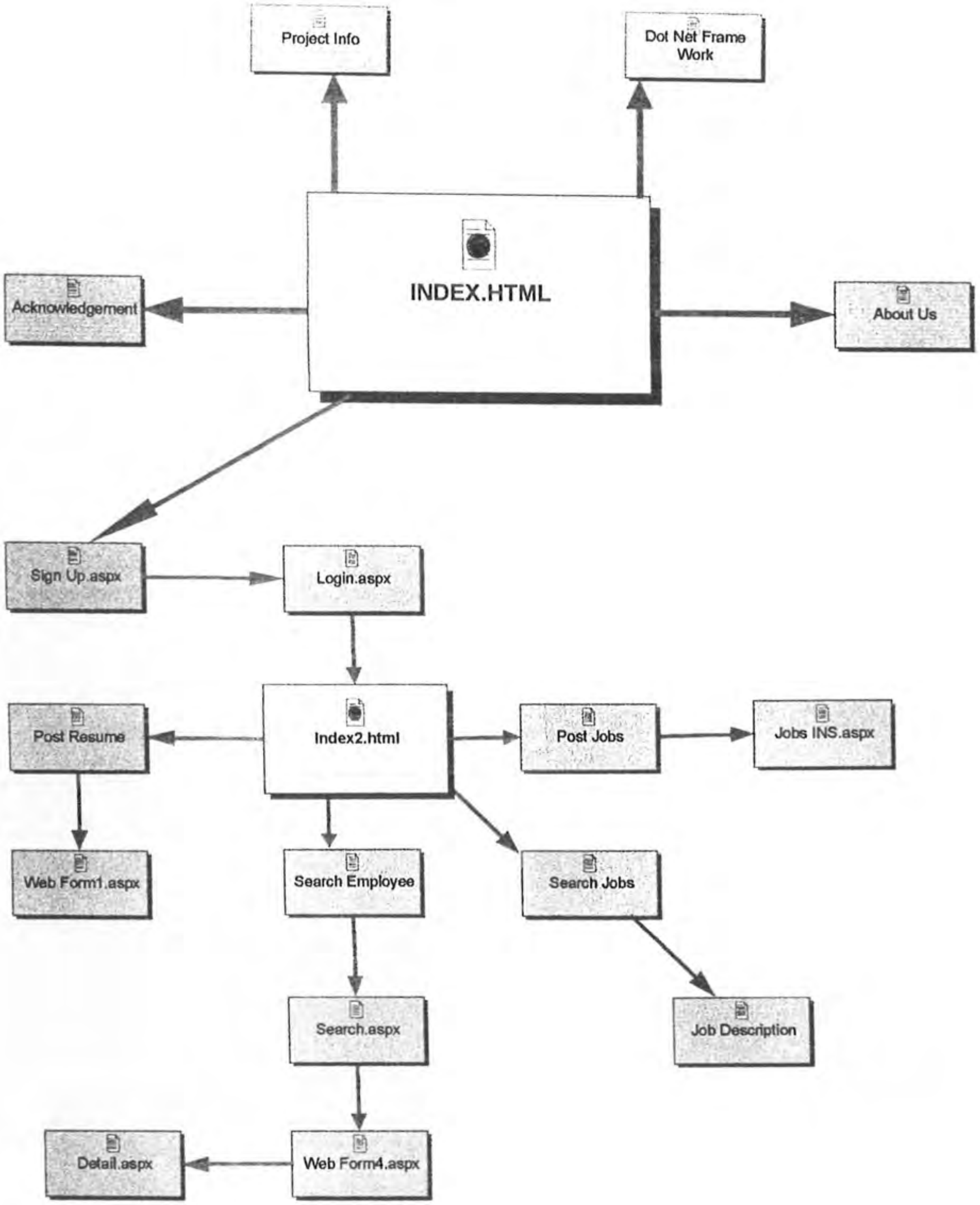
In this chapter, we have seen how to define the design goals of Microsoft Visual Studio .NET. You also got an idea about the Visual Studio .NET tools and the components of the .Net development environment. Finally, describe the productivity features of Visual Studio .NET.

APPENDIX B

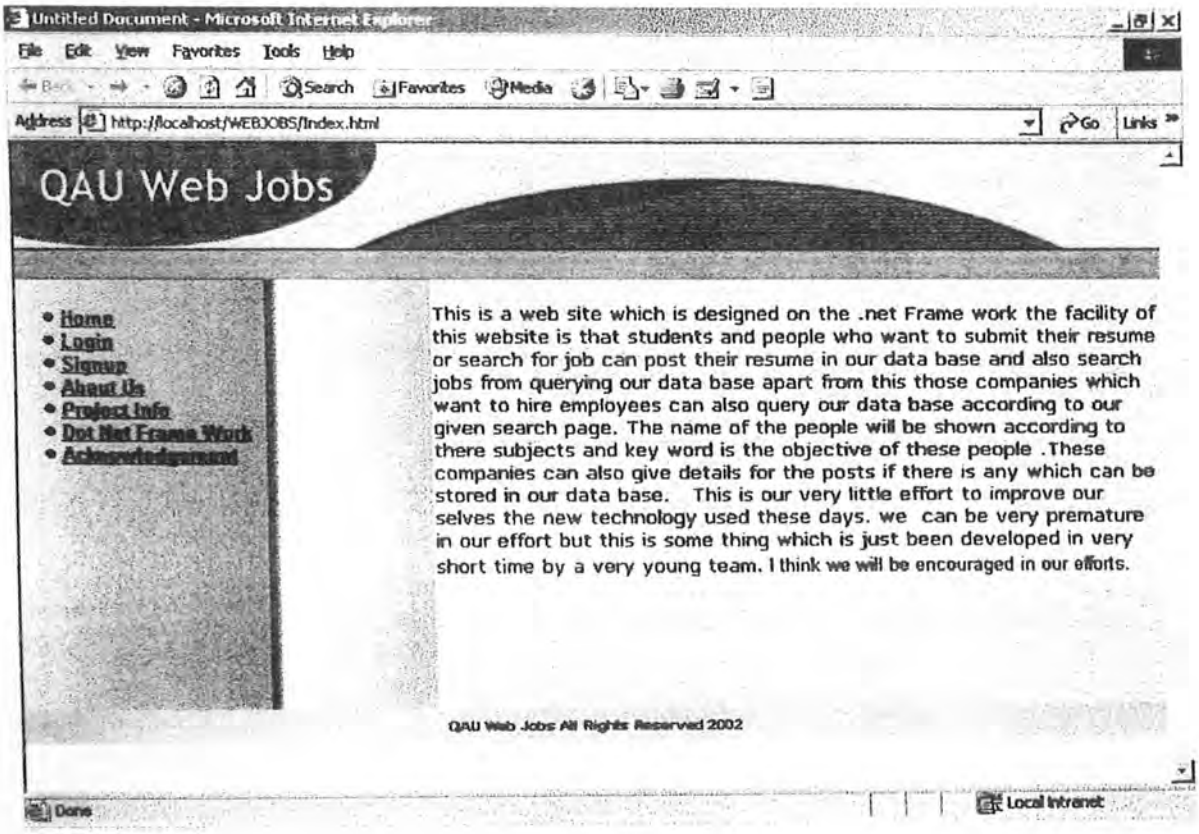
CHAPTER 9

APPLICATION PAGES

Site Map:


Project Info
Dot Net Frame Work
INDEX.HTML
Acknowledgement
About Us
Sign Up.aspx
Login.aspx
Post Resume
Index2.html
Post Jobs
Jobs INS.aspx
Web Form1.aspx
Search Employee
Search Jobs
Search.aspx
Job Description
Detail.aspx
Web Form4.aspx

INDEX.HTML:


Untitled Document - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Index.html
Go Links
QAU Web Jobs
Home
Login
Signup
About Us
Project Info
Dot Net Frame Work
Acknowledgement
Done
Local intranet

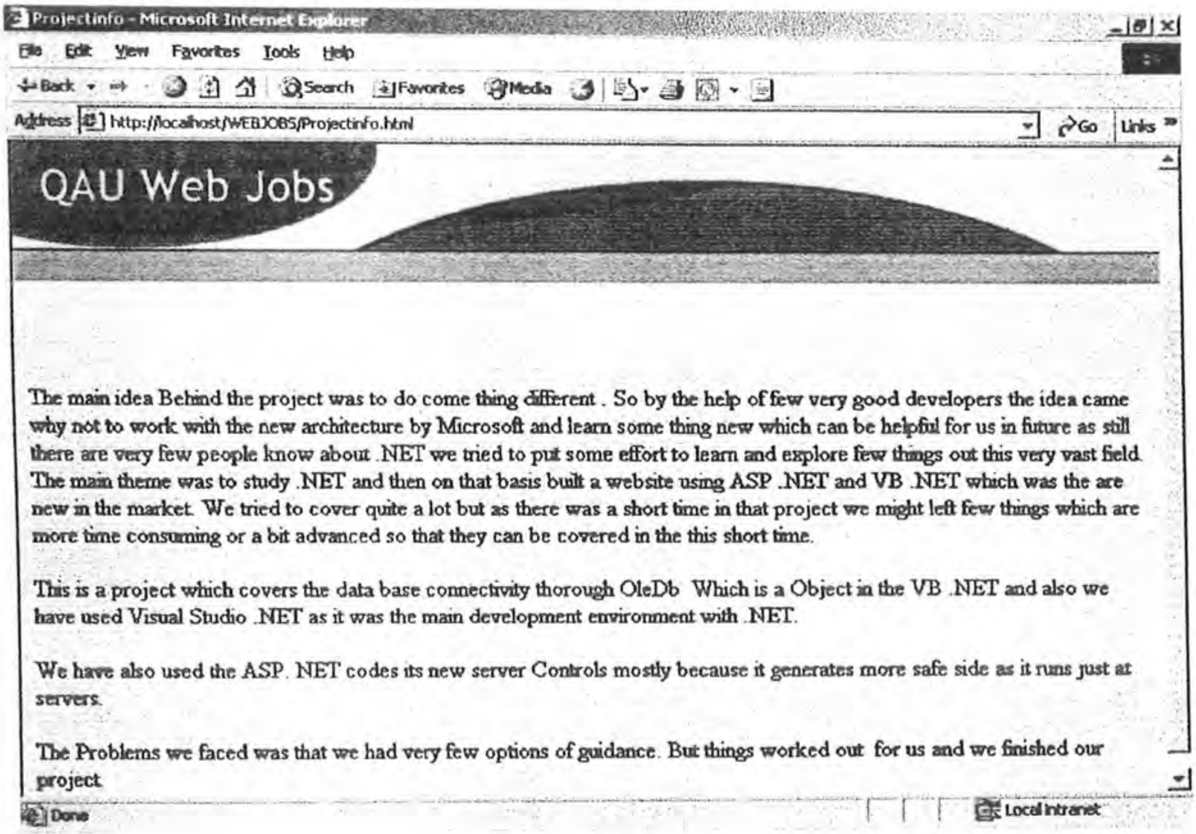
This is a web site which is designed on the .net Frame work the facility of this website is that students and people who want to submit their resume or search for job can post their resume in our data base and also search jobs from querying our data base apart from this those companies which want to hire employees can also query our data base according to our given search page. The name of the people will be shown according to there subjects and key word is the objective of these people .These companies can also give details for the posts if there is any which can be stored in our data base. This is our very little effort to improve our selves the new technology used these days. we can be very premature in our effort but this is some thing which is just been developed in very short time by a very young team. I think we will be encouraged in our efforts.

QAU Web Jobs All Rights Reserved 2002

Figure 9.1

This is our home page and from here user have to navigate on words. All functions of the website are given on the page and user can easily get on. User have to first signup to login and use the web site.

PROJECTINFO.HTML


ProjectInfo - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Projectinfo.html
Go Links
QAU Web Jobs
Done
Local intranet

The main idea Behind the project was to do come thing different . So by the help of few very good developers the idea came why not to work with the new architecture by Microsoft and learn some thing new which can be helpful for us in future as still there are very few people know about .NET we tried to put some effort to learn and explore few things out this very vast field. The main theme was to study .NET and then on that basis built a website using ASP .NET and VB .NET which was the are new in the market. We tried to cover quite a lot but as there was a short time in that project we might left few things which are more time consuming or a bit advanced so that they can be covered in the this short time.

This is a project which covers the data base connectivity thorough OleDb Which is a Object in the VB .NET and also we have used Visual Studio .NET as it was the main development environment with .NET.

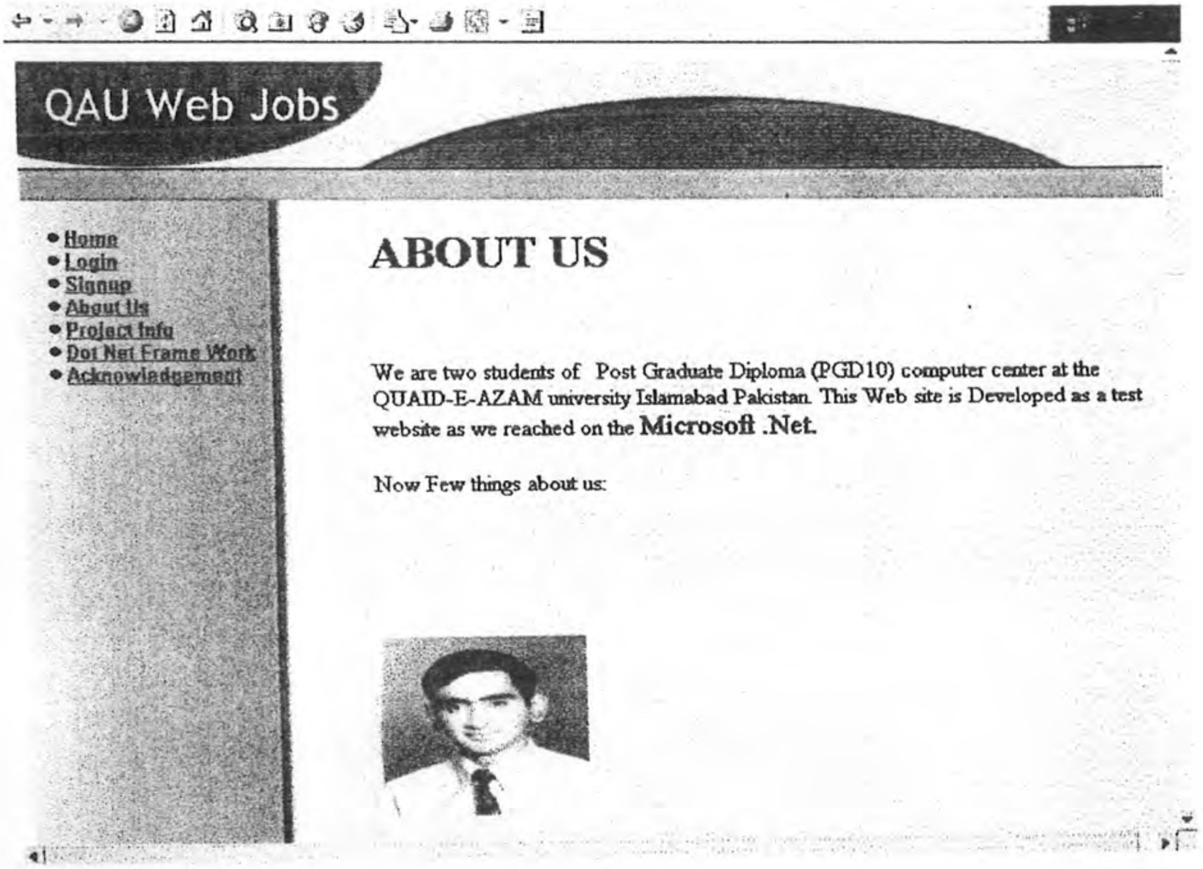
We have also used the ASP. NET codes its new server Controls mostly because it generates more safe side as it runs just at servers.

The Problems we faced was that we had very few options of guidance. But things worked out for us and we finished our project

Figure 9.2

This page shows the project information how we got the idea of doing this project and related stuff.

ABOUTUS.HTML



ABOUT US

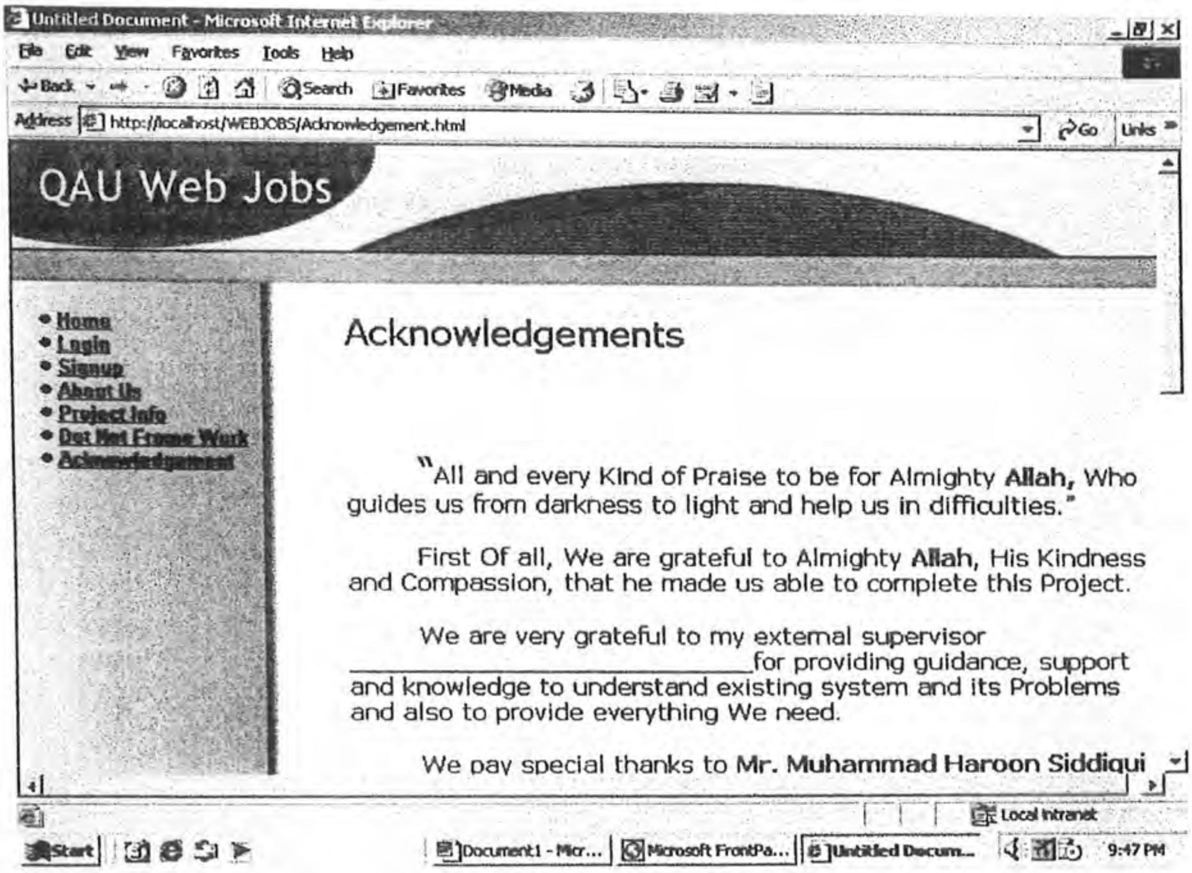
We are two students of Post Graduate Diploma (PGD10) computer center at the QUAID-E-AZAM university Islamabad Pakistan. This Web site is Developed as a test website as we reached on the Microsoft .Net.

Now Few things about us:

Figure 9.3

This page provide a brief information about us and few comments about the web site by us

Acknowledgements.HTML



Acknowledgements

"All and every Kind of Praise to be for Almighty **Allah,** Who guides us from darkness to light and help us in difficulties."

First Of all, We are grateful to Almighty **Allah**, His Kindness and Compassion, that he made us able to complete this Project.

We are very grateful to my external supervisor ______________________ for providing guidance, support and knowledge to understand existing system and its Problems and also to provide everything We need.

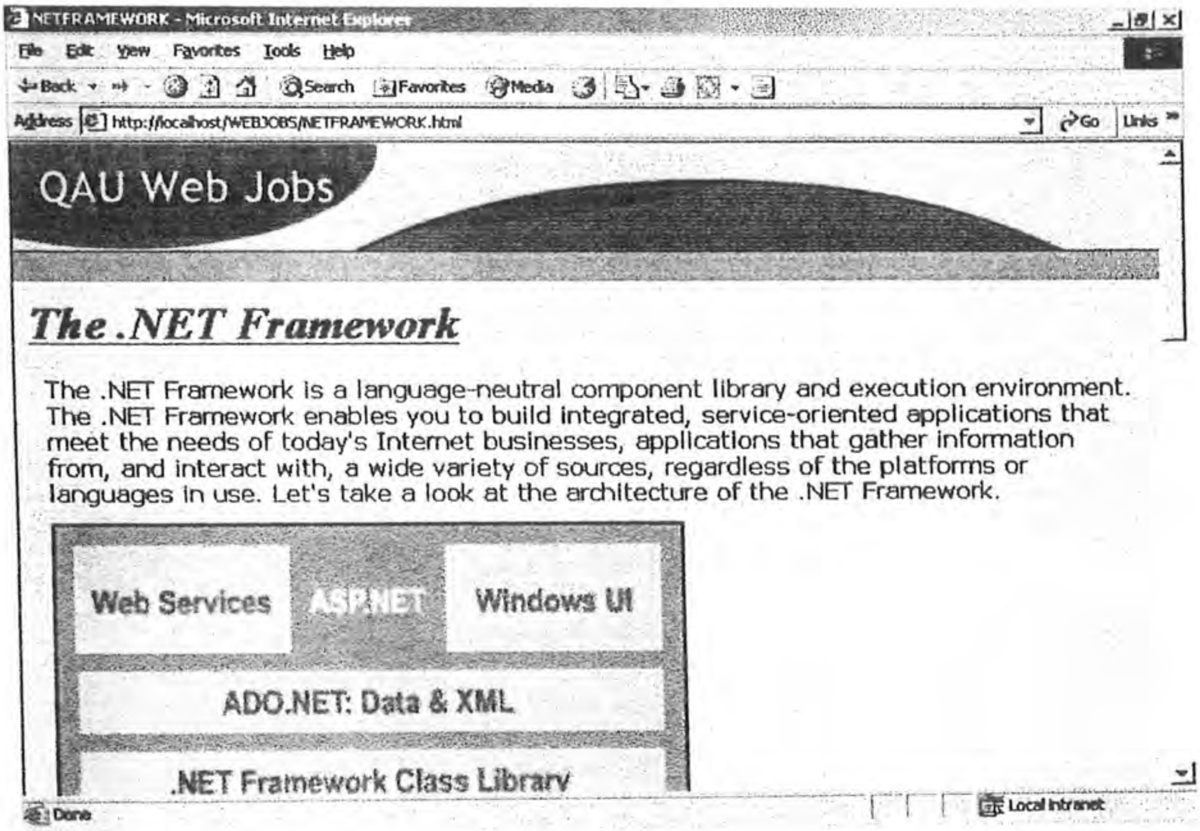
We pay special thanks to **Mr. Muhammad Haroon Siddiqui**

Figure 9.4

Here on this page we have acknowledged those who helped us out concerning about project.

User can easy go back to home page by clicking back button.

NETFRAMEWORK.HTML


NETFRAMEWORK - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/NETFRAMEWORK.html
Go Links
QAU Web Jobs
Web Services
ASP.NET
Windows UI
ADO.NET: Data & XML
.NET Framework Class Library
Done
Local intranet

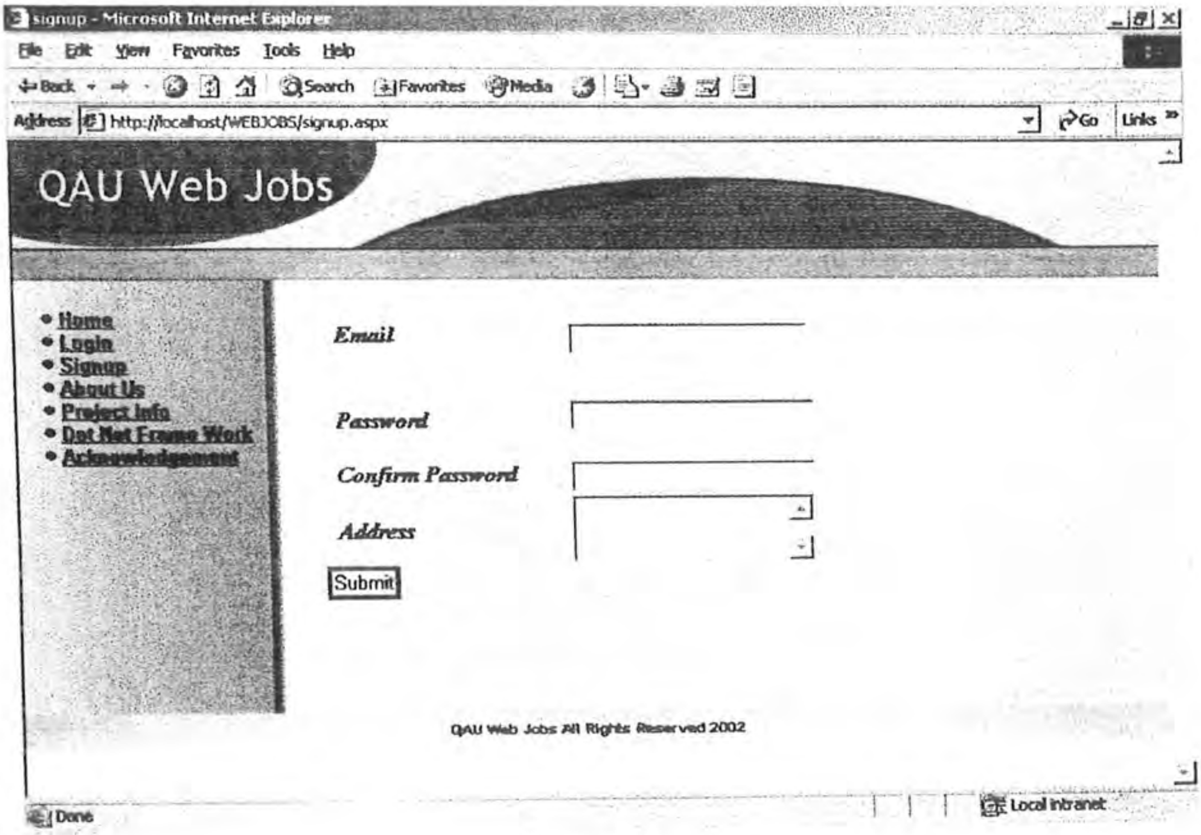
The .NET Framework

The .NET Framework is a language-neutral component library and execution environment. The .NET Framework enables you to build integrated, service-oriented applications that meet the needs of today's Internet businesses, applications that gather information from, and interact with, a wide variety of sources, regardless of the platforms or languages in use. Let's take a look at the architecture of the .NET Framework.

Figure 9.5

This page shows the little information about the .net frame work. user can easy go back to home page by clicking back button.

SIGNUP.ASPX



signup - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/signup.aspx
Go Links
QAU Web Jobs
Home
Login
Signup
About Us
Project Info
Dot Net Frame Work
Acknowledgement
Email
Password
Confirm Password
Address
Submit
QAU Web Jobs All Rights Reserved 2002
Done
Local intranet

Figure 9.6

The above page open when user click on the signup link and then gives in his valid email and password to be a permanent user of web site.

VB.NET CODE:

```
Public Class signup1
    Inherits System.Web.UI.Page
    Protected WithEvents Label1 As System.Web.UI.WebControls.Label
    Protected WithEvents Label2 As System.Web.UI.WebControls.Label
    Protected WithEvents Label3 As System.Web.UI.WebControls.Label
    Protected WithEvents email As System.Web.UI.WebControls.TextBox
    Protected WithEvents password As System.Web.UI.WebControls.TextBox
    Protected WithEvents Address As System.Web.UI.WebControls.TextBox
    Protected WithEvents Label4 As System.Web.UI.WebControls.Label
    Protected WithEvents confirmpassword As
System.Web.UI.WebControls.TextBox
    Protected WithEvents Button1 As System.Web.UI.WebControls.Button
    Public Sub check()
        MsgBox("hello")

    End Sub

#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        If password.Text <> confirmpassword.Text Then

            Response.Redirect("WebForm8.aspx")

        End If

        Dim lsStr As String
        Dim lnPos As Integer
        Dim lnPos1 As Integer
        lsStr = email.Text
        lnPos = InStr(lsStr, "@")
        lnPos1 = InStr(lsStr, ".")
        If lnPos = 0 Then
            Response.Redirect("webform7.aspx")

        End If

        If lnPos1 = 0 Then
            Response.Redirect("webform5.aspx")

        End If

        If Not (lsStr = "" Or password.Text = "" Or Address.Text = "")
Then
            Dim ds As New DataSet("User")
            Dim dr As System.Data.OleDb.OleDbDataReader
            Dim lbTest As Boolean
            lbTest = False
            Dim str As String
            str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False"
            Dim cont As New OleDb.OleDbConnection()
            cont.ConnectionString = str
            Dim insstr As String
            cont.Open()
            Dim cmd1 As New System.Data.OleDb.OleDbCommand("select *
from signup where email='" & email.Text & "'", cont)

            dr = cmd1.ExecuteReader()

            While dr.Read
                lbTest = True
            End While
            dr.Close()

            If lbTest = True Then
                Response.Redirect("WebForm6.aspx")
            End If
            Dim dc As New OleDb.OleDbDataAdapter("select * from signup
where email='" & email.Text & "'", cont)

            insstr = "insert into signup(Email, [Pass], Address )
values('" & email.Text & "','" & password.Text & "','" & Address.Text &
"')"

            ' Label8.Text = insstr

            Dim cmd As New OleDb.OleDbCommand(insstr, cont)

            cmd.ExecuteNonQuery()

            cont.Close()
        Else
            Response.Redirect("WebForm5.aspx")
        End If
    End Sub
```

```
Private Sub address_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs)

    End Sub
End Class
```

WEBFORM2.ASPX

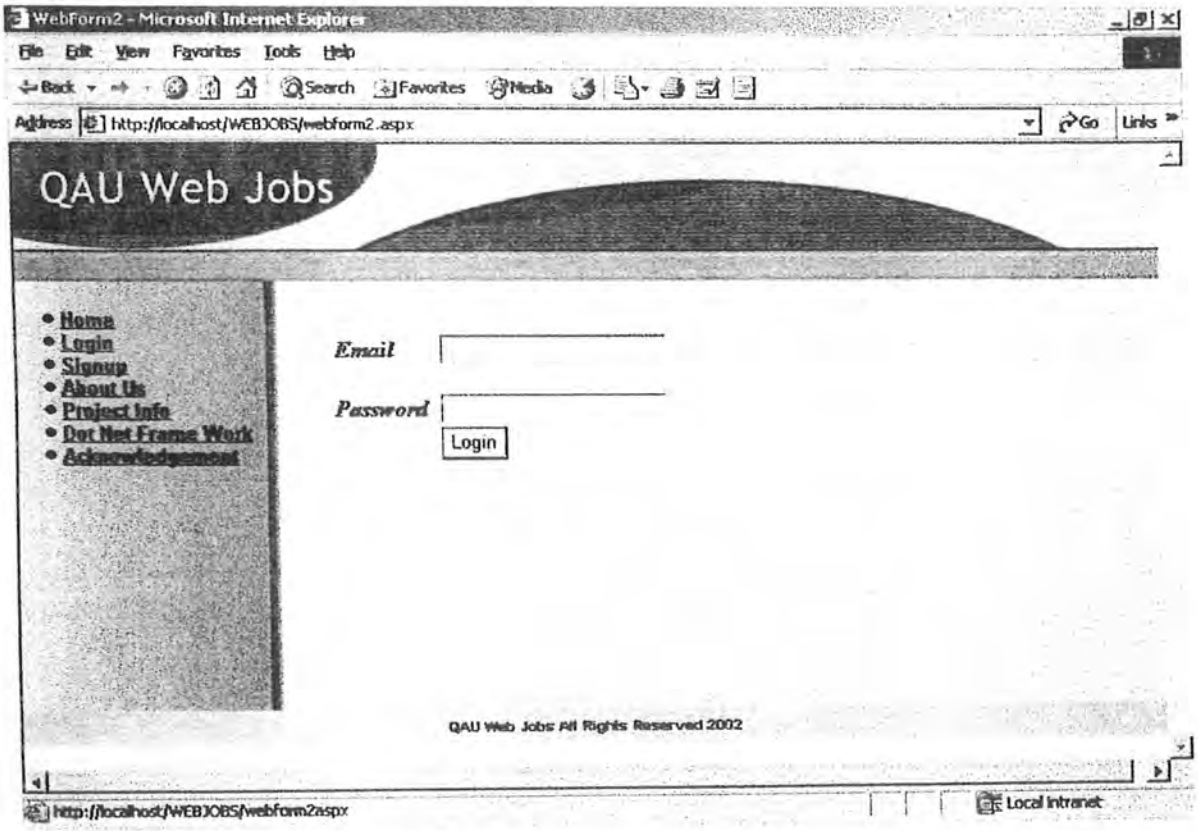

WebForm2 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/webform2.aspx
Go Links
QAU Web Jobs
Home
Login
Signup
About Us
Project Info
Dot Net Frame Work
Acknowledgement
Email
Password
Login
QAU Web Jobs All Rights Reserved 2002
http://localhost/WEBJOBS/webform2aspx
Local intranet

Figure 9.7

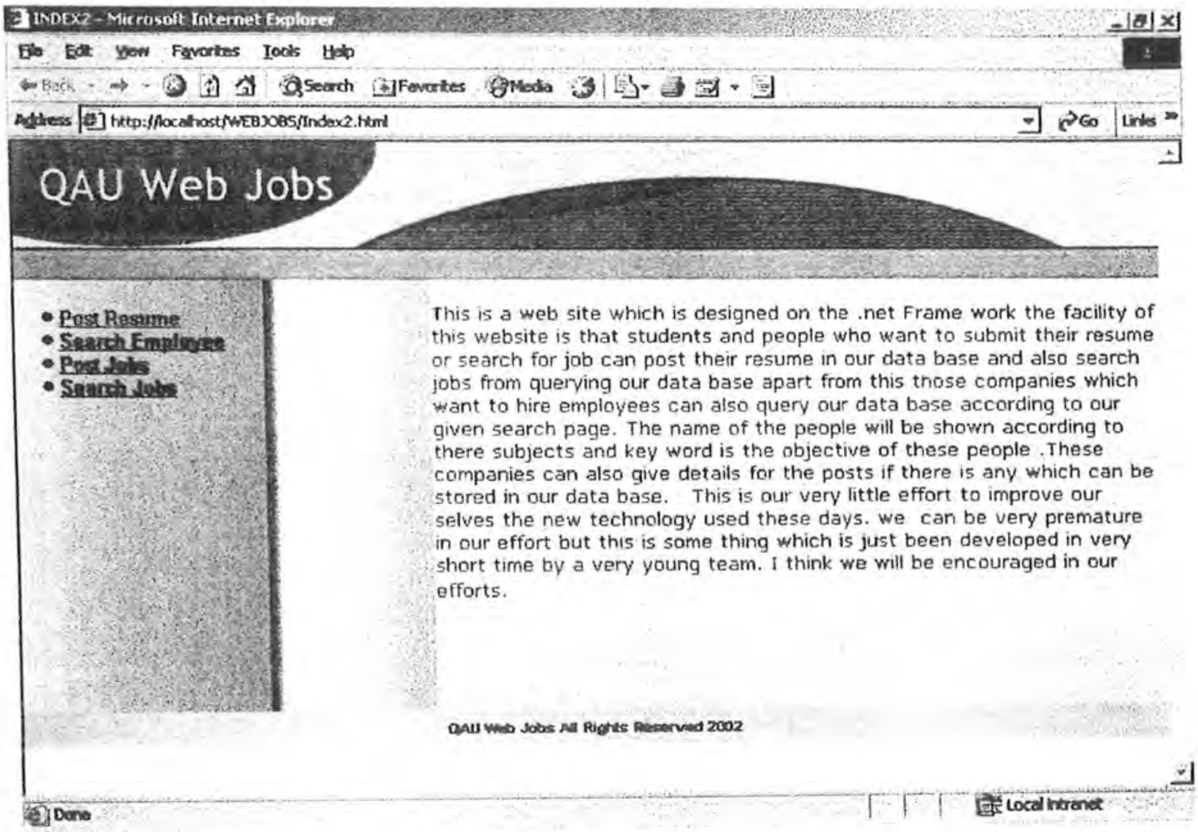
The above web form is used to login to the site after you signup where user have to enter the email address he gave while signup and related password.

VB.NET CODE :

```
Public Class WebForm2
    Inherits System.Web.UI.Page
    Protected WithEvents Label1 As System.Web.UI.WebControls.Label
    Protected WithEvents submit As System.Web.UI.WebControls.Button
    Protected WithEvents txtemail As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtpassword As
System.Web.UI.WebControls.TextBox
    Protected WithEvents Label2 As System.Web.UI.WebControls.Label

#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here

    End Sub

    Private Sub submit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles submit.Click

        'If txtemail.Text = "" Then
        'Response.Redirect("webform5.aspx")

        'End If

        'If
        'txtpassword.Text = "" Then
        'Response.Redirect("webform5.aspx")

        'End If


        Dim ds As New DataSet("User")
        Dim dr As System.Data.OleDb.OleDbDataReader

        Dim str As String
        str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False"
```

```
     Dim cont As New OleDb.OleDbConnection()
        cont.ConnectionString = str
        Dim insstr As String
        cont.Close()

        cont.Open()

        Dim cmd As New System.Data.OleDb.OleDbCommand("select * from
signup where email = '" & txtemail.Text & "'  AND pass='" &
txtpassword.Text & "'", cont)

        dr = cmd.ExecuteReader()

        Dim dc As New OleDb.OleDbDataAdapter("select * from signup",
cont)
        While dr.Read
            Dim wf As New WebForm2()
            Response.Redirect("index2.html")

        End While
        Exit Sub


    End Sub
End Class
```

INDEX2.HTML


INDEX2 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Index2.html
Go Links
Done
Local intranet

QAU Web Jobs

- Post Resume
- Search Employee
- Post Jobs
- Search Jobs

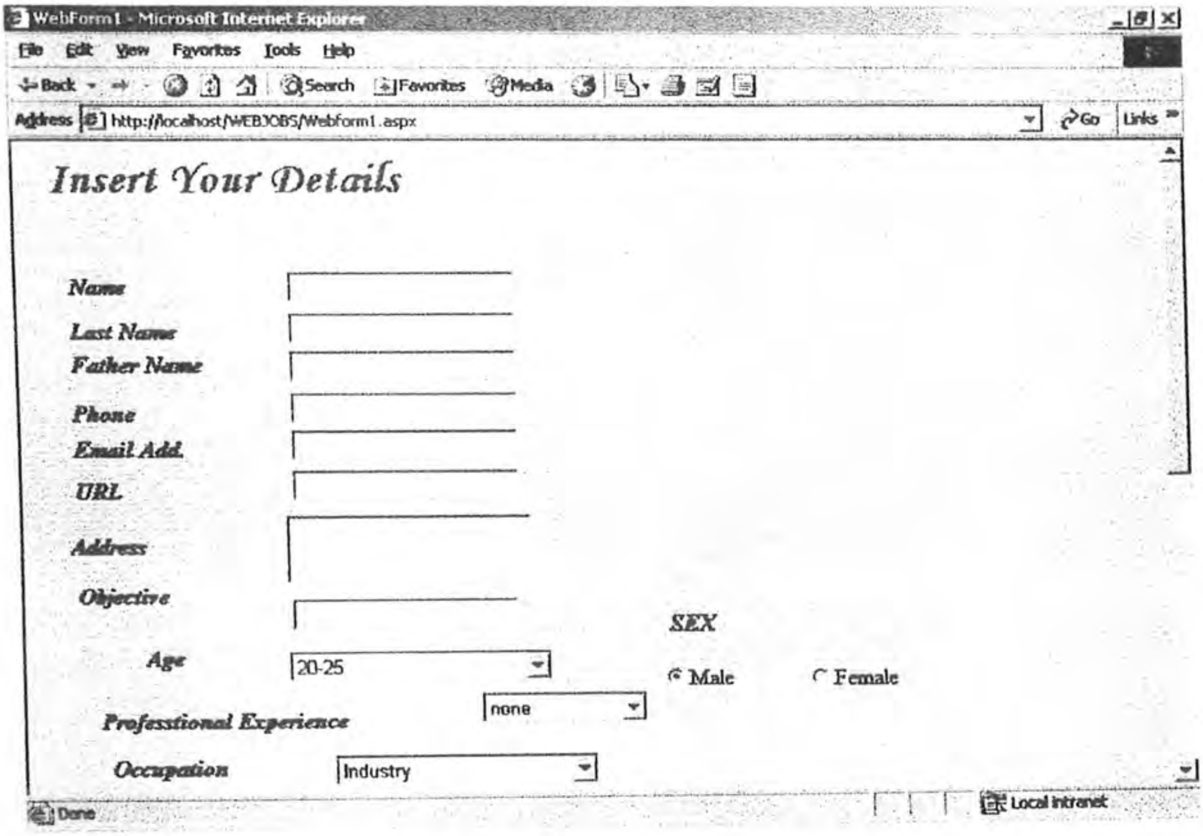
This is a web site which is designed on the .net Frame work the facility of this website is that students and people who want to submit their resume or search for job can post their resume in our data base and also search jobs from querying our data base apart from this those companies which want to hire employees can also query our data base according to our given search page. The name of the people will be shown according to there subjects and key word is the objective of these people .These companies can also give details for the posts if there is any which can be stored in our data base. This is our very little effort to improve our selves the new technology used these days. we can be very premature in our effort but this is some thing which is just been developed in very short time by a very young team. I think we will be encouraged in our efforts.

QAU Web Jobs All Rights Reserved 2002

Figure 9.8

After Login the above page open and guide user through the option given on side bar user can simply click on desired link to move further.

WEBFORM1.ASPX


WebForm1 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Webform1.aspx
Go Links
Done
Local intranet

Insert Your Details

Name

Last Name

Father Name

Phone

Email Add.

URL

Address

Objective

Age 20-25

SEX ● Male ○ Female

Professtional Experience none

Occupation Industry

Figure 9.9

If user select to post resume in index2.html the above form will open and it is connected to data base where all information given by user in this form will be stored after user click add button to submit its filled from.

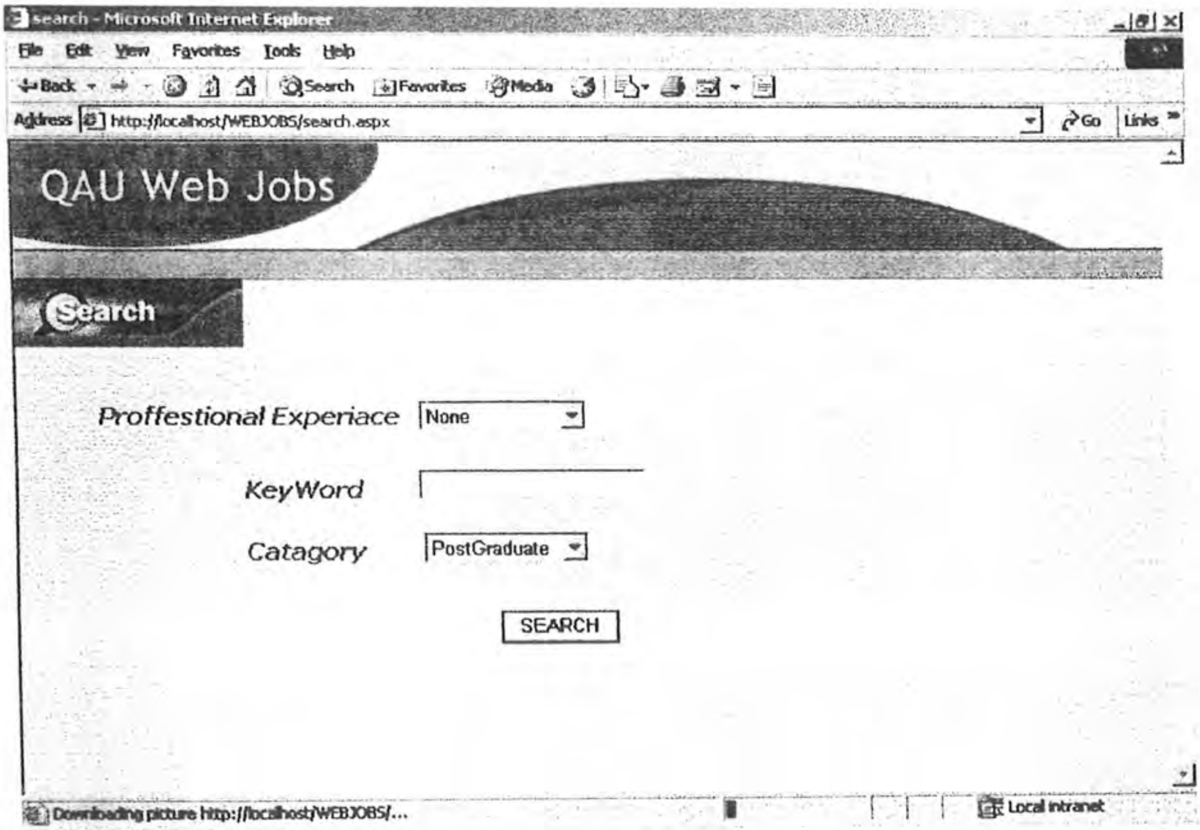
NOTE: user must enter objective carefully because that will be checked in its query.

VB.NET CODE :

```
Public Class WebForm1
    Inherits System.Web.UI.Page
    Protected WithEvents Label9 As System.Web.UI.WebControls.Label
    Protected WithEvents Label11 As System.Web.UI.WebControls.Label
    Protected WithEvents Male As System.Web.UI.WebControls.RadioButton
    Protected WithEvents Female As
System.Web.UI.WebControls.RadioButton
    Protected WithEvents firstName As System.Web.UI.WebControls.Label
    Protected WithEvents Fathername As System.Web.UI.WebControls.Label
    Protected WithEvents Address As System.Web.UI.WebControls.Label
    Protected WithEvents Objective As System.Web.UI.WebControls.Label
    Protected WithEvents Age As System.Web.UI.WebControls.Label
    Protected WithEvents Gender As System.Web.UI.WebControls.Label
    Protected WithEvents Academics As System.Web.UI.WebControls.Label
    Protected WithEvents txtfname As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtadd As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtobj As System.Web.UI.WebControls.TextBox
    Protected WithEvents cboage As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents cbope As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents txtskills As System.Web.UI.WebControls.TextBox
    Protected WithEvents submit As System.Web.UI.WebControls.Button
    Protected WithEvents Label2 As System.Web.UI.WebControls.Label
    Protected WithEvents cboOccupation As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents txtPhone As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtURL As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtemail As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtLname As System.Web.UI.WebControls.TextBox
    Protected WithEvents Label3 As System.Web.UI.WebControls.Label
    Protected WithEvents Label4 As System.Web.UI.WebControls.Label
    Protected WithEvents Label5 As System.Web.UI.WebControls.Label
    Protected WithEvents Label6 As System.Web.UI.WebControls.Label
    Protected WithEvents txtach As System.Web.UI.WebControls.TextBox
    Protected WithEvents Achivements As System.Web.UI.WebControls.Label
    Protected WithEvents Label7 As System.Web.UI.WebControls.Label
    Protected WithEvents txtinstitute As
System.Web.UI.WebControls.TextBox
    Protected WithEvents Label8 As System.Web.UI.WebControls.Label
    Protected WithEvents cbohe As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents Label1 As System.Web.UI.WebControls.Label
    Protected WithEvents ckit As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckcom As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckeco As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckman As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckcomm As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckbuss As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckeng As System.Web.UI.WebControls.CheckBox
    Protected WithEvents ckoh As System.Web.UI.WebControls.CheckBox
    Protected WithEvents txtss As System.Web.UI.WebControls.TextBox
    Protected WithEvents TextBox2 As System.Web.UI.WebControls.TextBox
    Protected WithEvents txthe As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtoccupation As
System.Web.UI.WebControls.TextBox
    Protected WithEvents txtPE As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtage As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtname As System.Web.UI.WebControls.TextBox

#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here

    End Sub

    Private Sub TextBox4_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtfname.TextChanged

    End Sub

    Private Sub submit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles submit.Click
        If txtemail.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtach.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If

        If txtname.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtLname.Text = "" Then
            Response.Redirect("webform5.aspx")
        End If

      If txtfname.Text = "" Then

       Response.Redirect("webform5.aspx")

        End If
        If txtPhone.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtURL.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtadd.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtobj.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtinstitute.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        If txtskills.Text = "" Then
            Response.Redirect("webform5.aspx")

        End If
        Dim str As String
        str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False"
        Dim cont As New OleDb.OleDbConnection()
        cont.ConnectionString = str
        Dim insstr As String
        Dim lsSubjects As String
        lsSubjects = " "

        If ckcom.Checked = True Then

            lsSubjects = lsSubjects & "Computers"

        End If

        If ckeng.Checked = True Then
            lsSubjects = lsSubjects & ","
            lsSubjects = lsSubjects & "English"

        End If

        If ckbuss.Checked = True Then
            lsSubjects = lsSubjects & ","
            lsSubjects = lsSubjects & "Business"
        End If
        If ckman.Checked = True Then

             lsSubjects = lsSubjects & ","
            lsSubjects = lsSubjects & "Managment"

     End If

        If ckit.Checked = True Then
            lsSubjects = lsSubjects & ","
            lsSubjects = lsSubjects & "Information Tecnology"

        End If

        If ckeco.Checked = True Then

            lsSubjects = lsSubjects & "Economics"

        End If

        If ckcomm.Checked = True Then

            lsSubjects = lsSubjects & "Commerece"

        End If

        If ckoh.Checked = True Then
            'lsSubjects = lsSubjects & ","
            lsSubjects = lsSubjects & "Orcaology And Hisyory"

        End If


        cont.Close()


        cont.Open()

        Dim ck As String
        If Male.Checked = True Then
            ck = "Male"
        Else
            ck = "Female"
        End If

        insstr = "insert into us(Email,  Name, LastName, FatherName,
age, gender, occupation, Phone, URL, objective, ProffestionalExperiece,
Academics, Skills, Achivements , Subjects ,institution) values('" &
txtemail.Text & "','" & txtname.Text & "','" & txtLname.Text & "','" &
txtfname.Text & "'," & cboage.SelectedItem.Value & ",'" & ck & "','" &
cboOccupation.SelectedItem.Value & "','" & txtPhone.Text & "','" &
txtURL.Text & "','" & txtobj.Text & "','" & cbope.SelectedItem.Value &
"','" & cbohe.SelectedItem.Value & "','" & txtskills.Text & "','" &
txtach.Text & "','" & lsSubjects & "','" & txtinstitute.Text & "')"
```

```
        ' Label8.Text = insstr

        Dim cmd As New OleDb.OleDbCommand(insstr, cont)

        cmd.ExecuteNonQuery()

        cont.Close()

    End Sub

    Private Sub txtacad_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cbohe.SelectedIndexChanged

    End Sub

    Private Sub Female_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Female.CheckedChanged

    End Sub

    Private Sub cbope_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cbope.SelectedIndexChanged

    End Sub

    Private Sub cboage_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cboage.SelectedIndexChanged

    End Sub
End Class
```

SEARCH.ASPX


search - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/search.aspx
Go Links
QAU Web Jobs
Search
SEARCH
Downloading picture http://localhost/WEBJOBS/...
Local intranet

Proffestional Experiace: None

KeyWord:

Catagory: PostGraduate

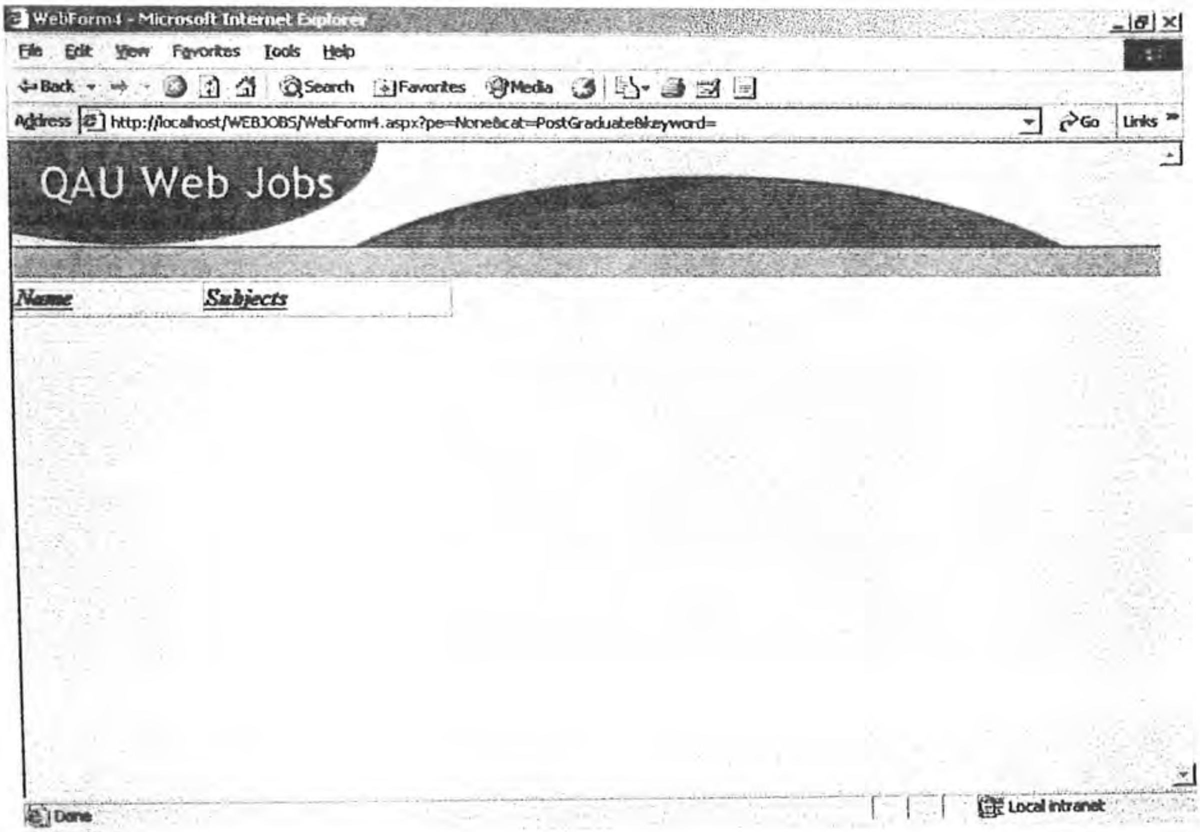
Figure 9.10

This is a query form if user select search employee at index2.html above form will be opened and here the user can give its requirements to choose the people form database Query.

VB.NET CODE

```
Public Class search
    Inherits System.Web.UI.Page
    Protected WithEvents TextBox1 As System.Web.UI.WebControls.TextBox
    Protected WithEvents Button1 As System.Web.UI.WebControls.Button
    Protected WithEvents Label1 As System.Web.UI.WebControls.Label
    Protected WithEvents ddl As System.Web.UI.WebControls.DropDownList
    Protected WithEvents Label3 As System.Web.UI.WebControls.Label
    Protected WithEvents ddl1 As System.Web.UI.WebControls.DropDownList
    Protected WithEvents Label2 As System.Web.UI.WebControls.Label

#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region



    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        'MsgBox("test")
        If TextBox1.Text = "" Then

        End If
        Response.Redirect("WebForm4.aspx?pe=" & ddl1.SelectedItem.Value
& "&cat=" & ddl.SelectedItem.Value & "&keyword=" & TextBox1.Text)
        '       datagrid1.Visible = True
        '        Dim ds As New DataSet("User")
        '       Dim dr As System.Data.OleDb.OleDbDataReader
        '
        '       Dim str As String
        '      str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False"
        '     Dim cont As New OleDb.OleDbConnection()
        '    cont.ConnectionString = str
        '   Dim insstr As String
        '  cont.Close()
        '
        '       cont.Open()
        '        Dim cmd As New System.Data.OleDb.OleDbCommand("select *
from us where objective like '" & TextBox1.Text & "'  AND Academics ='"
& ddl.SelectedItem.Text & "'  AND ProffestionalExperiece = '" &
ddl1.SelectedItem.Text & "'", cont)
        '
        '        dr = cmd.ExecuteReader()

        '        Dim dc As New OleDb.OleDbDataAdapter("select * from us",
cont)
        '       datagrid1.DataSource = dr
        '      datagrid1.DataBind()
        '


    End Sub

    Private Sub datagrid1_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs)

    End Sub

    Private Sub datagrid1_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs)

    End Sub

    Private Sub AdRotator1_AdCreated(ByVal sender As System.Object,
ByVal e As System.Web.UI.WebControls.AdCreatedEventArgs)

    End Sub
End Class
```

WEBFORM4.ASPX



WebForm4 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/WebForm4.aspx?pe=None&cat=PostGraduate&keyword=
Go Links
QAU Web Jobs
Done
Local intranet

Name	*Subjects*

Figure 9.11

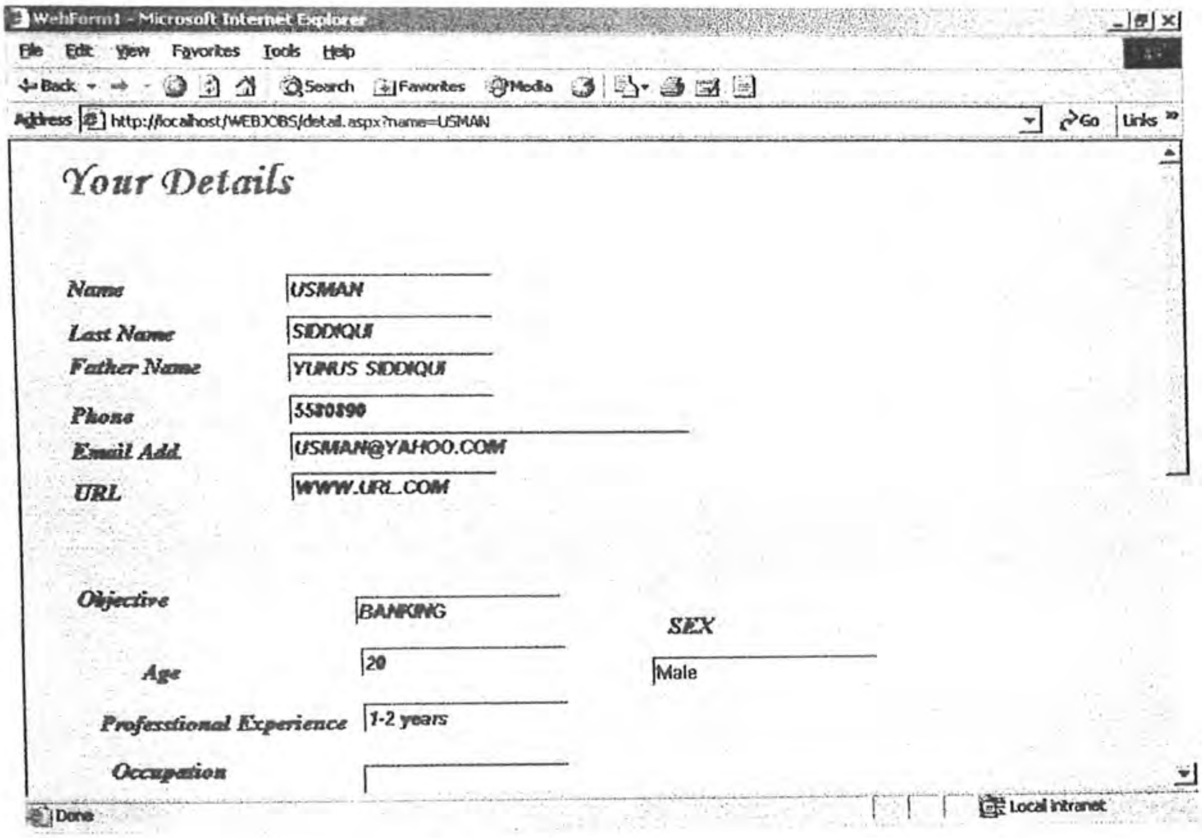
The results of the query in the search will be showed here with names and subjects of the people and from here on in names are in links and user can click the name if the subjects of the person are matching to his demands. and next page will open holding all the information of the person.

Which is

VB.NET CODE & ASP.NET CODE

```
<%
dim rst dim conn conn =createobject("ADODB.Connection") 'Dim ds As New
DataSet("User") 'Dim dr As System.Data.OleDb.OleDbDataReader Dim str As
String str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False" conn.open (str) Dim insstr As
String 'conn.Open() rst=CreateObject("ADODB.Recordset") dim lsSqlString
lsSqlString="select * from us where 1=1" if not request.QueryString("pe")="" then
lsSqlString=lsSqlString & " and ProffestionalExperiece ='" &
request.QueryString("pe") & "'" end if if not request.QueryString("cat")="" then
lsSqlString=lsSqlString & " and Academics ='" & request.QueryString("cat") & "'"
end if if not request.QueryString("keyword")="" then lsSqlString=lsSqlString & "
and objective ='" & request.QueryString("keyword") & "'" end if rst.open
(lsSqlString ,conn) ' where objective like '" & TextBox1.Text & "' AND Academics
='" & ddl.SelectedItem.Text & "' AND ProffestionalExperiece = '" &
ddl1.SelectedItem.Text & "'",conn 'Dim cmd As New
System.Data.OleDb.OleDbCommand("select * from us where objective like '" &
TextBox1.Text & "' AND Academics ='" & ddl.SelectedItem.Text & "' AND
ProffestionalExperiece = '" & ddl1.SelectedItem.Text & "'", cont) while not rst.eof
%>
<a href = detail.aspx name=?
<%=rst.fields(2).value%>
<%=rst.fields(2).value%> </a>
<%=rst.fields(15).value%>
<% rst.movenext end While
%>
```

DETAILS.ASPX


WebForm1 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go
Links
Done
Local intranet

http://localhost/WEBJOBS/detail.aspx?name=USMAN

Your Details

Name USMAN

Last Name SIDDIQUI

Father Name YUNUS SIDDIQUI

Phone 5580890

Email Add. USMAN@YAHOO.COM

URL WWW.URL.COM

Objective BANKING

SEX Male

Age 20

Professtional Experience 1-2 years

Occupation

Figure 9.12

Here you can see the details of the person and user can simply contact the person through email or at mailing address.

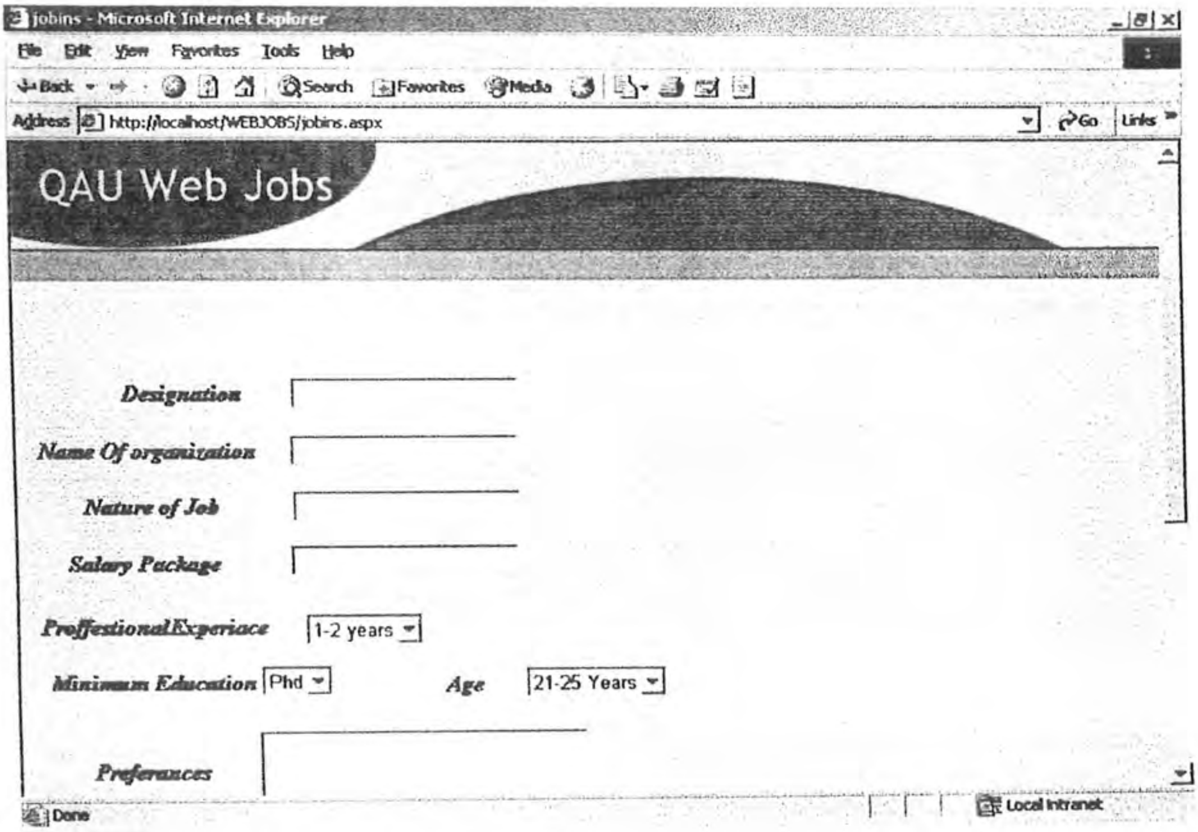
VB.NET CODE

```
Public Class detail
    Inherits System.Web.UI.Page
    Protected WithEvents Label9 As System.Web.UI.WebControls.Label
    Protected WithEvents Label11 As System.Web.UI.WebControls.Label
    Protected WithEvents firstName As System.Web.UI.WebControls.Label
    Protected WithEvents Fathername As System.Web.UI.WebControls.Label
    Protected WithEvents Objective As System.Web.UI.WebControls.Label
    Protected WithEvents Age As System.Web.UI.WebControls.Label
    Protected WithEvents Gender As System.Web.UI.WebControls.Label
    Protected WithEvents Academics As System.Web.UI.WebControls.Label
    Protected WithEvents txtfname As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtobj As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtskills As System.Web.UI.WebControls.TextBox
    Protected WithEvents submit As System.Web.UI.WebControls.Button
    Protected WithEvents Label2 As System.Web.UI.WebControls.Label
    Protected WithEvents txtPhone As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtURL As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtemail As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtLname As System.Web.UI.WebControls.TextBox
    Protected WithEvents Label3 As System.Web.UI.WebControls.Label
    Protected WithEvents Label4 As System.Web.UI.WebControls.Label
    Protected WithEvents Label5 As System.Web.UI.WebControls.Label
    Protected WithEvents Label6 As System.Web.UI.WebControls.Label
    Protected WithEvents txtach As System.Web.UI.WebControls.TextBox
    Protected WithEvents Achivements As System.Web.UI.WebControls.Label
    Protected WithEvents Label7 As System.Web.UI.WebControls.Label
    Protected WithEvents txtinstitute As
System.Web.UI.WebControls.TextBox
    Protected WithEvents Label8 As System.Web.UI.WebControls.Label
    Protected WithEvents Label1 As System.Web.UI.WebControls.Label
    Protected WithEvents txtss As System.Web.UI.WebControls.TextBox
    Protected WithEvents txthe As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtoccupation As
System.Web.UI.WebControls.TextBox
    Protected WithEvents txtPE As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtage As System.Web.UI.WebControls.TextBox
    Protected WithEvents TextBox4 As System.Web.UI.WebControls.TextBox
    Protected WithEvents txthighedu As
System.Web.UI.WebControls.TextBox
    Protected WithEvents agetxt As System.Web.UI.WebControls.TextBox
    Protected WithEvents petxt As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtsubj As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtgender As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtname As System.Web.UI.WebControls.TextBox

#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
```

```
 <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here
        txtfname.Text = Request.QueryString("Name")
        '        datagrid1.Visible = True
        Dim ds As New DataSet("User")
        Dim dr As System.Data.OleDb.OleDbDataReader

        Dim str As String
        str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
 Source=C:\DB1.mdb;Persist Security Info=False"
        Dim cont As New OleDb.OleDbConnection()
        cont.ConnectionString = str
        Dim insstr As String
        cont.Close()

        cont.Open()

        Dim cmd As New System.Data.OleDb.OleDbCommand("select *  from
 us where NAME like '" & Request.QueryString("Name") & "'", cont)

        dr = cmd.ExecuteReader()
        While dr.Read
            txtname.Text = dr.Item("Name")
            txtfname.Text = dr.Item("fathername")
            txtLname.Text = dr.Item("lastname")
            txtPhone.Text() = dr.Item("Phone")
            txtemail.Text() = dr.Item("email")

            txtobj.Text() = dr.Item("objective")
            txtURL.Text() = dr.Item("url")
            'txtinstitute.Text() = dr.Item("institute")
            txtach.Text() = dr.Item("Achivements")
            txthighedu.Text() = dr.Item("Academics")
            txtskills.Text() = dr.Item("Skills")
            txtgender.Text() = dr.Item("gender")



            txtsubj.Text() = dr.Item("subjects")
            agetxt.Text() = dr.Item("age")
            petxt.Text() = dr.Item("ProffestionalExperiece")

        End While
    End Sub

    Private Sub TextBox4_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtfname.TextChanged

    End Sub

    Private Sub submit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles submit.Click
        Dim str As String
        str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False"
        Dim cont As New OleDb.OleDbConnection()
        cont.ConnectionString = str
        Dim insstr As String
        Dim lsSubjects As String
        lsSubjects = " "

        cont.Close()


        cont.Open()




        ' Label8.Text = insstr

        Dim cmd As New OleDb.OleDbCommand(insstr, cont)

        cmd.ExecuteNonQuery()

        cont.Close()

    End Sub

    Private Sub txtacad_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs)

    End Sub

    Private Sub Female_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs)

    End Sub

    Private Sub cbope_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs)

    End Sub

    Private Sub cboage_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs)
```

```
    End Sub

    Private Sub TextBox6_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs)

    End Sub

    Private Sub txtemail_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtemail.TextChanged

    End Sub

    Private Sub txtadd_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs)

    End Sub
End Class
```

JOBSINS.ASPX


jobins - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/jobins.aspx
Go Links
QAU Web Jobs
Done
Local intranet

Designation

Name Of organization

Nature of Job

Salary Package

ProffestionalExperiace 1-2 years

Minimum Education Phd *Age* 21-25 Years

Preferances

Figure 9.13

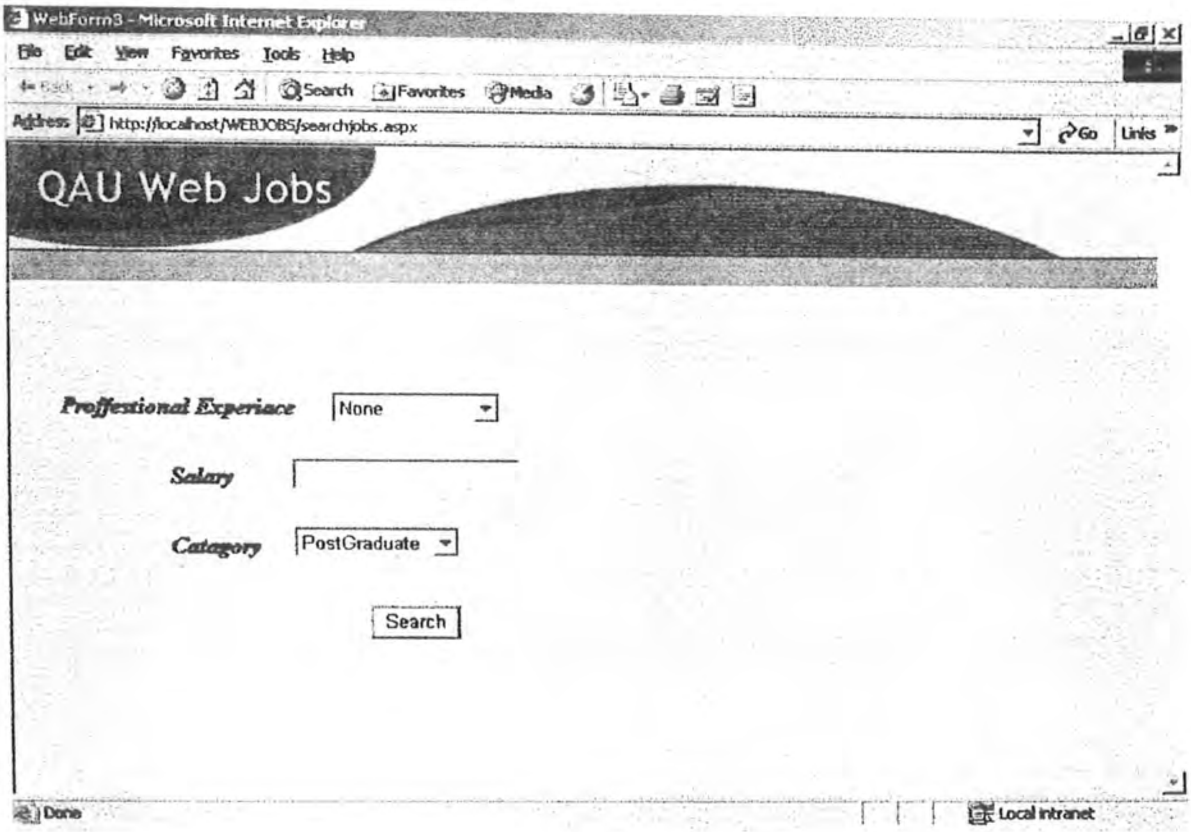
If user select to post jobs in index2.html the above form will open and it is connected to data base where all information given by user in this form will be stored after user click add button to submit its filled from.

VB.NET CODE:

```
Public Class jobins
    Inherits System.Web.UI.Page
    Protected WithEvents Label1 As System.Web.UI.WebControls.Label
    Protected WithEvents Label2 As System.Web.UI.WebControls.Label
    Protected WithEvents Label3 As System.Web.UI.WebControls.Label
    Protected WithEvents Label4 As System.Web.UI.WebControls.Label
    Protected WithEvents Label5 As System.Web.UI.WebControls.Label
    Protected WithEvents Label6 As System.Web.UI.WebControls.Label
    Protected WithEvents Label7 As System.Web.UI.WebControls.Label
    Protected WithEvents Label8 As System.Web.UI.WebControls.Label
    Protected WithEvents Label9 As System.Web.UI.WebControls.Label
    Protected WithEvents Button1 As System.Web.UI.WebControls.Button
    Protected WithEvents txtorg As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtdes As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtpre As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtphone As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtemail As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtnoj As System.Web.UI.WebControls.TextBox
    Protected WithEvents txtadd As System.Web.UI.WebControls.TextBox
    Protected WithEvents ddlpe As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents ddlme As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents txtsal As System.Web.UI.WebControls.TextBox
    Protected WithEvents ddlage As
System.Web.UI.WebControls.DropDownList
    Protected WithEvents Label12 As System.Web.UI.WebControls.Label
    Protected WithEvents Label10 As System.Web.UI.WebControls.Label

#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here
    End Sub
```

```
  Private Sub DropDownList5_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs)

    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click

        Dim str As String
        str = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\DB1.mdb;Persist Security Info=False"
        Dim cont As New OleDb.OleDbConnection()
        cont.ConnectionString = str
        Dim insstr As String
        cont.Close()


        cont.Open()


        insstr = "insert into jobs(designation , nameoforganization
,natureofjob , salary, pe , minedu, preferances , Address, Phone,
email , age  ) values('" & txtdes.Text & "','" & txtorg.Text & "','" &
txtnoj.Text & "','" & txtsal.Text & "','" & ddlpe.SelectedItem.Value &
"' ,'" & ddlme.SelectedItem.Value & "','" & txtpre.Text & "','" &
txtadd.Text & "','" & txtphone.Text & "','" & txtemail.Text & "','" &
ddlage.SelectedItem.Value & "')"

        'Label8.Text = insstr

        Dim cmd As New OleDb.OleDbCommand(insstr, cont)

        cmd.ExecuteNonQuery()

        cont.Close()

    End Sub
End Class
```

SEARCHJOBS.ASPX:


WebForm3 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/searchjobs.aspx
Go Links
QAU Web Jobs
Done
Local intranet

Proffestional Experiace None

Salary

Catagory PostGraduate

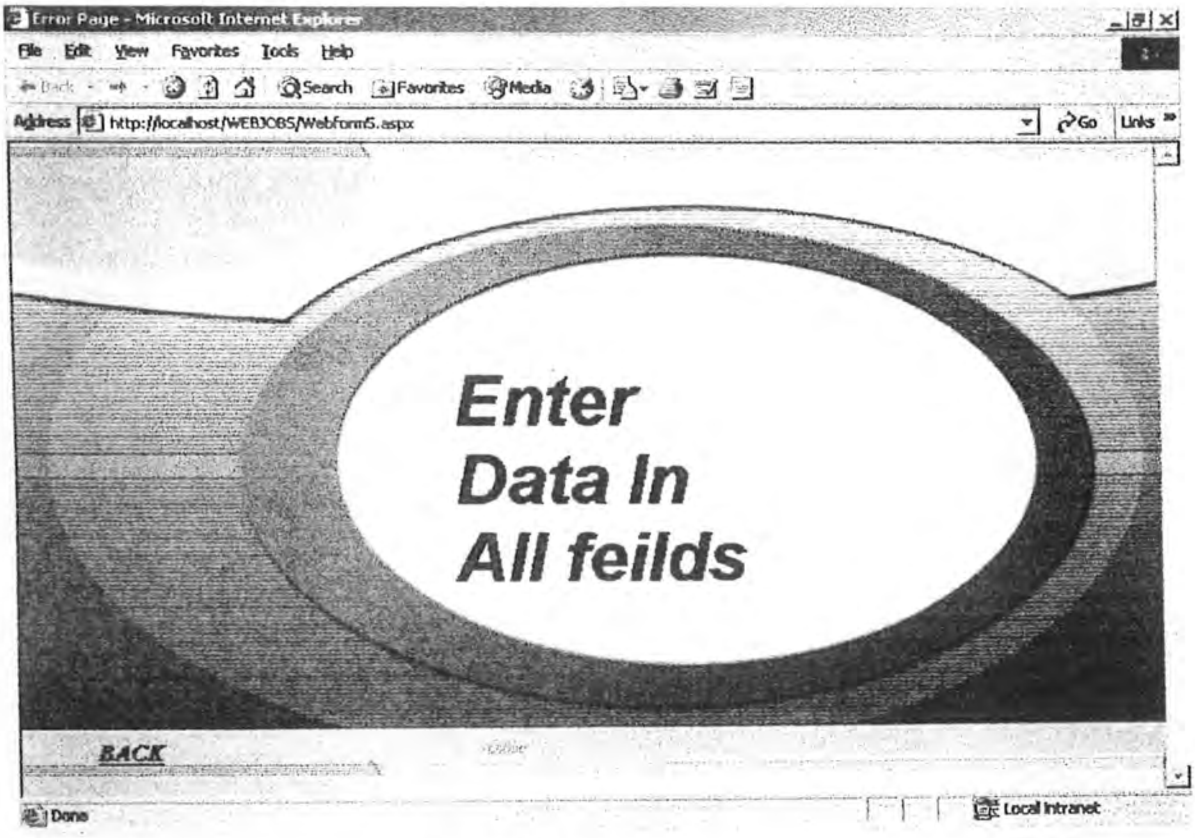
Search

Figure 9.14

The above page open when the user select the search jobs at index2.html and here user can enter its query that how much professional experience he has and how much education he has and related jobs will be shown in a data grid on the same page.

ERROR PAGES:

WEBFORM5:

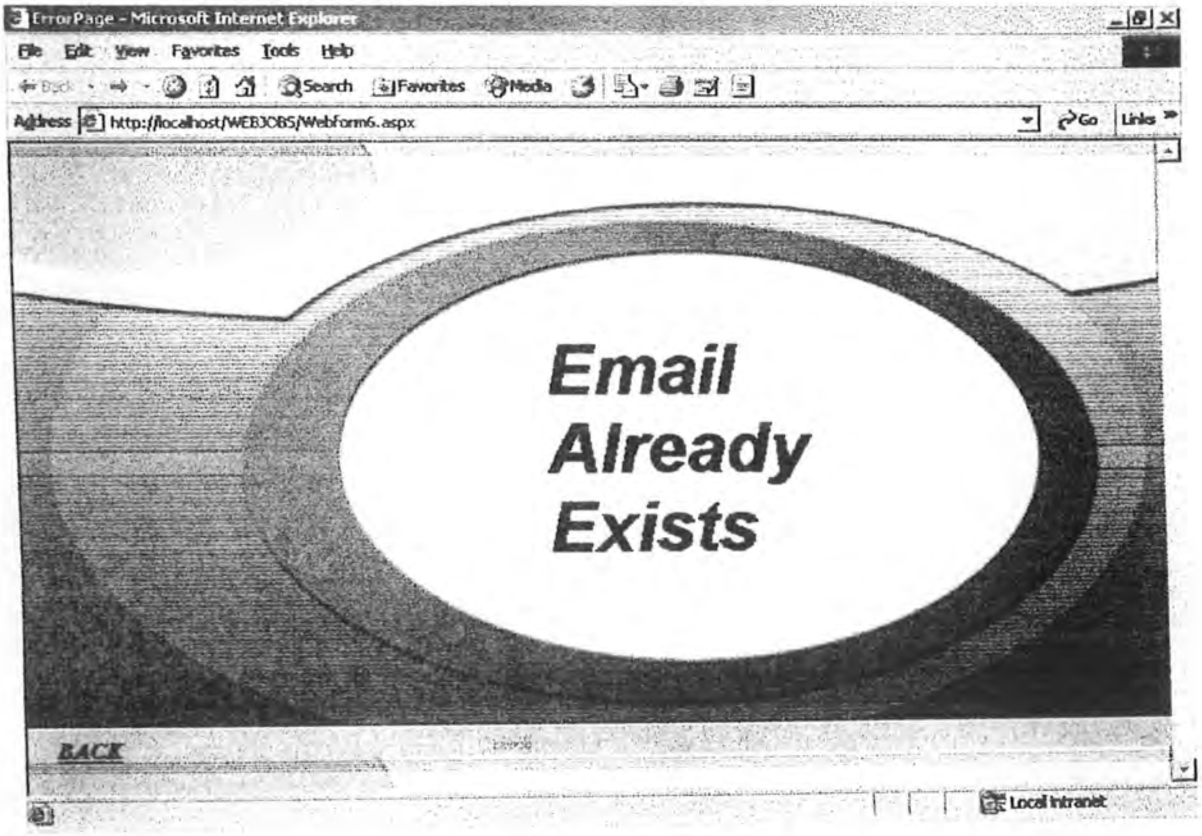

Error Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/WebForm5.aspx
Go Links
BACK
Done
Local intranet

Enter Data In All feilds

Figure 9.15

This is error page, which will open if user have not fill in data in some where in the form.

WEBFORM6.ASPX


ErrorPage - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Webform6.aspx
Go Links
BACK
Local intranet

Email Already Exists

Figure 9.16

This error page will open while signing up if user enter email that already have been signed up with the website.

WEBFORM7.ASPX

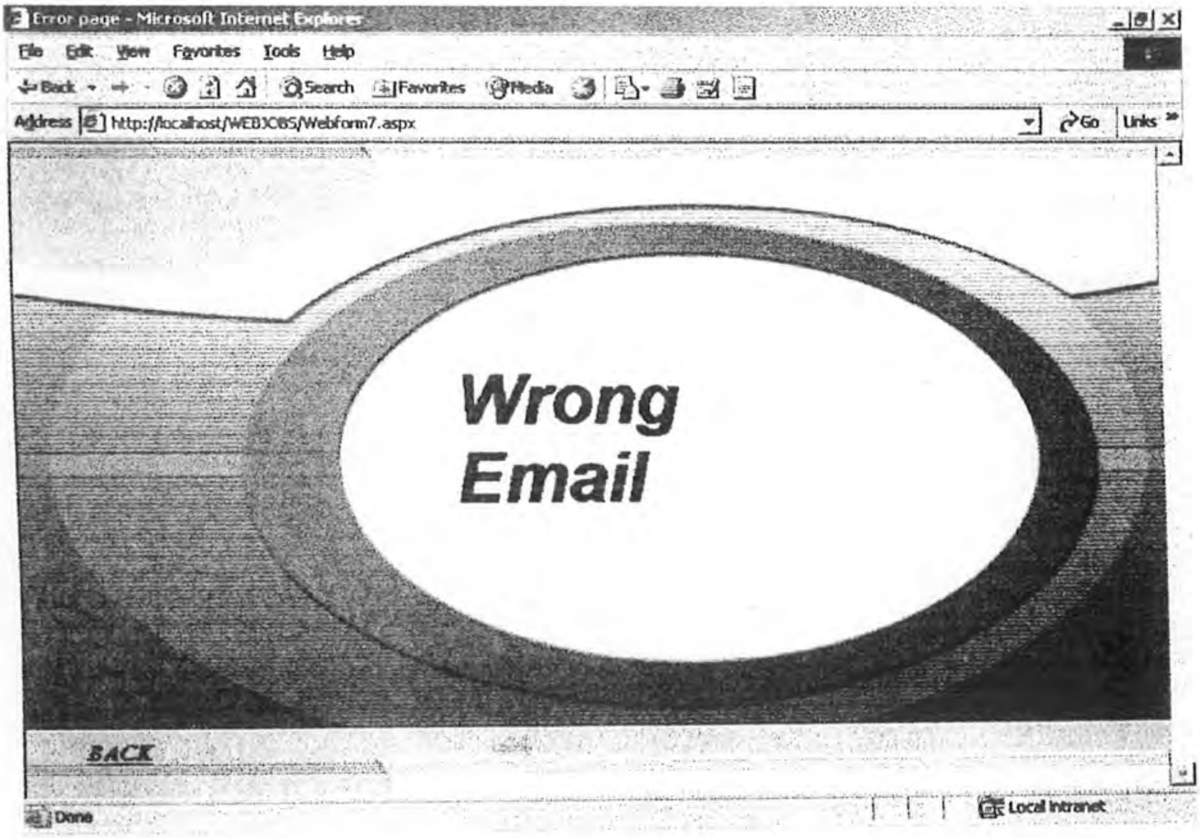

Error page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Webform7.aspx
Go Links
Wrong Email
BACK
Done
Local intranet

Figure 9.17

When user will enter wrong email address with our a "@" or "." Sign the program code will check it and if its wrong above page will be displayed.

WEBFORM8.ASPX

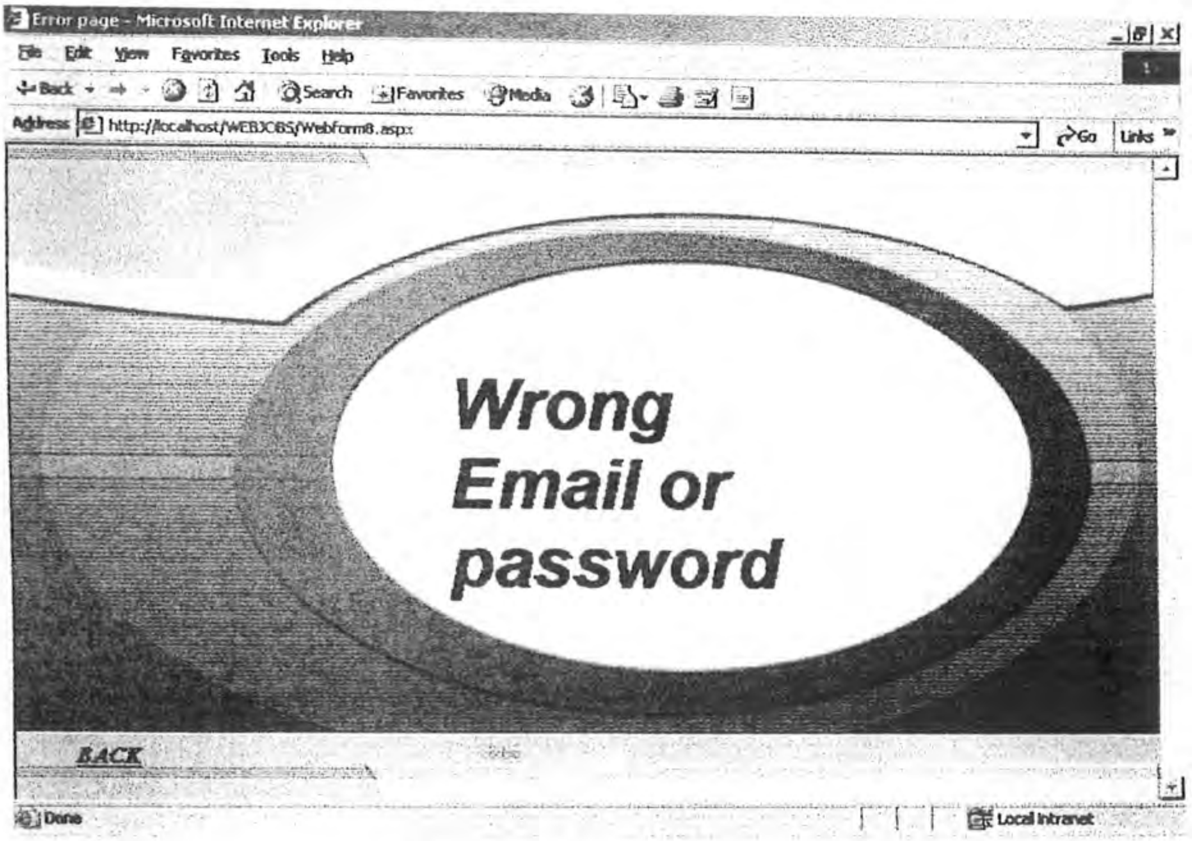

Error page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/WEBJOBS/Webform8.aspx
Go Links
Wrong Email or password
BACK
Done
Local intranet

Figure 9.18

This page opens when some one enter a invalid password or email while login to the site so user must enter the correct email or password while login.

DATA BASE:

Database Used: Microsoft Access
Tables Used: us , jobs , signup.

US TABLE:

Field Name	Data Type
Id	TEXT
Name	TEXT
Last Name	TEXT
Father Name	TEXT
Address	TEXT
Professional Experience	TEXT
Gender	TEXT
Occupation	TEXT
Phone	TEXT
URL	TEXT
Objective	TEXT
Age	TEXT
Academics	TEXT
Skills	TEXT
Achivements	TEXT
Institution	TEXT
Subjects	TEXT

JOBS TABLE:

Field Name	Data Type
Designation	TEXT
Nameoforganization	TEXT
Natureofjob	TEXT
Professional Experience	TEXT
Salary	TEXT
Minimumeducation	TEXT
Preferences	TEXT
Address	TEXT
Phone	TEXT
Duedate	Date/Time
Email	TEXT
Age (required)	TEXT

Signup Table:

Field Name	Data Type
Email	TEXT
Password	TEXT
Address	TEXT

APPENDIX C

Conclusion:

The world of communication has changed rapidly in the past few years. It has become a necessity, not a luxury, to be able to keep in touch with customers and colleagues even when you are traveling. Another significant change in communication in the past few years is the use of Internet.

Microsoft .NET is a platform that provides all the tools and technologies we need to build distributed Web applications that use standard Web protocols. These applications are able to communicate with a wide range of sophisticated clients, such as cellular phones and palm PCs. At the same time, the .NET platform allows unprecedented integration between programming languages as well as a variety of runtime services.

Microsoft also provides a high-level development environment for building applications on the .NET Framework. It provides key enabling technologies to simplify the creation, deployment, and ongoing evolution of secure, scalable, highly available Web applications.

We have studied the .NET technologies at very basic level because of the shortage of time but we have seen that it has really made things easy for the developers to make a web application very rapidly. It is really a revolution in the race to make the world a smaller place and we think Microsoft has been so far successful.

There is another technology by **SUN MICROSYETEM** which is named as **J2EE** and Microsoft .NET is better than that of **J2EE**.and it will bring revolution to highly distributed internet applications.

Future Recommendations

Among hot most technologies of the future are XML & Web services. Due to the shortage of time we couldn't use them in our project but we suggest other students to work in this area. XML stands for Extensible Mark Up language. Whole world is now moving towards XML. It has multi-dimensional benefits. It not only allows defining your own customized tags but also removes barriers to data sharing and software integration. It makes it easy to exchange data, and .NET software gives users the ability to work with the data once it's received.

XML Web services provide a way to create Web applications that can be accessed and utilized by different customers using different platforms. It is an intrinsic mechanism to build any Web site or service so that it will collaborate seamlessly with other Web sites and services. XML Web services provide a simple, flexible, and standards-based model for binding Web applications together over the Internet that takes advantage of existing infrastructure and applications.

We would recommend other students to explore these latest cutting edge technologies. Those who would learn it now would definitely get benefit, because it's the future.

REFRENCES:

These are the following references. These are basically the references of those sources, which we used to develop the application and was extremely helpful to us during the project.
The list is as under:

- **MSDN Library**
- **Beginning ASP.NET** (Wrox Series Book)
- **www.wrox.com**
- **Microsoft Solution Developer kit** (SDK)
- **A Training CD by MSDN named "Upgrade From VB 6.0 To VB.NET"**