[Chapter-1 4](#_Toc402183216)

[1. Introduction: 4](#_Toc402183217)

[**1.1.** **Background** 4](#_Toc402183218)

[**1.2.** **Objectives of Project** 4](#_Toc402183219)

[**1.3.** **Purpose:** 5](#_Toc402183220)

[**1.4.** **Scope** 5](#_Toc402183221)

[**1.5.** **Limitation:** 5](#_Toc402183222)

[Chapter-2 6](#_Toc402183223)

[2. Feasibility Study 6](#_Toc402183224)

[**2.1.** **Introduction:** 6](#_Toc402183225)

[**2.2.** **Technical Feasibility:** 6](#_Toc402183226)

[**2.3.** **Economical Feasibility:-** 7](#_Toc402183227)

[2.3.1. Cost Avoidance: 7](#_Toc402183228)

[2.3.2. Operational Feasibility: 7](#_Toc402183229)

[2.3.3. Clients Supports: 8](#_Toc402183230)

[2.3.4. User Support: 8](#_Toc402183231)

[Chapter–3 9](#_Toc402183232)

[3. Requirements and Analysis 9](#_Toc402183233)

[**3.1.** **Problem Definition** 9](#_Toc402183234)

[**3.2.** **Requirement Specification** 9](#_Toc402183235)

[**3.3.** **Functional Requirements:** 10](#_Toc402183236)

[**3.4.** **Non-Functional Requirements** 10](#_Toc402183237)

[Chapter-4 11](#_Toc402183238)

[4. Planning and Scheduling 11](#_Toc402183239)

[**4.1.** **Gantt chart:** 11](#_Toc402183240)

[**4.2.** **Context Diagram:** 11](#_Toc402183241)

[**4.3.** **ER Diagram** 12](#_Toc402183242)

[Chapter–5 13](#_Toc402183243)

[5. Survey of Technologies 13](#_Toc402183244)

[**1.** **ASP.NET** 13](#_Toc402183245)

[**2.** **Framework 4.0** 13](#_Toc402183246)

[**3.** **Microsoft-SQL Server 2008** 14](#_Toc402183247)

[Chapter-6 15](#_Toc402183248)

[6. Project Methodology 15](#_Toc402183249)

[**6.1. Data Collection Methods and Sources** 15](#_Toc402183250)

[**6.1.1** **Use primary and secondary data source:** 15](#_Toc402183251)

[**6.1.2** **Questionnaires, survey and field visit:** 15](#_Toc402183252)

[**6.1.3** **Software resources used** 15](#_Toc402183253)

[**6.1.4** **Hardware resources used** 15](#_Toc402183254)

[Chapter-7 16](#_Toc402183255)

[7. System Implementation 16](#_Toc402183256)

[**7.1.1.** **System Implementation** 16](#_Toc402183257)

[**7.1.2.** **Coding and Debugging Phase** 16](#_Toc402183258)

[**7.1.3.** **Verification and Validation** 16](#_Toc402183259)

[**7.1.4.** **Testing** 17](#_Toc402183260)

[7.1.4.1. Unit Testing 17](#_Toc402183261)

[7.1.4.2. Integrating Testing 17](#_Toc402183262)

[7.1.4.3. System Testing 17](#_Toc402183263)

[Chapter–8 18](#_Toc402183264)

[8. Data Dictionary 18](#_Toc402183265)

[**8.1.** **Exam** 18](#_Toc402183266)

[**8.2.** **Faculty** 18](#_Toc402183267)

[**8.3.** **Member** 18](#_Toc402183268)

[**8.4.** **Question** 19](#_Toc402183269)

[**8.5.** **Student** 19](#_Toc402183270)

[**8.6.** **StudentExam** 19](#_Toc402183271)

[Chapter–9 20](#_Toc402183272)

[9. Snapshots and Codes 20](#_Toc402183273)

[**9.1.** **Codes** 20](#_Toc402183274)

[9.1.1. Student.aspx.cs 20](#_Toc402183275)

[9.1.2. ChangePasswordStudent.aspx.cs 21](#_Toc402183276)

[9.1.3. Questions.aspx.cs 22](#_Toc402183277)

[9.1.4. LogoutStudent.aspx.cs 23](#_Toc402183278)

[9.1.5. AdminMasterPage.master 24](#_Toc402183279)

[9.1.6. AdminMasterPage.master.cs 25](#_Toc402183280)

[9.1.7. AddStudent.aspx.cs 25](#_Toc402183281)

[9.1.8. DisplayQuestion.aspx 26](#_Toc402183282)

[**9.2.** **Snapshorts** 29](#_Toc402183283)

[9.2.1. Default.aspx 29](#_Toc402183284)

[9.2.2. Student.aspx 29](#_Toc402183285)

[9.2.3. ChangePasswordStudent.aspx 30](#_Toc402183286)

[9.2.4. Exam.aspx (Design) 30](#_Toc402183287)

[9.2.5. Questions.aspx (Design) 31](#_Toc402183288)

[9.2.6. Admin.aspx (Design) 31](#_Toc402183289)

[9.2.7. ManageStudent.aspx (Design) 32](#_Toc402183290)

[9.2.8. Subject.aspx (Design) 32](#_Toc402183291)

[9.2.9. DisplayQuestion.aspx (Design) 33](#_Toc402183292)

[Chapter–10 34](#_Toc402183293)

[10. Future Scope of the project 34](#_Toc402183294)

[Bibliography 34](#_Toc402183295)

# Chapter-1

## Introduction:

Online Examination itself represents the examination through the internet. This project will help the university/ Institute to evaluate the question having multiple options with one correct answer.

This project helps the management to conducting online examination and save resources. The university/ Institution can conduct the online examination and announce the status or result immediately after the test is complete.

### **Background**

As the name of the project “Online Examination” suggests, the system/application has to manage the examination for various department of an academy. Earlier it was very time consuming and tedious process where both students as well as university used to perform all tasks manually like Student Registration, Setting Question Paper, Setting Schedules to finally declaring the result. Later the process became a lot easier when web application were developed which converts all of these tasks atomized. There are applications already working in this area like MCSE, CCNA which conducts their examination online.

I (6th sem/B.Sc.IT) am making this project by taking help from these already working and tested application as well as making my own set of improvements. This application is coded in .Net. This software project is very much helpful to students, Examination department and Exam controller. It stores the information about Students, Exams, and Results. As we know that this software is coded in .Net programming language, it is user friendly. That means this software is easy to operate. Even a person having less knowledge of computer can easily operate it and will be guided throughout by the software to operate it. This software is a best effort of changing in paper work to electronic work. Another advantage of this software is the database that is designed very strongly.

### **Objectives of Project**

The objective of the project “Online Examination System” is to make evaluation and conduction of examination simple, cost effective and faster. They are as follows:

* To provide an interface through which student can appear for examination online for objective type questions.
* Answer will be checked automatically by the system form the database.
* To provide username and password facility and credentials should be checked properly at the time of login for student and Exam Dept. Admin.
* To provide an interface from where Exam Dept. Admin can register new student, set new question paper.
* The authority to modify the student profile, question paper.

### **Purpose:**

Through this package I provide a fully customized web application. This system is an automated system so that the functional working of it is effective and time saving. In this modern ear time is the most precious thing, so in context of time the new system will be effective to do a group to task in easy and secure manner.

To appear for an exam in manual system in a university is very time consuming process. Now the purpose of this system is to overcome the shortfall faced in the previous system already working in the area.

The website will have to be secure, and properly working on WAN. It should be speedy with good interface. The university may conduct examination in various countries and in different language. It should support multiple platforms at least those used by people commonly.

### **Scope**

Methodology of the system is clear that result in the complete and proper working of the system from the requirements achieved. It contains the working model of the development system. A sequential step of model is followed to develop the system so that it will maintain various software engineering features/parameters.

No need to take leave, visit a different town, city or even states in some cases, People can appear right from their laptop or home desktop.

### **Limitation:**

* Students cannot register by themselves.
* When the student starts the exam there is no time limit
* The question will be static nature i.e. it will be not displayed randomly from question bank unless the Exam Dept. Admin change question order manually.
* Student can only see their Score not Remarks

# Chapter-2

## Feasibility Study

### **Introduction:**

The feasibility study of any system is mainly intended to study and analyze the proposed system and to decide whether the system under consideration will be viable or not after implementation. That is it determines the usability of the project after deployment. To come to result a set of query is answered keeping the efficiency of the software and its impact on the domain for which it was developed. It main emphasis is on the following three questions elucidated below as:

What are the user’s requirements and how does a candidate system meet them?

What resources are available for the proposed systems? Is it worth solving the problem?

What is the likely impact of the proposed system on the organization? I.e. how does the proposed system fit with in the organization?

Thus since the feasibility study may lead to commitment of large resources, it becomes necessary that it should be conducted competently and no fundamental errors of judgment are made. Different types of feasibility study and the way we performed on our project “On Line Examination”.

### **Technical Feasibility:**

In technical feasibility, we study all technical issues regarding the proposed system. It is mainly concerned with the specifications of the equipments and the software, which successfully satisfies the end-user’s requirement. The technical needs of the system may vary accordingly but include:

* The feasibility to produce outputs in a given time.
* Response time under certain conditions.
* Ability to process a certain volume of the transaction at a particular speed.
* Facility to communicate data.

Under this analysis process questions like (i) does the compatible platform exist within our domain or can we procure it? (ii) Does the proposed equipment have the technical capacity to hold the data required using the new system?.

Both at the development site and at server where we will be hiring the space for the website, and also the database would it be possible to upgrade the system after it is developed and implemented, if necessary? And would the recommended technology guarantee the reliability, accuracy and data security? This analysis process requires more emphasis on system configuration given more importance rather than the actual hardware specifications.

The configuration of the existing systems is:

* Processor   : Pentium III, 500 MHz (or above)
* Memory    : 128 MB (or above)
* Secondary storage  : 20 GB (or above)

For Software there are following alternatives:

* Operating System   :Window 98,2000,XP,NT
* Development tools  :ASP.Net ,C# ,HTML,DHTML
* Database     :Microsoft SQL server 7.0
* Documentation tool: MS-Word

### **Economical Feasibility:-**

Meaning: re there sufficient benefits in creating the system to make the acceptable? Or are the costs of not creating the system so great that it is advisable to undertaken the project.

This will include three major costs as described below:

* Cost of  Hardware and Software
* Cost of Software to be acquired to build and run the product is a one time cost.
* Buying a back and database is the major part of hardware and Software cost. Comparison between the oracle database high cost and better features with the SQL server low cost and better support for the same vendor operating system make this decision need oriented.

Benefits in reduced cost, error and saving will be made by reduction of present system expenses, time saving and increased accuracy.

#### Cost Avoidance:

Future cost reduction in form of reduction in the number of administrative staff needed and manual records maintains in organization.

Rise in cost will be avoided.

#### Operational Feasibility:

Meaning: The system will be used if it is developed well then be resistance from users that undermine the possible application benefits.

#### Clients Supports:

Client and user support for present system is there, as the current procedure used takes more time and effort than proposed system.

No major training and new skills are required as it is based on DBMS model.

It will help in the time saving and fast processing and dispersal of user request and application.

New product will provide all the benefits of present system with better performance such as improved information, better management and collection of the reports.

#### User Support:

User involvement in the building of present system is sought to keep in mind the user specific requirement and needs.

User will have control over own information. Important information such as Test result can be generated at the click of a button.

# Chapter–3

## Requirements and Analysis

### **Problem Definition**

The problem here is to develop a web application which completely automize the examination process for an university.

To perform this complete problem has been divided into five sub-problems so that they can be solved easily and after that can be integrated to make it an integrated working application

Online Registration / Enrollment of Student

* Online Login
* Online Schedule
* Online Manage the question bank
* Online Examination
* Online Result declaration
* Online display randomly question.
* The questions will be display randomly from question bank.

At the Login module user name and password facility and credentials should be checked properly at the time of login for student, expert, controller and Exam Dept Admin.

In the schedule Module send the exam schedule to the student.

In question bank module for the examination could be changed dynamically.  
Automatic checking of objective answers and manual checking of descriptive answers should be facilitated, if any.  
In the Objective answers module will be checked automatically by the system from the database and the subjective answer manually checking by exam department will take place.

In the exam module When the student  starts the exam the timer will start automatically and show the student how much time is left.

At result module display the result of each student and send by the exam. Dept.

### **Requirement Specification**

On the basis of system study performed in an organization about all the functions that deals with online examination following requirements are specified.

### **Functional Requirements:**

The Functional Requirements of the project are as follows:

* To provide the functionality to students to appear for the exam online.
* To provide login interface through which only authorized user can pass by.
* The timer facility should be provide once the student appears in the Exam.
* The web application provide  question to the student randomly.
* The system should support WAN so that exam can given from anywhere.
* This system should handle multiple exam at the same time.
* To functionally enable Exam Controller to define criteria and create Admin and Experts.
* To functionally enable Exam Dept Admin to manage Question Bank
* To functionally enable Exam Dept Admin to declare schedule and Results.
* To functionally enable Expert to send set of Questions.

### **Non-Functional Requirements**

The Non-Functional Requirements of the project are as follows:

* The system should be reliable and robust.
* The system should be User friendly.
* The system should be completely Consistent and Secure.

Interface requirements

* Interface should be easy to follow,
* Interface shouldn’t have  very much graphics,
* Interface shouldn’t have  hidden buttons, and
* Interface should produce  relevant error message.

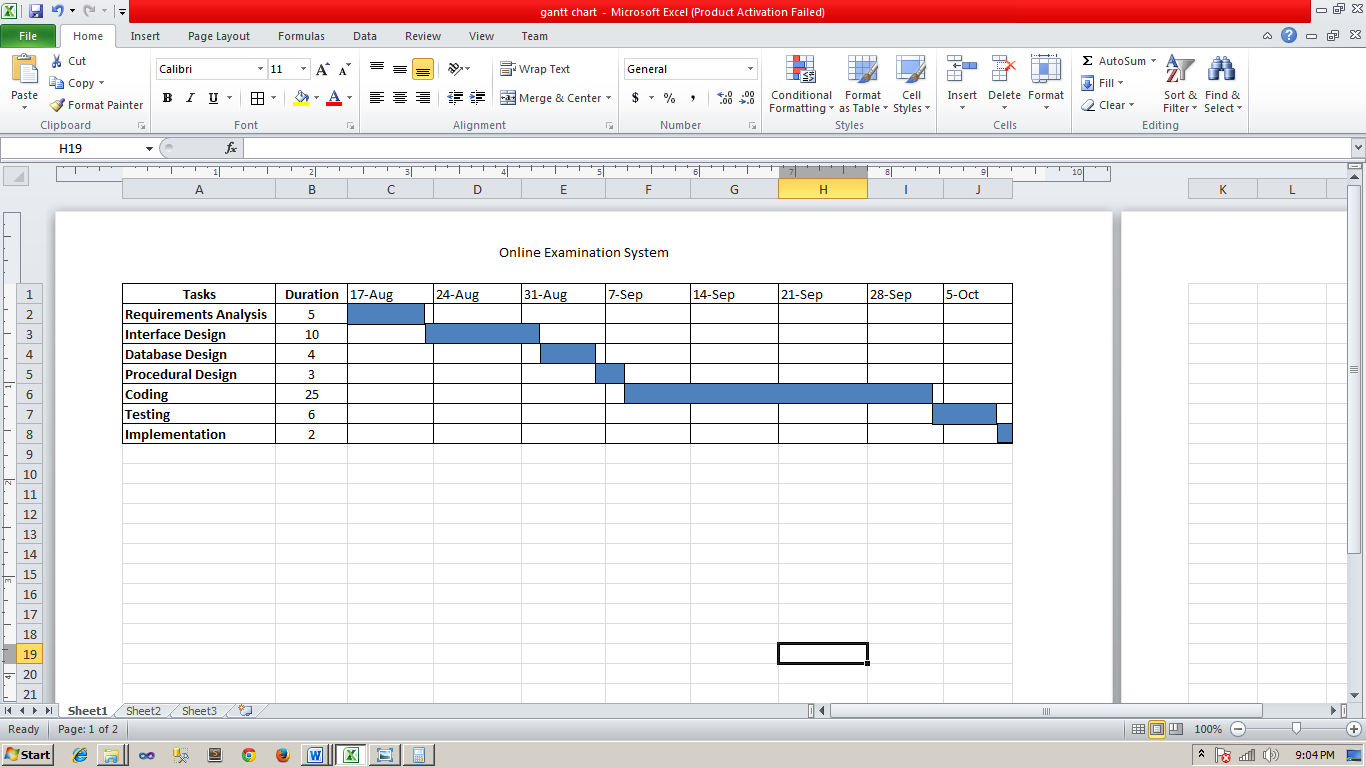
In a complete way, we’re expected to build a web based application which       accommodates all functionality of an organization that deal with online examination.

# Chapter-4

## Planning and Scheduling

### **Gantt chart:**

A Gantt chart is a type of bar chart, developed by Henry Gantt in the 1910s, that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project.



### **Context Diagram:**

This diagram represents what are the bounders and scope of **Online ExamSystem** project. It describes the main objective of the system and its entities involved.

**Administrator**

**Public**

**Student**

**Online Examination System**

Figure: the context diagram of Online Exam System

**The Administrator can do the following:**

* Create/delete accounts
* Change passwordfor for current user
* Create/ delete/updateStudents
* Create/ delete/update Courses
* Create/ delete/update Faculty
* Create/ delete/update Questions

**The Student can dothe following:**

* Change password.
* Give Exam

**The Public can do the following:**

* View Website
* Send feedbacks

### **ER Diagram**

Entity Relationship Diagram is used to model the data in the system. It is detailed and the data elements for a system. The ER representation of the entities, associations, diagram is shown in figure.

M

M

M

M

M

M

M

1

1

M

**StudentExam**

**Result**

**Question**

**Has**

**Has**

**Creates**

**Exam**

**Add** to Cart

**Faculty**

**Student**

**Member**

# Chapter–5

## Survey of Technologies

To develop this package different types of tools and database are used which are as follows:

1. ASP.NET
2. Framework 4.0
3. Microsoft SQL Server 2008
4. Microsoft word processor

### **ASP.NET**

ASP.NET is a Microsoft technology for building the web based application and services.ASP.NET application contains of forms, controls, classes and procedures.

Forms are windows upon which you build your user interface. Controls are called activeX controls are interface tools, such as labels, textbox and command buttons that you use to display information to the user, gather information from the user, and respond to user actions. Classes are templates from which you can create your own objects at run time.

Procedure are small routine you write that are called from anywhere in your application. These routines will perform a function for you that you write once but can call many times.

An application is made up of forms, modules and classes. A form is made up of properties, events and controls. Controls are made up of properties and events

Some Features of ASP.NET

* Language interoperability: a one language code is to interact with other language.
* DEP (Event driven programming language): it is an Event Driven Programming Language (i.e. we code on its event and drive the program).
* Rapid Application Development (RAID).
* Allow the creation of COM components such as activeX controls.
* Includes good debugging facilities.

### **Framework 4.0**

Dot net framework is a platform that is provided by Microsoft technology in which we can develop web based application, window based application and console base application with C# (c sharp), VB and J# (J sharp) and other 22 language.

### **Microsoft-SQL Server 2008**

Microsoft SQL Server fulfills these responsibilities:

* Reduction and redundancy:

Centralized control of the DBA avoids the unnecessary duplication of data and effectively. Reduces the total amount of data storage required. It also eliminates the extra processing and of the inconsistencies

* Sharing Data

Any number of application programs of user’s can share the same database.

* Data Integrity:

Data integrity means that the data contains in the database both accurate and consistent

* Data Security

In this facility the confidential data must not be accessed by unauthorized person.

* Rapid Accessing Data:

SQL Server provides rapid access to data by utilizing indexes and storing frequently accessed data in memory.

* SQL Server Enterprise Manager:

The Enterprise Manage is the central console from which most SQL Server database-management tasks can be coordinated. SQL Enterprise Manager provides a single interface from which all servers in a company can be managed.

# Chapter-6

## Project Methodology

### **6.1. Data Collection Methods and Sources**

There are number of data collection methods. The methods and sources used during the development of this system are described here.

### **Use primary and secondary data source:**

The collection of requirement data are collected from primary and secondary data sources. The actual observation of the scenario has helped in primary data collection. Data collected previously were used as the secondary source of data. Hence from these two sources we are able to visualize the requirements for the new system.

### **Questionnaires, survey and field visit:**

Questionnaires are provided to the people of some schools/colleges and to know about the current system and the system they want in better to the current system. The existing system is observed in the form of field visit. These helped in finding out the requirements for the new Question Preparation Application system.

### **Software resources used**

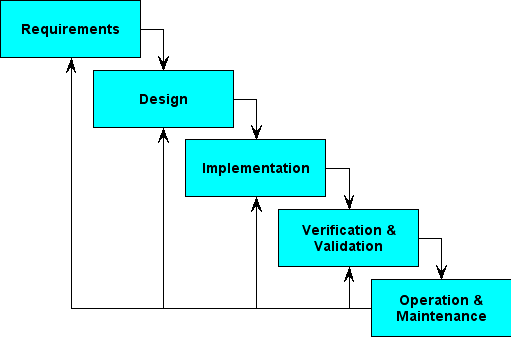
* Front End design : Microsoft Visual Studio 2012
* Back End : MS SQL Server R2
* Code-Behind Language : C#
* Other Requirement : .NET Framework 4.5

### **Hardware resources used**

* Processor : Intel (recommended)
* Memory : 128MB (minimum)
* Storage Devices : Hard disk 20 GB or above

# Chapter-7

## System Implementation



**Figure 6: classic life cycle or the waterfall model**

### **System Implementation**

System **implementation** fundamentally focuses on the coding, testing, installation and documentation of the system. The purpose of these activities is to convert the physically designed specification into the actual working of that system. It helps to convert the specification into reality.

### **Coding and Debugging Phase**

Coding is the process whereby the physical design specification created by the analysis and designed by the programming team. While implementing our system the coding is done in ASP.Net.

While coding phase was running debugging was also carried out in parallel. The iterative process of detection of new errors and their correction went thoroughly during the whole phase.

### **Verification and Validation**

Software testing is one of the important phases of the software development cycle referred to as verification and validation. This phase helps to remove the logical and theoretical error if systems fail to meet its objective. The system is tested with different module wise after completion of each module.

Verification refers to the process of evaluating system component to determine whether the products of a given development phase satisfy the condition ignored at the start of that phase.

Validation is process of evaluating a system or component during or at the end of the development process to determine whether it satisfy specified requirement.

### **Testing**

Testing is the verification and validation activity for the software product. It provides a final measure of quality assurance for the software product during the later phase of the system development cycle. Different types of testing technique were implemented during different phase of our system development cycle.

#### Unit Testing

Unit testing was undertaken when a module was coded and successfully reviewed. Various driver modules and stubs were prepared for the same. Modules required to provide the necessary environment were not available so stubs and drivers were designed to provide complete environment for the modules under test. Stub procedures were developed for testing purpose that had the same I/O parameters as units under test but these has a highly simplified behavior.

Unit testing was applied in each form after completion of designing and coding. Many errors and hazards were remove through each module and then corrected.

#### Integrating Testing

Integrating testing was carried out where related module was individually tested. They are merging together and tested with correlated data. Integration planning was performed before integrating all modules. With the help of Integrating testing uncovered error during module testing were fixed. Debugging of errors reported during this testing was challenging and it was interesting to fix them as it gave a clearer picture of the whole system.

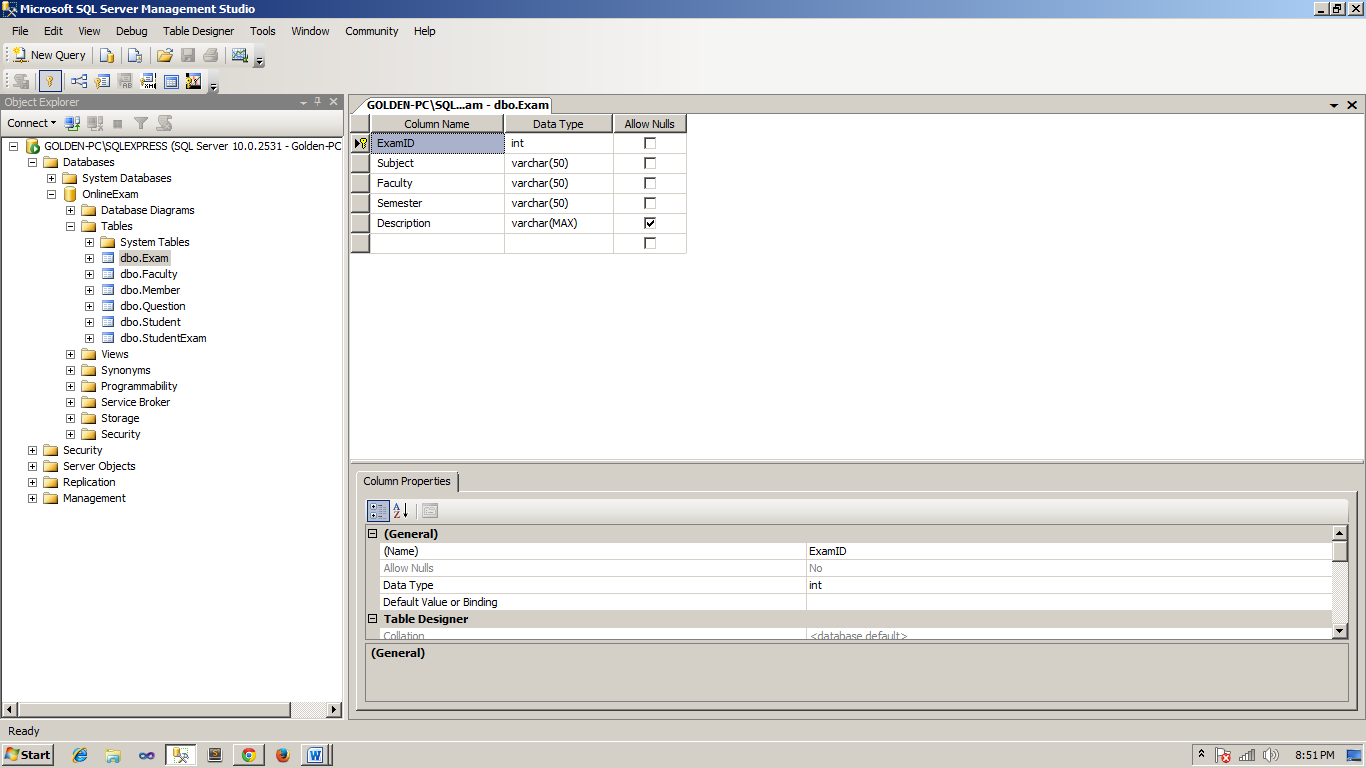
#### System Testing

It is the testing of whole system as a whole. The multiple modules were integrated into a single component and these components were tested. After completion of whole system, the system is tested in its original system. System is then need to tested in original environment. There are mainly two types of system testing.

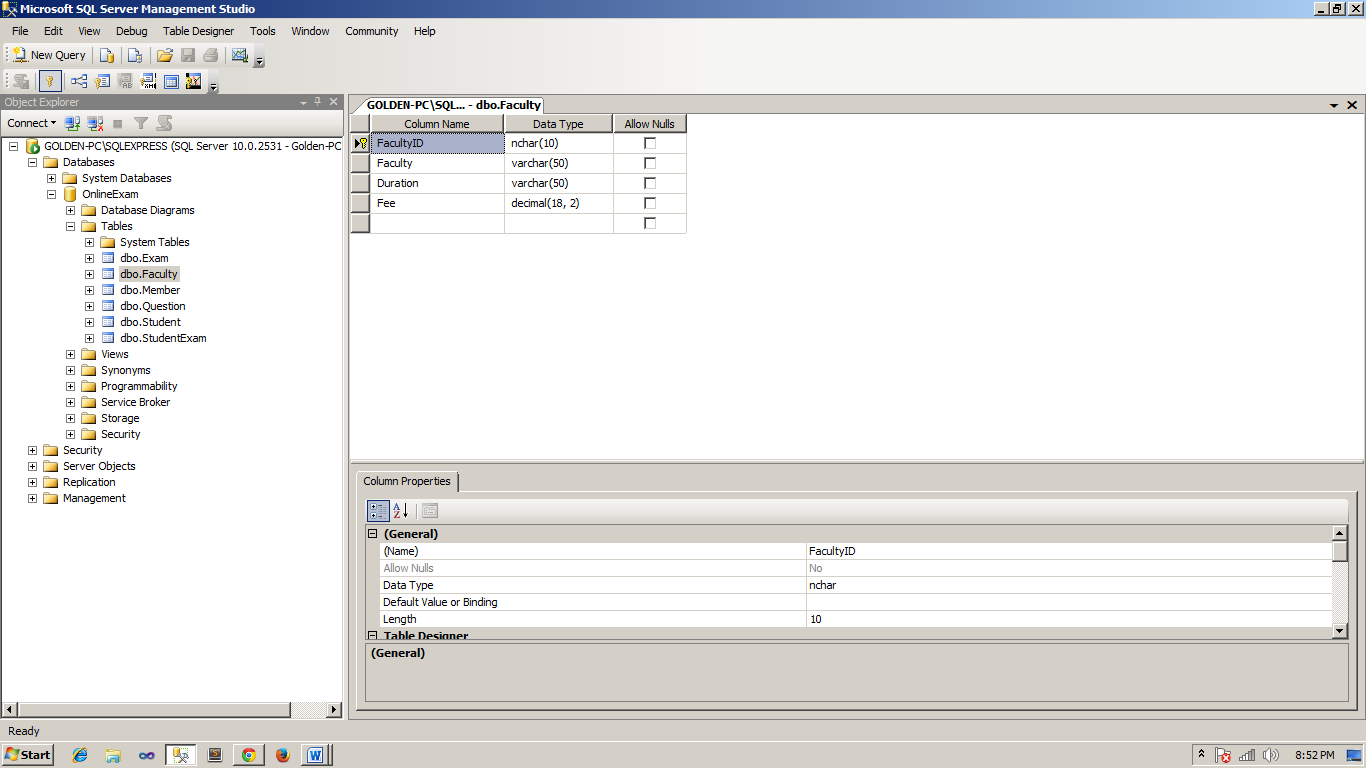
# Chapter–8

## Data Dictionary

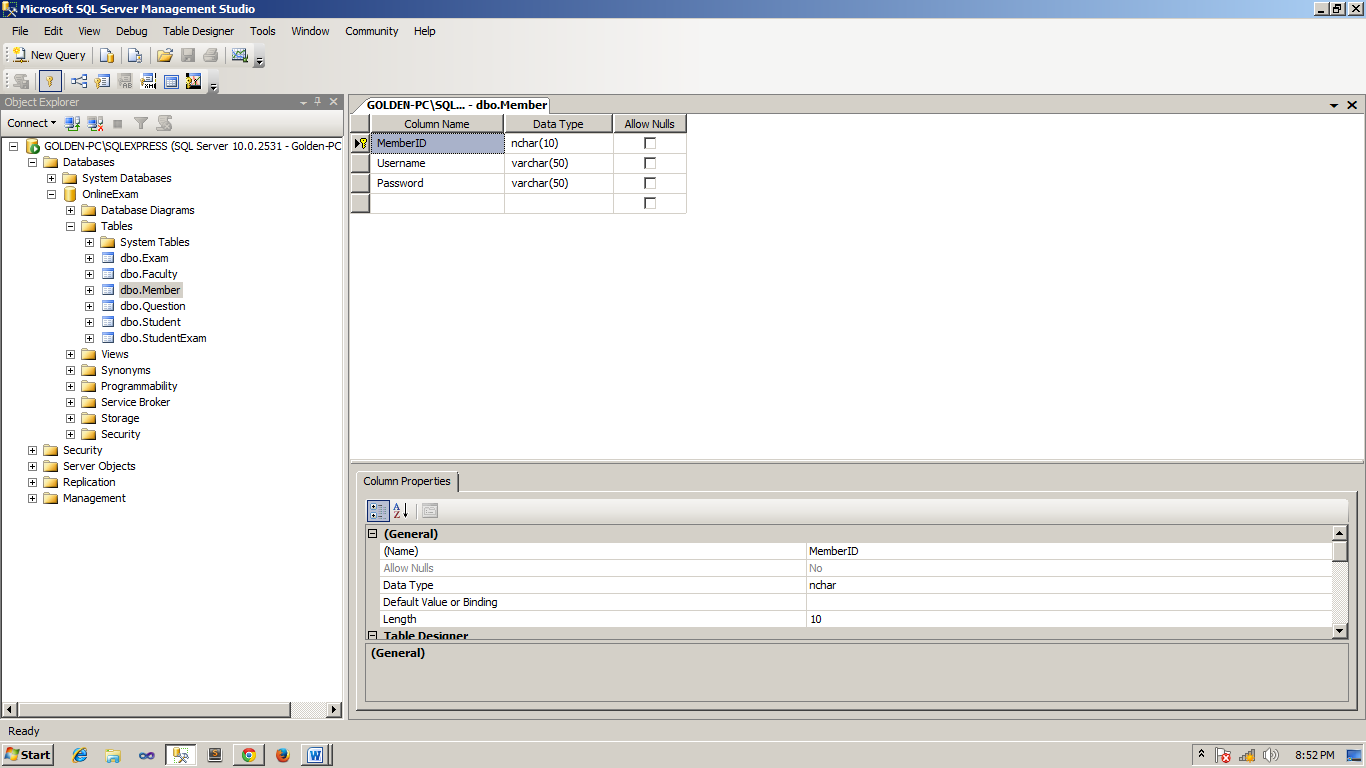
### **Exam**



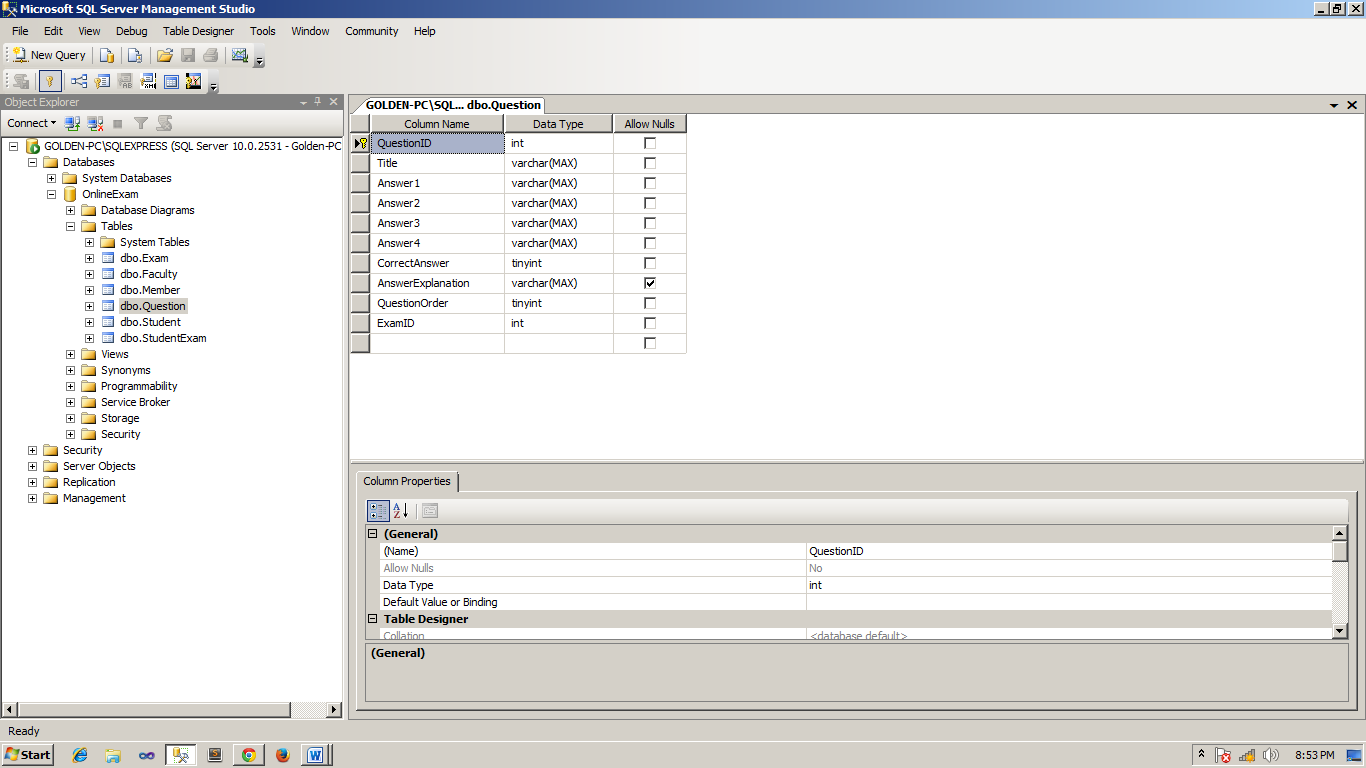
### **Faculty**



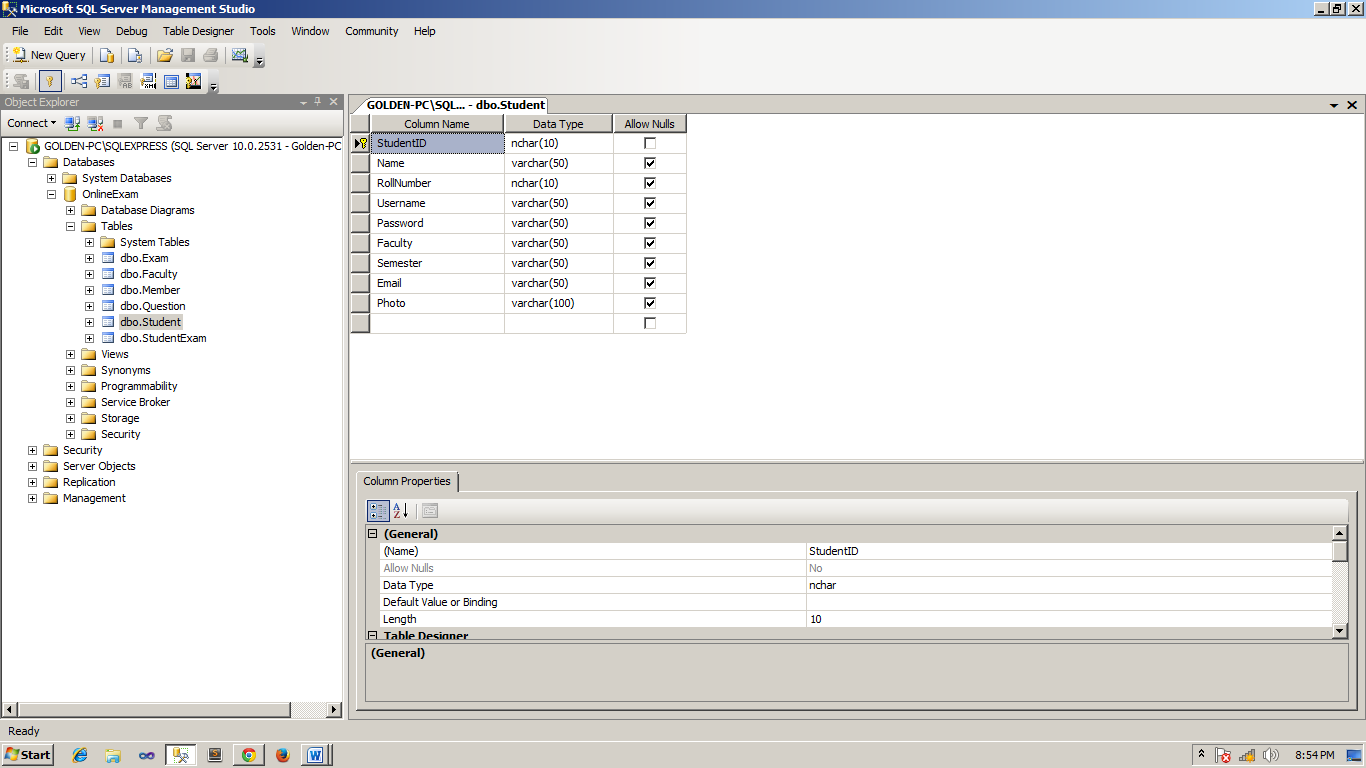
### **Member**



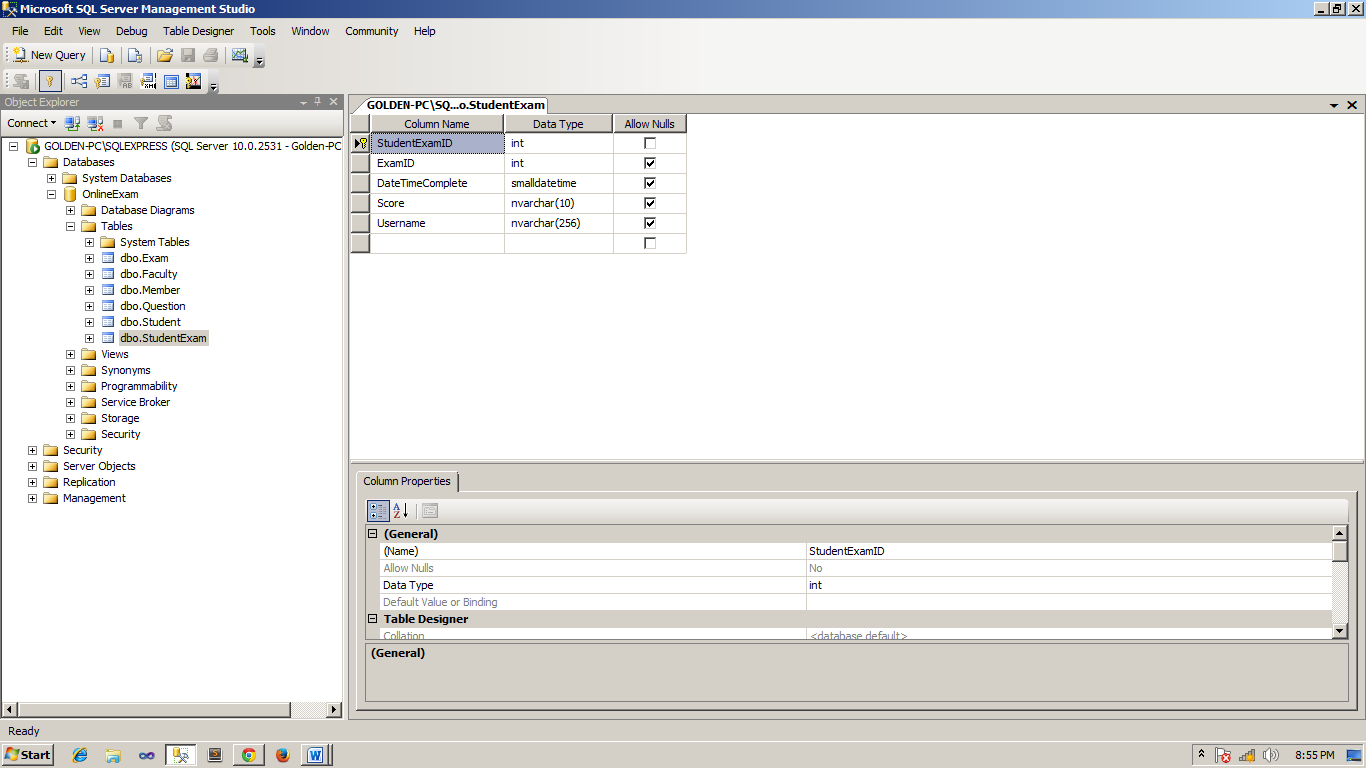
### **Question**



### **Student**



### **StudentExam**



# Chapter–9

## Snapshots and Codes

### **Codes**

#### Student.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Web.Security;

publicpartialclassstudent : System.Web.UI.Page

{

protectedvoid Page\_Load(object sender, EventArgs e)

{

if (!Page.IsPostBack)

{

FacultyDetail fd = newFacultyDetail();

DataTable dt = fd.BindFaculty();

ddlFaculty.DataSource = dt;

ddlFaculty.DataTextField = "Faculty";

ddlFaculty.DataBind();

}

txtUsername.Focus();

}

protectedvoid btnLogin\_Click(object sender, EventArgs e)

{

UserDetail ud = newUserDetail();

try

{

string hashedPassword = FormsAuthentication.HashPasswordForStoringInConfigFile(txtPassword.Text, "SHA1");

DataTable dt = ud.CheckStudent(txtUsername.Text, hashedPassword, ddlFaculty.SelectedItem.Value, ddlSemester.SelectedItem.Value);

if (dt.Rows.Count > 0)

{

Session["Student"] = txtUsername.Text;

Session["Faculty"] = ddlFaculty.SelectedItem.Value;

Session["Semester"] = ddlSemester.SelectedItem.Value;

Response.Redirect("MyProfile.aspx");

}

else

{

ltrMessage.Text = "Invallid User or Password!";

}

}

catch (Exception ex)

{

ltrMessage.Text = ex.Message;

}

}

}

#### ChangePasswordStudent.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.Security;

using System.Data;

using System.Data.SqlClient;

using System.Configuration;

publicpartialclassChangePassword : System.Web.UI.Page

{

protectedvoid Page\_Load(object sender, EventArgs e)

{

}

protectedvoid btnUpdate\_Click(object sender, EventArgs e)

{

if (ChangeUserPassword())

{

lblMessage.Text = "Password Changed Successfully!";

txtCurrentPassword.Text = "";

txtNewPassword.Text = "";

txtConfirmNewPassword.Text = "";

txtCurrentPassword.Focus();

}

else {

lblMessage.ForeColor = System.Drawing.Color.Red;

lblMessage.Text = "Invalid Current Password";

txtCurrentPassword.Text = "";

txtNewPassword.Text = "";

txtConfirmNewPassword.Text = "";

txtCurrentPassword.Focus();

}

}

privatebool ChangeUserPassword() {

List<SqlParameter> paramList = newList<SqlParameter>(){

newSqlParameter(){

ParameterName="@Username",

Value=Session["Student"].ToString()

//Value=User.Identity.Name

},

newSqlParameter(){

ParameterName="@CurrentPassword",

//Value=txtCurrentPassword.Text

Value=FormsAuthentication.HashPasswordForStoringInConfigFile(txtCurrentPassword.Text,"SHA1")

},

newSqlParameter(){

ParameterName="@NewPassword",

//Value=txtNewPassword.Text

Value=FormsAuthentication.HashPasswordForStoringInConfigFile(txtNewPassword.Text,"SHA1")

}

};

return ExecuteSP("spChangePasswordStudent", paramList);

}

privatebool ExecuteSP(String SPName, List<SqlParameter> SPParameters)

{

string CS = ConfigurationManager.ConnectionStrings["OnlineExamConnectionString"].ConnectionString;

using (SqlConnection con = newSqlConnection(CS))

{

SqlCommand cmd = newSqlCommand(SPName, con);

cmd.CommandType = CommandType.StoredProcedure;

foreach (SqlParameter parameter in SPParameters)

{

cmd.Parameters.Add(parameter);

}

con.Open();

returnConvert.ToBoolean(cmd.ExecuteScalar());

}

}

}

#### Questions.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data.SqlClient;

using System.Collections;

publicpartialclassQuestions : System.Web.UI.Page

{

protectedvoid Page\_Load(object sender, EventArgs e)

{

questionDetails.DataBind();

GlobalConnection gc = newGlobalConnection();

SqlCommand cmd = newSqlCommand("SELECT Subject FROM Exam WHERE ExamID='" + Session["ExamID"].ToString() + "'", gc.cnn);

SqlDataReader dr;

dr = cmd.ExecuteReader();

while (dr.Read())

{

lblCurrentTest.Text = dr["Subject"].ToString();

}

}

protectedvoid nextButton\_Click(object sender, EventArgs e)

{

System.Data.DataRowView dr = (System.Data.DataRowView)questionDetails.DataItem;

Answer a = newAnswer();

a.QuestionID = dr["QuestionOrder"].ToString();

a.CorrectAnswer = dr["CorrectAnswer"].ToString();

a.UserAnswer = ddlAnswer.SelectedValue.ToString();

ArrayList al = (ArrayList)Session["AnswerList"];

al.Add(a);

Session.Add("AnswerList", al);

if (questionDetails.PageIndex == questionDetails.PageCount - 1)

{

Response.Redirect("Results.aspx");

}

else

{

questionDetails.PageIndex++;

}

if (questionDetails.PageIndex == questionDetails.PageCount - 1)

{

btnNext.Text = "Finished";

}

}

}

#### LogoutStudent.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

publicpartialclassStudentLogout : System.Web.UI.Page

{

protectedvoid Page\_Load(object sender, EventArgs e)

{

Session["Student"] = null;

Response.Redirect("LoginStudent.aspx");

}

}

#### AdminMasterPage.master

<%@MasterLanguage="C#"AutoEventWireup="true"CodeFile="AdminMasterPage.master.cs"Inherits="AdminMasterPage"%>

<!DOCTYPEhtmlPUBLIC"-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<htmlxmlns="http://www.w3.org/1999/xhtml">

<headrunat="server">

<title></title>

<asp:ContentPlaceHolderid="head"runat="server">

</asp:ContentPlaceHolder>

<linkhref="Stylesheet/style2.css"rel="stylesheet"type="text/css"/>

<styletype="text/css">

.style1

{

color: #FFFFFF;

font-size:small;

}

</style>

</head>

<body>

<formid="form1"runat="server">

<divid="container">

<divid="header">

<h1>OeS</h1>

</div><!--end of header -->

<divid="navigation">

<ul>

<li><ahref="Home.aspx">Home</a></li>

<li>Settings

<ul>

<li><ahref="ManageStudent.aspx">Student</a></li>

<li><ahref="Member.aspx">Member</a></li>

<li><ahref="ChangePasswordMember.aspx">Change Password</a></li>

</ul>

</li>

<li><ahref="Faculty.aspx">Faculty</a></li>

<li><ahref="Subject.aspx">Subject</a></li>

<li><ahref="DisplayQuestion.aspx">Question</a></li>

<li><ahref="LogoutMember.aspx">Logout</a></li>

</ul>

</div><!-- end of naviagaion -->

<divstyle="text-align:right;">

<spanclass="style1"><strong>Welcome! </strong></span>

<asp:Label

ID="lblWelcome"runat="server"style="color: #FF6600"></asp:Label>

</div><!-- user Status-->

<divid="content"style="padding:20px;">

<asp:ContentPlaceHolderID="ContentPlaceHolder1"runat="server">

</asp:ContentPlaceHolder>

</div><!-- end of content -->

<divstyle="clear:both;"></div>

</div><!--end of container -->

<divid="footer">

<p>&copy;Copy Right 2014. All Right Reserved <ahref="#">Prakash Tamang</a></p>

</div><!--end of footer-->

</form>

</body>

</html>

#### AdminMasterPage.master.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

publicpartialclassAdminMasterPage : System.Web.UI.MasterPage

{

protectedvoid Page\_Load(object sender, EventArgs e)

{

if (!Page.IsPostBack)

{

if (Session["Member"] != null)

{

lblWelcome.Text += Session["Member"].ToString();

}

else

{

Response.Redirect("admin.aspx");

}

}

}

}

#### AddStudent.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Web.Security;

publicpartialclassAddStudent : System.Web.UI.Page

{

protectedvoid Page\_Load(object sender, EventArgs e)

{

if (!Page.IsPostBack)

{

FacultyDetail fd = newFacultyDetail();

DataTable dt = fd.BindFaculty();

ddlFaculty.DataSource = dt;

ddlFaculty.DataTextField = "Faculty";

ddlFaculty.DataBind();

}

}

protectedvoid btnCreateStudent\_Click(object sender, EventArgs e)

{

try

{

UserDetail ud = newUserDetail();

string hashedPassword = FormsAuthentication.HashPasswordForStoringInConfigFile(txtRollNumber.Text, "SHA1");

ud.AddStudent(txtStudentID.Text, txtStudentName.Text, txtRollNumber.Text, txtRollNumber.Text, hashedPassword, ddlFaculty.SelectedItem.Value, ddlSemester.SelectedItem.Value, txtEmail.Text, "Student/" + FileUpload1.FileName);

FileUpload1.SaveAs(Server.MapPath("~/Student/" + FileUpload1.FileName));

ltrMessage.Text = "Saved Successfully!";

txtStudentID.Text = "";

txtStudentName.Text = "";

txtRollNumber.Text = "";

txtEmail.Text = "";

txtStudentID.Focus();

}

catch(Exception){

ltrMessage.Text = "Student Already Exists";

}

}

}

#### DisplayQuestion.aspx

<%@PageTitle="Add Questions"Language="C#"MasterPageFile="~/AdminMasterPage.master"AutoEventWireup="true"CodeFile="DisplayQuestion.aspx.cs"Inherits="Default2"%>

<asp:ContentID="Content1"ContentPlaceHolderID="head"Runat="Server">

<styletype="text/css">

.style2

{

color: #000000;

}

.style3

{

font-size: 22pt;

color: #FF6600;

}

</style>

</asp:Content>

<asp:ContentID="Content2"ContentPlaceHolderID="ContentPlaceHolder1"Runat="Server">

<p>

<asp:ImageButtonID="ImageButton1"runat="server"

ImageUrl="~/Images/Button-Add-icon.png"onclick="ImageButton1\_Click"/>

&nbsp;<spanclass="style3"><strong>Questions</strong></span></p>

<p>

<spanclass="style2">Select ExamID:

<asp:DropDownListID="ddlExamID"runat="server"AutoPostBack="True">

</asp:DropDownList>

</span>

&nbsp;<asp:SqlDataSourceID="SqlDataSource1"runat="server"

ConnectionString="<%$ ConnectionStrings:OnlineExamConnectionString %>"

SelectCommand="SELECT DISTINCT [ExamID] FROM [Question]">

</asp:SqlDataSource>

</p>

<p>

<asp:GridViewID="GridView1"runat="server"AllowSorting="True"

AutoGenerateColumns="False"BackColor="White"BorderColor="#336666"

BorderStyle="Double"BorderWidth="3px"CellPadding="4"

DataKeyNames="QuestionID"DataSourceID="SqlDataSource2"GridLines="Horizontal">

<Columns>

<asp:TemplateFieldShowHeader="False">

<EditItemTemplate>

<asp:LinkButtonID="LinkButton1"runat="server"CausesValidation="True"

CommandName="Update"Text="Update"></asp:LinkButton>

&nbsp;<asp:LinkButtonID="LinkButton2"runat="server"CausesValidation="False"

CommandName="Cancel"Text="Cancel"></asp:LinkButton>

</EditItemTemplate>

<ItemTemplate>

<asp:LinkButtonID="LinkButton1"runat="server"CausesValidation="False"

CommandName="Edit"Text="Edit"></asp:LinkButton>

&nbsp;

<spanonclick="return confirm('Are you sure you want to Delete the Record?')">

<asp:LinkButtonID="LinkButton2"runat="server"CausesValidation="False"

CommandName="Delete"Text="Delete"></asp:LinkButton>

</span>

</ItemTemplate>

</asp:TemplateField>

<asp:BoundFieldDataField="QuestionID"HeaderText="QuestionID"ReadOnly="True"

SortExpression="QuestionID"/>

<asp:BoundFieldDataField="Title"HeaderText="Title"SortExpression="Title"/>

<asp:BoundFieldDataField="Answer1"HeaderText="Answer1"

SortExpression="Answer1"/>

<asp:BoundFieldDataField="Answer2"HeaderText="Answer2"

SortExpression="Answer2"/>

<asp:BoundFieldDataField="Answer3"HeaderText="Answer3"

SortExpression="Answer3"/>

<asp:BoundFieldDataField="Answer4"HeaderText="Answer4"

SortExpression="Answer4"/>

<asp:BoundFieldDataField="QuestionOrder"HeaderText="QuestionOrder"

SortExpression="QuestionOrder"/>

</Columns>

<FooterStyleBackColor="White"ForeColor="#333333"/>

<HeaderStyleBackColor="#336666"Font-Bold="True"ForeColor="White"/>

<PagerStyleBackColor="#336666"ForeColor="White"HorizontalAlign="Center"/>

<RowStyleBackColor="White"ForeColor="#333333"/>

<SelectedRowStyleBackColor="#339966"Font-Bold="True"ForeColor="White"/>

<SortedAscendingCellStyleBackColor="#F7F7F7"/>

<SortedAscendingHeaderStyleBackColor="#487575"/>

<SortedDescendingCellStyleBackColor="#E5E5E5"/>

<SortedDescendingHeaderStyleBackColor="#275353"/>

</asp:GridView>

<asp:SqlDataSourceID="SqlDataSource2"runat="server"

ConnectionString="<%$ ConnectionStrings:OnlineExamConnectionString %>"

DeleteCommand="DELETE FROM [Question] WHERE [QuestionID] = @QuestionID"

InsertCommand="INSERT INTO [Question] ([QuestionID], [Title], [Answer1], [Answer2], [Answer3], [Answer4], [QuestionOrder]) VALUES (@QuestionID, @Title, @Answer1, @Answer2, @Answer3, @Answer4, @QuestionOrder)"

SelectCommand="SELECT [QuestionID], [Title], [Answer1], [Answer2], [Answer3], [Answer4], [QuestionOrder] FROM [Question] WHERE ([ExamID] = @ExamID)"

UpdateCommand="UPDATE [Question] SET [Title] = @Title, [Answer1] = @Answer1, [Answer2] = @Answer2, [Answer3] = @Answer3, [Answer4] = @Answer4, [QuestionOrder] = @QuestionOrder WHERE [QuestionID] = @QuestionID">

<DeleteParameters>

<asp:ParameterName="QuestionID"Type="Int32"/>

</DeleteParameters>

<InsertParameters>

<asp:ParameterName="QuestionID"Type="Int32"/>

<asp:ParameterName="Title"Type="String"/>

<asp:ParameterName="Answer1"Type="String"/>

<asp:ParameterName="Answer2"Type="String"/>

<asp:ParameterName="Answer3"Type="String"/>

<asp:ParameterName="Answer4"Type="String"/>

<asp:ParameterName="QuestionOrder"Type="Byte"/>

</InsertParameters>

<SelectParameters>

<asp:ControlParameterControlID="ddlExamID"DefaultValue="NULL"Name="ExamID"

PropertyName="SelectedValue"Type="Int32"/>

</SelectParameters>

<UpdateParameters>

<asp:ParameterName="Title"Type="String"/>

<asp:ParameterName="Answer1"Type="String"/>

<asp:ParameterName="Answer2"Type="String"/>

<asp:ParameterName="Answer3"Type="String"/>

<asp:ParameterName="Answer4"Type="String"/>

<asp:ParameterName="QuestionOrder"Type="Byte"/>

<asp:ParameterName="QuestionID"Type="Int32"/>

</UpdateParameters>

</asp:SqlDataSource>

</p>

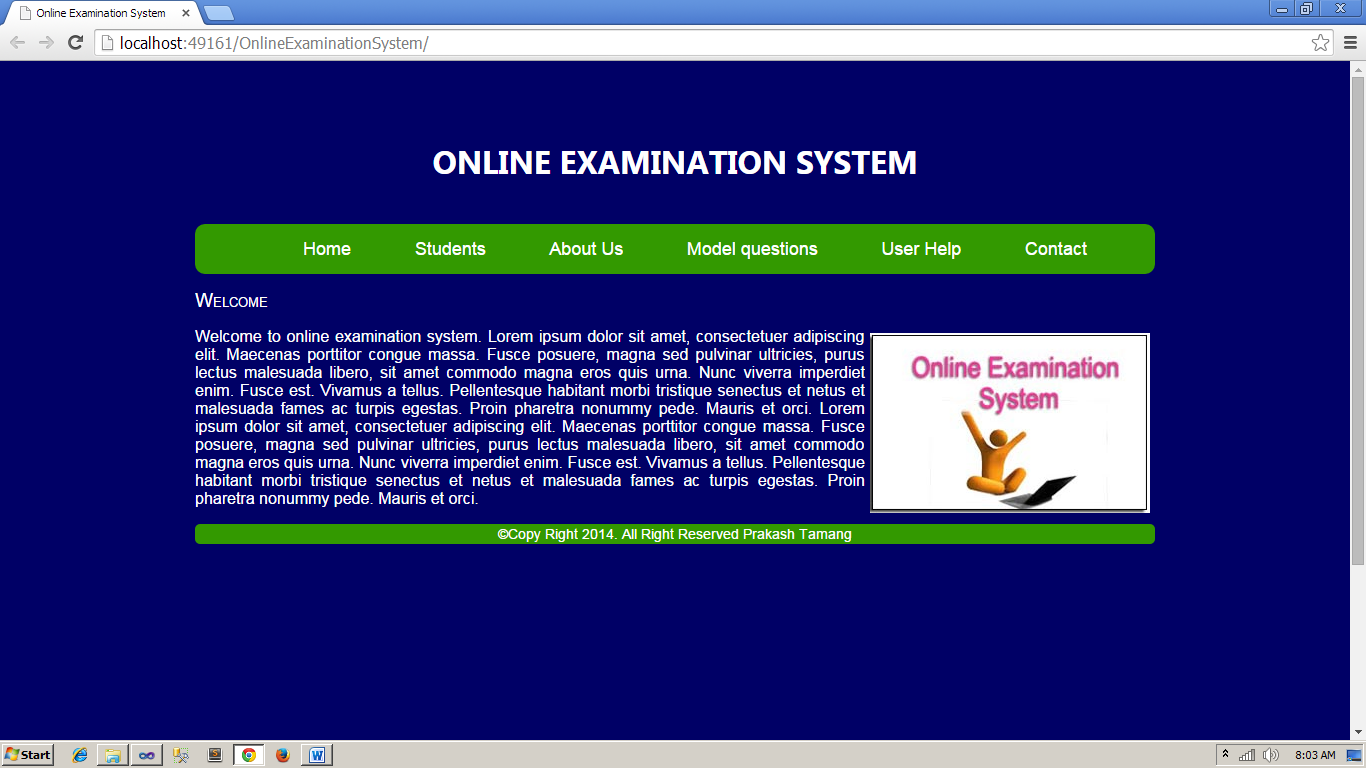
<p>

&nbsp;</p>

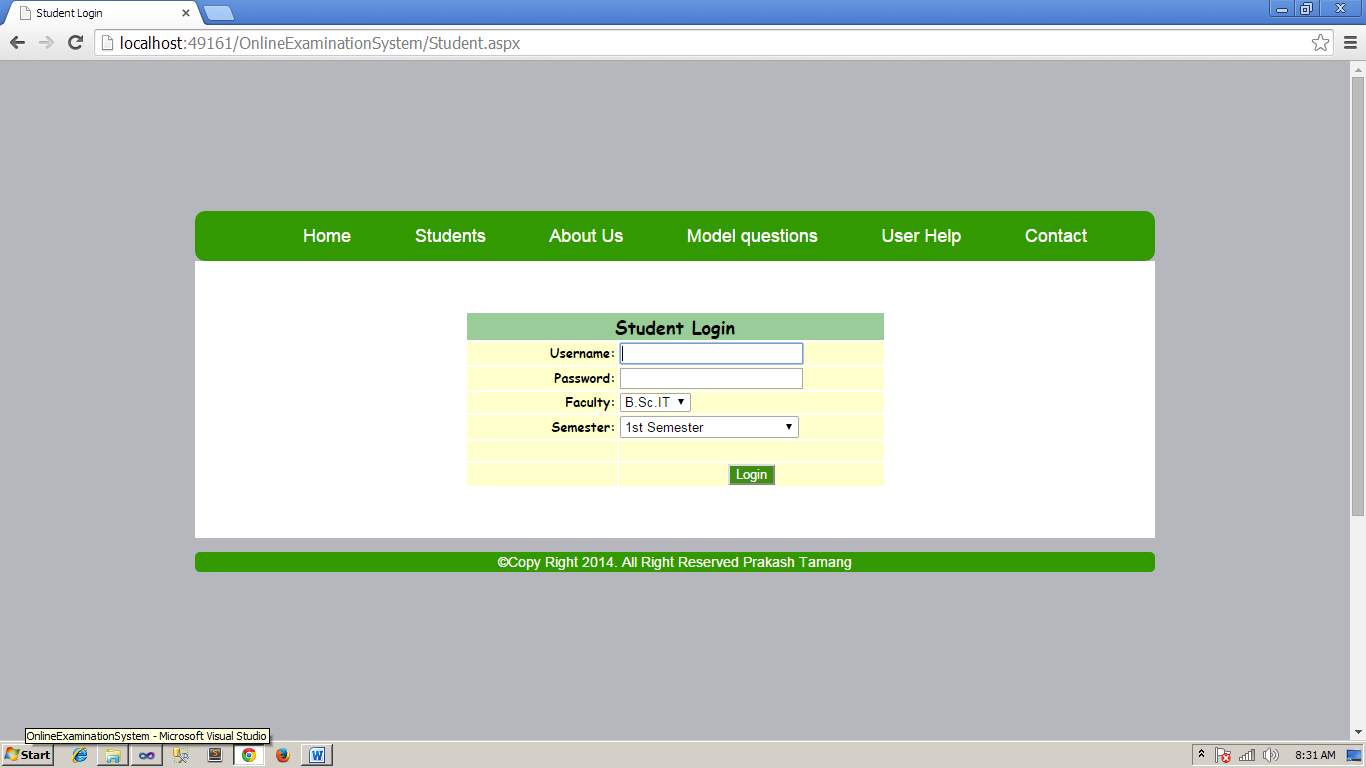
</asp:Content>

### **Snapshorts**

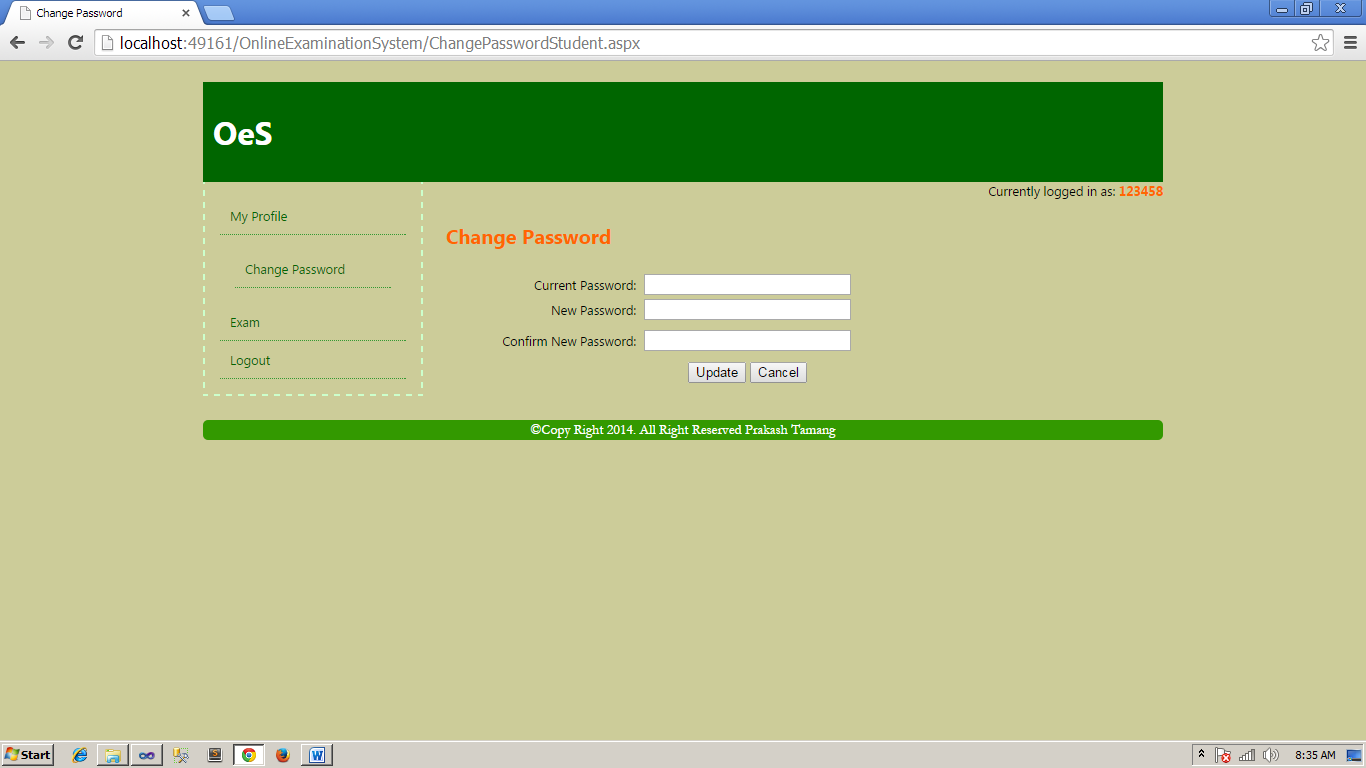
#### Default.aspx



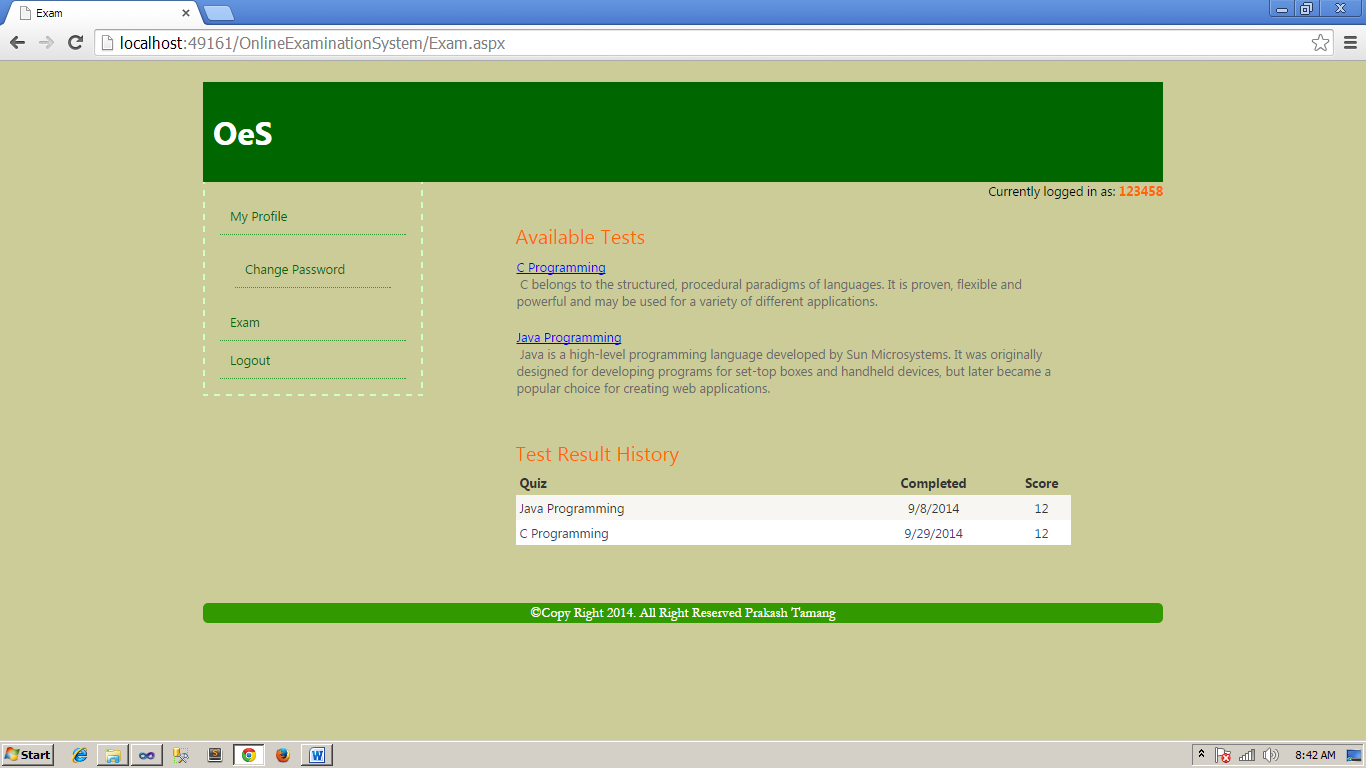
#### 9.2.2. Student.aspx



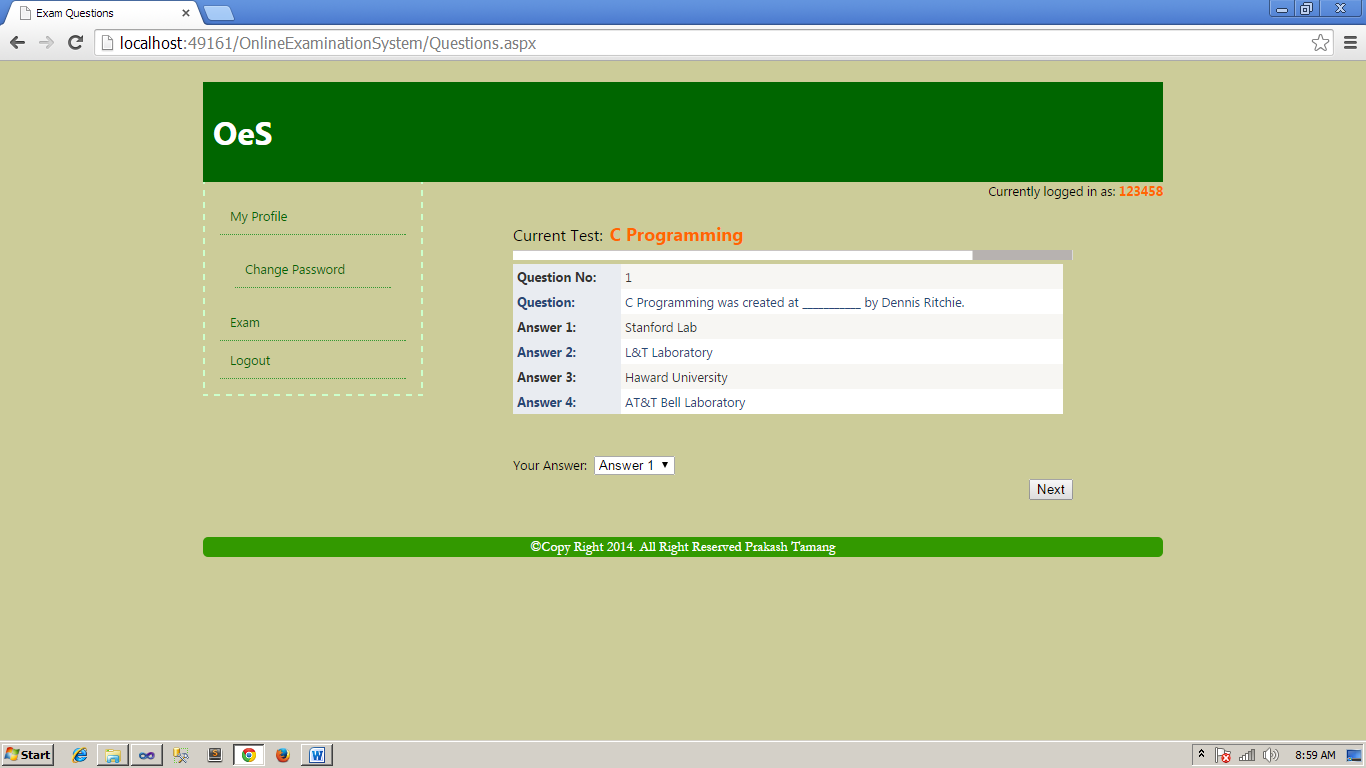
#### ChangePasswordStudent.aspx



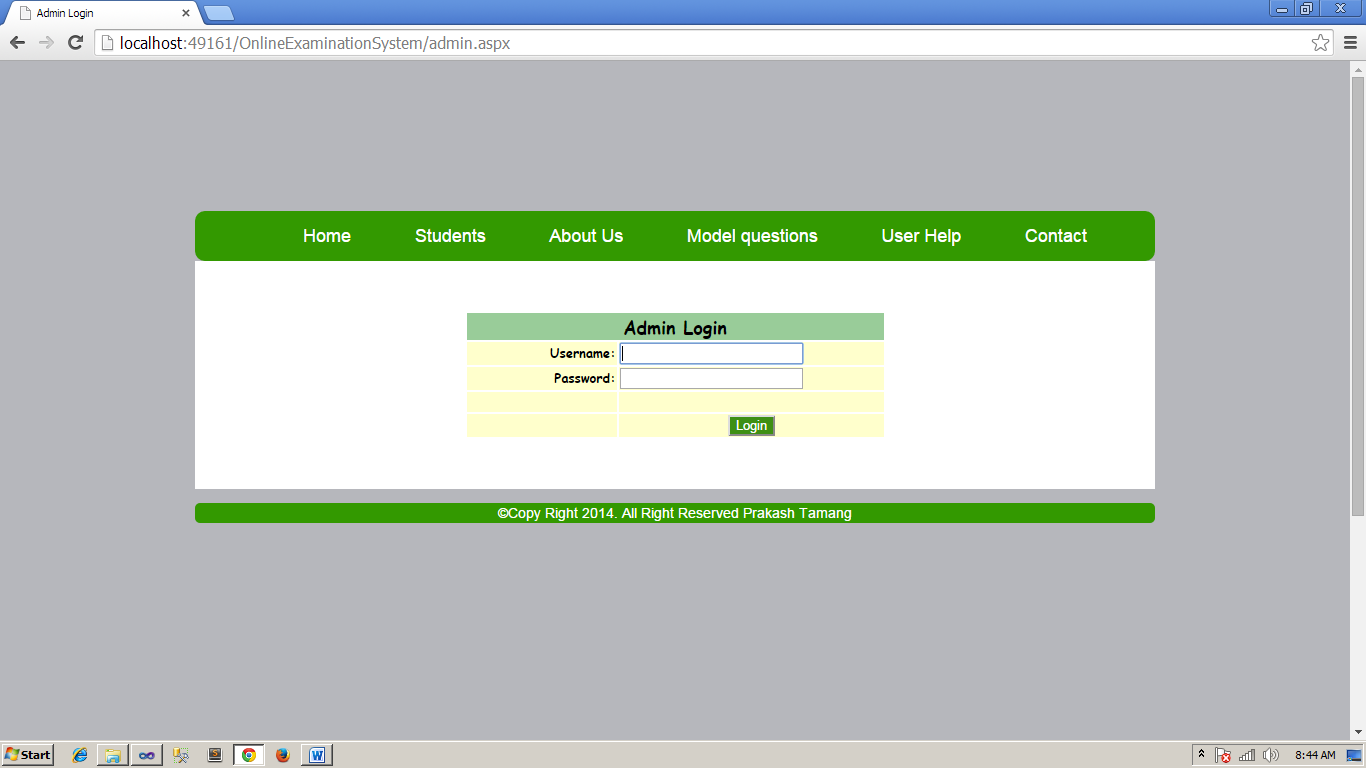
#### Exam.aspx (Design)



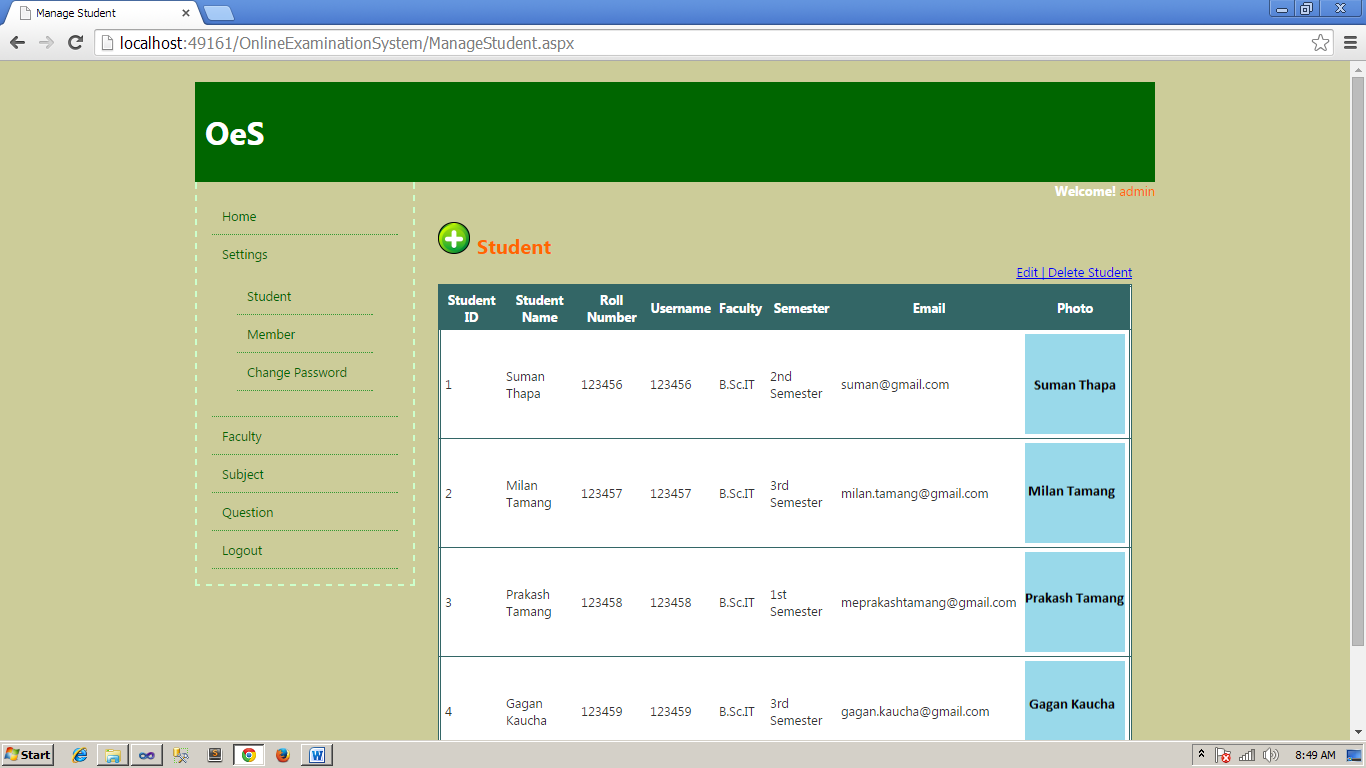
#### Questions.aspx (Design)



#### Admin.aspx (Design)



#### ManageStudent.aspx (Design)



#### Subject.aspx (Design)



#### DisplayQuestion.aspx (Design)



# Chapter–10

## Future Scope of the project

The development of this project surely prompts many new areas of investigation. This project has wide scope to implement it in any University/Institution having multiple paper there. This project covers all functionalities related to On-Line Examination Hall Hence it can be implemented any-where else after minute organization level customization

Moreover some parts of the project have remained uncompleted due to some reasons. First of all limitations of our project, which has been discussed in previous topic make place for future enhancements. Though that was not the part of objective of our project but it would have great to implement that provided we’d enough time.

Some of them are all those which are enumerated in the limitation section like:

* Online Payment
* A module from where entities can communicate
* Subjective paper evaluation

## Bibliography

* “ASP.NET Black Book”
* ”A Programmer’s Guide to ADO.NET in C#” by Mahesh Chand.
* ASP.Net Tutorial, http://www.lynda.com
* ASP.Net Tutorial, <http://www.asp.net.com>