PROJECT REPORT ON

Online Examination System

in partial fulfillment of the requirements for the award of the diploma in

SOFTWARE DEVELOPMENT AND SYSTEM ADMINISTRATION

Level-5 Diploma Certificate National Skill Development Corporation

New Delhi

Submitted by

Noufiya Nazrin P N

ROLL NO: 192539

DEPARTMENT OF SOFTWARE DEVELOPMENT
AND SYSTEM ADMINISTRATION
(2019)

M.E.S. COLLEGE MARAMPALLY ALUVA -7
AFFILIATED TO MAHATMA GANDHI UNIVERSITY
KERALA

M E S COLLEGE, MARAMPALLY
ALUVA -7

DEPARTMENT OF SOFTWARE DEVELOPMENT AND
SYSTEM ADMINISTRATION

Certificate

This to certify that the report entitled

Online Examination System

Has been submitted by

NOUFIYA NAZRIN P N

ROLL NO: 192539

In partial fulfillment of the award of the diploma in

SOFTWARE DEVELOPMENT AND SYSTEM ADMINISTRATION

Level-5 Diploma Certificate National Skill Development Corporation

New Delhi

During the academic year 2019

Register No: 190041000903

Staff in Charge   B.Voc. Coordinator   Nodal Officer   Head of the Department

Submitted for the examination held on ……………………………..

Examiners

1.

2.

ACKNOWLEDGEMENT

First of all, I bow my head before the God Almighty in prayer and gratitude for all the blessings showered on me to complete the project work successfully.

I express my heartfelt gratitude to **Dr. MANZUR ALI P P**, Principal and **Mrs. ASHIKA K A**, Head of the Department, Software Development and System Administration, MES College Marampally for their cooperation and guidance throughout the course of the project work.

I am deeply indebted to my project guide **Mrs. JYOTHI LAKSHMI S S,** Assistant Professor, SDSA for his continuous encouragement and valuable guidance given to me. I am also very thankful to all the faculties of the department of Software Development and System Administration for their timely suggestions and encouragement.
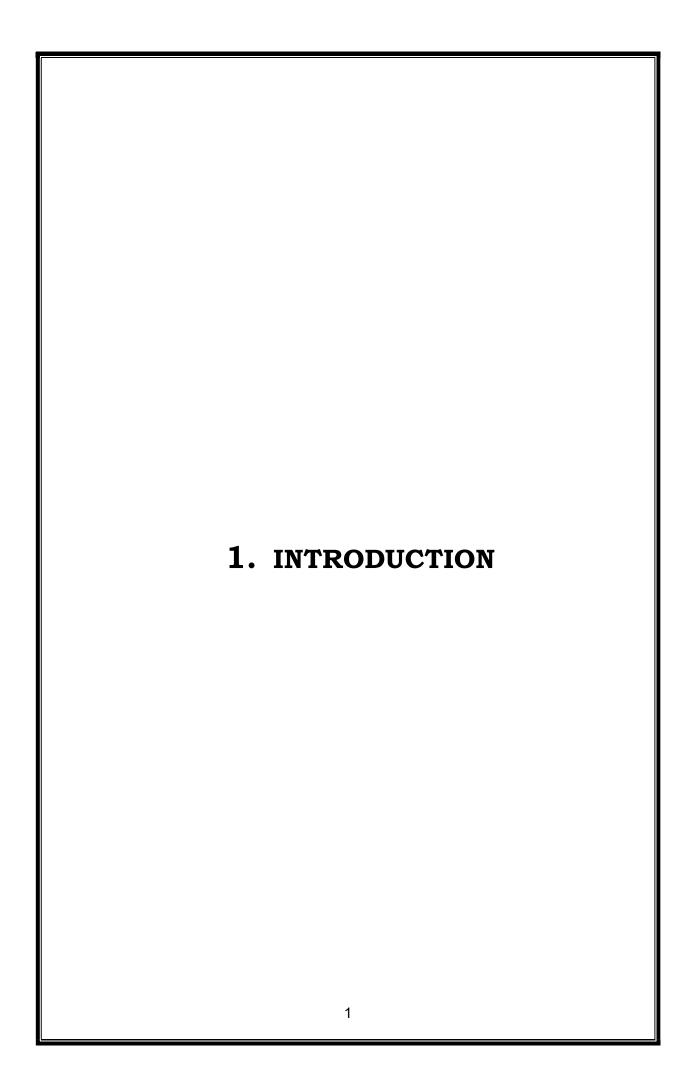
I extend my sincere thanks to all the friends for their provoking suggestions and enlightening discussions which helped me to complete my project successfully.

And last but not the least, I extend my heartfelt thanks to my beloved parents, without their support I could never have reached this point.

NOUFIYA NAZRIN P N

Contents

1. INTRODUCTION

1.1. Overview of the system

Online examinations are an important method of evaluating the success potential of students. This research effort the individuals under consideration were students who would be enrolling in computer courses or Technologies Registrations. A prototype of a web-based placement examination system is described from the standpoint of the research effort, end user, and software development.

An on-line educational system including exam processing and electronic journal features. An instructor builds a course based questions which on-line contain in identification of assignments. Which are compiled into an on-line exam syllabus?

Users enrolled in the platform may access the electronic details they provided and perform various functions with the on-line educational system in order to participate in the on-line examinations. Users can receive an on-line exam, having multimedia content, for the course, and they can electronically provide answers for the exam. And after Completion of their duration of exam they are provided the grade or marks secured in their examinations.

1.2. Problem definition and objective
Problem definition

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

To perform this, the 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. That is,

1. Online login
2. Online schedule
3. Question bank management
4. Conducting examination
5. Online result declaration

At the login module, username and password facility and credentials should be checked properly at the time of login for student, faculty and Examination Department administrator.

In the schedule Module, it sends the exam schedule to the student.

Question bank module for the examination could be changed dynamically.

Automatic checking of objective answers and manual checking of single-sentenced answers should be facilitated, if any. The Objective answers module will be checked automatically by the system from the database.

In the exam module, when the student starts examination the timer will starts automatically and shows the student how much time is left.

At result module, it displays the result of each student and send by the examination department.

Objective

- This can be used in educational institutions
- No restriction that examiner has to be present when the candidate takes the test.

2. SYSTEM ANALYSIS

2.1. Introduction

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

To perform this, the 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. That is,

1. Online login
2. Online schedule
3. Question bank management
4. Conducting examination
5. Online result declaration

2.2. Existing system

The existing system of conducting examination process in our institution is manual. Existing system is a large ma power process and is difficult to implement it in different platform. It has so many problems. So, we introduced an online examination system, which is fully computerized. Existing system is a large man power and is difficult to implement.

Disadvantages

- Time consuming
- Difficult to analyze the exam manually
- Results are not precise as evaluations are done annually
- Result processing after summation of exam tales more time as it is done manually,

These problems can be solved by introducing a new system. To overcome these problems we are implementing a computerized system with combination of MVC, HTML, Java Script, J Query, CSS and bootstrap.

2.3. Proposed system

The present system has limitations. The proposed system is designed to eliminate all the disadvantages of the existing system. The proposed system is an interactive system, highly user friendly and designed exclusively for the Authority. The system is developed in MVC and SQL SERVER as backend.

The proposed system reduces manpower, paper wastage etc. it is very efficient and saves time. The system is having many added advantages which are having higher number of economic features for user interface. The system modular structure will allow the system to pair with other system for easy data transfer with high control levels in user hands. The proposed system will be more appropriate and genuine without many errors and by using this application question paper and result can be generated easily.

It is designed keeping in mind all the drawbacks of the present system. The proposed system overcomes all the hurdles faced by the existing system.

2.3.1. Justification of the proposed system

The modern computerized system is developed with the aim to overcome the drawbacks of existing manual system. The proposed system has got many advantages. People from different parts of the world can register very easily. The new system is more personalized. It is a maze in such a manner that all the new users can understand all the options in it very easily. It is made in a quick and easy referential manner. Access to all important matters is not always locked and can be opened easily at the time of urgency. The advantages of proposed system are that security is maintained in the new system. Securities for all important data are maintained confidentially. As it is easily understandable and user-friendly, quickly entries can be made in this system.

2.3.2. Benefit of the proposed system

- Provides complete online web-based solution, including student login, giving tests, storing of results.
- Complete web-based administration, administrator can manage examination, student registration and question bank from web interface.
- Student ca give examination without pen and paper
- 100% accuracy in result calculation
- Faculty can manage courses and question bank

2.4. Feasibility study

The purpose of the feasibility study is to investigate the present system, evaluate the cost and effectiveness of the proposed system, evaluate the

possible applications of computer-based methods, select a tentative system, evaluate the impact of the proposed system on existing personnel and ascertain the need for new personnel.

Economic Feasibility

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and savings that are expected from a system and compare them with costs, decisions are made to design and implement the system. This part of the feasibility study gives the top management the economic justification for the new system. This is an important input to the management, because very often the top management does not like to get confused by the various technicalities that are bound to be associated with a project of this kind. A simple economic analysis that gives the actual comparison of costs and benefits is much more meaningful in such cases. In the system, the organization is most satisfied by economic feasibility. Because, if the organization implements this system, it need not require any additional hardware resources as well as it will be saving a lot of time.

Technical Feasibility

Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to the feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities.

Operational Feasibility

The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since, processing speed is very high and the work is reduced in the maintenance point of view, management is convinced that the project is operationally feasible.

2.5 System specification

System specifications help to define the operational and performance guidelines for a system. It may outline how the system is expected to perform, and what that may include. Key specifications may include interface definitions, document design rules and functional areas.

2.5.1. Software specification

Operating system: Windows 10
Front-End: MVC
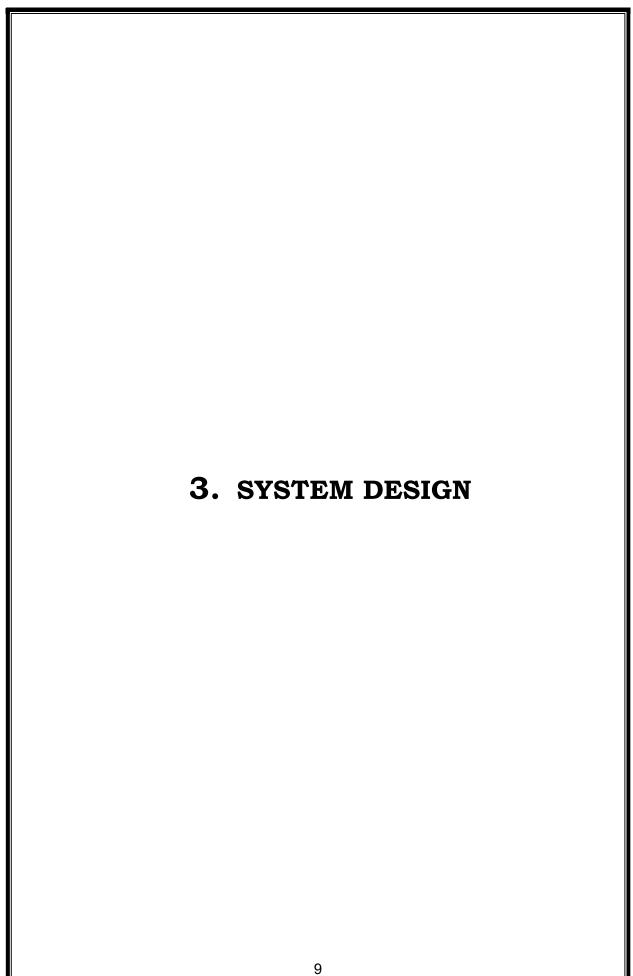Back-End: SQL Server 2017

2.5.2. Tools/platform

Visual Studio 2019
Internet Explorer

2.5.3. Hardware specification

Processor: Intel® Pentium® CPU N3540 @2.16GHz 2.16GHz
RAM: 4.00 GB
HDD: 250 GB or Above

3. SYSTEM DESIGN

3.1. Introduction

The system is a solution "how to" approach to the creation of new system. The basis for the system design is a good system analysis. The important phase is composed of several steps. The first step is to determine how the output should be produced and in what format. Second the database, the input data and the master files have to be designed to meet the requirements of the proposed system. It provides for the understanding and procedural details necessary for implementing the candidate system o the user and the organization are documented and evaluated by the management as steps towards implementation. System design goes through phases development, logical and physical design.

DFD shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the inputs, outputs, database and procedures. The design covers the review of the current physical system, prepares output specifications, prepares input specifications and specifies the implementation plan, reviews benefits, costs, and target dates and constrains.

System design objectives include specify logical design elements, support business activities ensure that systems features meet user requirements, provides engineered for ease of use by people, provides detailed software development specification, confirm to design standards.

The logical design specifics output, input file and screen layout. The physical design procedures the working system by defining the design specification that tells the programmer exactly what the candidate system must do.

3.2. Output design

One of the important features of an information system for users is the output it produces. Output is the information delivered to users through the information system. Without quality output, the entire system appears to be unnecessary that users will avoid using it. Uses generally merit the system solely by its output. In order to create the most useful output possible. One works closely with the user order to create the most useful output possible. One works closely with the user through an interactive process, until the result is considered to be satisfactory.

Output design has been an ongoing activity almost from the beginning of the project. In the study phase, outputs were identified and described general in the project directive. A tentative output medium was then selected and sketches made for each output. In the feasibility analysis, a "best" new system was selected; its description identified the input and output media. In the design phase the system has included an evolution and selection of specific equipment for the system.

Outputs from computer system are required primarily to communicate the results of processing to the user.

Admin_tbl

admin_id	admin_name	admin_password
1	Noufiya	noufiya
2	Sameesha	sameesha

Candidate_tbl

candidate_id	candidate_name	candidate_password	candidate_email	candidate_contact	candidate_institution
1	Anu	anu@1234	anuanu@gmail.com	7896543210	MES COLLEGE MARAMPALLY
2	Ajith	ajith123	ajith@gmail.com	9876054321	MES COLLEGE MARAMPALLY

Exam_tbl

exam_id	subject_id	exam_title
1	2	Test1

Question_tbl

all_qus_id	qus_id	admin_id	exam_id	qus_title
1	1	1	1	HTML stands for
2	2	1	1	The correct sequence of HTML tags for starting a webpage is -
3	3	1	1	Which of the following element is responsible for making the text bold in HTML
4	4	1	1	Which of the following tag is used for inserting the largest heading in HTML
5	5	1	1	Which of the following tag is used to insert a line-break in HTML
6	6	1	1	How to create an unordered list (a list with

				the list items in bullets) in HTML
7	7	1	1	Which character is used to represent the closing of a tag in HTML
8	8	1	1	How to create an ordered list (a list with the list items in numbers) in HTML
9	9	1	1	How to insert an image in HTML
10	10	1	1	input is -

ans1	ans2	ans3	ans4	correct_ans	Selectedvalue
HyperText Markup Language	HighText and links Markup Language	HighText Machine Language	None of these	1	NULL
Head, Title, HTML, body	HTML, Body, Title, Head	HTML, Title, Head, Body	HTML, Head, Title, Boday	4	NULL
Pre	a	b	br	3	NULL
h3	h1	h5	h6	2	NULL
Br	a	pre	b	1	NULL
Ul	Ol	li	i	1	NULL
\	!	/	.	3	NULL
Ul	Ol	li	i	2	NULL
img href="image.jpg"	img url="image.png"	img link="image.jpg"	img src="image.png"	4	NULL
a format tag	an empty tag	All of the above	None of the above	2	NULL

Result_tbl

result_id	exam_id	candidate_id	total	grade	quality
1	1	1	10	A+	Best
2	1	1	7	B	Good

Subject_tbl

subject_id	subject_title0	subject_code
1	HTML	S01
2	CSS	S02

3.3. Input design

Input design, user oriented data are converted to a computer based format. It is the link between user and the information system. The input design involves determining what the inputs are, how the data should be performed, how to validate data, how to minimize data entry and how to provide a multiuser facility.

The main objectives of the input design are:
1. Produce cost effective method of input
2. Achieve highest possible level of accuracy
3. Ensure that the input is acceptable to and understood by user

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering users like manager need to know the allocated space for each field; field sequence and which must match with that in the source document.

Text1   Text2   Text3

Text4

Text5 [   ]
Text6 [   ]

[Button]

Text7

Text1

Text2   Text3

Text10

Text11

Text4

Text5

Text6

Text7

Text8

Text9

Text12

Text1

Text2   Text3

Text10

Text11

Text4

Text5

Text6

Text7

Text8

Text9

Text12

Text13   Text14
subtext   subtext

```
┌─────────────────────────────────────────────────────────┐
│  Text1                                                   │
│                                         Text10           │
│        Text2    Text3                      Text11        │
├──────────┬──────────────────────────────────────────────┤
│ Text4    │   Text12                                      │
│          │                                               │
│ Text5    │   Text13  ┌──────────┐                        │
│          │   Text14  ├──────────┤                        │
│ Text6    │           └──────────┘                        │
│          │                                               │
│ Text7    │                                               │
│          │                                               │
│ Text8    │                                               │
│          │                                               │
│ Text9    │                                               │
│          │                                               │
└──────────┘                                               
```

3.4. Database design

Table: Admin_tbl

admin_id	Int	Primary key
admin_name	varchar(50)	Null
admin_password	varchar(50)	Null

Table: Candidate_tbl

candidate_id	int	Primary key
candidate_name	varchar(50)	Null
candidate_password	varchar(50)	Null
candidate_email	varchar(50)	Null
candidate_contact	varchar(50)	Null
candidate_institution	varchar(50)	Null

Table: Exam_tbl

exam_id	int	Primary key
subject_id	int	Foreign key from the table Subject_tbl
exam_title	varchar(50)	Null

Table: Questions_tbl

all_qus_id	int	Primary key
qus_id	int	Null
admin_id	int	Foreign key from the table Admin _tbl

15

exam_id	int	Foreign key from the table Exam_tbl
qus_title	varchar(300)	Null
ans1	varchar(300)	Null
ans2	varchar(300)	Null
ans3	varchar(300)	Null
ans4	varchar(300)	Null
correct_ans	int	Null
selectedvalue	int	Null

Table: Result_tbl

result_id	int	Primary key
exam_id	int	Foreign key from the table Exam_tbl
candidate_id	int	Foreign key from the table Candidate_tbl
Total	int	Null
Grade	varchar(50)	Null
Quality	varchar(50)	Null

Table: Subject_tbl

subject_id	int	Primary key
subject_title	varchar(50)	Null
subject_code	varchar(50)	Null

3.5. ER Diagram

3.6. Data Flow Diagram

Level 0:

Level 1:

Fig 1.1: Level 1 ADMIN'S side Data Flow Diagram

Fig 1.2: Level 1 CANDIDATE'S side Data Flow Diagram

Level 2:

Admin → Request For Login → LOGIN → Check For Login → admin

LOGIN → Response → Admin

admin → Reply → LOGIN

LOGIN → CREATE CANDIDATE → Insert Data → candidate

candidate → Reply → CREATE CANDIDATE

READ CANDIDATE → Read Data → candidate

candidate → Reply → READ CANDIDATE

UPDATE CANDIDATE → Update Data → candidate

candidate → Reply → UPDATE CANDIDATE

DELETE CANDIDATE → Delete Data → candidate

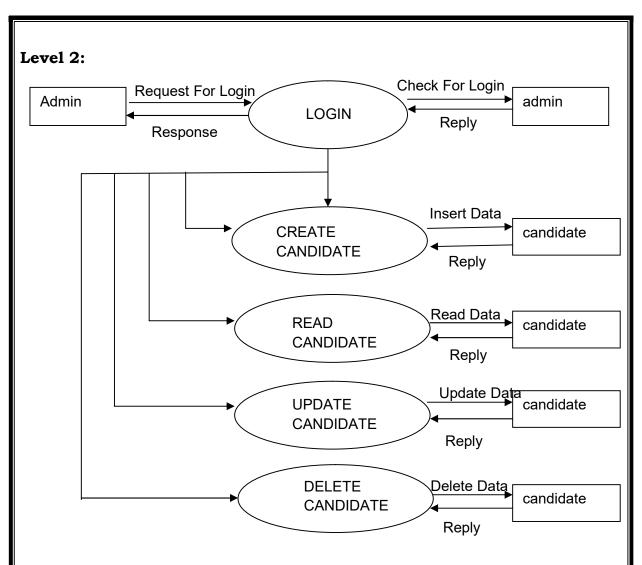candidate → Reply → DELETE CANDIDATE

Fig 2.1.1: Level 2 ADMIN'S side Data Flow Diagram

Fig 2.1.2: Level 2 ADMIN'S side Data Flow Diagram

Fig 2.1.3: Level 2 ADMIN'S side Data Flow Diagram

Fig 2.1.4: Level 2 ADMIN'S side Data Flow Diagram

Fig 2.2.1: Level 2 ADMIN'S side Data Flow Diagram

4. SYSTEM DEVELOPMENT

4.1. Process description

The procedure of "**Online Examination**" begins with a selection among candidates. After login, the student is the directed to the page showing available exams and by selectin**g o**ne among them he/she can attend the exam. The student is provided a set of questions and an amount of time to answer the questions. Different subjects hold different set of questions and each question have 4 choices with answer. After the exam is over the form will automatically submitted and the result is generated.

4.2. Source code

Timer

```
@model Online_Exam_Project_Main.Models.Questions_tbl

@{
    ViewBag.Title = @ViewBag.score;
}
<br>

<h1></h1>



<div id="clockdiv">
    <h1 style="color: red">

        <div class="minutes" style="float: left"> </div>
        <p style="float: left">:</p>
        <div class="seconds"></div>
    </h1>
</div>

<script>
    function getTimeRemaining(endtime) {
        var t = Date.parse(endtime) - Date.parse(new Date());
        var seconds = Math.floor((t / 1000) % 60);
        var minutes = Math.floor((t / 1000 / 60) % 1);

        return {
            'total': t,
            'minutes': minutes,
            'seconds': seconds
        };
    }

    function initializeClock(id, endtime) {
        var clock = document.getElementById(id);

        var minutesSpan = clock.querySelector('.minutes');
        var secondsSpan = clock.querySelector('.seconds');

        function updateClock() {
            var t = getTimeRemaining(endtime);
```
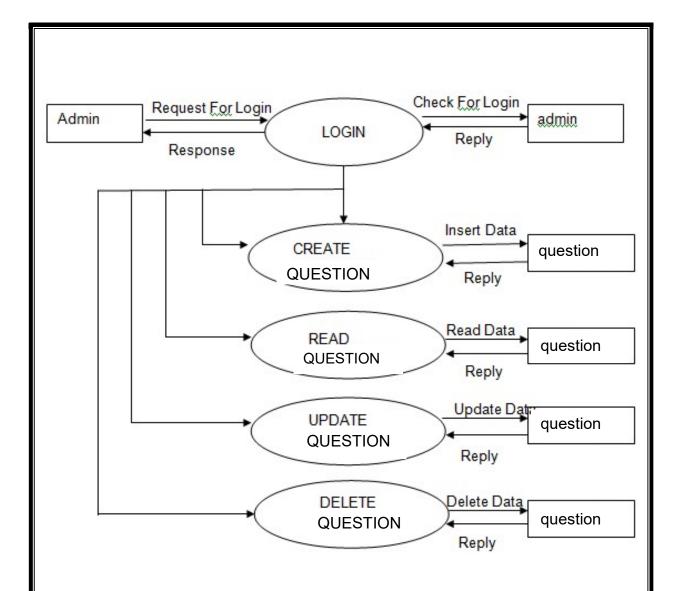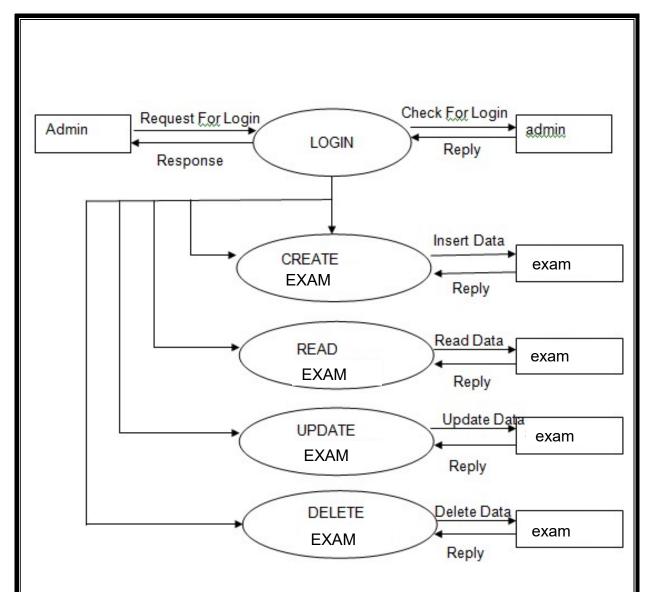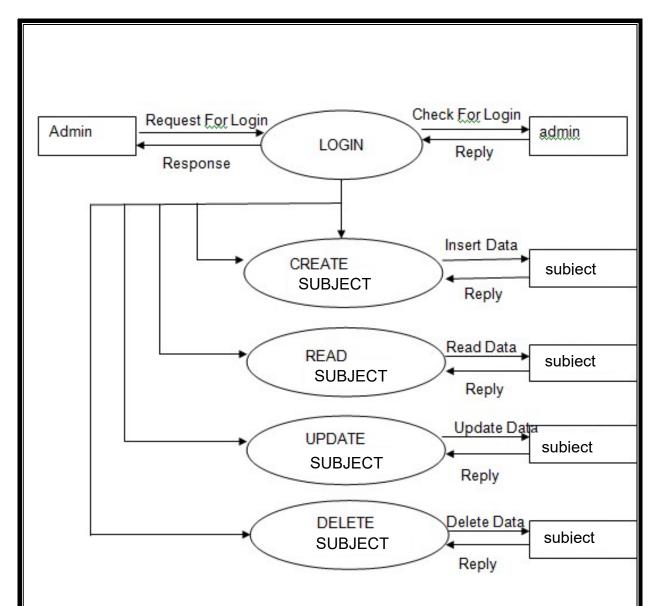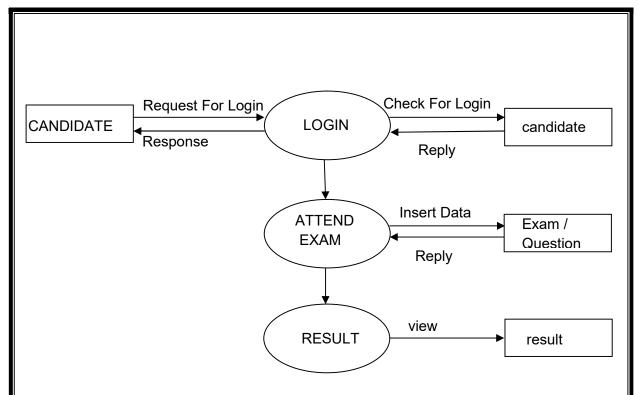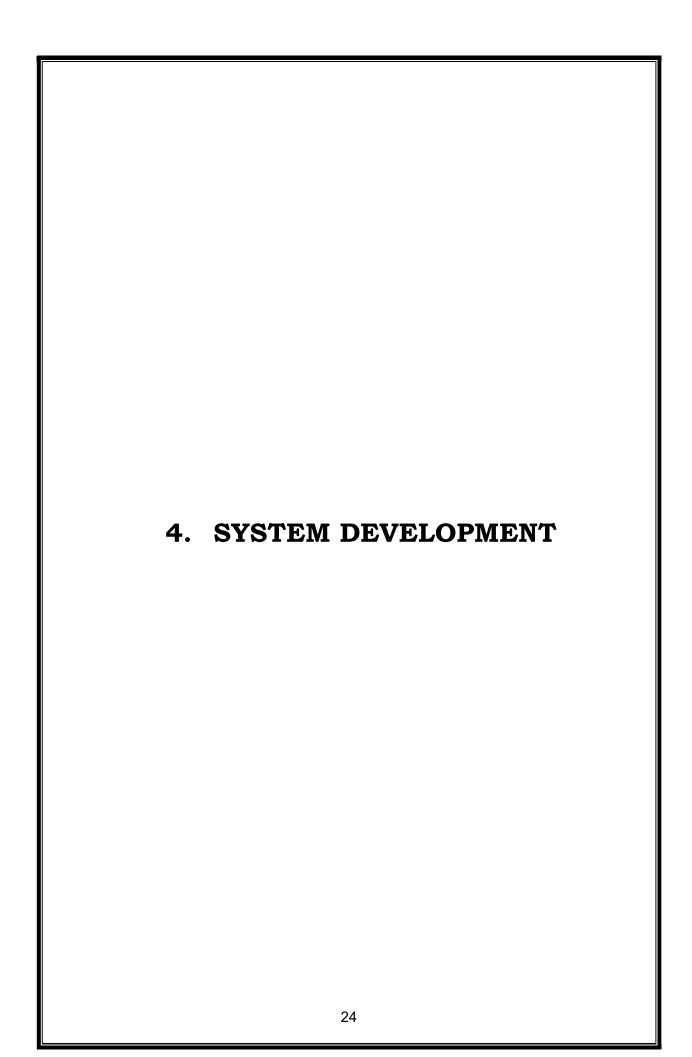
```
        minutesSpan.innerHTML = ('0' + t.minutes).slice(-2);
        secondsSpan.innerHTML = ('0' + t.seconds).slice(-2);

        if (t.total <= 0) {
            clearInterval(timeinterval);
        }
    }

    updateClock();
    var timeinterval = setInterval(updateClock, 1000);
}

var deadline = new Date(Date.parse(new Date()) + 60 * 60 * 1000);
initializeClock('clockdiv', deadline);

// ...................................
var _second = 1000;
var _minute = _second * 60;
var _hour = _minute * 60;
var _day = _hour * 24;
var timer;
var end = new Date().getTime() + (60 * _second);
//

function showRemaining() {
    var now = new Date();
    var distance = end - now;
    if (distance < 0) {

        clearInterval(timer);
        document.getElementById('countdown') +.innerHTML = 'EXPIRED!';

        return;
    }
    var days = Math.floor(distance / _day);
    var hours = Math.floor((distance % _day) / _hour);
    var minutes = Math.floor((distance % _hour) / _minute);
    var seconds = Math.floor((distance % _minute) / _second);

    document.getElementById('countdown').innerHTML = days + 'days ';
    document.getElementById('countdown').innerHTML += hours + 'hrs ';
    document.getElementById('countdown').innerHTML += minutes + 'mins ';
    document.getElementById('countdown').innerHTML += seconds + 'secs';
}

timer = setInterval(showRemaining, 1000);
</script>

<div>
    @Html.Partial("_QuestionPartial")
</div>
```

Admin Login

```
@model Online_Exam_Project_Main.Models.Admin_tbl

@{
    ViewBag.Title = "Login";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>Admin Login</h2>
```

```razor
@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()

    <div class="form-horizontal">
        <hr />
        @Html.ValidationSummary(true)

        <div class="form-group">
            <Label class = "control-label col-md-2">Admin Name:</Label>
            <div class="col-md-10">
                @Html.EditorFor(model => model.admin_name)
                @Html.ValidationMessageFor(model => model.admin_name)
            </div>
        </div>

        <div class="form-group">
            <Label class="control-label col-md-2">Admin Password:</Label>
            <div class="col-md-10">
                <input type="password" name="admin_password" autocomplete="off"/>
            </div>
        </div>

        <div class="form-group">
            <div class="col-md-offset-2 col-md-10">
                <input type="submit" value="Login" class="btn btn-default" />
            </div>
        </div>
    </div>
}

<div>
    @Html.ActionLink("New Admin", "Create")
</div>

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
}
```

Admin Controller

```csharp
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Online_Exam_Project_Main.Models;


namespace Online_Exam_Project_Main.Controllers
{
    public class AdminController : Controller
```

27

```csharp
    {
        private Exam_QuizEntities db = new Exam_QuizEntities();

        // GET: /Admin/
        public ActionResult Index()
        {
            return View(db.Admin_tbl.ToList());
        }

        // GET: /Admin/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Admin_tbl admin_tbl = db.Admin_tbl.Find(id);
            if (admin_tbl == null)
            {
                return HttpNotFound();
            }
            return View(admin_tbl);
        }

        // GET: /Admin/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: /Admin/Create
        // To protect from overposting attacks, please enable the specific
properties you want to bind to, for
        // more details see http://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult
Create([Bind(Include="admin_id,admin_name,admin_password")] Admin_tbl
admin_tbl)
        {
            if (ModelState.IsValid)
            {
                db.Admin_tbl.Add(admin_tbl);
```

```csharp
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        return View(admin_tbl);
    }

    // GET: /Admin/Edit/5
    public ActionResult Edit(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Admin_tbl admin_tbl = db.Admin_tbl.Find(id);
        if (admin_tbl == null)
        {
            return HttpNotFound();
        }
        return View(admin_tbl);
    }

    // POST: /Admin/Edit/5
    // To protect from overposting attacks, please enable the specific
properties you want to bind to, for
    // more details see http://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult
Edit([Bind(Include="admin_id,admin_name,admin_password")] Admin_tbl
admin_tbl)
    {
        if (ModelState.IsValid)
        {
            db.Entry(admin_tbl).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        return View(admin_tbl);
    }

    // GET: /Admin/Delete/5
    public ActionResult Delete(int? id)
```

```
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Admin_tbl admin_tbl = db.Admin_tbl.Find(id);
            if (admin_tbl == null)
            {
                return HttpNotFound();
            }
            return View(admin_tbl);
        }


        // POST: /Admin/Delete/5
        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public ActionResult DeleteConfirmed(int id)
        {
            Admin_tbl admin_tbl = db.Admin_tbl.Find(id);
            db.Admin_tbl.Remove(admin_tbl);
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        public ActionResult Login()
        {

            return View();

        }


        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Login(Admin_tbl ut)
        {
            if (ModelState.IsValid)
            {
                using (Exam_QuizEntities db = new Exam_QuizEntities())
                {

                    var log = db.Admin_tbl.FirstOrDefault(model =>
model.admin_name.Equals(ut.admin_name) &&
model.admin_password.Equals(ut.admin_password));
                    if (log != null)
```

```csharp
                {
                    Session["admin_name"] = log.admin_name;

                    return RedirectToAction("AdminProfile", "Admin");
                }
                else
                {
                    ViewBag.Message = "Username or password incorrect";
                    Response.Write("<script> alert('User Name or Password is
wrong')</script>");

                }
            }

        }

        return View();
    }
    public ActionResult Logout()
    {
        Session.Clear();
        return RedirectToAction("Login", "Admin");


    }

    public ActionResult AdminProfile()
    {
        return View();
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
  }
}
```

Question Controller

```csharp
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using Online_Exam_Project_Main.Models;

namespace Online_Exam_Project_Main.Controllers
{
    public class QuestionController : Controller
    {
        private Exam_QuizEntities db = new Exam_QuizEntities();

        // GET: /Question/
        public ActionResult Index()
        {
            var questions_tbl = db.Questions_tbl.Include(q =>
q.Admin_tbl).Include(q => q.Exam_tbl);
            return View(questions_tbl.ToList());
        }

        // GET: /Question/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Questions_tbl questions_tbl = db.Questions_tbl.Find(id);
            if (questions_tbl == null)
            {
                return HttpNotFound();
            }
            return View(questions_tbl);
        }

        // GET: /Question/Create
        public ActionResult Create()
        {
```

```csharp
        ViewBag.admin_id = new SelectList(db.Admin_tbl, "admin_id",
"admin_name");
        ViewBag.exam_id = new SelectList(db.Exam_tbl, "exam_id",
"exam_title");
        return View();
    }


    // POST: /Question/Create
    // To protect from overposting attacks, please enable the specific
properties you want to bind to, for
    // more details see http://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult Create(Questions_tbl questions_tbl)
    {
        Questions_tbl qtbl = new Questions_tbl();

        if (ModelState.IsValid)
        {
            qtbl.all_qus_id = questions_tbl.all_qus_id;
            qtbl.qus_id = questions_tbl.qus_id;
            qtbl.admin_id = questions_tbl.admin_id;
            qtbl.exam_id = questions_tbl.exam_id;
            qtbl.qus_title = questions_tbl.qus_title;
            qtbl.ans1 = questions_tbl.ans1;
            qtbl.ans2 = questions_tbl.ans2;
            qtbl.ans3 = questions_tbl.ans3;
            qtbl.ans4 = questions_tbl.ans4;
            qtbl.correct_ans = questions_tbl.correct_ans;
            db.Questions_tbl.Add(questions_tbl);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        ViewBag.admin_id = new SelectList(db.Admin_tbl, "admin_id",
"admin_name", questions_tbl.admin_id);
        ViewBag.exam_id = new SelectList(db.Exam_tbl, "exam_id",
"exam_title", questions_tbl.exam_id);
        return View(questions_tbl);
    }

    // GET: /Question/Edit/5
    public ActionResult Edit(int? id)
```
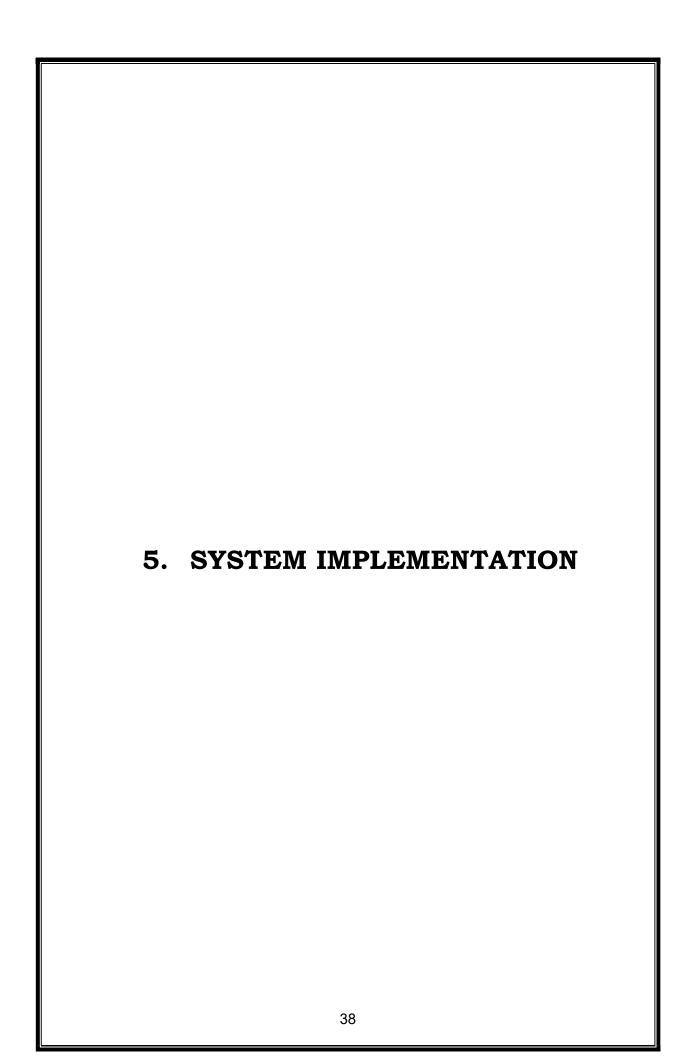
```
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Questions_tbl questions_tbl = db.Questions_tbl.Find(id);
            if (questions_tbl == null)
            {
                return HttpNotFound();
            }
            ViewBag.admin_id = new SelectList(db.Admin_tbl, "admin_id",
"admin_name", questions_tbl.admin_id);
            ViewBag.exam_id = new SelectList(db.Exam_tbl, "exam_id",
"exam_title", questions_tbl.exam_id);
            return View(questions_tbl);
        }


        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult
Edit([Bind(Include="all_qus_id,qus_id,admin_id,exam_id,qus_title,ans1,ans2
,ans3,ans4,correct_ans")] Questions_tbl questions_tbl)
        {
            if (ModelState.IsValid)
            {
                db.Entry(questions_tbl).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
            }
            ViewBag.admin_id = new SelectList(db.Admin_tbl, "admin_id",
"admin_name", questions_tbl.admin_id);
            ViewBag.exam_id = new SelectList(db.Exam_tbl, "exam_id",
"exam_title", questions_tbl.exam_id);
            return View(questions_tbl);
        }

        // GET: /Question/Delete/5
        public ActionResult Delete(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
```

```
        Questions_tbl questions_tbl = db.Questions_tbl.Find(id);
        if (questions_tbl == null)
        {
            return HttpNotFound();
        }
        return View(questions_tbl);
    }


    // POST: /Question/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public ActionResult DeleteConfirmed(int id)
    {
        Questions_tbl questions_tbl = db.Questions_tbl.Find(id);
        db.Questions_tbl.Remove(questions_tbl);
        db.SaveChanges();
        return RedirectToAction("Index");
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
    }
}
```

Show Question Controller

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

namespace Online_Exam_Project_Main.Controllers
{
    public class ShowQuestionController : Controller
    {
        //
        // GET: /ShowQuestion/
```

```csharp
        Exam_QuizEntities db = new Exam_QuizEntities();



    [HttpPost]
    public ActionResult Index(DrpList drp)
    {

        ViewBag.drpData = drp;
        ViewBag.questionNo = drp.QuestionNo;

        TempData["a"] = drp.QuestionNo;

        if (drp.QuestionNo == 1)
        //if (questionNo == null || id == null)
        {
            Session["exid"] = drp.examid;

            Questions_tbl SingleQuestion =
db.Questions_tbl.SingleOrDefault(m => m.qus_id == 1 && m.exam_id ==
drp.examid);


            TempData["qData"] = SingleQuestion;
            return RedirectToAction("NextQuestion");
            //return View(SingleQuestion);
        }
        else
        {


            Questions_tbl SingleQuestion =
db.Questions_tbl.SingleOrDefault(m => m.qus_id == drp.QuestionNo &&
m.exam_id == drp.examid);
            int qus = (int)drp.QuestionNo;

            //for (int i = 0; i <= drp.QuestionNo; i++)
            //{
            //    ViewBag.questionNo = qusno + i;
            //}


            return View(SingleQuestion);
```
36

```csharp
        }

    }


    int add = 0;

    public ActionResult NextQuestion()
    {
        int qNo = (int)TempData["a"];
        ViewBag.questionNo = qNo;
        Questions_tbl a = (Questions_tbl)TempData["qData"];

        return View(a);

    }
    [HttpPost]
    public ActionResult NextQuestion(Questions_tbl aaa)
    {
        if (aaa.correct_ans == aaa.selectedvalue && aaa.qus_id!=1)
        {
            Session["correctAns"] = Convert.ToInt32(Session["correctAns"]) +
1;
        }
        else if (aaa.correct_ans == aaa.selectedvalue && aaa.qus_id == 1)
        {
            Session["correctAns"] = 1;
        }

        if (aaa.qus_id == 10)
        {return RedirectToAction("Create","Result");
}
        int qId = (int)aaa.qus_id + 1;
        Questions_tbl SingleQuestion = db.Questions_tbl.SingleOrDefault(m
=> m.qus_id == qId && m.exam_id == aaa.exam_id);

        ViewBag.questionNo = qId;
        TempData["a"] = SingleQuestion.qus_id;
        TempData["qData"] = SingleQuestion;
        return RedirectToAction("NextQuestion");

    }
}}
```

5. SYSTEM IMPLEMENTATION

5.1 Testing

Testing is the process of examining the software to compare the actual behavior with that of the expected behavior. The major goal of the software is to demonstrate that faults are not present. In order to achieve this goal the tester executes the program with the intent of finding errors. Through testing cannot show absence of errors but by not showing their presence it is considered that these are not present.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operations commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.

Levels of testing
1. Unit testing
2. Integration testing
3. Validation testing
4. Output testing

Unit testing

In this each module is tested individually before integrating it to final system. Unit test focuses verification in the smallest unit of the software design in each module. This is known as module testing as here each module is tested to check whether it is produced the desire output and to see if any error occurs.

Integration testing

Integration testing sometimes called integration and testing, abbreviation I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test to plan to those aggregates, and delivers as its output the integrated system ready for system testing. The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items.

Validation testing

In software project management, software testing and software engineering, validation is the process of checking that a software system meets specifications and that it fulfills its intended purpose. The errors which are uncovered during integration testing are connected during this phase
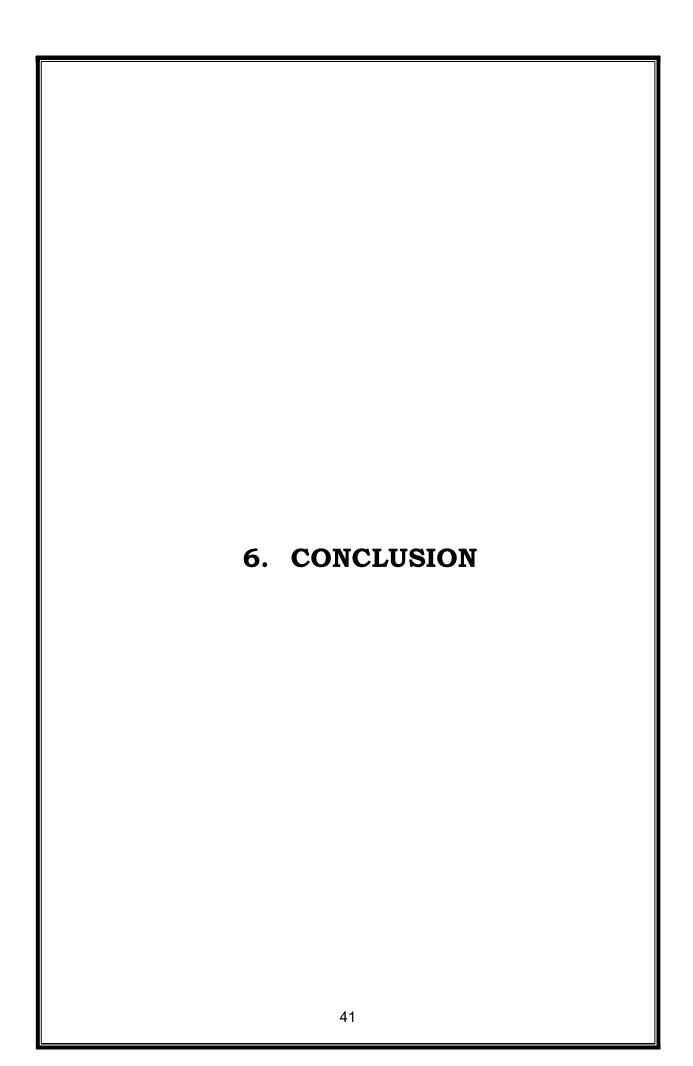
Output testing

No system could be useful if it does not produce the required output in the specific format. Output testing is performed to ensure the correctness of the output and its format. The output generated or displayed by the system is tested asking the users about the format required by them.

5.2. System implementation

Implementation is the stage in the project where theoretical design is turned into a working system and is giving confidence on the new system for the users which will work efficiently. It involves careful planning, investigation of the current system and it's constrains on implementations, design of methods to achieve the changeover, an evolution, of change over methods. Apart from planning major tasks of preparing the implementation coordination committee based of the system. According to this are to be carried out discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. Implementation is the final and important phase. The most critical stage in achieving a successful new system and giving the users confidence that the new system will work and be efficient. The system can be implement only after thorough testing is done and if found to working according to the specification.
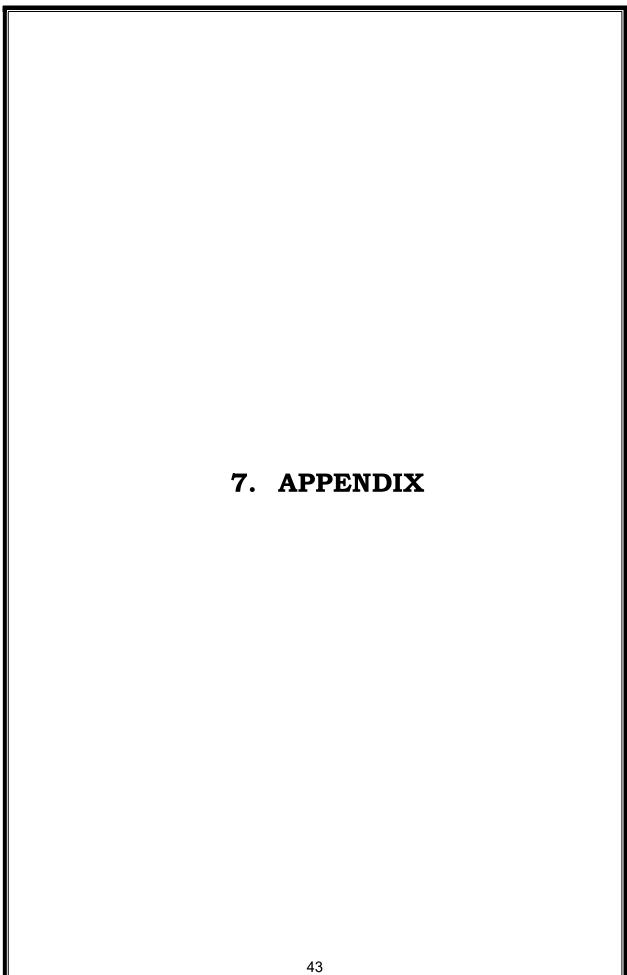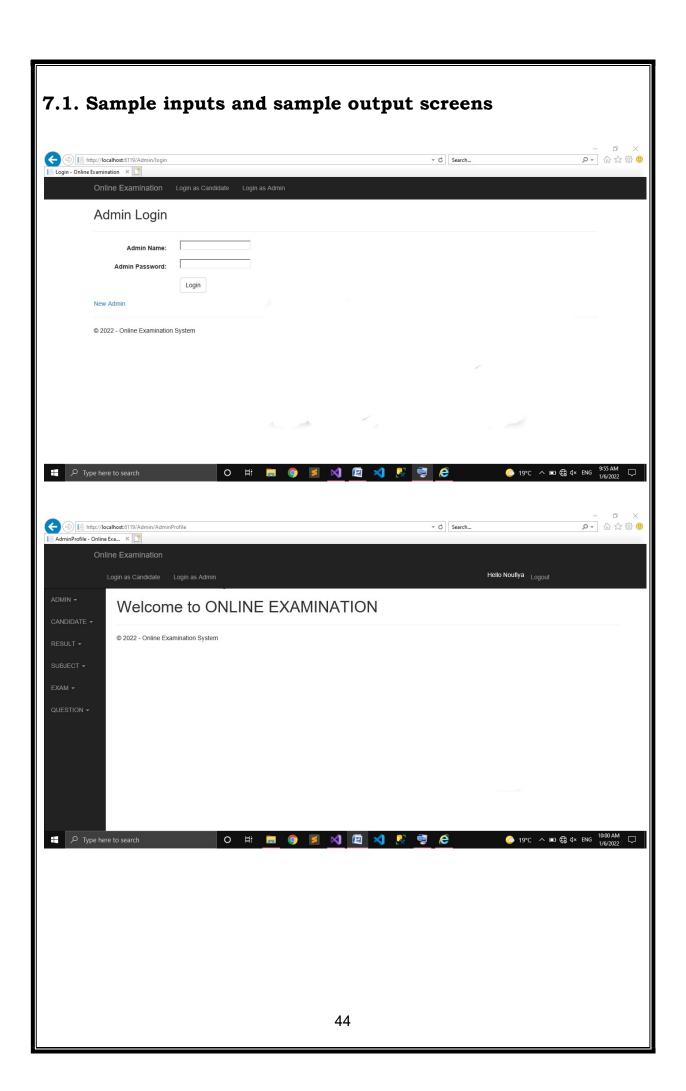
5.3. Security

In any organizations data is the most important element and the main issue related to it is the security of those valuable data. One of the major areas in development process of a system is providing security to all its data in an efficient way. In my work, as it is for an institution it is tightly protected by password system. Only the admin and user can use it with userid and password. The database server is equipped with efficient password security system. So the entire system is provided with tight security and I am sure that data should not be hacked by an unauthorized person
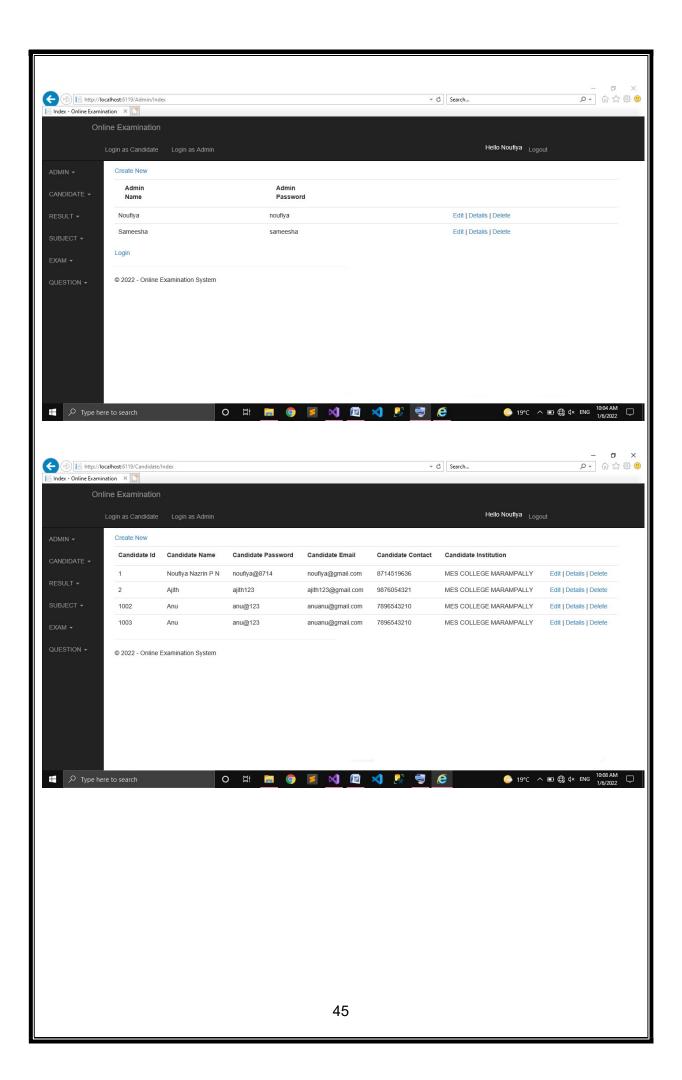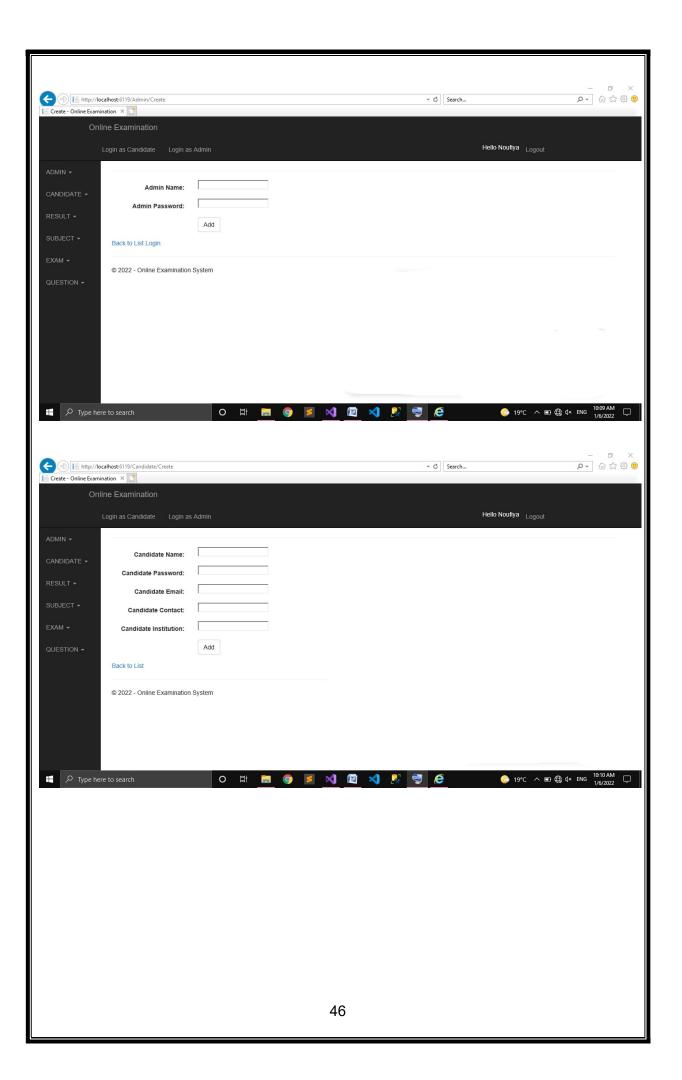
6. CONCLUSION

In this new world, Computers have made their presence in each field. Computer simplified procedures in almost all fields. In my project, I provide almost all facilities for entering the details of the employee, which is presently done manually. The facilities include adding of new user in the database, and other necessary details like finding salary, printing salary, removing employee, updating details etc. The system also provides a platform to perform fetching employee details and so on.

This system reduces the effort and saves time and has a user friendly platform for the user. Users can efficiently use the system by accepting the appropriate warning messages and they can ensure that the details are stored into the database by reading the messages provided by the system also provide by the system. I have tried my level best to make a perfect user friendly system and I'm sure that it is with less percentage of errors. I expect the implementation of this system will provide them an active working environment and can reduce the manpower. I expect implementing my system will improve the efficiency of the institutions for managing employees.
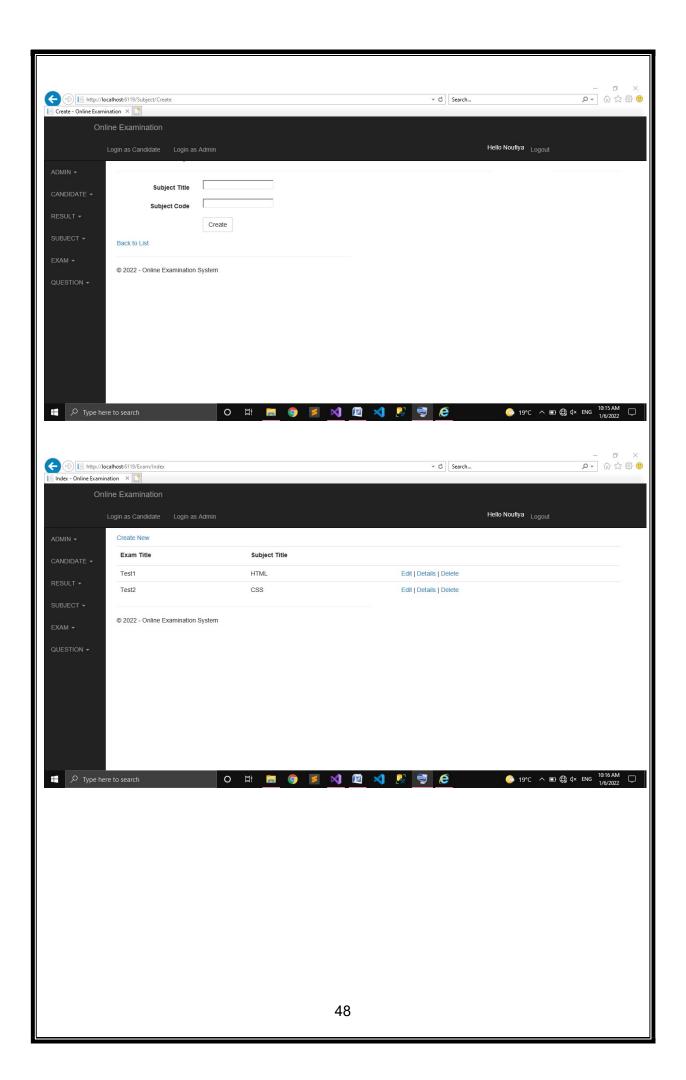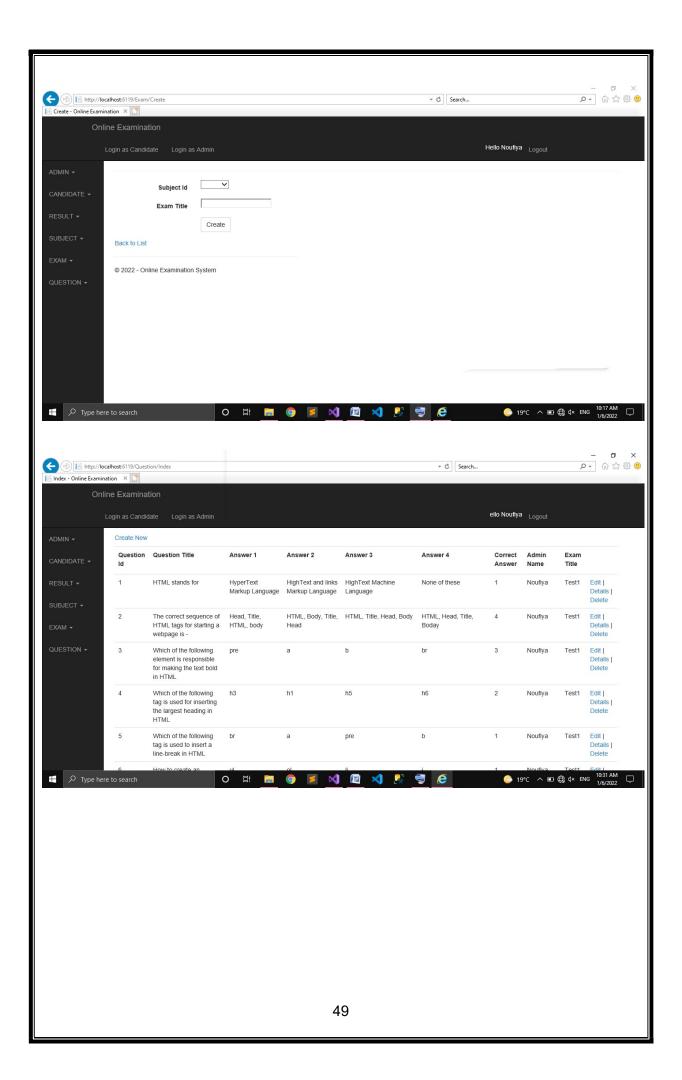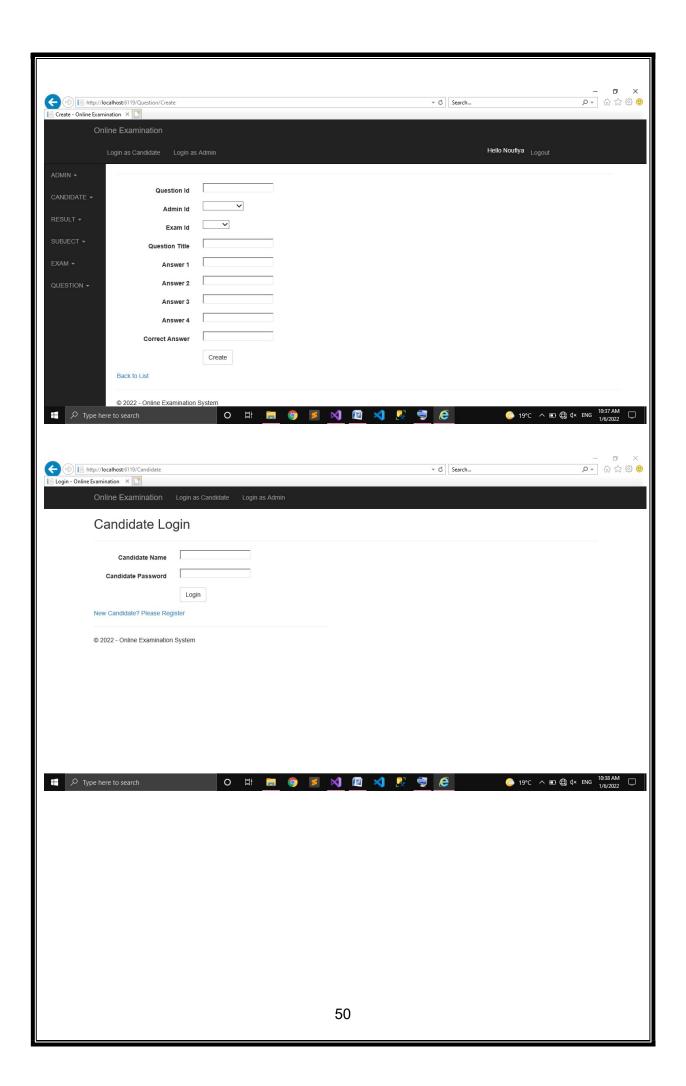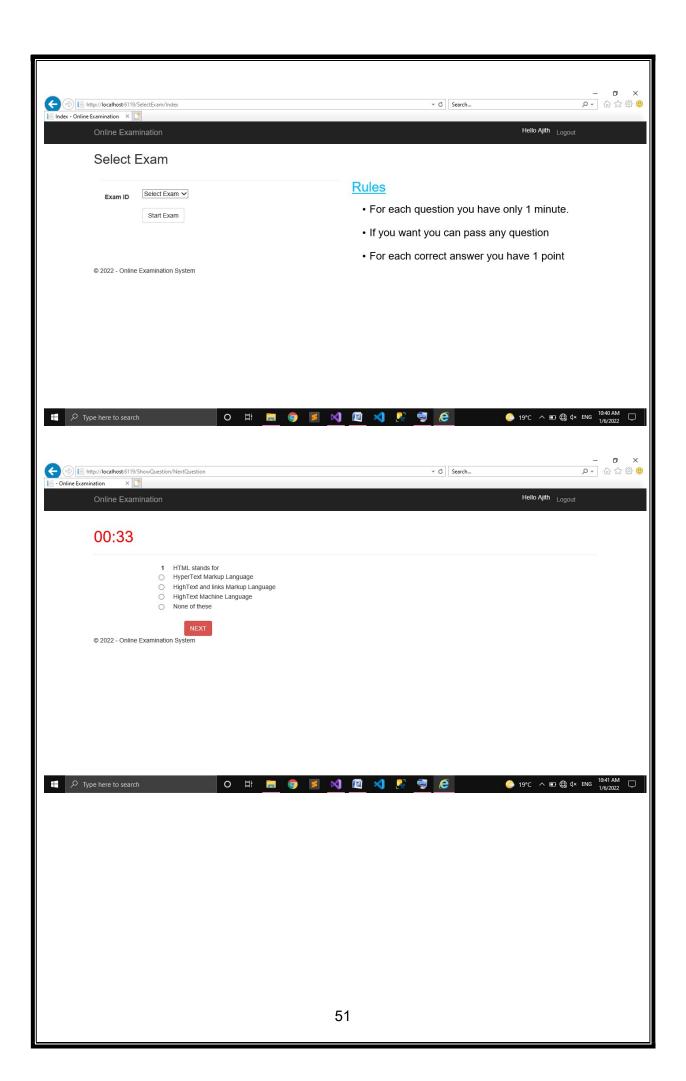
7. APPENDIX

7.1. Sample inputs and sample output screens

8. BIBLIOLOGY

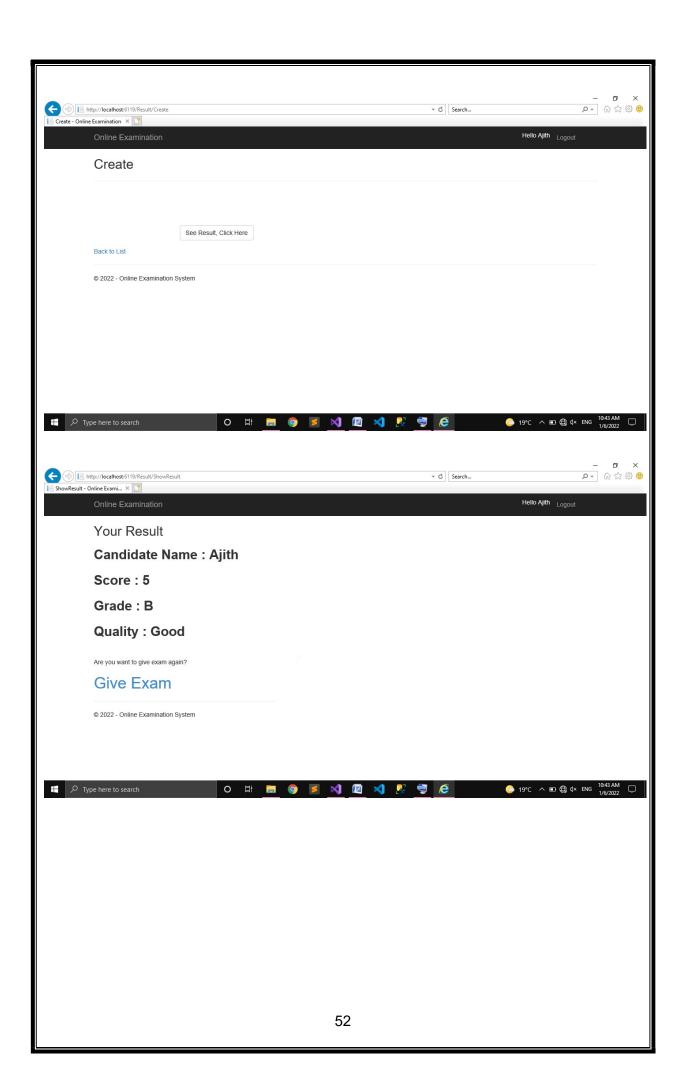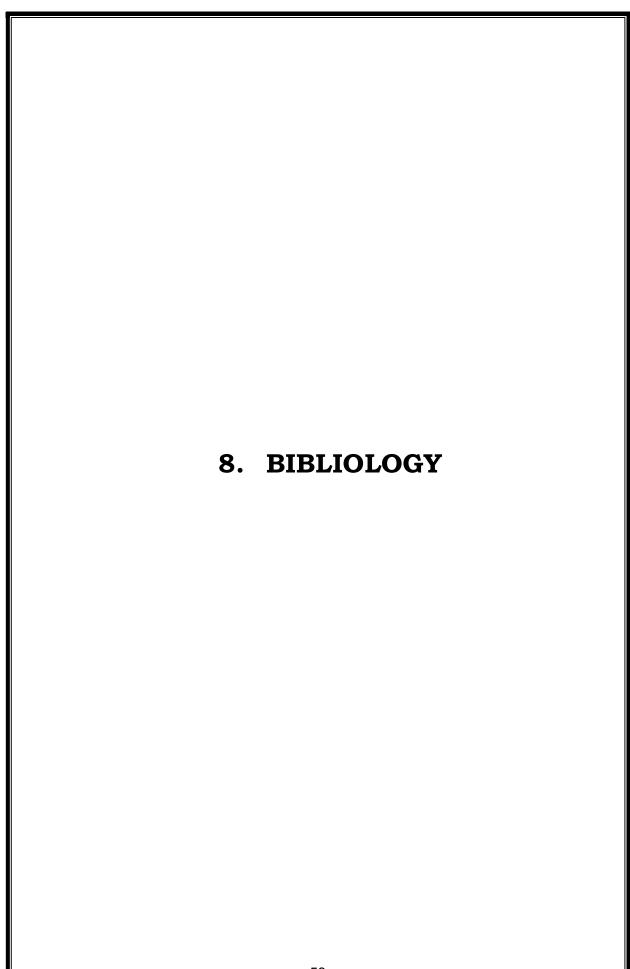**Electronic Materials**

MVC

https://www.tutorialsteacher.com/mvc/mvc-architecture

https://www.c-sharpcorner.com/UploadFile/18585c/mvc-architecture-mvc-life-cycle/

https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm

YouTube

Books Referred

Adam Freeman -Pro ASP.NET MVC 5, 5th ed. Edition, Apress.

Ian Griffiths- Programming C# 5.0, Shroff Publishers & Distributers Pvt. Ltd.