INDIRA GANDHI NATIONAL OPEN UNIVERSITY

PROJECT REPORT

MCS-044

"Online Entrance Exam"

By

PYAR MOHAMMAD

Enrollment No.-166527535

Submitted to the School of Computer and Information Sciences,
IGNOU
in partial fulfillment of the requirements
for the award of the degree
Master of Computer Applications (MCA)

2020
Indira Gandhi National Open University
Maidan Garhi
New Delhi – 110068.

TABLE OF CONTENT

INTRODUCTION

The Online Entrance Exam System is an electronic application. This framework will help the University to assess the inquiry have different alternative with one right reply. The University can direct the online examination and report the outcome in a couple time. The Online Entrance Exam office is in charge of the creating the inquiry paper and it would be totally secure. Online Entrance Exam framework give remotely access to understudies. It assists the inspector with reducing the work

Objective

Online entrance exam system is a non removable examination pattern of today's life. We need more time saving and more accurate entrance exam system as the number of applicants is increasing day by day. For all IT students and professionals, it is very important to have some basic understanding about the online entrance exam system. On this site you will get source code with the running project. It will help you to understand the concept of the project. Here **you find project in ASP.NET free download**.

PURPOSE ANS SCOPE

PURPOSE: The purpose of on-line test simulator is to take online test in an efficient manner and no time wasting for checking the paper. The main objective of on-line test simulator is to efficiently evaluate the candidate thoroughly through a fully automated system that not only saves lot of time but also gives fast results.

For students they give papers according to their convenience and time and there is no need of using extra thing like paper, pen etc.

SCOPE: Scope of this project is very broad in terms of other manually taking exams. Few of them are :-
This can be used in educational institutions as well as in corporate world.
Can be used anywhere any time as it is a web based application(user Location does not matter).
No restriction that examiner has to be present when the candidate takes the test.

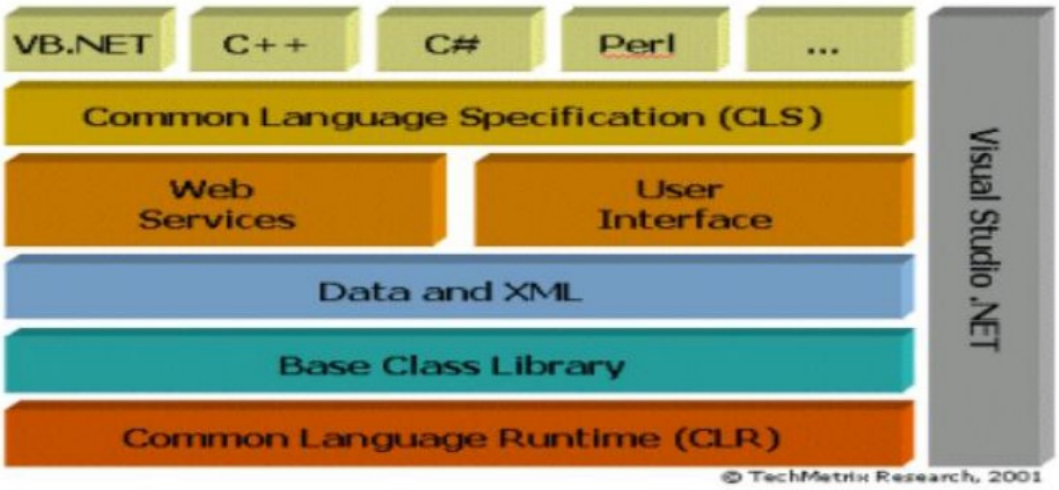
FEATURES:

- Secure
- Easy to use
- Reliable and accurate
- No need of examiner

SURVEY OF TECHNOLOGIES

About.NET

Microsoft recently announced the .NET platform as their latest vision for building, deploying, and running distributed applications and systems across the Internet. At its core is a virtual machine that turns intermediate language (IL) into machine code. High-level language compilers for C#, VB.NET and ASP.NET are provided to turn source code into IL. The .NET platform takes advantage of several new technology standards, such as extended Markup Language (XML) and Simple Object Access Protocol (SOAP), to fully utilize the abundance of computing and communications resources available and in use today. It is a set of common services which can be used from a number of object languages.



Benefits of .NET

What are the benefits of the .NET Framework? Put simply, faster time to make, easier deployment and administration, and improved performance. Here are some of the key benefits:

Any Language. The .NET Framework enables custom software developers to use any programming language.

Reliability. The .NET Framework includes technologies to make applications more reliable.

Mobility. The .NET Framework makes large advances with regards to mobile devices.

Manageability. The .NET Framework goes to incredible lengths to make it easy to deploy, run, and manage applications.

Security. The .NET Framework includes an evidence-based security system.

The Microsoft .NET Framework

The .NET Framework is the infrastructure for the Microsoft .NET platform.
The .NET Framework is an environment for building, deploying, and running Web applications and Web Services.
Microsoft's first server technology ASP (Active Server Pages), was a powerful and flexible "programming language". But it was too code oriented. It was not an application framework and not an enterprise development tool.
The Microsoft .NET Framework was developed to solve this problem.

.NET Frameworks keywords:

- Easier and quicker programming
- Reduced amount of code
- Declarative programming model
- Richer server control hierarchy with events
- Larger class library
- Better support for development tools

The .NET Framework consists of 3 main parts:

- Entity Framework
- C# (Pronounced C sharp)
- Visual Basic (VB .NET)
- ASP .NET MVC (Active Server Pages)

- Windows Forms (Windows desktop solutions)
- Compact Framework (PDA / Mobile solutions)
- Development environments:
- Visual Studio .NET (VS .NET)

ASP MVC.NET

ASP MVC.NET Features:-

- Form authentication
- Window authentication
- URL authentication
- Membership and roles
- Output and data caching
- Session and profitable management
- Provider architecture

DATABASE MICROSOFT SQL SERVER

Microsoft SQL Server is a full-featured relational database management system (RDBMS) that offers a variety of administrative:

Enterprise Manager
Query Analyzer
SQL Profiles
Service Manager
Data Transformation Services (DTS)

Hardware and Software Requirements:

Hardware Requirements:-

Processor : P4 or higher

RAM : 512MB

HDD : 1GB of free space

3.4.2. Software Requirements:-

Front-end : ASP.NET MVC

Back-end : Microsoft SQL Server

Platform : .NET framework(Entity Framework)

Operating system : Windows XP or Higher Version

System Analysis

- Systems analysis is a process of collecting factual data, understand the processes involved, identifying problems and recommending feasible suggestions for improving the system functioning. This involves studying the business processes, gathering operational data, understand the information flow, finding out bottlenecks and evolving solutions for overcoming the weaknesses of the system so as to achieve the organizational goals.

- Systems analysis and design, as performed by systems analysts, seeks to understand what humans need to analyze data input or data flow systematically, process or transform data, store data, and output information in the context of a particular business.

Feasibility Study

An analysis of the ability to complete a project successfully, taking into account legal, economic, technological, scheduling and other factors.
A feasibility study is to determine the viability of a business venture in a specific area or sector of business.

There are three types of feasibility study

1. Economical feasibility
2. Operational feasibility
3. Technical feasibility

Economic Feasibility

- In economic feasibility, the most important is cost-benefit analysis. As the name suggests, it is an analysis of the costs to be incurred in the system and benefits derivable out of the system.

- System is economically feasible due to following points :- Benefits in reducing the cost are in the form of staff cut off. The cost incurred to implemented the system are the payment of the data entry operator, a little maintenance required for the hardware and software from time to time consistency in efficiency.

Technical Feasibility

Technical feasibility determines whether the work for the project can be done with the existing equipment, software technology and available personnel. Technical feasibility is concerned with specifying equipment and software that will satisfy the user requirement. The proposed system can run on any amplifier and also connect on laptop, pc, mobile etc. Using this we play a songs online through internet.

I used of programming language enables me to develop software that can help End-User to operate system easily. The use of Front end ASP.net using C# & back-end Link Database

Operational Feasibility

It is all about problem that may arise during the operations. It is related to the operational issue related to the accuracy, response time, security, easy to use, services etc. Generally a software is not rejected on the basic of operational feasibility as they can be improved.

As the system is user-friendly, throughout the system is also well liked and approved by the user showing no resistance what so ever at all. If the user wants facility, it can be provided. The application without causing any harm to the organization will enhance the results in the better respect of the new system and will avoid the confusion and resistance by catching the user's attention.

Fact Finding

Fact finding is process of collection of data and information based on techniques which contain sampling of existing documents, research, observation, questionnaires, interviews, prototyping and joint requirements planning.

Interview

This technique of fact-finding is most popular, productive for good analysts and most probably widely used. Interviews are a fact-finding technique where by the systems analyst collects information from individual fact to face. Interviewing can be used to find-facts; verify facts; clarify facts; general enthusiasm etc.

Some question Like

- How you do get the query from the customer?
- How do you do get the payment of the services and hardware parts from the customers / organizations?
- How do you do maintain the machine records and their bills?
- How do you do assign the complaints to the engineers? And much more.

On-site Observation

Observation could be Formal or Informal. This is most effective when and analyst wants to obtain an understanding of a system. This technique used when analyst wants either participates in or watches a person perform activities to learn about the system.

First of all I went to the local computer shop and store. I met every staff member in the computer shop and store. I went to the complaint desk and I ask how do you do solve the complaints and manage their data, payments etc? After that i went the document store. I observed the documents, objects, occurrences of events, forms etc.

Questionnaires

This is a special purpose document that allows the analyst to collect information and opinions from respondents. Questionnaires become useful when a little information is required from a number of people.

Sampling

To follow this particular method of fact-finding, Analyst has to study well existing documentation, forms, and files of existing system. A good analyst gets fact first from existing documentation rather than from people.

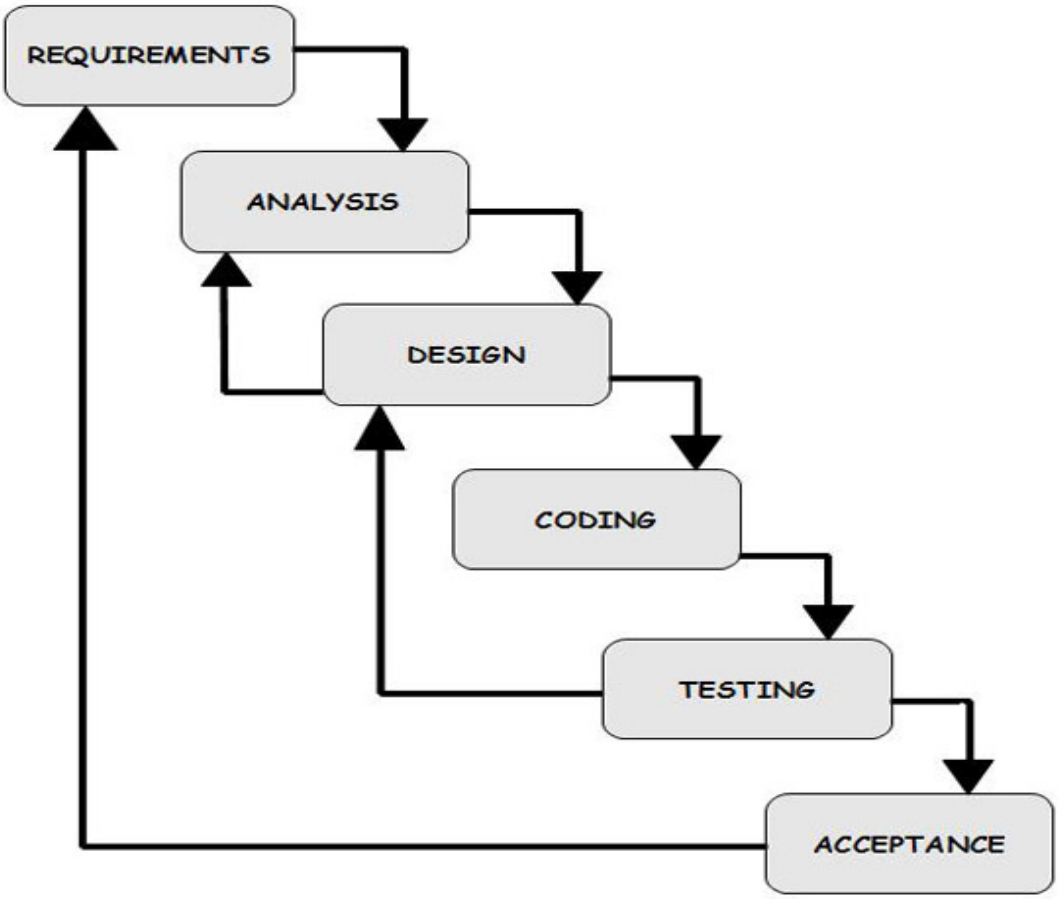
The analyst collects samples of all relevant documents, such as sample payment slips, complaint forms, report forms, and other relevant forms. To create a records of machines, complaints, engineers etc. The assembled documents help me understand what data the new Tech-Com System must collect and process.

LIFE CYCLE METHODOLOGY

WATERFALL MODEL

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed fully before the next phase can begin.

This type of model is basically used for the for the project which is small and there are no uncertain requirements. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project. In this model the testing starts only after the development is complete. In waterfall model phases do not overlap.



Functional Requirements

- Describes what a system is expected to do **(*Functionality*).**
- Describes the processes that system will carry out.
- Details of the inputs into the system from paper forms and documents and other systems.
- Details of the output expected from the system on screen display and as printouts on the paper.
- Describe the usability factors and facts between the system and users.
- Who can enter the data into the system?
- How the system meets applicable regulatory requirements

Non-Functional Requirements

Extensible- Ability of a software system (such as a database system) to allow and accept significant extension of its capabilities, without major rewriting of code or changes in its basic architecture. See also expand ability and scaling.

Maintainability- Characteristic of design and installation which determines the probability that a failed equipment, machine. or system can be restored to its normal operable state within a given time frame, using the prescribed practices and procedures.

Portability- Probability is estimated usually through repeated random sampling, and is represented numerically as between 0 (impossibility) and 1 (certainty).

Correctness- Correctness from software engineering perspective can be defined as the adherence to the specifications that determine how users can interact with the software and how the software should behave when it is used correctly.

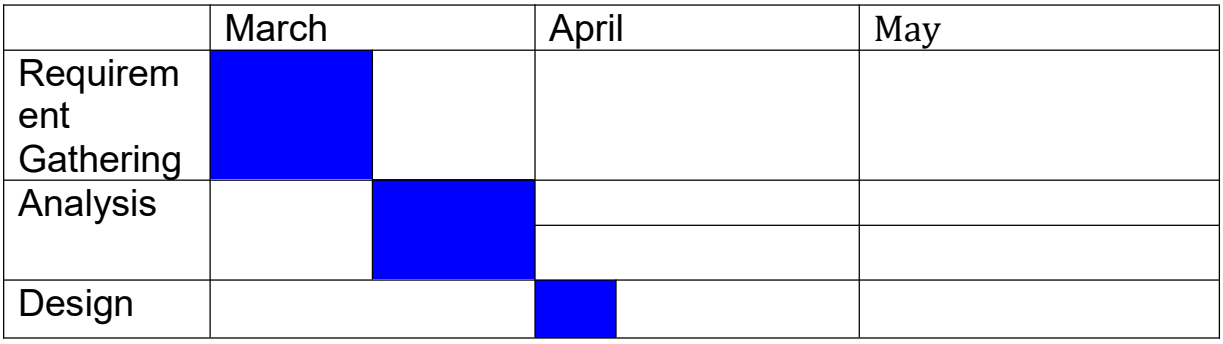
Use-friendly- Easy to understand and easy to maintenance.

SCHEDULING TECHNIQUES

The success of a project will depend critically upon the effort, care and skill you apply in its initial planning. A specification is the definition of your project: a statement of the problem, not the solution. Normally, the specification contains errors, ambiguities, misunderstandings and enough rope to hang you and your entire team. Thus before you embark upon the next six months of activity working on the wrong project, you must assume that a numbly was the chief author of the specification you received and you must read, worry, revise and ensure that everyone concerned with the project (from originator, through the workers, to the end-customer) is working with the same understanding. The outcome of this deliberation should be a written definition of what is required, by when; and this must be agreed by all involved. There are no short-cuts to this; if you fail to spend the time initially, it will cost you far more later on. The discipline for stating how to complete a project within a certain time frame, usually with define stages and with designated resources is known as Project Scheduling.

Two types of scheduling technique-

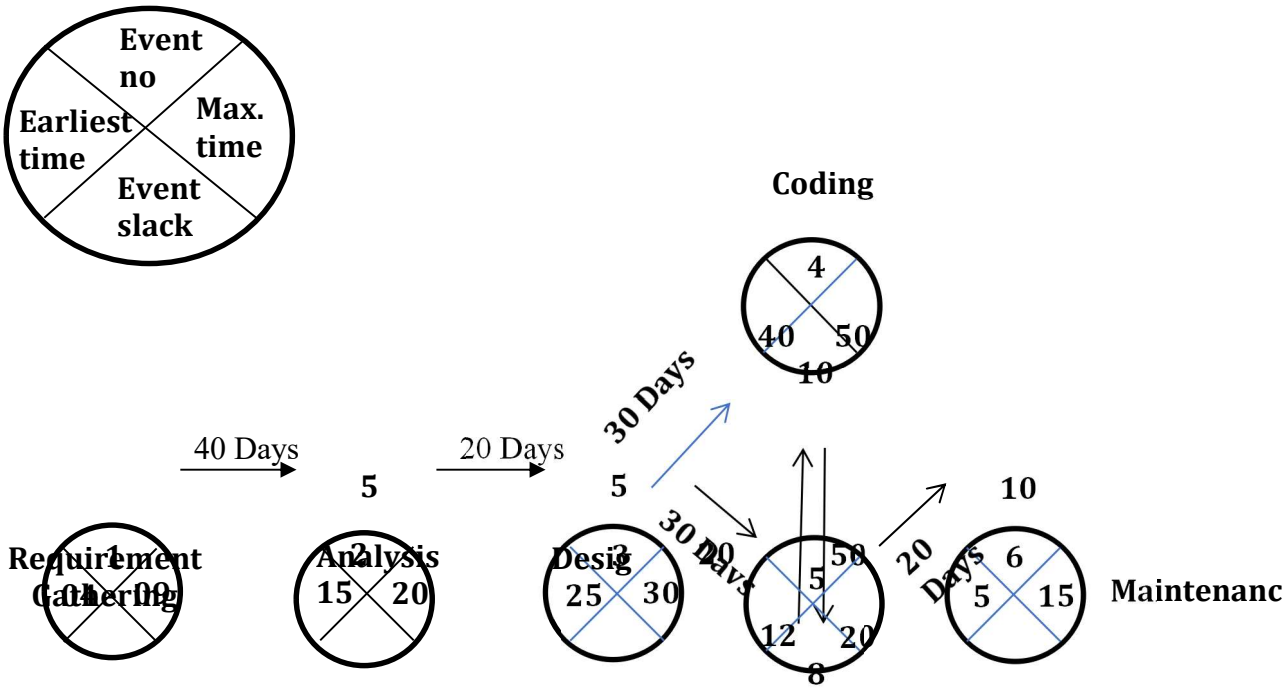
Gantt Chart:- A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity.



Coding										
Testing										
Implement										
	W 1	W 2	W 3	W 4	W 1	W 2	W 3	W 4	W 1	

We are weeks of the months, for i =1, 2, 3, 4

PERT CHART :- A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project.PERT stands for Program Evaluation Review Technique, a methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program. A similar methodology, the Critical Path Method (CPM) was developed for project management in the private sector at about the same time.



SYSTEM DESIGN

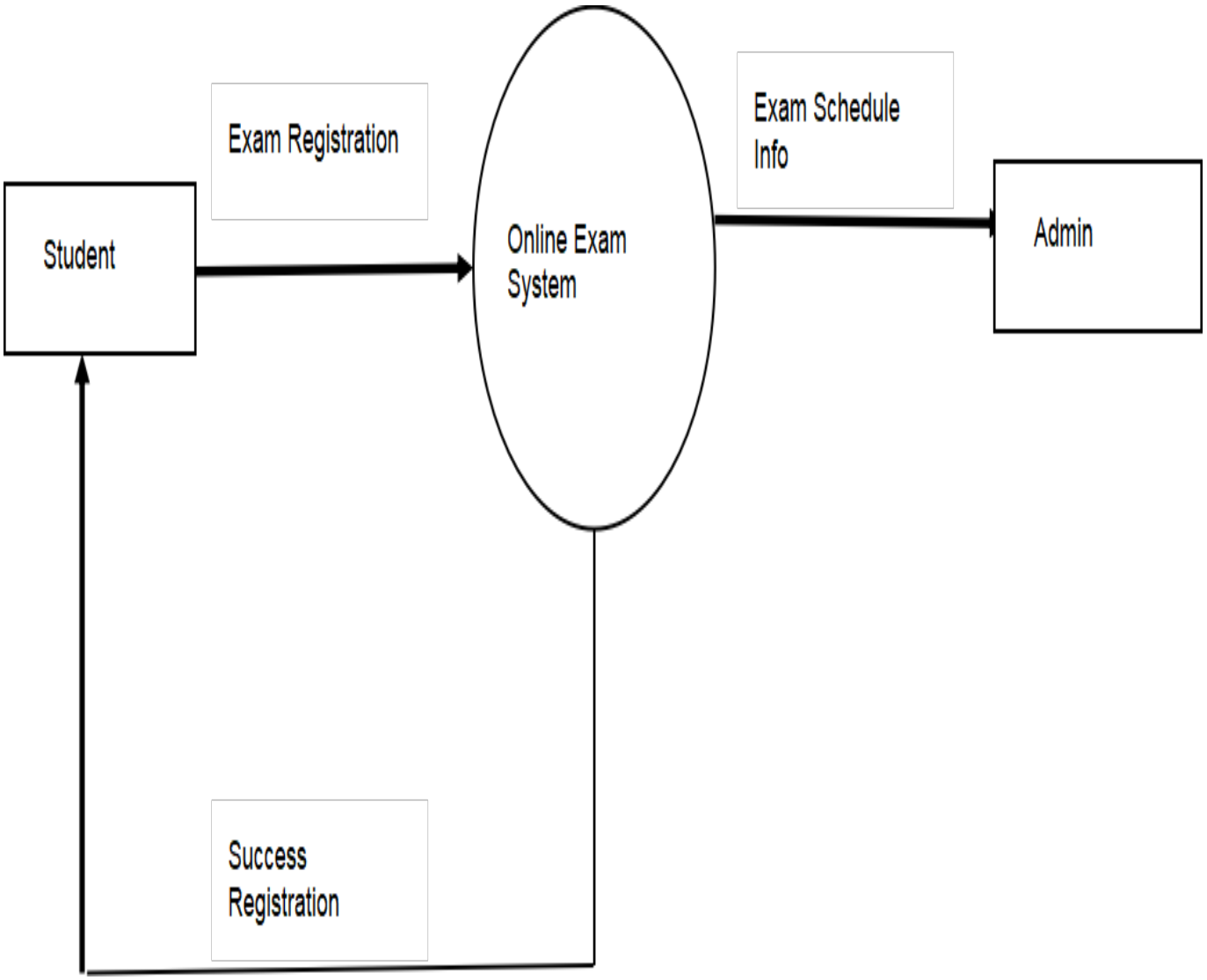
System design is to deliver the requirements as specified in the feasibility report.

The main objectives of the design are: E-RD

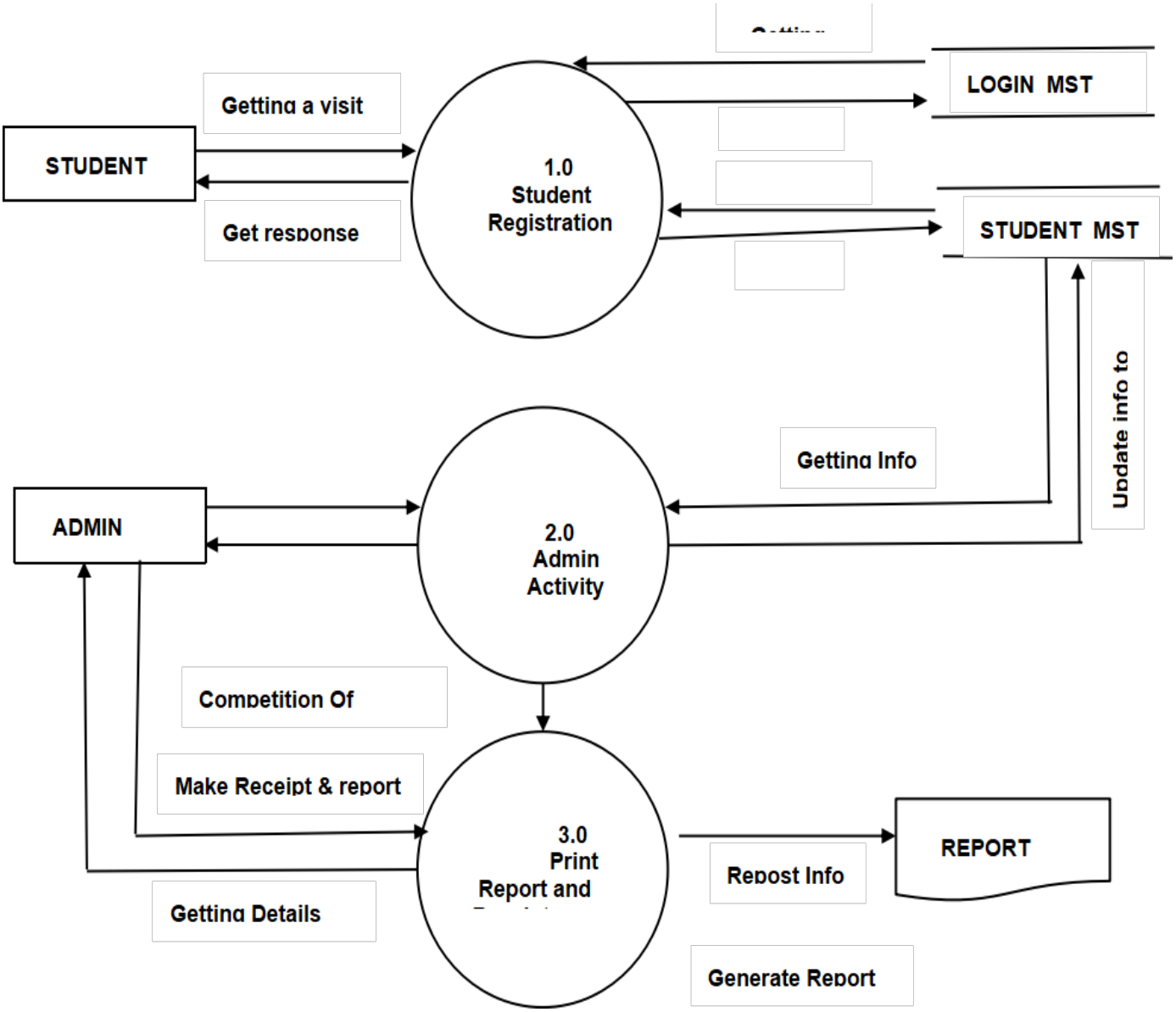
- Modules
- DFD
- ER-D
- Class diagram
- Use case diagram
- Data dictionary

Data Flow Diagram (DFD)

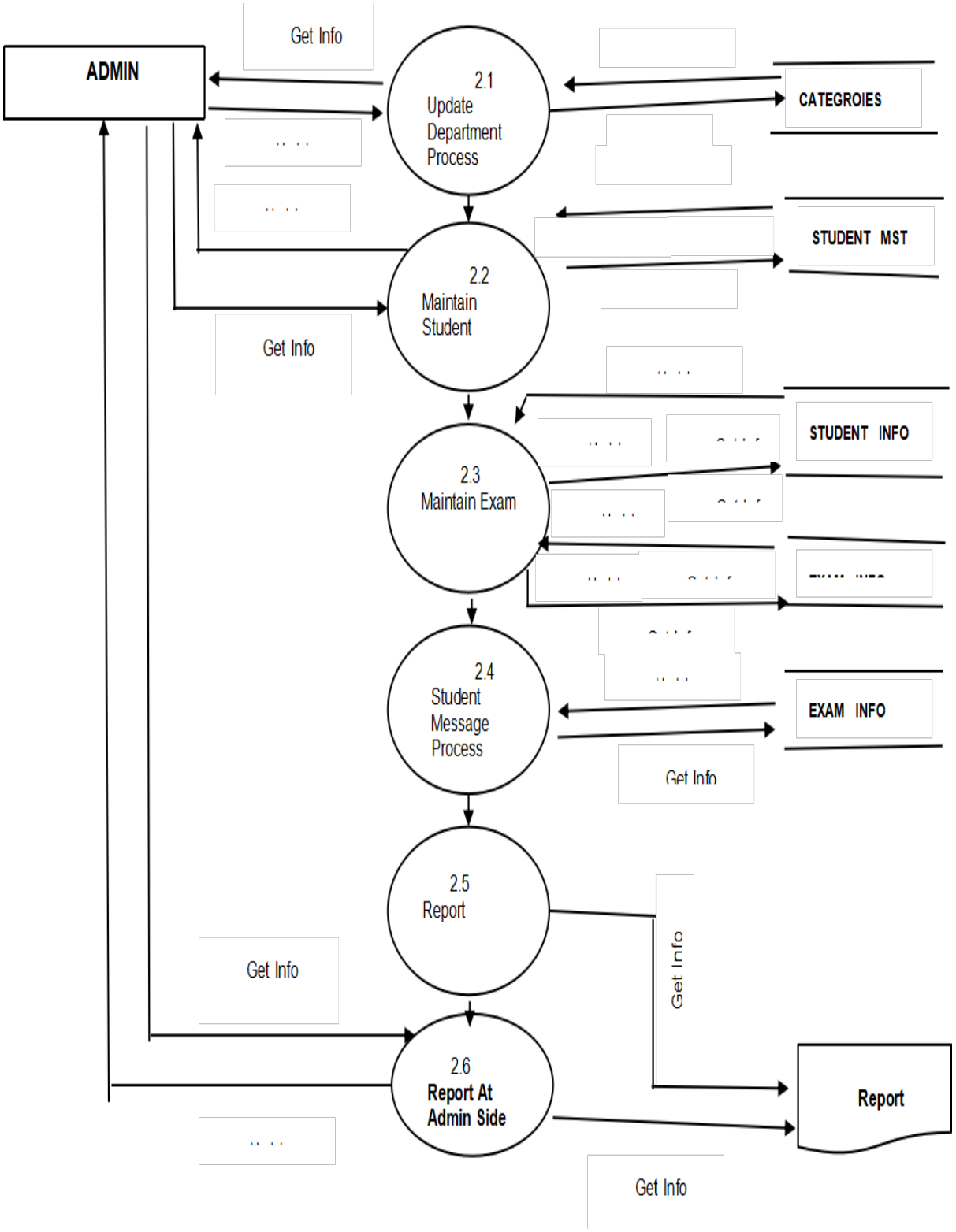
0 LEVEL DFD:-



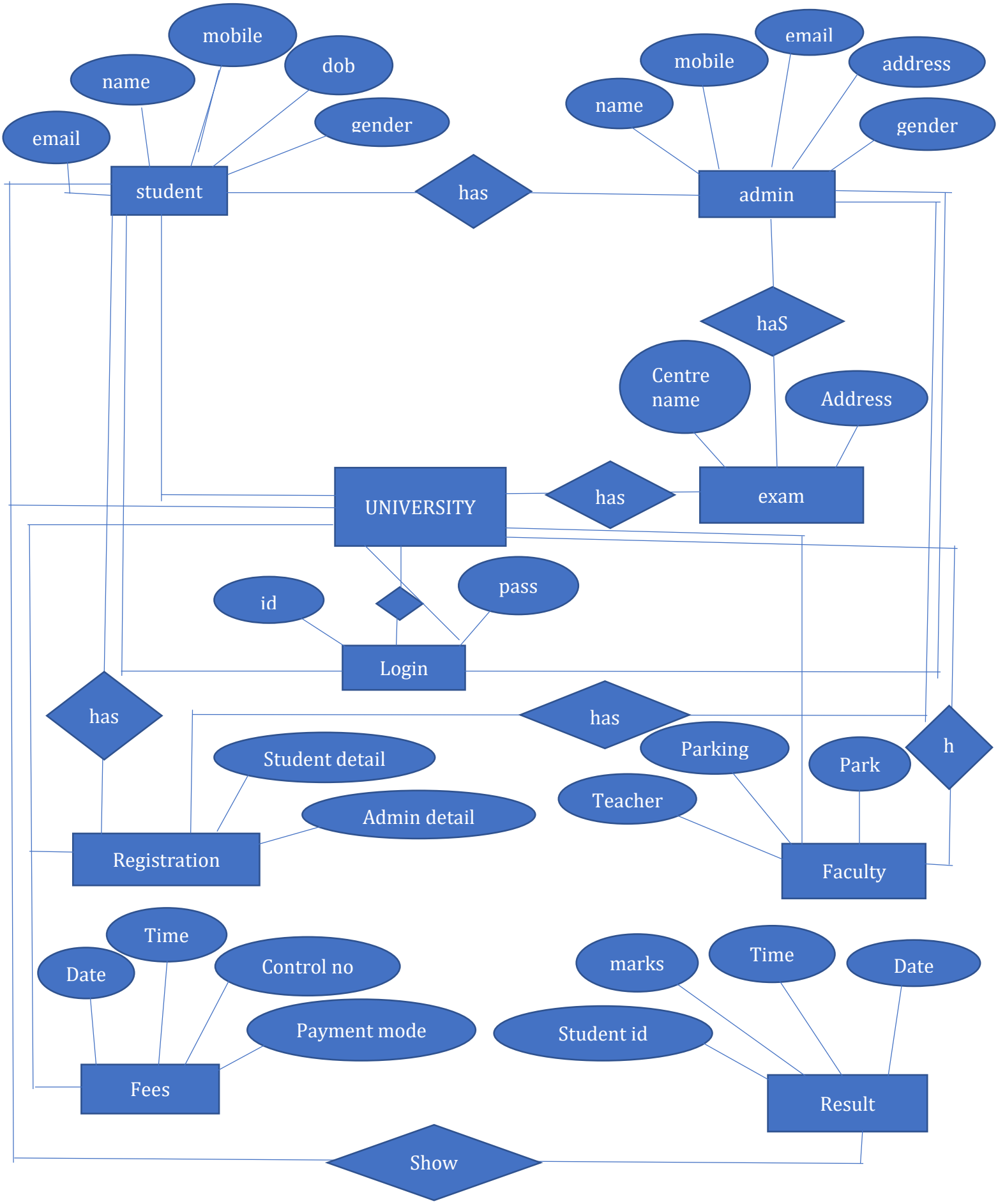
1st Level DFD:-



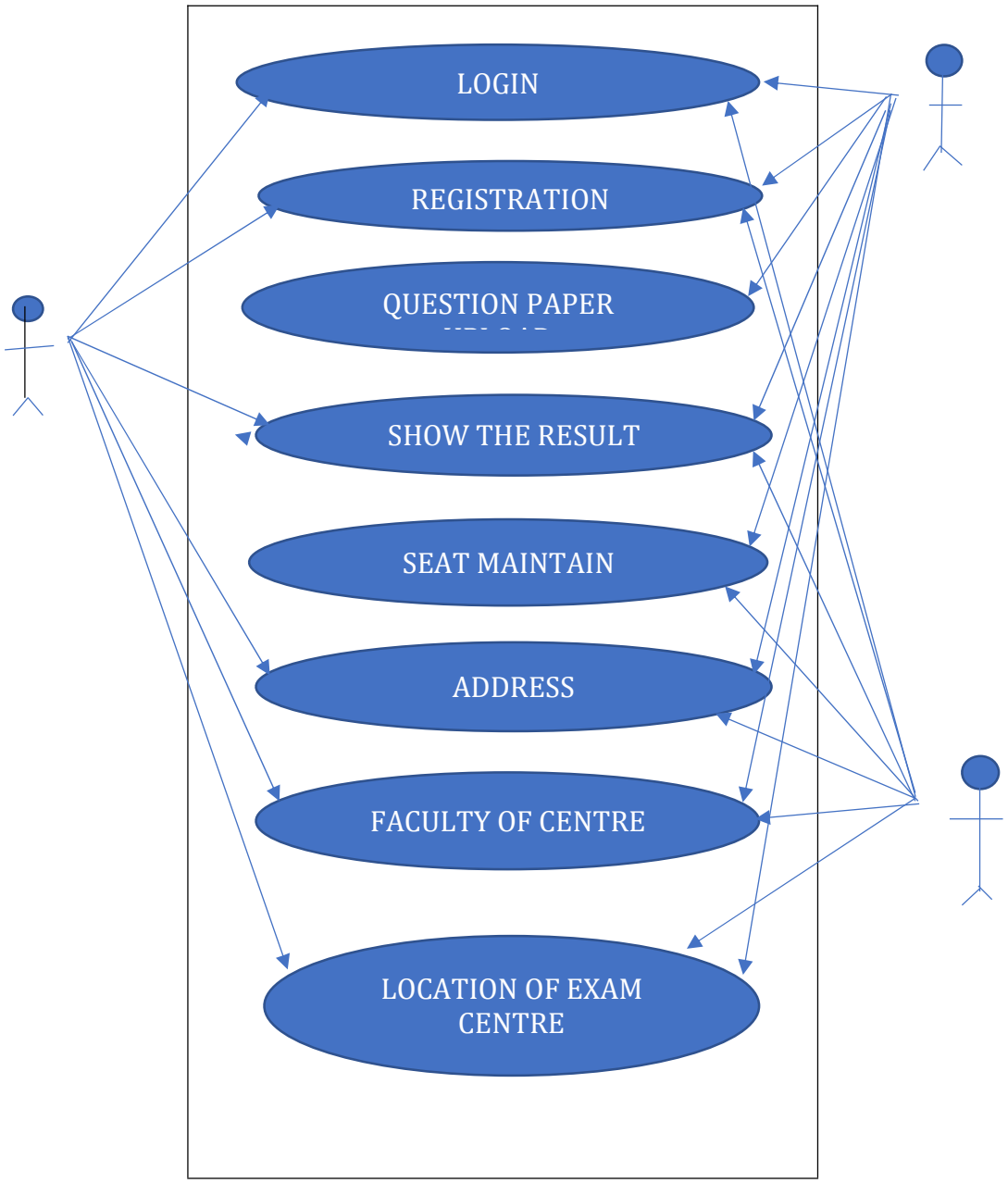
2nd LEVEL DFD



ER-Diagram:-


mobile
dob
name
email
gender
student
has
email
mobile
address
name
gender
admin
haS
Centre name
Address
UNIVERSITY
has
exam
id
pass
Login
has
has
Parking
Park
h
Student detail
Teacher
Admin detail
Registration
Faculty
Time
Date
Control no
Payment mode
Fees
marks
Time
Date
Student id
Result
Show

USE CASE DIAGRAM



Modules

- **Admin Login module**
- **Student Login module**
- **Admin Registration module**
- **Student Registration module**
- **Exam centre module**
- **Programme module**
- **Result module**
- **Fee module**

DATA DICTIONARY

Admin Login Page :-

Field name	Data type	Description
a_email	varchar	Primary key
a_password	varchar	Pass

Student Login Page :-

Field name	Data type	Description
s_id	int	Primary key
s_password	varchar	Pass

Admin registration :-

Field name	Data type	Description
arid	int	Id of admin
arname	varchar	Name of admin
aremail	varchar	Email of admin
arpassword	varchar	Password of admin
arcpassword	varchar	Confirm Password of admin
armobile	varchar	Mobile of admin

Student registration :-

Field name	Data type	Description
srid	int	Id of student
srname	varchar	Name of student
srenroll	int	Enroll of student
srprogram	varchar	Program of student
sremail	varchar	Email of student
srpassword	varchar	Password of student
srcpassword	varchar	Confirm Password of student

Exam centre :-

Field name	Data type	Description
S.No	int	Serial Number
Reginal centre	varchar	Reginal centre
Exam centre	varchar	Exam centre
Address	varchar	Address

Programme :-

Field name	Data type	Description
Program Name	varchar	Program Name

Result :-

Field name	Data type	Description
rsid	int	Student id
rsenrollment	int	Student Enrollment
rscourse	varchar	Student Course
rsmarks	int	Student marks
rsmaxmarks	int	Student max marks
rsmonthyear	date	Student month and year
rsremark	varchar	Student remarks

Fees :-

Field name	Data type	Description
programname	varchar	Name
programcode	int	Code
programfees	varchar	Amount of fee

Coding

AdminLoginIndex.cshtml:-

```
@{
    ViewBag.Title = "AdminLoginIndex";
}

<table style="background-color:blue; color:white; text-align:center" width="20.5%">
    <tr>
        <td colspan="2">
            Admin Login Page
        </td>
    </tr>
</table>

<script src="~/jquery.min.js"></script>
<script src="~/AdminLogin.js"></script>
<script src="~/AdminLoginValidation.js"></script>

<table style="background-color:maroon; color:white">
    <tr>
        <td>Email :</td>
        <td>
            <input type="text" id="txtemail" placeholder="Email" />
        </td>
    <tr>
    <tr>
        <td>Password :</td>
        <td>
            <input type="password" id="txtpassword" placeholder="Password" />
        </td>
    </tr>
    <tr>
        <td></td>
```

```
        <td>
            <input type="button" id="btnadminlogin" value="AdminLogin" onclick="return
AdminLoginValid()" />
            <!-- <a href="../Reg/RegIndex"><input type="button" id="btnsignup"
value="SignUp" onclick="Signup()" /></a> -->
        </td>
    </tr>
</table>
```

AdminLoginController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class AdminLoginController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult AdminLoginIndex()
        {
            return View();
        }
        public JsonResult AdminLoginUser(tblAregistration obj)
        {
            var data = (from x in db.tblAregistrations where x.aremail == obj.aremail
&& x.arpassword == obj.arpassword select x).ToList();
            return Json(data, JsonRequestBehavior.AllowGet);
        }
    }
}
```

AdminLogin.js

```
function AdminLoginData() {
    $.ajax({
        url: '../AdminLogin/AdminLoginUser',
        data: { aremail: $("#txtemail").val(), arpassword: $("#txtpassword").val() },
        success: function (data) {
            if (data.length > 0) {
                window.location.href = "../ShowAdminLoginData/ShowAdminLoginIndex?QS="
+ data[0].arid;
            }
            else {
                alert("Admin detail wrong");
            }
            Clear();
        },
        error: function () {
```

```
            alert("Admin  fail!");
        }
    });
}


function Clear() {
    $("#txtemail").val("");
    $("#txtpassword").val("");
    $("#btnadminlogin").val("AdminLogin");
}
```

ShowAdminLoginIndex.cshtml

```
@{
    ViewBag.Title = "ShowAdminLoginIndex";
}

<script src="~/jquery.min.js"></script>

<table id="tbl" border="1" style="background-color:yellow;color:red">
    <tr style="background-color:red;color:white">
        <th>Admin ID</th>
        <th>Admin Name</th>
        <th>Admin Email</th>
        <th>Admin Password</th>
        <th>Admin Confirm_Password</th>
        <th>Admin Mobile</th>
    </tr>
</table>

<script type="text/javascript">

    $(document).ready(function () {
        BindSingleUser();
    });

    function BindSingleUser() {
        $.ajax({
            url: '../ShowAdminLoginData/GetDataById',
            data: {A: @ViewBag.data},
            success: function (data) {
                $("#tbl").append('<tr> <td>' + data[0].arid + '</td> <td>' +
data[0].arname + '</td> <td>' + data[0].aremail + ' </td> <td>' + data[0].arpassword +
' </td> <td>' + data[0].arcpassword + ' </td> <td>' + data[0].armobile + '</td> < tr
>');
            },
            error: function () {
                alert("Data not found by ID!");
            }
        });
    }

</script>
```

ShowAdminLoginDataController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class ShowAdminLoginDataController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult ShowAdminLoginIndex(string QS)
        {
            ViewBag.data = QS;
            return View();
        }
        public JsonResult GetDataById(int A)
        {
            var data = (from x in db.tblAregistrations where x.arid == A select
x).ToList();
            return Json(data, JsonRequestBehavior.AllowGet);
        }
    }
}
```

StudentLoginIndex.cshtml

```
@{
    ViewBag.Title = "StudentLoginIndex";
}

    <table style="background-color:blue; color:white; text-align:center"
width="20.5%">
        <tr>
            <td colspan="2">
                Student Login Page
            </td>
        </tr>
    </table>

<script src="~/jquery.min.js"></script>
<script src="~/StudentLogin.js"></script>
<script src="~/StudentLoginValidation.js"></script>


<table style="background-color:maroon; color:white">
    <tr>
        <td>Enroll :</td>
        <td>
            <input type="text" id="txtenroll" placeholder="Enrollment" />
        </td>
    <tr>
    <tr>
        <td>Password :</td>
        <td>
            <input type="password" id="txtpassword" placeholder="Password" />
        </td>
```

```
    </tr>
    <tr>
        <td></td>
        <td>
            <input type="button" id="btnstudentlogin" value="StudentLogin"
onclick="return StudentLoginValid()" />
            <!-- <a href="../Reg/RegIndex"><input type="button" id="btnsignup"
value="SignUp" onclick="Signup()" /></a> -->
        </td>
    </tr>
</table>
```

StudentLoginController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class StudentLoginController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult StudentLoginIndex()
        {
            return View();
        }
        public JsonResult StudentLoginUser(tblSregistration obj)
        {
            var data = (from x in db.tblSregistrations where x.srenroll ==
obj.srenroll && x.srpassword == obj.srpassword select x).ToList();
            return Json(data, JsonRequestBehavior.AllowGet);
        }
    }
}
```

StudentLogin.js

```
function StudentLogin() {
    $.ajax({
        url: '../StudentLogin/StudentLoginUser',
        data: { srenroll: $("#txtenroll").val(), srpassword: $("#txtpassword").val() },
        success: function (data) {
            if (data.length > 0) {
                window.location.href =
"../ShowStudentLoginData/ShowStudentLoginDataIndex?QS=" + data[0].srid;
            }
            else {
                alert("Student detail wrong");
            }
            Clear();
        },
        error: function () {
```

```
            alert("Login  fail!");
        }
    });
}


function Clear() {
    $("#txtenroll").val("");
    $("#txtpassword").val("");
    $("#btnstudentlogin").val("StudentLogin");
}
```

ShowStudentLoginDataIndex.cshtml

```
@{
    ViewBag.Title = "ShowStudentLoginDataIndex";
}
    <script src="~/jquery.min.js"></script>

<table id="tbl" border="1" style="background-color:yellow;color:red">
    <tr style="background-color:red;color:white">
        <th>Student ID</th>
        <th>Student Name</th>
        <th>Student Enroll</th>
        <th>Student Program</th>
        <th>Student Degree</th>
        <th>Student Email</th>
        <th>Student Password</th>
        <th>Student Confirm_Password</th>
    </tr>
</table>


<script type="text/javascript">

    $(document).ready(function () {
        BindSingleUser();
    });

    function BindSingleUser() {
        $.ajax({
            url: '../ShowStudentLoginData/GetDataById',
            data: {A: @ViewBag.data},
            success: function (data) {
                $("#tbl").append('<tr> <td>' + data[0].srid + '</td> <td>' +
data[0].srname + '</td> <td>' + data[0].srenroll + '</td> <td>' + data[0].srprogram +
'</td> <td>' + data[0].srdegree + '</td> <td>' + data[0].sremail + ' </td> <td>' +
data[0].srpassword + ' </td> <td>' + data[0].srcpassword + ' </td> <tr >');
            },
            error: function () {
                alert("Data not found by ID!");
            }
        });
    }

</script>
```

ShowStudentLoginDataController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class ShowStudentLoginDataController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult ShowStudentLoginDataIndex(string QS)
        {
            ViewBag.data = QS;
            return View();
        }
        public JsonResult GetDataById(int A)
        {
            var data = (from x in db.tblSregistrations where x.srid == A select
x).ToList();
            return Json(data, JsonRequestBehavior.AllowGet);
        }
    }
}
```

AregistrationIndex.cshtml

```
@{
    ViewBag.Title = "AregistrationIndex";
}

    <table style="background-color:blue; color:white; text-align:center"
width="29.2%">
        <tr>
            <td colspan="2">
                Admin Registration Page
            </td>
        </tr>
    </table>

<script src="~/jquery.min.js"></script>
<script src="~/AdminRegistration.js"></script>
<script src="~/AdminRegistrationValidation.js"></script>

<table style="background-color:maroon; color:white">
    <tr>
        <td>Admin Name:</td>
        <td><input type="text" id="txtname" /></td>
    </tr>
    <tr>
        <td>Admin Email :</td>
        <td>
```

```
            <input type="text" id="txtemail" />
        </td>
    </tr>
    <tr>
        <td>Admin Password :</td>
        <td>
            <input type="text" id="txtpwd" />
        </td>
    </tr>
    <tr>
        <td>Admin Confirm Password :</td>
        <td>
            <input type="password" id="txtconpwd" />
        </td>
    </tr>
    <tr>
        <td>Admin Mobile No:</td>
        <td><input type="text" id="txtmobileno" /></td>
    </tr>
    <tr>
        <td></td>
        <td><input type="button" id="btnaregistration" value="AdminRegister"
onclick="return AdminRagisterdValid()" /></td>
    </tr>
</table>
```

AregistrationController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class AregistrationController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult AregistrationIndex()
        {
            return View();
        }
        public void Insert(tblAregistration obj)
        {
            db.tblAregistrations.Add(obj);
            db.SaveChanges();
        }
    }
}
```

Aregistration.js

```
function Clear() {
    $("#txtname").val("");
    $("#txtemail").val("");
    $("#txtpwd").val("");
    $("#txtconpwd").val("");
    $("#txtmobileno").val("");
    $("#btnaregistration").val("AdminRegister");
}

function AdminRagisterdData() {
    $.ajax({
        url: '../Aregistration/Insert',
        data: { arname: $("#txtname").val(), aremail: $("#txtemail").val(), arpassword:
$("#txtpwd").val(), arcpassword: $("#txtconpwd").val(), armobile:
$("#txtmobileno").val() },
        success: function () {
            alert("AdminRegistration is successfull !");
            Clear();
        },
        error: function () {
            alert("AdminRegistration fail!!");
        }
    });
}
```

SregistrationIndex.cshtml

```
@{
    ViewBag.Title = "SregistrationIndex";
}

<table style="background-color:blue; color:white; text-align:center" width="29.5%">
    <tr>
        <td colspan="2">
            Student Registration Page
        </td>
    </tr>
</table>

<script src="~/jquery.min.js"></script>
<script src="~/StudentRegistration.js"></script>
<script src="~/StudentRegistrationValidation.js"></script>

<table style="background-color:maroon; color:white">
    <tr>
        <td>Student Name:</td>
        <td><input type="text" id="txtname" /></td>
    </tr>
    <tr>
        <td>Student Enroll No:</td>
        <td><input type="text" id="txtenrollno" /></td>
```

```
    </tr>
    <tr>
        <td>Student Program:</td>
        <td><input type="text" id="txtprogram" /></td>
    </tr>
    <tr>
        <td>Student Email :</td>
        <td>
            <input type="text" id="txtemail" />
        </td>
    </tr>
    <tr>
        <td>Student Password :</td>
        <td>
            <input type="text" id="txtpwd" />
        </td>
    </tr>
    <tr>
        <td>Student Confirm Password :</td>
        <td>
            <input type="password" id="txtconpwd" />
        </td>
    </tr>
    <tr>
        <td></td>
        <td><input type="button" id="btnsregistration" value="StudentRegister"
onclick="return StudentRagisterdValid()" /></td>
    </tr>
</table>
```

SregistrationController .cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class SregistrationController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult SregistrationIndex()
        {
            return View();
        }
        public void Insert(tblSregistration obj)
        {
            db.tblSregistrations.Add(obj);
            db.SaveChanges();
        }
    }
}
```

Sregistration.js

```
function Clear() {
    $("#txtname").val("");
    $("#txtenrollno").val("");
    $("#txtprogram").val("");
    $("#txtemail").val("");
    $("#txtpwd").val("");
    $("#txtconpwd").val("");
    $("#btnsregistration").val("StudentRegister");
}

function StudentRagisterdData() {
    $.ajax({
        url: '../Sregistration/Insert',
        data: { srname: $("#txtname").val(), srenroll: $("#txtenrollno").val(),
srprogram: $("#txtprogram").val(), sremail: $("#txtemail").val(), srpassword:
$("#txtpwd").val(), srcpassword: $("#txtconpwd").val() },
        success: function () {
            alert("StudentRegistration is successfull !");
            Clear();
        },
        error: function () {
            alert("StudentRegistration fail!!");
        }
    });
}
```

ExamCentreIndex.cshtml

```
@{
    ViewBag.Title = "ExamCentreIndex";
}

<table style="background-color:blue; color:white" border="1" width="100%">
    <tr>
        <th width="1%">S.No</th>
        <th width="10%">Regional Centre</th>
        <th width="14%">Exam Centre</th>
        <th>Address</th>
    </tr>
</table>
<br />
<table border="1" style="background-color:maroon; color:white">
    <tr>
        <td>1</td>
        <td>HYDERBAD</td>
        <td>0105 WARANGAL</td>
        <td>
            LAL BHADUR COLLAGE
            SARDAR PATEL ROAD
        </td>
    </tr>
    <tr>
        <td>2</td>
        <td>GUWAHTI</td>
```

```
    <td>
        0401
        JALUKBARI, GUWAHATI
    </td>
    <td>
        MAIN ARTS & LAW BUILDING
        GUWAHATI UNIVERSITY CAMPUS
        JALUKBARI
        GUWAHATI
        ASSAM - 781014
    </td>
</tr>
<tr>
    <td>3</td>
    <td>PATNA</td>
    <td>
        0511
        GAYA
    </td>
    <td>
        IGNOU STUDY CENTRE
        MANVIKI BHAWAN
        GAYA COLLEGE
        GAYA
        - 823001
    </td>
</tr>
<tr>
    <td>4</td>
    <td>CHANDIGARH</td>
    <td>
        2203
        PATIALA
    </td>
    <td>
        DEPT. OF DISTANCE EDUCATION
        PUNJABI UNIVERSITY
        PATIALA
        PUNJAB
        - 147002
    </td>
<tr>
    <td>5</td>
    <td>DELHI-I</td>
    <td>
        0765
        JAMIA NAGAR
    </td>
    <td>
        JAMIA MILLIA ISLAMIA
        JAMIA NAGAR
        NEW DELHI
        - 110025
    </td>
<tr>
    <td>6</td>
    <td>AHMEDABAD</td>
    <td>
        0905
        SURAT
    </td>
    <td>
```

```
        IGNOU STUDY CENTRE
        ROOM NO-4, M.T.B. ARTS COLLEGE
        ATHWA LINES, SURAT
    </td>
<tr>
    <td>7</td>
    <td>KARNAL</td>
    <td>
        1002
        SONEPAT
    </td>
    <td>
        IGNOU STUDY CENTRE, ART BLOCK
        HINDU COLLEGE
        SONEPAT
        - 131001
    </td>
<tr>
    <td>8</td>
    <td>SHIMLA</td>
    <td>
        1107
        KULLU
    </td>
    <td>
        IGNOU STUDY CENTRE
        GOVT. COLLEGE
        KULLU
        HIMACHAL PRADESH - 175101
    </td>
<tr>
    <td>9</td>
    <td>JAMMU</td>
    <td>
        1232
        JAMMU
    </td>
    <td>
        RECREATION HALL,
        GOVT. MAM COLLEGE,
        BABA SAHEB AMBEDKAR ROAD
        JAMMU
        J&K - 180006
    </td>
<tr>
    <td>10</td>
    <td>BANGALORE</td>
    <td>
        319
        TUMKUR
    </td>
    <td>
        SRI SIDDHARTHA INSTITUTE OF
        TECHNOLOGY,
        MARALURU, KUNIGAL ROAD
        TUMAKURU
        KARANATAKA - 572105
    </td>
<tr>
    <td>11</td>
    <td>BHOPAL</td>
    <td>
```

```
        15113D
        INDORE
    </td>
    <td>
        MOOK BADHIR SANGATHAN
        SCHEME NO. 71-B
        BEHIND RANJEET, HANUMAN TEMPLE
        INDORE, MP
        - 452009
    </td>
<tr>
    <td>1</td>
    <td>HYDERBAD</td>
    <td>0105 WARANGAL</td>
    <td>
        LAL BHADUR COLLAGE
        SARDAR PATEL ROAD
    </td>
<tr>
    <td>12</td>
    <td>PUNE</td>
    <td>
        16145
        OSMANABAD
    </td>
    <td>
        IGNOU STUDY CENTRE
        TERNA MAHAVIDYALAYA ARTS,
        SCIENCE & COMMERCE) PLOT NO. 1
        MIDC AREA AURANGABAD ROAD
        OSMANABAD - 413501
    </td>
<tr>
    <td>13</td>
    <td>BHUBANESHWAR</td>
    <td>
        2108
        SAMBALPUR
    </td>
    <td>
        IGNOU SC
        G.M. UNIVERSITY
        DIST-SAMBALPUR
        AT. FATAK, PO-BUDHARAJA
        SAMBALPUR, ODISHA - 768004
    </td>
<tr>
    <td>14</td>
    <td>JAIPUR</td>
    <td>
        23012
        AJMER
    </td>
    <td>
        SHRI GOVIND SINGH
        GURJAR GOVT. COLLEGE,
        BEAWER ROAD, NASIRABAD,
        AJMER, RAJ.
    </td>
<tr>
    <td>15</td>
    <td>CHENNAI</td>
```

```
    <td>
        2578
        VELACHERY
    </td>
    <td>
        GURU NANAK COLLEGE
        VELACHERY
        CHENNAI
        - 600042
    </td>
</tr>
<tr>
    <td>16</td>
    <td>LUCKNOW</td>
    <td>
        2712
        JHANSI
    </td>
    <td>
        BIPIN BIHARI COLLEGE
        OUTSIDE SAINYAR GATE
        JHANSI
        UTTAR PRADESH - 284001
    </td>
</tr>
<tr>
    <td>17</td>
    <td>KOLKATA</td>
    <td>
        2863
        MEDINIPUR
    </td>
    <td>
        INDIAN CENTRE FOR ADVANCEMENT
        OF RESEARCH & EDUCATION, ICARE
        COMPLEX, HIT CAMPUS, HALDIA
        DIST. PURBA MEDINIPUR - 721657
    </td>
</tr>
<tr>
    <td>18</td>
    <td>DELHI-II</td>
    <td>
        0712
        DILSHAD GARDEN
    </td>
    <td>
        ARWACHIN INTERNATIONAL SCHOOL
        POCKET-B, OPP GURUDWARA
        DILSHAD GARDEN,
        DELHI - 110095
    </td>
</tr>
<tr>
    <td>19</td>
    <td>SRINAGAR</td>
    <td>
        1203
        LADAKH
    </td>
    <td>
        GOVT. BOYS HIGHER
        SECONDARY INSTITUTE
        LEH, LADAKH
        J&K - 194101
    </td>
```

```
    <tr>
        <td>20</td>
        <td>DEHRADUN</td>
        <td>
            2711
            HALDWANI,NAINITAL
        </td>
        <td>
            IGNOU STUDY CENTRE,
            M B PG COLLEGE, HALDWANI
            NAINITAL
            UTTARAKHAND - 263141
        </td>
    <tr>
</table>
```

ExamcentreController .cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace OEES.Controllers
{
    public class ExamcentreController : Controller
    {
        public ActionResult ExamCentreIndex()
        {
            return View();
        }
    }
}
```

ProgramIndex.cshtml

```
@{
    ViewBag.Title = "ProgramIndex";
}

    <table style="background-color:blue; color:white" width="36.8%">
        <tr>
            <th>
                Program Name
            </th>
        </tr>
    </table>

<table border="1" style="background-color:maroon; color:white">
    <tr>
        <td>
            Bachelor Of Arts
        </td>
```

```
</tr>
<tr>
    <td>
        Bachelor Of Computer Application
    </td>
</tr>
<tr>
    <td>
        Doctor Of Philosophy In Geography
    </td>
</tr>
<tr>
    <td>
        Ph D In Arabic
    </td>
</tr>
<tr>
    <td>
        Ma In Distance Education
    </td>
</tr>
<tr>
    <td>
        Master Of Arts In Journalism & Mass Communication
    </td>
</tr>
<tr>
    <td>
        Master Of Business Administration
    </td>
</tr>
<tr>
    <td>
        Master Of Commerce
    </td>
</tr>
<tr>
    <td>
        Certificate In Arabic Language
    </td>
</tr>
<tr>
    <td>
        Certificate In General Duty Assistance
    </td>
</tr>
<tr>
    <td>
        Certificate In General Duty Assistance
    </td>
</tr>
<tr>
    <td>
        Certificate In Guidance
    </td>
</tr>
<tr>
    <td>
        Certificate In Information Technology
    </td>
</tr>
<tr>
```

```
        <td>
            Awareness Course On Goods And Services Tax
        </td>
    </tr>
    <tr>
        <td>
            Post Graduate Certificate In Medical Management Of Cbrne Disasters
        </td>
    </tr>
</table>
```

ProgramController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace OEES.Controllers
{
    public class ProgramController : Controller
    {
        public ActionResult ProgramIndex()
        {
            return View();
        }
    }
}
```

ResultIndex.cshtml

```
@{
    ViewBag.Title = "ResultIndex";
}

<table style="background-color:blue; color:white" width="19%">
    <tr>
        <th>
            Result Page
        </th>
    </tr>
</table>

<script src="~/jquery.min.js"></script>
<script src="~/ResultData.js"></script>
<script src="~/ResultValidation.js"></script>

<table style="background-color:maroon; color:white">
    <tr>
        <td>Result :</td>
        <td>
            <input type="text" id="txtresult" placeholder="Result" />
```

```
        </td>
    <tr>
    <tr>
        <td></td>
        <td>
            <input type="button" id="btnresult" value="SUBMIT" onclick="return
ResultValid()" />
        </td>
    </tr>
</table>
```

ResultController .cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class ResultController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult ResultIndex()
        {
            return View();
        }
        public JsonResult ResultStudent(tblResult1 obj)
        {
            var data = (from x in db.tblResult1 where x.rsenrollmentno ==
obj.rsenrollmentno select x).ToList();
            return Json(data, JsonRequestBehavior.AllowGet);
        }
    }
}
```

Result.js

```
function ResultData() {
    $.ajax({
        url: '../Result/ResultStudent',
        data: { rsenrollmentno: $("#txtresult").val() },
        success: function (data) {
            if (data.length >0) {
                window.location.href = "../ShowResultData/ShowResultDataIndex?QS=" +
data[0].rsenrollmentno;
            }
            else {
                alert("Result enrollment no wrong");
            }
            Clear();
```

```
        },
        error: function () {
            alert("Result  fail!");
        }
    });
}
```

ShowResultDataIndex.cshtml

```
function Clear() {
    $("#txtresult").val("");
    $("#btnresult").val("SUBMIT");
}



@{
    ViewBag.Title = "ShowResultDataIndex";
}

    <script src="~/jquery.min.js"></script>

<table id="tbl" border="1" style="background-color:yellow;color:red">
    <tr style="background-color:red;color:white">
        <th>Student Enrollment</th>
        <th>Student Course</th>
        <th>Student Marks</th>
        <th>Student Maxmarks</th>
        <th>Student MonthYear</th>
        <th>Student Remark</th>
    </tr>
</table>

<script type="text/javascript">

    $(document).ready(function () {
        BindSingleUser();
    });

    function BindSingleUser() {
        $.ajax({
            url: '../ShowResultData/GetDataById',
            data: { A: @ViewBag.data},
            success: function (data) {
                for (var i = 0; i < data.length; i++) {
                    $("#tbl").append('<tr> <td>' + data[i].rsenrollmentno + '</td>
<td>' + data[i].rscourse + '</td> <td>' + data[i].rsmarks + '</td> </td> <td>' +
data[i].rsmaxmark + '</td> <td>' + data[i].rsmonthyear + '</td> <td>' +
data[i].rsremark + ' </td> < tr >');
                }
            },
            error: function () {
                alert("Data not found by ID!");
            }
        });
    }

</script>
```

ShowResultDataController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class ShowResultDataController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult ShowResultDataIndex(string QS)
        {
            ViewBag.data = QS;
            return View();
        }
        public JsonResult GetDataById(int A)
        {
            var data = (from x in db.tblResult1 where x.rsenrollmentno == A select
x).ToList();
            return Json(data, JsonRequestBehavior.AllowGet);
        }
    }
}
```

FeesIndex.cshtml

```
@{
    ViewBag.Title = "FeesIndex";
}
    <table style="background-color:blue; color:white" width="47.2%">
        <tr>
            <th>
                Program Name
            </th>
            <th>
                Program Code
            </th>
            <th>
                Program Fees
            </th>
        </tr>
    </table>

<table border="1" style="background-color:maroon; color:white">
    <tr>
        <td>
            Bachelor Of Arts
        </td>
        <td>
            BA
```

```
    </td>
    <td>
        6000
    </td>
</tr>
<tr>
    <td>
        Bachelor Of Computer Application
    </td>
    <td>
        BCA
    </td>
    <td>
        40200
    </td>
</tr>
<tr>
    <td>
        Doctor Of Philosophy In Geography
    </td>
    <td>
        PHDGEOG
    </td>
    <td>
        16800
    </td>
</tr>
<tr>
    <td>
        Ph D In Arabic
    </td>
    <td>
        PhD (A)
    </td>
    <td>
        5000
    </td>
</tr>
<tr>
    <td>
        Ma In Distance Education
    </td>
    <td>
        MADE
    </td>
    <td>
        10800
    </td>
</tr>
<tr>
    <td>
        Master Of Arts In Journalism & Mass Communication
    </td>
    <td>
        MAJMC
    </td>
    <td>
        25000
    </td>
</tr>
<tr>
    <td>
```

```
            Master Of Business Administration
        </td>
        <td>
            MBA
        </td>
        <td>
            37800
        </td>
    </tr>
    <tr>
        <td>
            Master Of Commerce
        </td>
        <td>
            MCom
        </td>
        <td>
            11000
        </td>
    </tr>
    <tr>
        <td>
            Certificate In Arabic Language
        </td>
        <td>
            CAL
        </td>
        <td>
            1800
        </td>
    </tr>
    <tr>
        <td>
            Certificate In General Duty Assistance
        </td>
        <td>
            CGDA
        </td>
        <td>
            6000
        </td>
    </tr>
    <tr>
        <td>
            Certificate In General Duty Assistance
        </td>
        <td>
            CGDA
        </td>
        <td>
            6000
        </td>
    </tr>
    <tr>
        <td>
            Certificate In Guidance
        </td>
        <td>
            CIG
        </td>
        <td>
            1400
```

```
        </td>
    </tr>
    <tr>
        <td>
            Certificate In Information Technology
        </td>
        <td>
            CIT
        </td>
        <td>
            6000
        </td>
    </tr>
    <tr>
        <td>
            Awareness Course On Goods And Services Tax
        </td>
        <td>
            GST
        </td>
        <td>
            3500
        </td>
    </tr>
    <tr>
        <td>
            Post Graduate Certificate In Medical Management Of Cbrne Disasters
        </td>
        <td>
            PGCMDM
        </td>
        <td>
            5500
        </td>
    </tr>
</table>
```

FeesController.cs

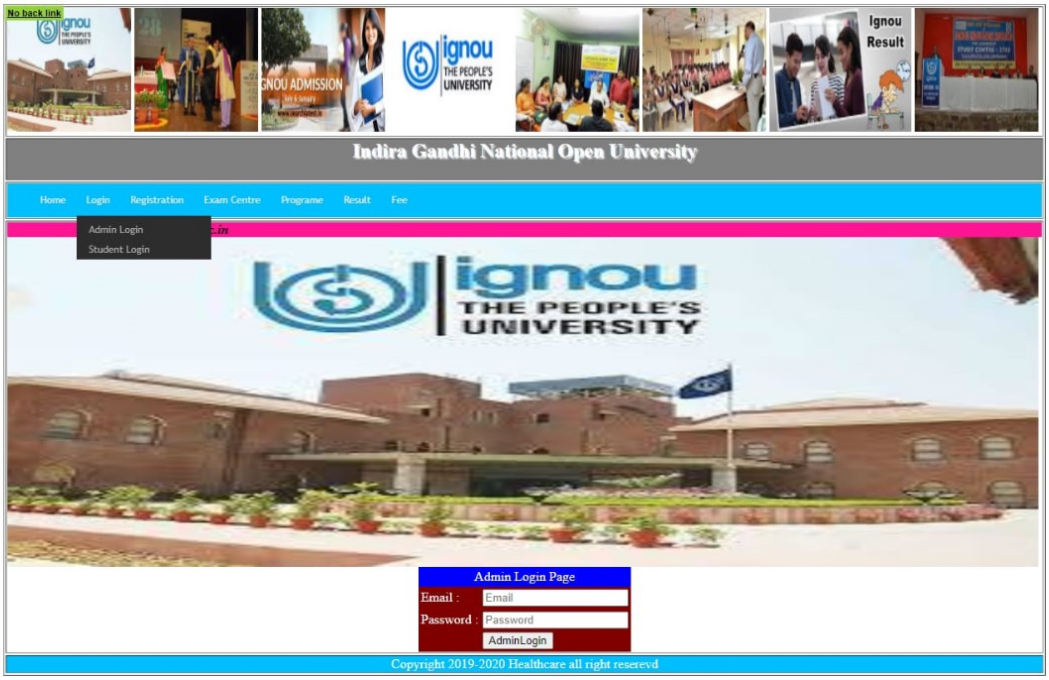
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using OEES.Models;

namespace OEES.Controllers
{
    public class FeesController : Controller
    {
        OEESEntities db = new OEESEntities();
        public ActionResult FeesIndex()
        {
            return View();
        }
    }
}
```

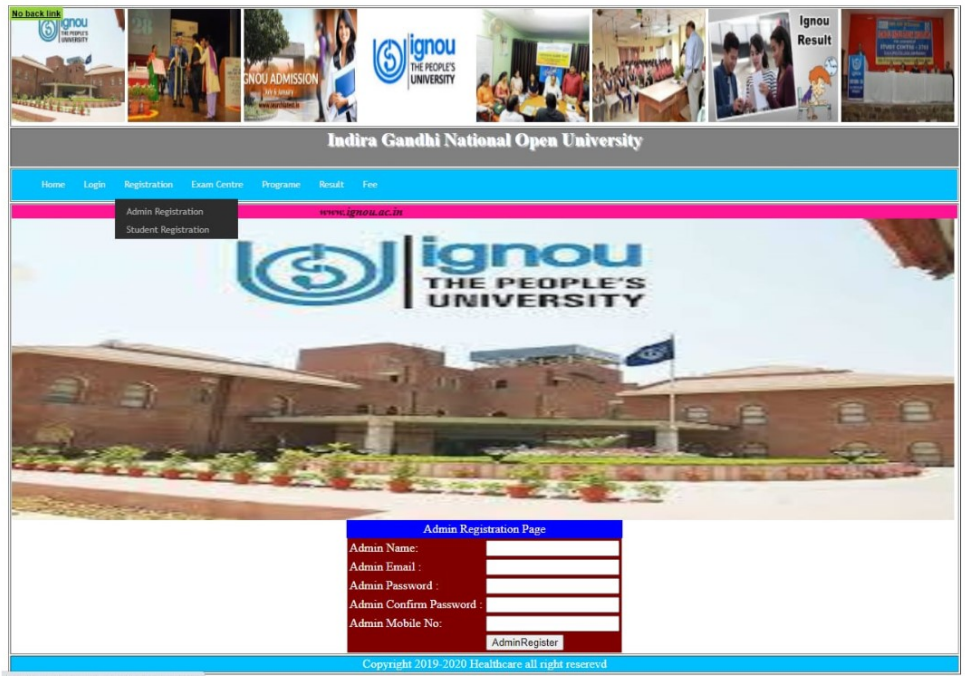
OUTPUT

HomePage:

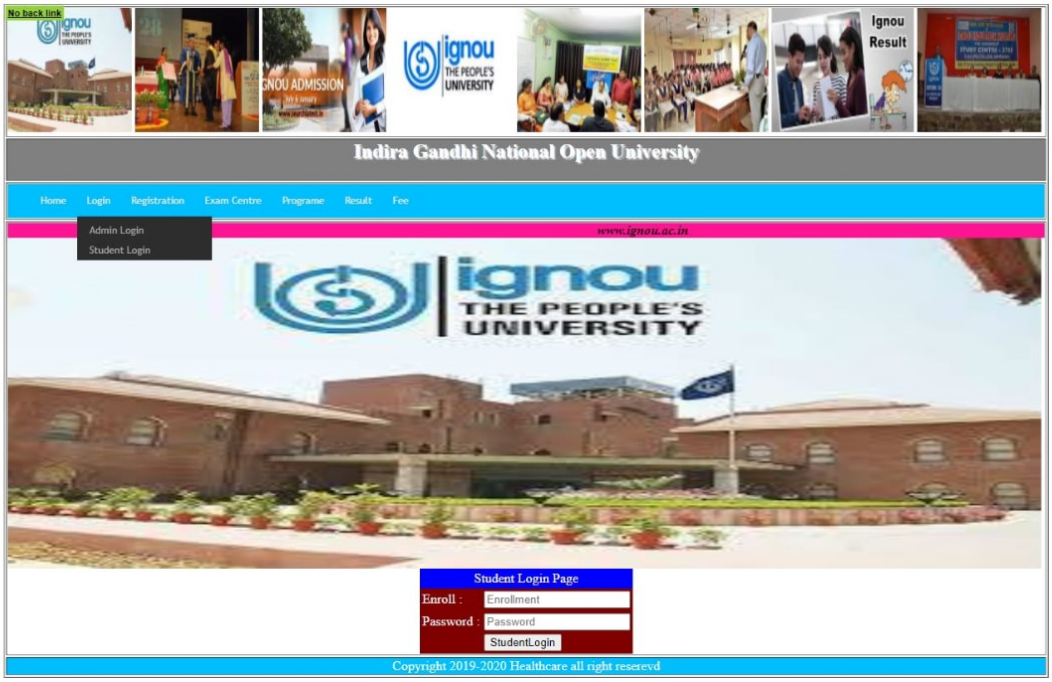
AdminLoginPage:-



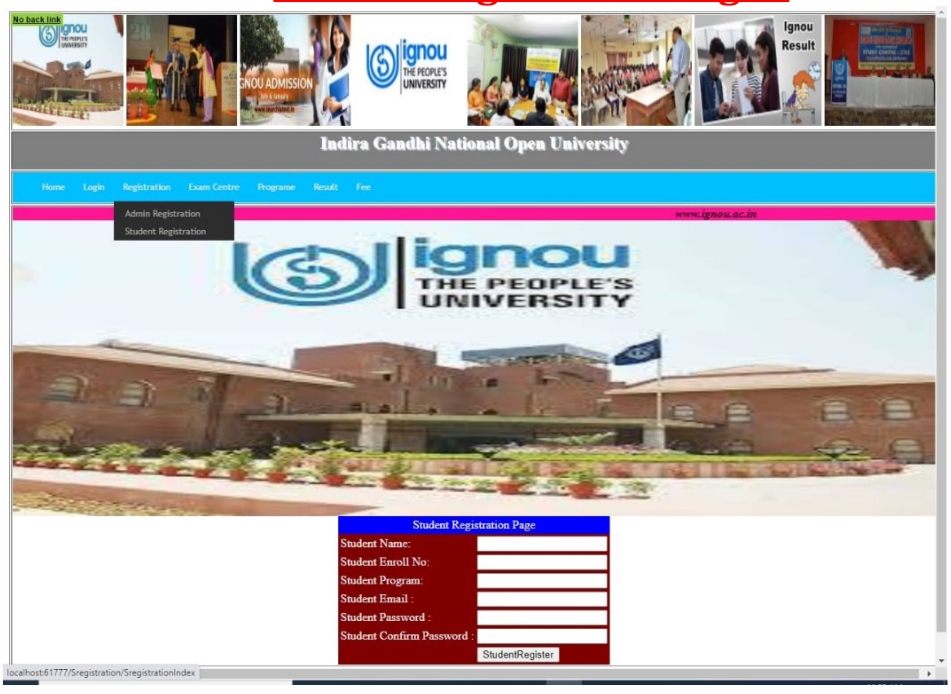
AdminRegistrationPage:-



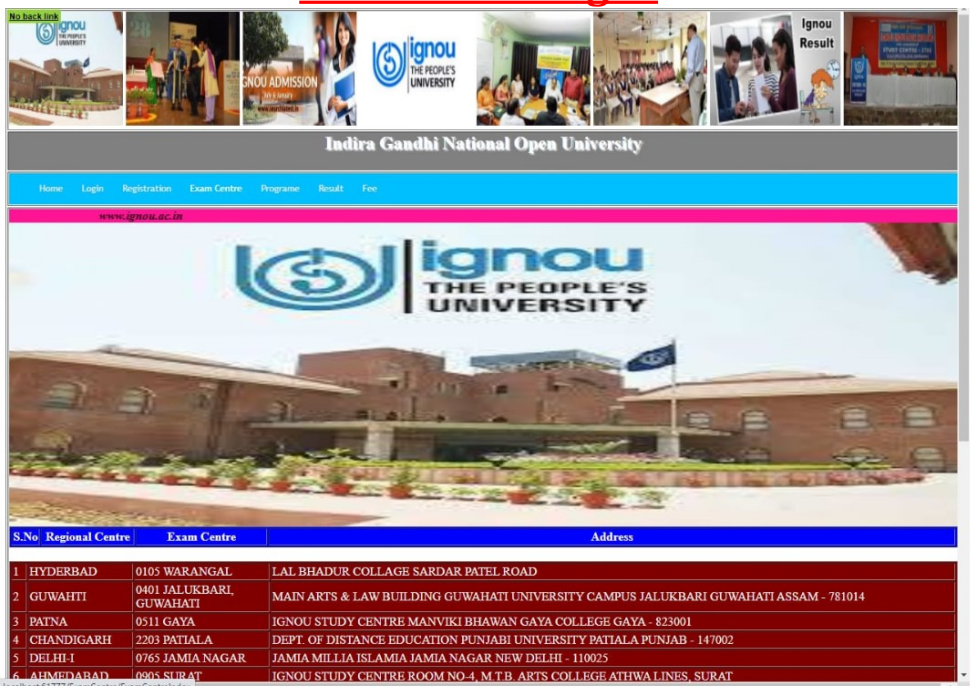
StudentLoginPage:-



StudentRegistrationPage:-



ExamCentrePage:-

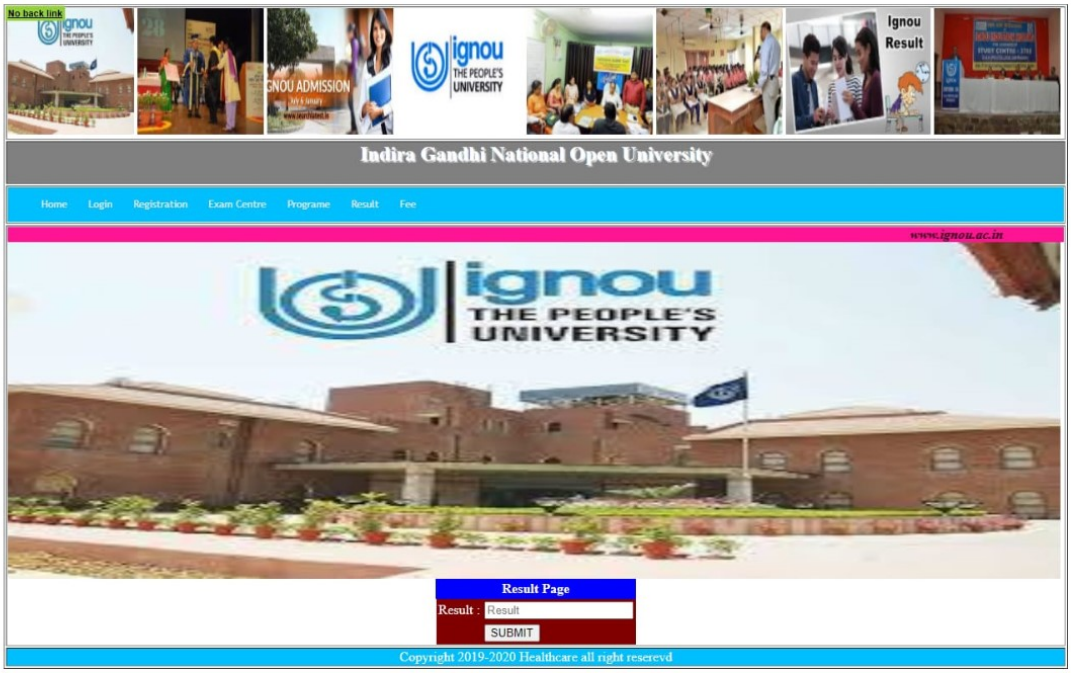


S.No	Regional Centre	Exam Centre	Address
1	HYDERBAD	0105 WARANGAL	LAL BHADUR COLLAGE SARDAR PATEL ROAD
2	GUWAHTI	0401 JALUKBARI, GUWAHATI	MAIN ARTS & LAW BUILDING GUWAHATI UNIVERSITY CAMPUS JALUKBARI GUWAHATI ASSAM - 781014
3	PATNA	0511 GAYA	IGNOU STUDY CENTRE MANVIKI BHAWAN GAYA COLLEGE GAYA - 823001
4	CHANDIGARH	2203 PATIALA	DEPT. OF DISTANCE EDUCATION PUNJABI UNIVERSITY PATIALA PUNJAB - 147002
5	DELHI-I	0765 JAMIA NAGAR	JAMIA MILLIA ISLAMIA JAMIA NAGAR NEW DELHI - 110025
6	AHMEDABAD	0905 SURAT	IGNOU STUDY CENTRE ROOM NO-4, M.T.B. ARTS COLLEGE ATHWA LINES, SURAT

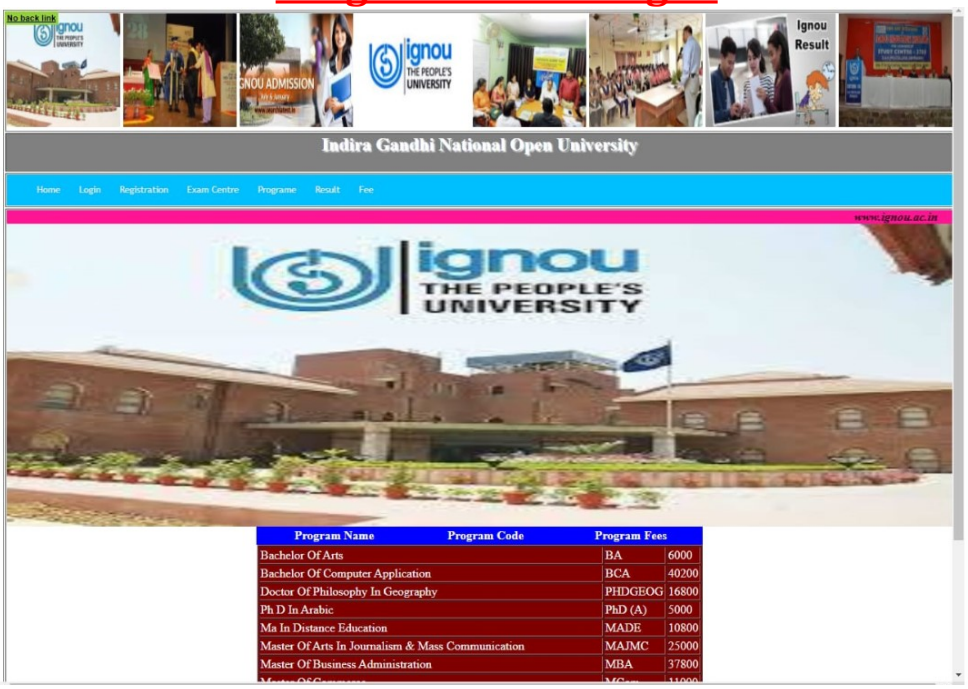
ProgrammePage:-

Program Name
Bachelor Of Arts
Bachelor Of Computer Application
Doctor Of Philosophy In Geography
Ph D In Arabic
Ma In Distance Education
Master Of Arts In Journalism & Mass Communication
Master Of Business Administration

ResultPage:-



ProgrammeFeesPage:-



Testing to be used

TESTING:- Testing is finding out how well something works. In terms of human beings, testing tells what level of knowledge or skill has been acquired. In computer hardware and software development, testing is used at key checkpoints in the overall process to determine whether objectives are being met. Most of us have had an experience with software that did not work as expected. Software that does not work can have a large impact on an organisation Software that does not work can have a large impact on an organization.
Similarly, Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of A software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process.

Types of Testing

WHITE BOX TESTING:-White-box testing is a verification technique software engineers can use to examine if their code works as expected. The use of equivalence partitioning and boundary value analysis to manage the number of test cases that needs to be written and to examine error prone/extreme “corner” test cases.

BLACK BOX TESTING:- Black Box Testing, also known as Behavioral Testing, is a software testing method in which the internal structure/ design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

LEVEL OF TESTING

UNIT TESTING:- **Black Box Testing**, also known as Behavioral Testing, is a software testing method in which the internal structure/ design/ implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

INTEGRATION TESTING:- **Integration testing**, also known as integration and testing (I&T), is a software development process which program units are combined and tested as groups in multiple ways.

SYSTEM TESTING:- **System testing** ,is the type of testing to check the behavior of a complete and fully integrated software product based on the software requirements specification (SRS) document.

USER ACCEPTANCE TESTING:-**User acceptance testing (UAT**) is the last phase of the software testing process. During UAT, actual software users test the software to make sure it can handle required tasks in real world scenarios, according to specifications.

TYPES OF ACCEPTANCE TEST

A) Alpha Testing:- Alpha testing is conducted by Customer at the developer's site, it is performed by potential users like developer, end users or organization users before it is released to external customers & report the defects found while Alpha testing.
This software product testing is not final version of software application, after fixing all reported bug (after bug triage) the new version of software application will release. Sometimes the Alpha Testing is carried out by client or an outsider with the attendance of developer and tester. The version of the release on which Alpha testing is perform is called "**Alpha Release**".

B)Beta Testing:-Most if times we have the sense of hearing term "Beta release/version", so it is linked to Beta Testing.

180 Basically the beta testing is to be carried out without any help of developers at
the end user's site by the end users &, so it is performed under uncontrolled environment. Beta testing is also known as Field testing. This is used to get feedback from the market.
This testing is conducted by limited users & all issues found during this testing are reported on continuous basis which helps to improve the system. Developers are taking actions on all issues reported in beta testing after bug triage & then the software application is ready for the final release. The version release after beta testing is called "**Beta Release**"
181 182

TEST PLAN

The test plan is a mandatory document. You can't test without one. For simple, straight-forward projects the plan doesn't have to be elaborate but it must address
certain items. As identified by the "American National Standards Institute and Institute for Electrical and Electronic Engineers Standard 829/1983 for Software Test Documentation", the following components should be covered in a software test plan. Test plan is a formal document that describes the following:
1. Scope, objectives, and the approach to testing.
2. People and equipment dedicated/allocated to testing.
3. Tools that will be used.
4. Dependencies and risks.
5. Categories of defects.
6. Test entry and exit criteria.
7. Measurements to be captured.
8. Reporting and communication
The release of a new application or an upgrade inherently carries a certain amount
of risk that it will fail to do what it's supposed to do. A good test plan goes a long way towards reducing this risk. By identifying areas that are riskier than others we can concentrate our testing efforts there.
When the problems are inevitably found, it's important that both the IT side and the business users have previously agreed on how to respond. It is very common
176 to use a set of rating categories that represent decreasing relative severity in terms
of business/commercial impact. In one system, '1' is the most severe and 6' has the least impact. A defect's category is as follows:

1. **Show Stopper –** It is impossible to continue testing because of the severity of the defect**.**
2. **Critical -** Testing can continue but the application cannot be released into

production until this defect is fixed.

3. **Major -** Testing can continue but this defect will result in a severe departure from the business requirements if released for production.

4. **Medium** - Testing can continue and the defect will cause only minimal departure from the business requirements when in production.

5. **Minor-** Testing can continue and the defect will not affect the release into production. The defect should be corrected but little or no changes to business requirements are envisaged.

6. **Cosmetic–** Minor cosmetic issues like colors, fonts, and pitch size that do not affect testing or production release. If, however, these features are important business requirements then they will receive a higher severity level.
177 178

Items Covered by a Test Plan

Component Description Purpose
Responsibilities Specific people who are and their assignments
Assigns responsibilities and keeps everyone on track and focused
Assumptions Code and systems status and availability
Avoids misunderstandings about schedules
Test Testing scope, schedule, duration, and prioritization
Outlines the entire process and maps specific tests
Communication Communications plan—who, what, when, how
Everyone knows what they need to know when they need to know it
Risk Analysis Critical items that will be tested
Provides focus by identifying areas that are critical for success
Defect Reporting How defects will be

logged and documented
Tells how to document a defect so that it can be reproduced, fixed, and retested

Test cases

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly.The process of developing test cases can also help find problems in the requirements or design of an application.

Login Page Test :

Test ID Objectives Description Expected Result
Actual Result Status
TC 1. To check login
Text box accepts valid username and password
1 . Enter valid username and invalid password.
2 . Enter invalid username and valid password.
3 . Enter invalid username and invalid password.
4 . Enter valid username and valid password.
Error message
Error message
Error message
Response to home page
Same
Same
Same
Same
Pass

Pass
Pass
Pass 183

Registration Page Test :

Test
ID
Objectives Description Expected
Result
Actual
Result
Status
TC 2.
TC 3.
To check some
important
textbox should
not leave blank.
Password and
re enter
password
values must be
same
1 . Left the
important
Text boxes blank.
2 . Fill the
Text boxes with
required data.
1 . Enter
dissimilar values.
2 . Enter same
password values
Error
message
Accepted
Error
message
Accepted
Same
Same
Same
Same
Pass
Pass
Pass
Pass 184

SYSTEM SECURITY AND VALIDATION

Secure Password

- As part of entity authentication on the matrix the confidential authentication information composed of a string of characters (passwords) are used.
- Passwords are only stored as SHA1 hash. Hash conversion is done at the client end

Automatic logoff:-

After a pre-determined time of inactivity (for example, 15 minutes), an electronic session is terminated.

Secure Storage of Login ID Password:-

If Passwords are stored in its original form, then the system administrator can easily know what password & ID Contractor has specified. To prevent this, electronic signature i.e. Message Digest of the password derived after running the hash algorithm, and the same is stored in the database.

Certificate Based Access

If User ID/Password is the only form of authentication then Contractor can refuse that he didn't log into that website or his User ID & Password was stolen or system administrator had logged in on his behalf. Further Universally Login ID/Password based access is considered as weak for of Authentication and hence Digital Certificate Based access is widely used.

Role Based Access

User who logs into the website gets to access the content (full or limited) depending on the rights/privileges he has, which further depends on his Job Role or Designation.

LIMITATION OF THE PROJECT

1. It is an intranet application and only authorized user can access the sites.

2. People who are not familiar with computer can't use this software.

3. No user can login or access the sites without the valid user name and password.

4. Maintenance cost is high because of its distribution over large
    a. network.

5. Some of the tabs are not working because their no need for need for them at this stage.

FUTURE SCOPE OF THE PROJECT

1. In future, if the client do some enhancement or new / change operations request in this software, then it can be easily done as the database is implemented using SQL Server which is extensible.

2. Also this software is implemented using proper comments in each module so that any programmer in future can understand it & upgrade it.

3. In future if the management opens some more branches for billing, complaints process in different parts/state of the country the same software can be used.

4. This project helps to increase the efficiency of work that can be done more accurately in comparison to manually. All the information can be stored for a long time for further enquiry.

5. In future, if the management decides to put all the information on the web page, the same table structure can be used to store records; only a web site is needed to be created. Further, if the management wants to add or delete rooms it can be done systematically.

6. This proposed software will also have the following scope for the future changes.

Security – This software will provide the online financial transactions to the student with their debit/credit cards. Hence, security is a major concern for this application. Regular efforts should be made to make this software more secure & reliable.

Mobile Alerts –This facility can also be added in this software. A student and admin can get regular updates on his/her mobile like exam date, time, address etc.

Updated GUI –Since this is a portal where different products are available for the user services, hence to provide an interactive environment to the user to make online entrance exam, there should be regular updates required to make GUI showing all products in an attractive & user friendly manner.

BIBLIOGRAPHY

1. Professional ASP.NET MVC 5 by Jon Galloway, Brad Wilson, K. Scott Allen, David Matson August 2014

2. Professional Team Foundation Server 2013 by Steven St. Jean, Damian Brady, Ed Blankenship, Martin Woodward, Grant Holiday May 2014
3. Professional C# 5.0 and .NET 4.5.1 by Christian Nagel, Jay Glynn, Morgan Skinner February 2014

4. Professional Microsoft SQL Server 2014 Integration Services by Brian Knight, Devin Knight, Jessica M. Moss, Mike Davis, Chris Rock April 2014

5. Professional SQL Server 2012 Internals and Troubleshooting by Christian Bolton, Justin Langford, Glenn Berry, Gavin Payne, Amit Banerjee, Rob Farley November 2012

WEB REFERENCES-

http://www.w3schools.com/sql/sql_quickref.asp

http://www.lynda.com/SQL-Server-training-tutorials/