Chapter 1

Project Overview

1.1 INTRODUCTION

A design of suitable automated system for generating question papers and managing related data may prove vital in an Educational Institute. In this paper, we have proposed an integrated automated system that stores questions related to a particular course and prints a question paper based on its syllabus and curriculum. We have implemented a role-based hierarchy which restricts access to the users. The system also deploys security mechanisms that prohibit duplication of question papers. There are provisions to enter and edit data suitable to any educational organization with complete freedom for specifying courses, semesters, syllabus and pattern. This enables an educational institute to generate question ensuring security and non-repetitiveness of question papers and is a boon for organizations with limited staff and resources. Our system aims to provide fast operations, data storage and high security for all its tasks. The evolution of traditional and existing Question Paper Generation systems and the need for an automated system is unraveled. We have proposed our revised system of Automated Question Paper Generation.

1.2 SCOPE AND OBJECTIVES

Free from tension of collecting question as Built in ready to use question bank is available. One can make may sets of paper from one database. This software assures no duplicity of questions in database. Question paper once generated can be downloaded or emailed to the admin who has logged in. this software provides well-arranged syllabus of related subjects in chapters or modules. Three level of difficulty of questions are provided i.e. Easy, Medium, and Difficult. Question paper is generated with great ease and accuracy in less than a minute. Keeps record of generated question paper. Question paper can be generated and viewed only by admin and password security is provided. Fully customization of test paper institute name, time, instructions and maximum marks. This software provides unique user authentication facilities. Database backup and restore backup facilities is also provided.

1.3 MODULES AND THEIR DESCRIPTION

This system is having 7 Modules:

1.   Admin Login

    2. Question Insertion

    3. Difficulty Choosing

    4. Random Paper generation

    5. Wide Chapter Coverage

    6. Doc File Creation

    7. Emailing

Description:

- Admin Login:

  Admin would be having a login account. He can add questions in the system.

- Add Question:

  User may insert questions as per weight age, difficulty, subject and chapter in the system.

- Difficulty Choosing:

  User may then choose the paper difficulty level.

- Random Paper generation:

  The system now automatically chooses random questions as per the selected difficulty level.

- Wide Chapter Coverage:

  The system tries to cover as many chapters as possible for that subject.

- Doc File Creation:

  The system now exports the well formatted question paper in a doc file.

- Emailing:

  The file is now emailed to various users i.e. college/exam centers just minutes before the exam.

1.4 EXISTING SYSTEM & PROPOSED SYSTEM

Problem with current scenario

1. Traditionally, there was no such system that would easily generate a question paper by just inserting set of question papers to the system.
2. In existing system, university use to generate question papers and distribute the question papers to respective colleges manually.
3. There might be chances of paper leak due to existing manual system.
4. The system is quite inefficient because question paper may not reach the respective colleges on time.

1.5 LIMITATION OF PAPER-BASED SYSTEMS

As most human working processes, this system suffers due to bias. There might be some questions which are repeated in many question papers as the professor has a personal inclination towards them. So there is no guarantee of pure randomly generated question paper. Other problems that may plague this system are non-availability of staff and resources, natural calamities and accidents. Also, the security of the system can be easily compromised if leverage over the person responsible for generating question papers is obtained. Other limitations include: -

- Lack of storage space
- Prone to damage
- Inefficient document transportation
- Supply costs
- Poor environmental credentials
- Limited collaboration
- Editing problems

1.6 PROPOSED SYSTEM

- To overcome the existing system anomalies, this question paper generator system is developed.

- In this system we present a smart question paper generating system for universities.

- It is made to allow universities to generate question papers with random but even questions to cover most chapters of subject with difficulty level within seconds and mail them to colleges instantly.

- In our system we allow administrator to input a set of questions and respective answers for option ticking.

- We also allow admin to provide weightage and complexity for each of thesequestions.

- After this the questions are stored in database along with their weightage.

- Now on question paper generating time the admin just has to select the percentage of difficulty.

- On this selection the system selects questions randomly in a way that their weightage makes up for 100 marks and according to difficulty that admin chooses the questions are chosen based on their complexity level.

- The questions are also added for various difficulty levels so that as soon admin chooses the type of paper difficulty (Easy, medium, difficult) the system automatically generates paper, prepares doc file as per selected paper format.

- Also emails it to other colleges. After this question paper is converted to pdf file and emailed to colleges on button click.

Features:

- Avoid repetition of questions.
- Create variants or versions of a particular question or the whole question paper.
- Easy to adapt and use, intuitive interface.
- In-built question bank so the software can be used immediately after installation.

Benefits:

- Limited or unlimited no. of question papers.
- Minimizes the dependency upon teachers, DTP staff, etc.
- Create partially or fully different versions of the same question paper.
- Saves Time, Money, Headache so that you can focus on other critical task.
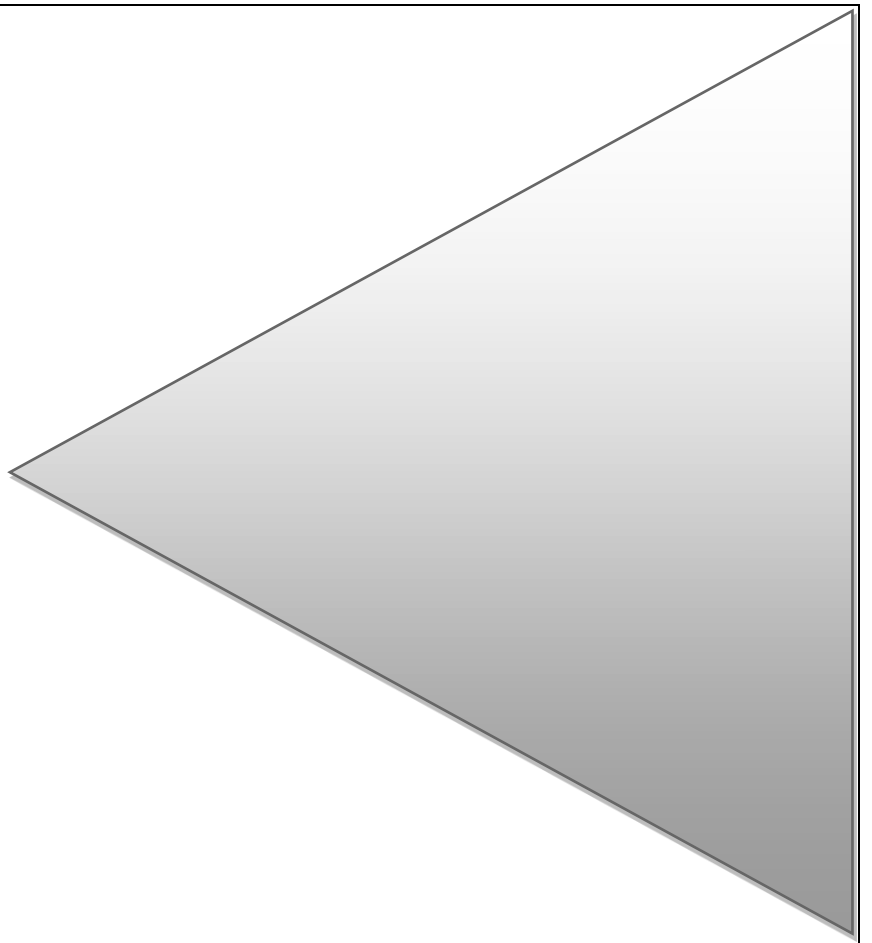
Chapter 2

Project Analysis

2.1 Gantt Chart
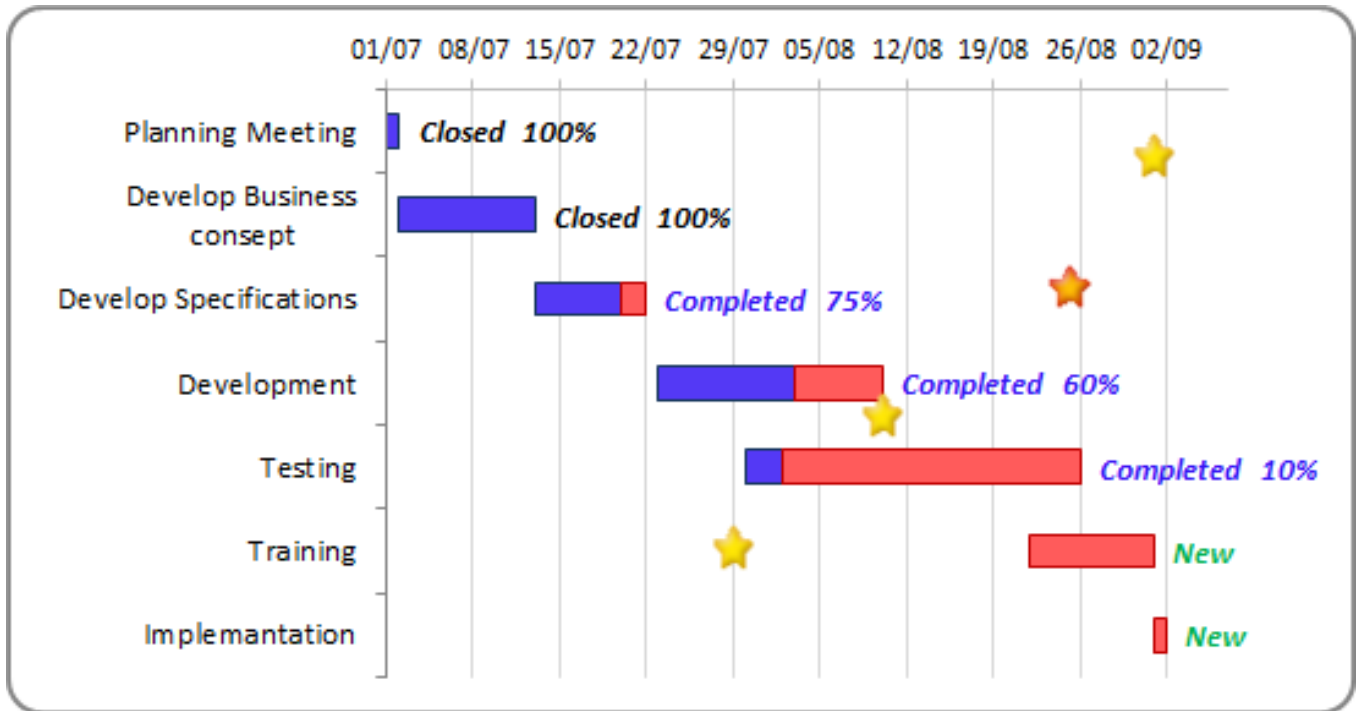
Fig 2.1 Gantt Chart
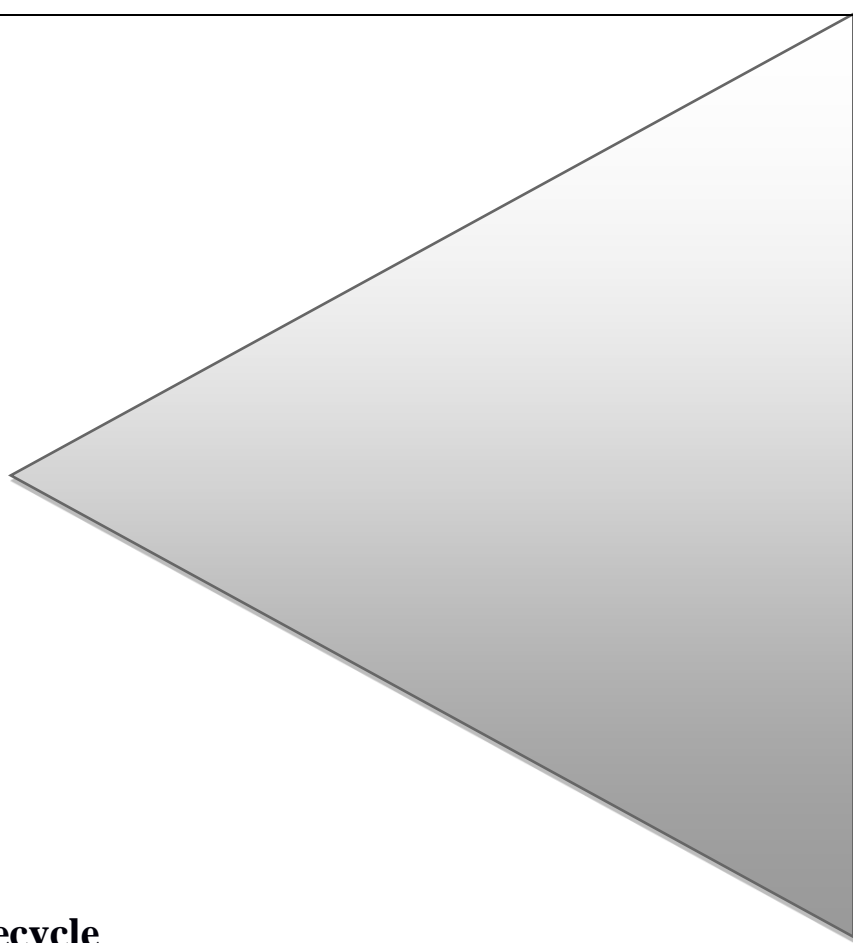
Chapter 3

Project Lifecycle

3.1 Project Lifecycle Details
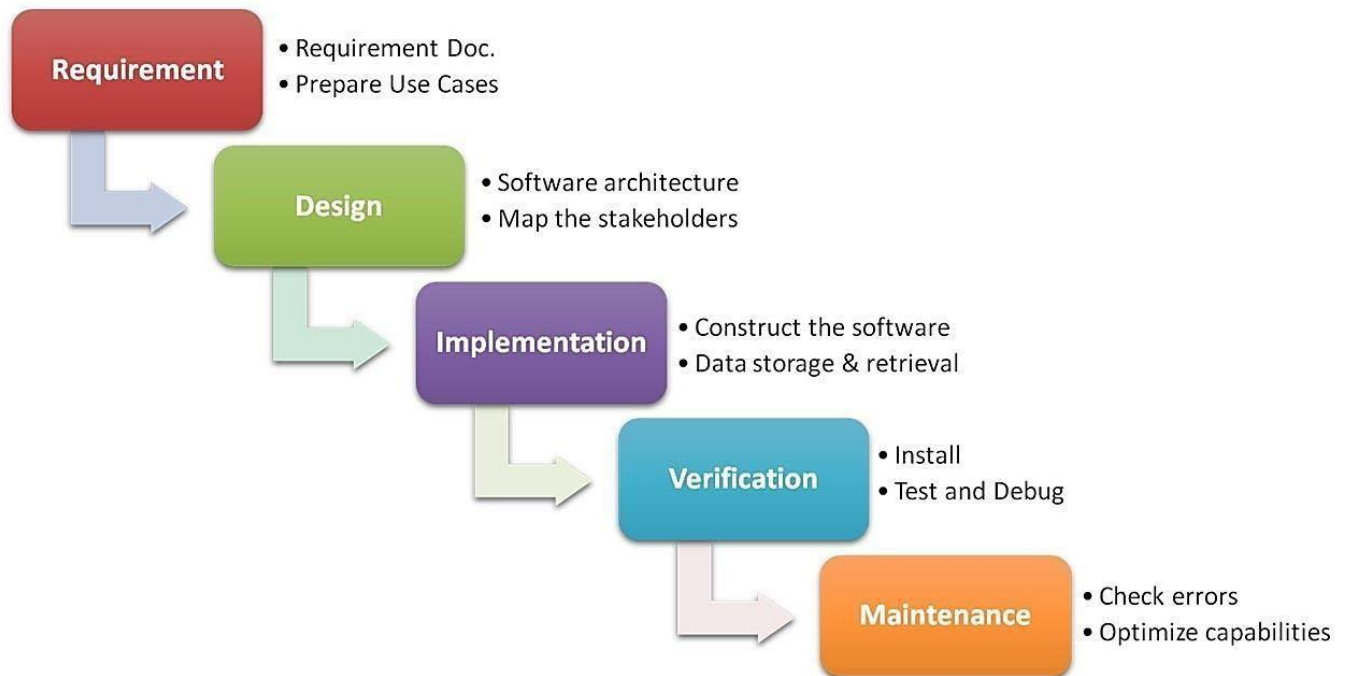
Waterfall Model

Fig 3.1 Waterfall Model

Description

The waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirement. The waterfall approach is the earliest approach that was used for software development.
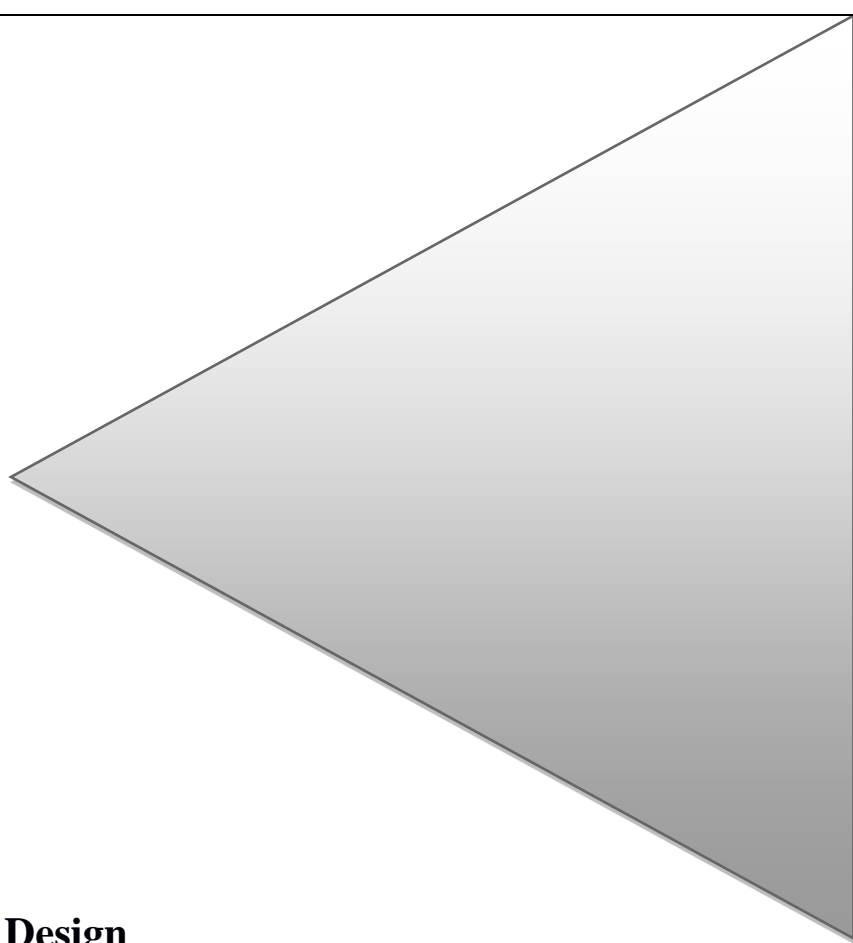
Chapter 4
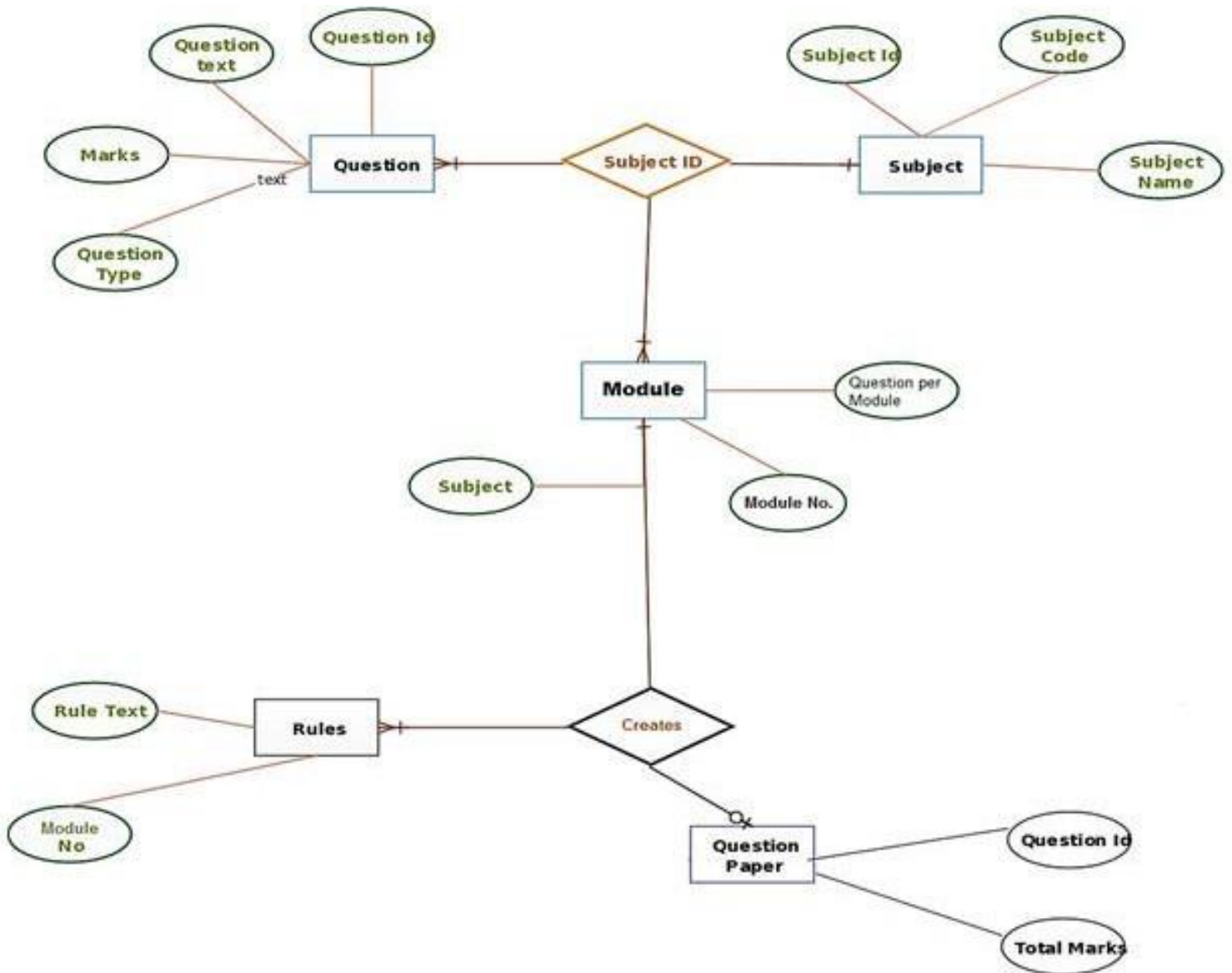
Project Design

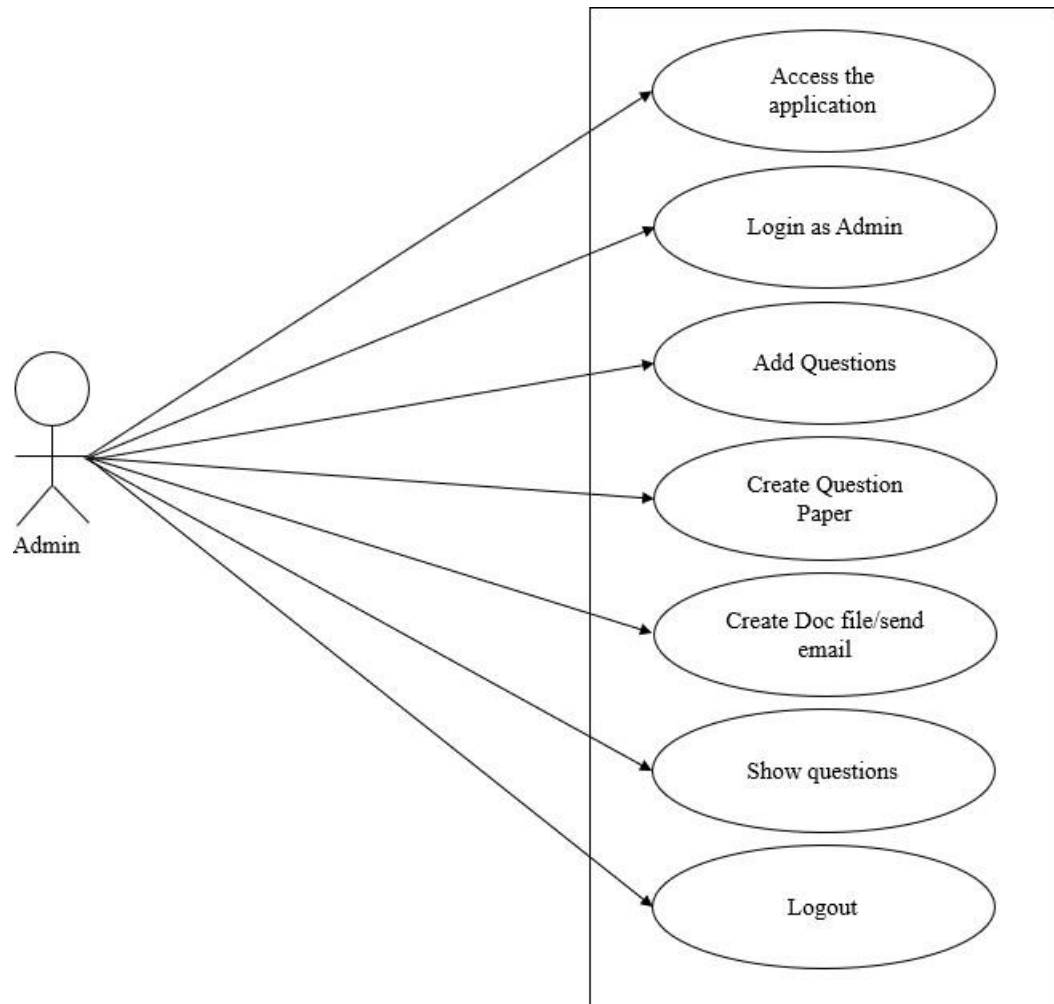## 4.1 E-R Diagram:

Fig 4.1 E-R Diagram
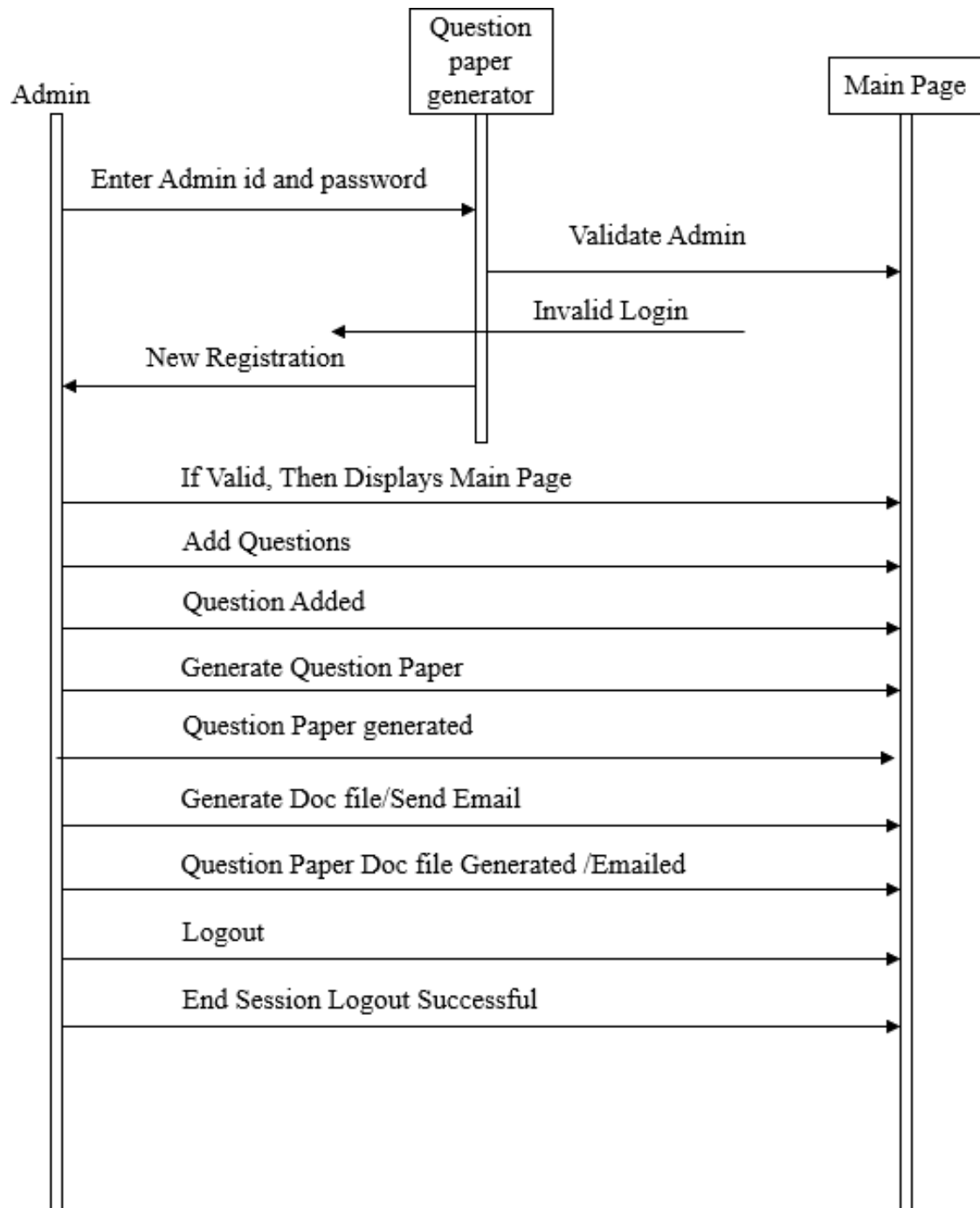
4.2 Use Case Diagram

Fig 4.2 Use Case Diagram

4.3 Sequence Diagram

Fig 4.3 Sequence Diagram
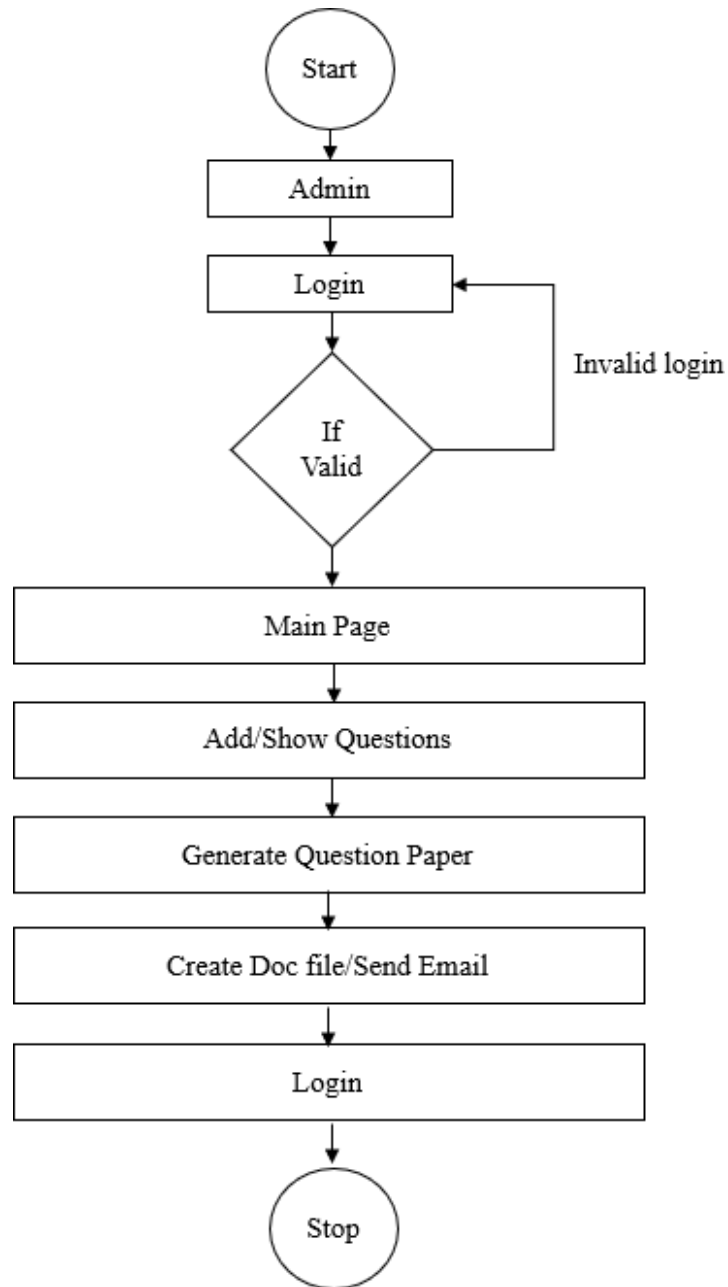
4.4 Activity Diagram

Fig 4.4 Activity Diagram

Class Diagram:

ADMIN

- Admin_id: String
- Password: String

+ Login()
+ btn_Click ()
+ Logout ()

Question Paper

- Subject   : String
- QP Code   : String
- Time   : Int
- Marks   : String
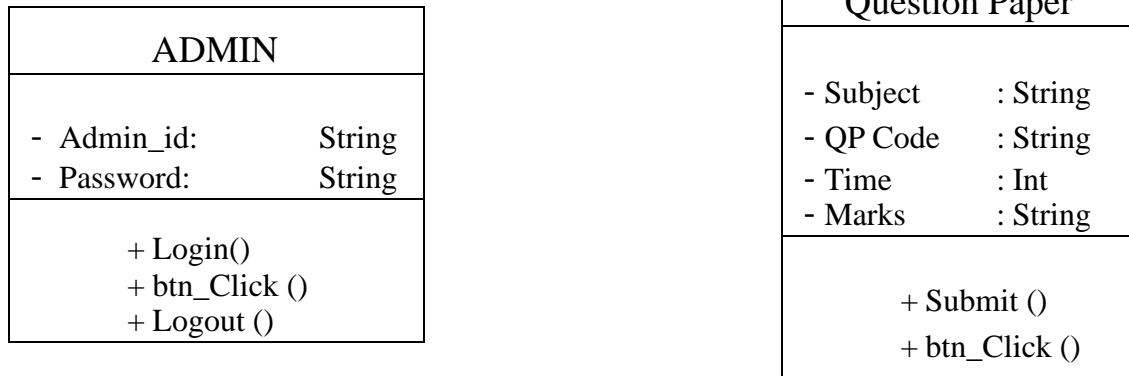
+ Submit ()
+ btn_Click ()

Fig 4.5 Class Diagram

4.5 Data Flow Diagram

A data flow diagram is graphical tool used to describe and analyze movement of data through a system. These are the central tool and the basis from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow diagrams. The physical data flow diagrams show the actual implements and movement of data between people, departments and workstations. A full description of a system actually consists of a set of data flow diagrams. Using two familiar notations Yourdon, Gane and Sarson notation develops the data flow diagrams. Each component in a DFD is labeled with a descriptive name. Process is further identified with a number that will be used for identification purpose. The development of DFD's is done in several levels. Each process in lower level diagrams can be broken down into a more detailed DFD in the next level. The lop-level diagram is often called context diagram. It consists a single process bit, which plays vital role in studying the current system. The process in the context level diagram is exploded into other process at the first level DFD.

The idea behind the explosion of a process into more process is that understanding at one level of detail is exploded into greater detail at the next level. This is done until further explosion is necessary and an adequate amount of detail is described for analyst to understand the process.
Larry Constantine first developed the DFD as a way of expressing system requirements in a graphical from, this led to the modular design.

A DFD is also known as a "bubble Chart" has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So, it is the starting point of the design to the lowest level of detail. A DFD consists of a series of bubbles joined by data flows in the system.

DFD SYMBOLS:

In the DFD, there are four symbols

1. A square defines a source(originator) or destination of system data

2. An arrow identifies data flow. It is the pipeline through which the information flows

3. A circle or a bubble represents a process that transforms incoming data flow into outgoing data flows.

4. An open rectangle is a data store, data at rest or a temporary repository of data

CONSTRUCTING A DFD:

Several rules of thumb are used in drawing DFD's:

- Process should be named and numbered for an easy reference. Each name should be representative of the process.

- The direction of flow is from top to bottom and from left to right. Data traditionally flow from source to the destination although they may flow back to the source. One way to indicate this is to draw long flow line back to a source. An alternative way is to repeat the source symbol as a destination. Since it is used more than once in the DFD it is marked with a short diagonal.

- When a process is exploded into lower level details, they are numbered.

- The names of data stores and destinations are written in capital letters. Process and dataflow names have the first letter of each work capitalized

- A DFD typically shows the minimum contents of data store.   Each data store should contain all the data elements that flow in and out.

- Questionnaires should contain all the data elements that flow in and out. Missing interfaces redundancies and like is then accounted for often through interviews.

SAILENT FEATURES OF DFD's

- The DFD shows flow of data, not of control loops and decision are controlled considerations do not appear on a DFD.

- The DFD does not indicate the time factor involved in any process whether the data flows take place daily, weekly, monthly or yearly.

- The sequence of events is not brought out on the DFD

TYPES OF DATA FLOW DIAGRAMS:

- Current Physical
- Current Logical
- New Logical
- New Physical

CURRENT PHYSICAL:

In Current Physical DFD process label include the name of people or their positions or the names of computer systems that might provide some of the overall system-processing label includes an identification of the technology used to process the data. Similarly, data flows and data stores are often labels with the names of the actual physical media on which data are stored such as file folders, computer files, business forms or computer tapes.

CURRENT LOGICAL:

The physical aspects at the system are removed as much as possible so that the current system is reduced to its essence to the data and the processors that transform them regardless of actual physical form.

NEW LOGICAL:

This is exactly like a current logical model if the user were completely happy with the user were completely happy with the functionality of the current system but had problems with how it was implemented typically through the new logical model will differ from current logical model while having additional functions, absolute function removal and inefficient flows recognized.

NEW PHYSICAL:
The new physical represents only the physical implementation of the new system.

RULES GOVERNING THE DFD'S

- No process can have only outputs.
- No process can have only inputs. If an object has only inputs than it must be asink.
- A process has a verb phrase label.

DATA STORE:

Data cannot move directly from one data store to another data store, a process must move data.

Data cannot move directly from an outside source to a data store, a process, which receives, must move data from the source and place the data into data store

A data store has a noun phrase label.

SOURCE OR SINK

The origin and /or destination of data.

Data cannot move direly from a source to sink it must be moved by a process

A source and /or sink has a noun phrase land

DATA FLOW

A Data Flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The later it usually indicated however by two separate arrows since these happen at different type.

A join in DFD means that exactly the same data comes from any of two or more different processes data store or sink to a common location.

A data flow cannot go directly back to the same process it leads. There must be at least one other process that handles the data flow produce some other data flow returns the original data into the beginning process.

A Data flow to a data store means update (delete or change).

A data Flow from a data store means retrieve or use.
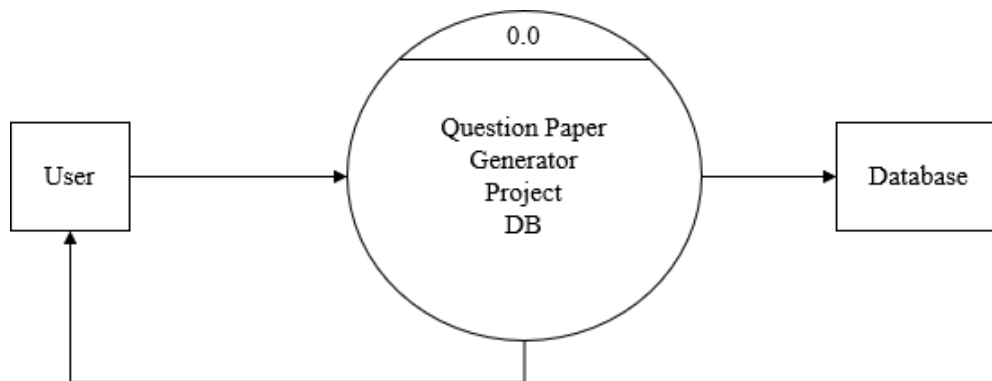
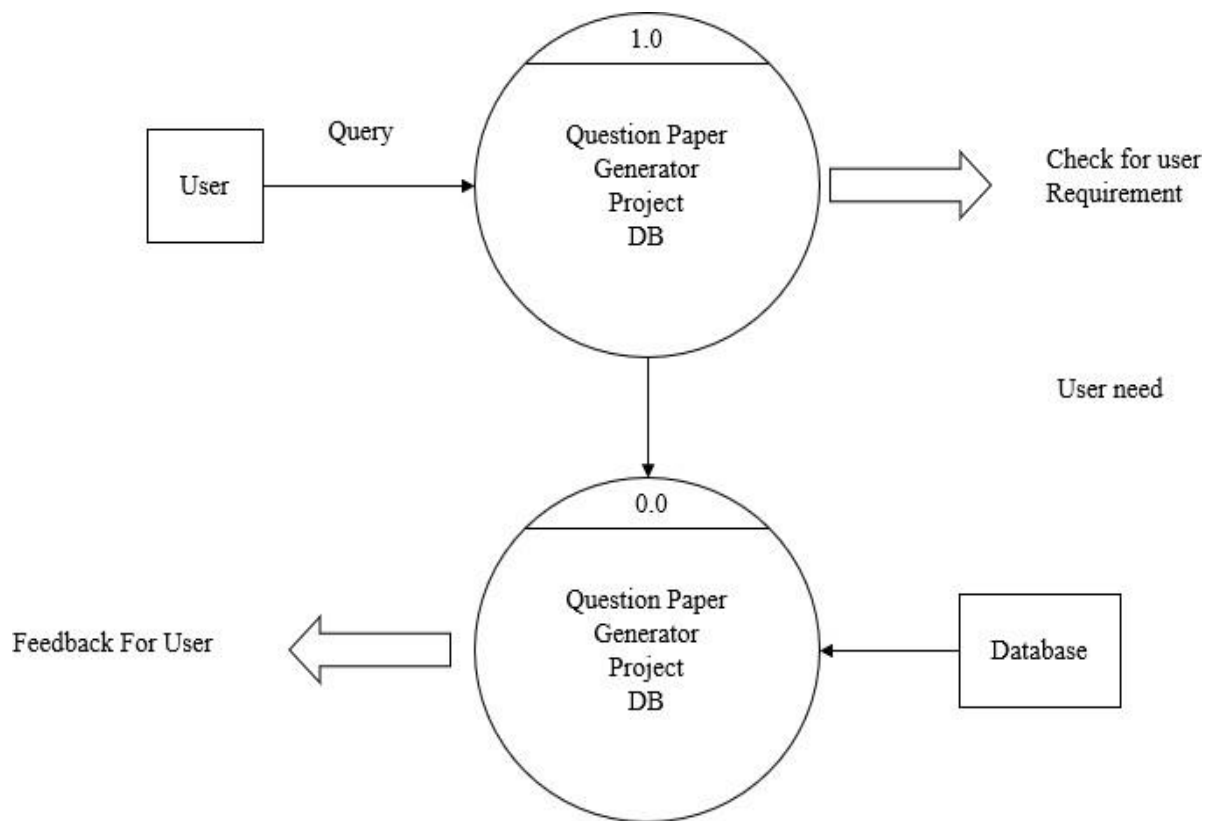**DATABASE DETAIL**

Fig 4.6 DATABASE DETAIL
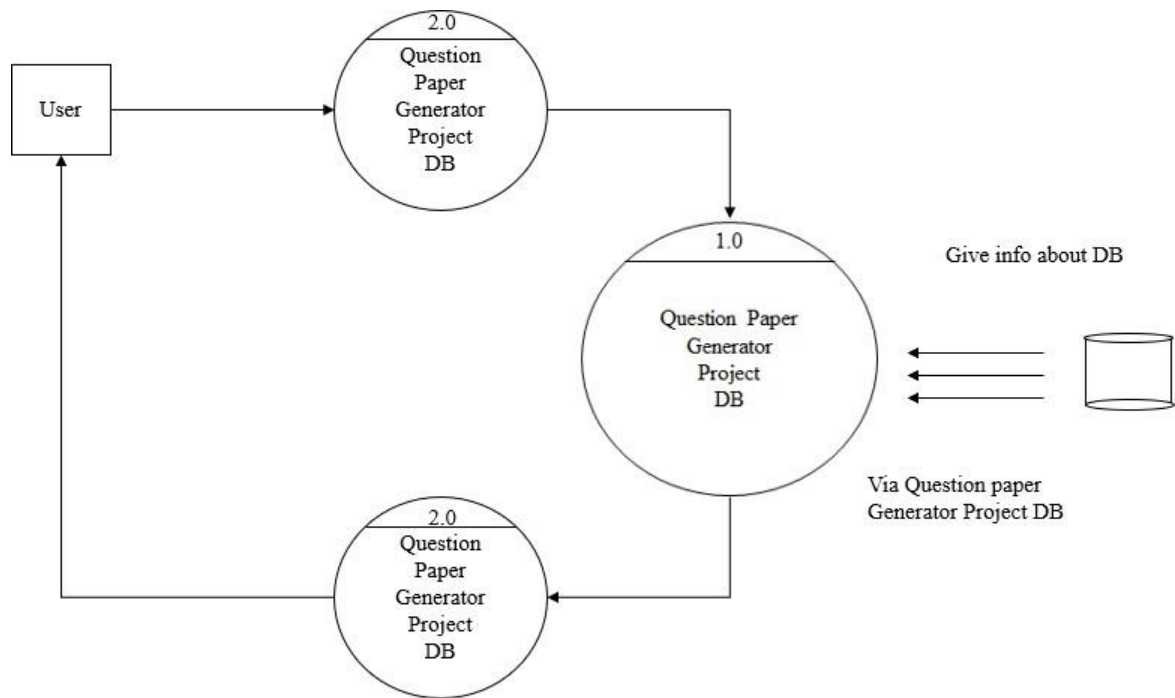
DATA FLOW DIAGRAMS

Fig 4.7 LEVEL 1 DFD

Fig 4.8 LEVEL 2 DFD
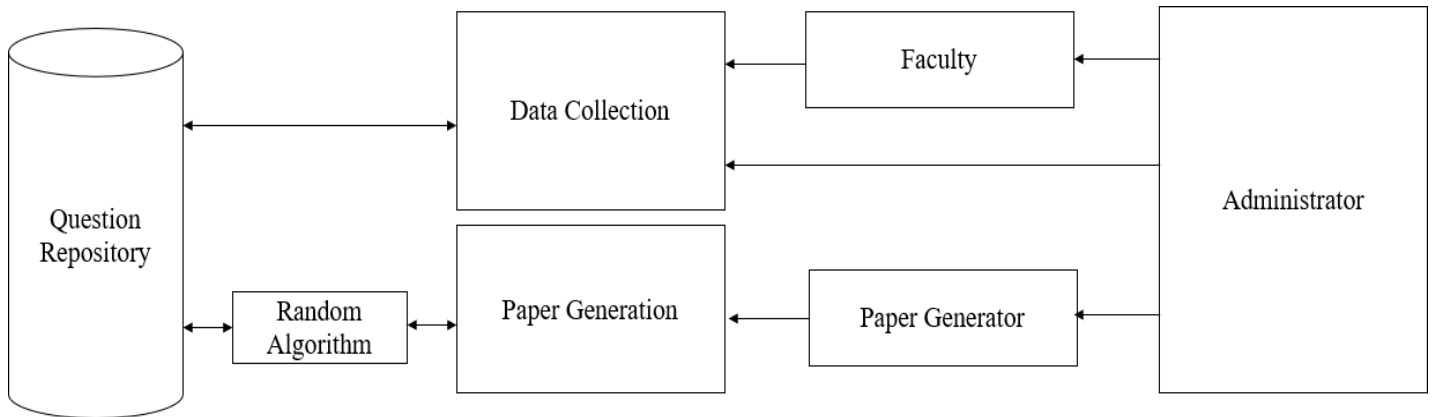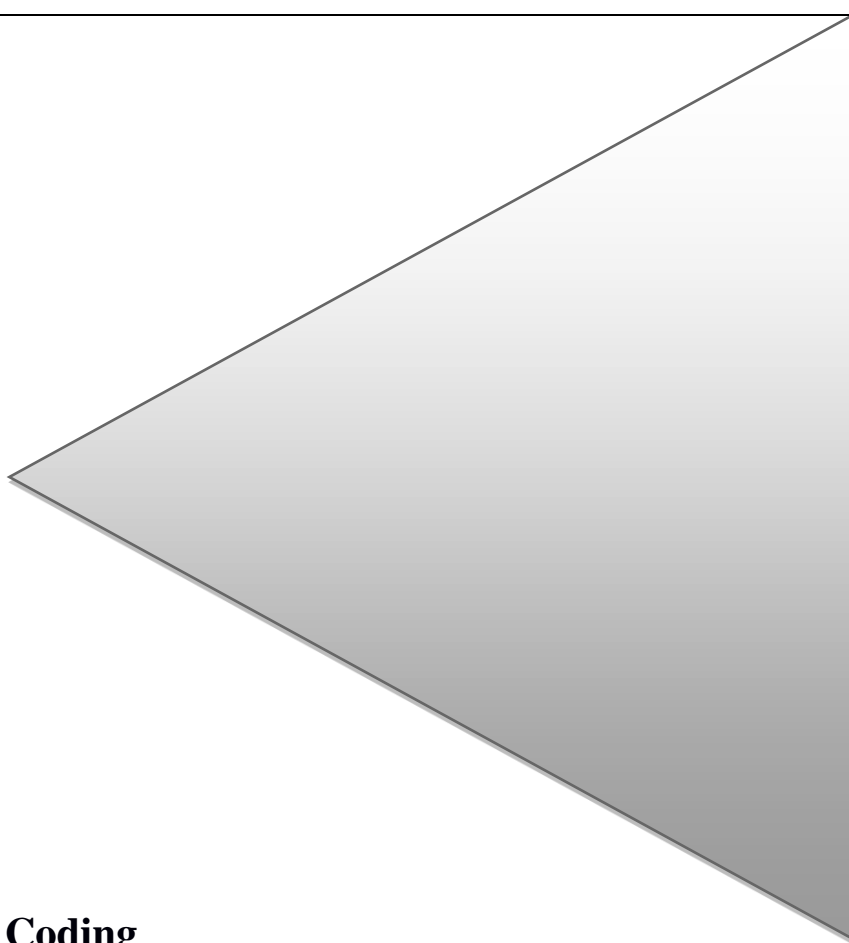
4.6 SYSTEM ARCHITECTURE

Fig 4.9 System Architecture

Chapter 5

Project Coding

5.1 CODING

MAIN QUESTION GENERATION PAGE CODES:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Text;
//using Microsoft.Office.Interop.Word;
using System.Diagnostics;
using System.IO;
using System.Data.SqlClient;
using System.Data;
using System.Net;
using System.Net.Mail;
using System.Configuration;
using System.Web.Security;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;


public partial class Create : System.Web.UI.Page
{
    public SqlConnection con = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\workspace\Question Paper Generator
system\Question\App_Data\Question.mdf;Integrated Security=True");
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            if (Session["name"] != null)
            {
                string sel = "select distinct(courseid), coursename from Course";
                con.Open();
                SqlDataAdapter da = new SqlDataAdapter(sel, con);
                DataSet ds = new DataSet();
                da.Fill(ds);
                int count1 = ds.Tables[0].Rows.Count;
                if (count1 > 0)
                {
                    for (int i = 0; i < count1; i++)
                    {
```

```
                string courseid = ds.Tables[0].Rows[i][0].ToString();
                string coursename = ds.Tables[0].Rows[i][1].ToString();
                course.Items.Add(new ListItem(coursename, courseid));
            }
        }
        con.Close();

    }
    else
    {
        Response.Redirect("Default.aspx");
    }

    }
}


protected void course_SelectedIndexChanged(object sender, EventArgs e)
{
    error.Visible = false;
    error.Text = "";
    subject.Items.Clear();
    string courseid = course.Text;
    string sel = "select subject from Course where courseid='" + courseid + "'";
    con.Open();
    SqlDataAdapter da = new SqlDataAdapter(sel, con);
    DataSet ds = new DataSet();
    da.Fill(ds);
    int count1 = ds.Tables[0].Rows.Count;
    if (count1 > 0)
    {
        for (int i = 0; i < count1; i++)
        {
            string subjectname = ds.Tables[0].Rows[i][0].ToString();
            subject.Items.Add(new ListItem(subjectname, subjectname));
        }
    }
}

protected void subject_SelectedIndexChanged(object sender, EventArgs e)
{
    error.Visible = false;
    error.Text = "";
```

```csharp
    }

    protected void download_Click(object sender, EventArgs e)
    {
        string sel = "select coursename from Course where courseid='"+course.Text+"'";
        con.Open();
        SqlDataAdapter da = new SqlDataAdapter(sel, con);
        DataSet ds = new DataSet();
        da.Fill(ds);
        string cname = ds.Tables[0].Rows[0][0].ToString();
        con.Close();

        string date = DateTime.Now.ToString("yyyy-MM-dd");
        string text = "";

 /* For UNIT Test */
        if(DropDownList1.Text == "1")
        {
            string errorText = "<span style='font-size: 20px; font-weight: bold;'>Question Paper can not be
generated due to following reasons:</span><br/>";
            int errorCount = 0;
            text += "<p align='center'><strong> MGM College of Engineering & Technology</strong></p>";
            text += "<p align='center'><strong> Navi Mumbai, Kamothe </strong></p>";
            text += "<p align = 'center'><u> TEST </u></p> ";
            text += "<table class='table' align ='right' style='width: 46%'><tbody><tr><td style='text-align:
right;'>Enrol No.</td><td colspan='2' style='border:2px solid black; width:
60%'></td></tr></tbody></table><br/> ";
            text += "<p><strong> Course : " + cname + "</strong ></p>";
            text += "<p><strong> Subject : Code: " + subject.Text + "[" + TextBox3.Text + "]</strong></p> ";
            text += "<p><strong> Duration : " + DropDownList7.Text + "</strong ></p> ";
            text += "<table class='table' style='width: 100%'><tbody><tr><td style='width: 80%;text-align:
left;'><strong> Total Marks:  30[Max.]</strong></td><td colspan = '2' style='width: 20%; text-align:
right;'><strong>Date:" + date + " </strong></td></tr></tbody></table><br/> ";
            text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.1) Attempt any five of the following.</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (10) </strong></td></tr></tbody></table> ";

            string fsel = "select top 6 * from Ques where Difficulty='" + difficulty.Text + "' and marks = '2' and
Course = '" + course.Text + "' and Subject = '" + subject.Text + "' Order by NEWID();";
            con.Open();
            SqlDataAdapter da1 = new SqlDataAdapter(fsel, con);
            DataSet ds1 = new DataSet();
            da1.Fill(ds1);
            con.Close();
            int count = ds1.Tables[0].Rows.Count;
            if (count > 0 && count == 6)
```

```
{
    int alpha = 097;
    for (int i = 0; i < count; i++)
    {
        char a = Convert.ToChar(alpha);
        string ques = ds1.Tables[0].Rows[i][2].ToString();
        text += "<p>" + a + ") " + ques + "</p>";
        alpha++;
    }
}
else
{
    errorText += "Kindly add more questions for " + subject.Text + " in " + difficulty.Text + " Level
for 2 marks<br/>";
    errorCount++;
}

string ssel = "select top 2 * from Ques where Difficulty='" + difficulty.Text + "' and marks = '4' and
Course = '" + course.Text + "' and Subject = '" + subject.Text + "' Order by NEWID();";
con.Open();
SqlDataAdapter da2 = new SqlDataAdapter(ssel, con);
DataSet ds2 = new DataSet();
da2.Fill(ds2);
con.Close();
int count1 = ds2.Tables[0].Rows.Count;
if (count1 > 0 && count1 == 2)
{
    string ques = ds2.Tables[0].Rows[0][2].ToString();
    text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.2) a) " + ques + "</strong></td><td colspan = '2' style='width: 20%; text-align:
right;'><strong> (04) </strong></td></tr></tbody></table> ";
    text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong>       b) Attempt any one of the following.
</strong></td><td colspan = '2' style='width: 20%; text-align: right;'><strong> (06)
</strong></td></tr></tbody></table> ";
}
else
{
    errorText += "Kindly add more questions for " + subject.Text + " in " + difficulty.Text + " Level
for 4 marks<br/>";
    errorCount++;
}

string tsel = "select top 4 * from Ques where Difficulty='" + difficulty.Text + "' and marks = '6' and
Course = '" + course.Text + "' and Subject = '" + subject.Text + "' Order by NEWID();";
con.Open();
```

```
            SqlDataAdapter da3 = new SqlDataAdapter(tsel, con);
            DataSet ds3 = new DataSet();
            da3.Fill(ds3);
            int count2 = ds3.Tables[0].Rows.Count;
            if (count2 > 0 && count2 == 4)
            {
                string ques1 = ds3.Tables[0].Rows[0][2].ToString();
                string ques2 = ds3.Tables[0].Rows[1][2].ToString();
                text +=
"<p>             &nbs
p;1) " + ques1 + "</p>";
                text +=
"<p>             &nbs
p;2) " + ques2 + "</p>";
                string ques3 = ds2.Tables[0].Rows[1][2].ToString();
                text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.3) a) " + ques3 + "</strong></td><td colspan = '2' style='width: 20%; text-align:
right;'><strong> (04) </strong></td></tr></tbody></table> ";
                text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong>       b) Attempt any one of the following.
</strong></td><td colspan = '2' style='width: 20%; text-align: right;'><strong> (06)
</strong></td></tr></tbody></table> ";
                string ques4 = ds3.Tables[0].Rows[2][2].ToString();
                string ques5 = ds3.Tables[0].Rows[3][2].ToString();
                text +=
"<p>             &nbs
p;1) " + ques4 + "</p>";
                text +=
"<p>             &nbs
p;2) " + ques5 + "</p>";
            }
            else
            {
                errorText += "Kindly add more questions for " + subject.Text + " in " + difficulty.Text + " Level
for 6 marks<br/>";
                errorCount++;
            }
            con.Close();

            if (errorCount != 0)
            {
                error.Visible = true;
                error.Text = errorText;
            }
            else
            {
```

```csharp
            string s = "select top 1 QId From QLog ORDER BY QId Desc";
            con.Open();
            string qid = "";
            SqlCommand cmd = new SqlCommand(s, con);
            object c = cmd.ExecuteScalar();
            if (c != null)
            {
                int i = Convert.ToInt32(c);
                i++;
                qid = i.ToString();
            }
            else
            {
                qid = "101";
            }
            con.Close();

            string ins = "insert into QLog (QId, QName, QCode, Date, Course, Subject) Values ('" + qid + "',
'" + TextBox2.Text + "', '" + TextBox3.Text + "', '" + date + "', '" + cname + "', '" + subject.Text + "')";
            SqlCommand cm = new SqlCommand(ins, con);
            con.Open();
            cm.ExecuteNonQuery();
            con.Close();

            var strBody = new StringBuilder();
            AddFormatting(strBody, text);
            DownloadWord(strBody.ToString());
        }

    }

/* For Semester */
    else
    {
        semester();
    }
  }

  private void semester()
  {
    string sel = "select coursename from Course where courseid='" + course.Text + "'";
    con.Open();
    SqlDataAdapter da = new SqlDataAdapter(sel, con);
    DataSet ds = new DataSet();
    da.Fill(ds);
```

```csharp
        string cname = ds.Tables[0].Rows[0][0].ToString();
        con.Close();

        string date = DateTime.Now.ToString("yyyy-MM-dd");
        string text = "";

        string errorText = "<span style='font-size: 20px; font-weight: bold;'>Question Paper can not be
generated due to following reasons:</span><br/>";
    int errorCount = 0;

        text += "<table class='table' style='width: 100%'><tbody><tr><td style='text-align: left;'>MGM
College of Engineering & Technology</td><td style='text-align: right;'>Enrol No.</td><td colspan='2'
style='border:2px solid black; width: 15%'></td></tr>";
        text += "<tr><td style='text-align: left;'>NAVI-MUMBAI KAMOTHE  </td><td style='text-align:
left;'><strong><u> WINTER  2022 </u></strong></td><td
colspan='2'></td></tr></tbody></table><br/>";
        text += "<p align='center'><strong> SEMESTER EXAMINATION IN "+cname+" </strong></p>";
        text += "<table class='table' style='width: 100%'><tbody><tr><td style='text-align: left;'>Time
Allowed: " + DropDownList7.Text + "</td><td style='text-align: right;'>Max Marks</td><td colspan='2'
style='width: 15%'>100</td></tr></tbody></table><br/>";
        text += "<p align='center'><strong> SUBJECT & CODE : "+subject.Text+" ["+TextBox3.Text+"]
</strong></p>";
        text += "<p align='left'><strong>INSTRUCTIONS</strong></p>";
        text += "<p>1) Instruction 1</p><p>2) Instruction 2</p><p>3) Instruction 3</p><p>4) Instruction
4</p> ";

 /*section1 */
        text += "<p align='center'><strong><u>SECTION I</u></strong></p>";

        text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.1) Attempt any six of the following.</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (18) </strong></td></tr></tbody></table> ";


        string fsel = "select top 16 * from Ques where Difficulty='" + difficulty.Text + "' and marks = '3' and
Course = '" + course.Text + "' and Subject = '" + subject.Text + "' Order by NEWID();";
    con.Open();
    SqlDataAdapter da1 = new SqlDataAdapter(fsel, con);
    DataSet ds1 = new DataSet();
    da1.Fill(ds1);
    con.Close();
    int count = ds1.Tables[0].Rows.Count;
    string section21 = "";
    string section22 = "";
    string section23 = "";
    string section24 = "";
```

```csharp
        if (count > 0 && count == 16)
        {

            int alpha = 097;
            for (int i = 0; i < 8; i++)
            {
                char a = Convert.ToChar(alpha);
                string ques = ds1.Tables[0].Rows[i][2].ToString();
                text += "<p>" + a + ") " + ques + "</p>";
                alpha++;
            }
            int alpha2 = 097;
            for (int i = 8; i < 16; i++)
            {
                char a = Convert.ToChar(alpha2);
                string ques = ds1.Tables[0].Rows[i][2].ToString();
                section21 += "<p>" + a + ") " + ques + "</p>";
                alpha2++;
            }
        }
        else
        {
            errorText += "Kindly add more questions for " + subject.Text + " in " + difficulty.Text + " Level
for 3 marks<br/>";
            errorCount++;
        }

        text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.2) Attempt any four of the following.</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (16) </strong></td></tr></tbody></table> ";
        string sell = "select top 14 * from Ques where Difficulty='" + difficulty.Text + "' and marks = '4' and
Course = '" + course.Text + "' and Subject = '" + subject.Text + "' Order by NEWID();";
        con.Open();
        SqlDataAdapter daa = new SqlDataAdapter(sell, con);
        DataSet dss = new DataSet();
        daa.Fill(dss);
        con.Close();
        int countt = dss.Tables[0].Rows.Count;
        if (countt > 0 && countt == 14)
        {
            int alpha = 097;
            for (int i = 0; i < 5; i++)
            {
                char a = Convert.ToChar(alpha);
                string ques = dss.Tables[0].Rows[i][2].ToString();
                text += "<p>" + a + ") " + ques + "</p>";
```

```
            alpha++;
        }
        text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.3) (A) Attempt the following</strong></td><td colspan = '2' style='width: 20%; text-
align: right;'><strong> (08) </strong></td></tr></tbody></table> ";
        int alpha1 = 097;
        for (int i = 5; i < 7; i++)
        {
            char a = Convert.ToChar(alpha1);
            string ques = dss.Tables[0].Rows[i][2].ToString();
            text += "<p>" + a + ") " + ques + "</p>";
            alpha1++;
        }
        int alpha3 = 097;
        for (int i = 7; i < 12; i++)
        {
            char a = Convert.ToChar(alpha3);
            string ques = dss.Tables[0].Rows[i][2].ToString();
            section22 += "<p>" + a + ") " + ques + "</p>";
            alpha3++;
        }
        int alpha4 = 097;
        for (int i = 12; i < 14; i++)
        {
            char a = Convert.ToChar(alpha4);
            string ques = dss.Tables[0].Rows[i][2].ToString();
            section23 += "<p>" + a + ") " + ques + "</p>";
            alpha4++;
        }
    }
    else
    {
        errorText += "Kindly add more questions for " + subject.Text + " in " + difficulty.Text + " Level
for 4 marks<br/>";
        errorCount++;
    }


    text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> (B) Attempt any one of the following</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (08) </strong></td></tr></tbody></table> ";
    string sel2 = "select top 4 * from Ques where Difficulty='" + difficulty.Text + "' and marks = '8' and
Course = '" + course.Text + "' and Subject = '" + subject.Text + "' Order by NEWID();";
    con.Open();
    SqlDataAdapter da2 = new SqlDataAdapter(sel2, con);
```

```csharp
        DataSet ds2 = new DataSet();
        da2.Fill(ds2);
        con.Close();
        int count2 = ds2.Tables[0].Rows.Count;
        if (count2 > 0 && count2 == 4)
        {
          int alpha = 097;
          for (int i = 0; i < 2; i++)
          {
            char a = Convert.ToChar(alpha);
            string ques = ds2.Tables[0].Rows[i][2].ToString();
            text += "<p>" + a + ") " + ques + "</p>";
            alpha++;
          }
          int alpha5 = 097;
          for (int i = 2; i < 4; i++)
          {
            char a = Convert.ToChar(alpha5);
            string ques = ds2.Tables[0].Rows[i][2].ToString();
            section24 += "<p>" + a + ") " + ques + "</p>";
            alpha5++;
          }
        }
        else
        {
          errorText += "Kindly add more questions for " + subject.Text + " in " + difficulty.Text + " Level
for 8 marks<br/>";
          errorCount++;
        }

/*section2 */
      text += "<p align='center'><strong><u>SECTION II</u></strong></p>";
      text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.1) Attempt any six of the following.</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (18) </strong></td></tr></tbody></table> ";
      text += section21;

      text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.2) Attempt any four of the following.</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (16) </strong></td></tr></tbody></table> ";
      text += section22;

      text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> Q.3) (A) Attempt the following</strong></td><td colspan = '2' style='width: 20%; text-
align: right;'><strong> (08) </strong></td></tr></tbody></table> ";
```

```csharp
text += section23;

text += "<table class='table' style='width: 100%'><tbody><tr><td style = 'width: 80%;text-align:
left;'><strong> (B) Attempt any one of the following</strong></td><td colspan = '2' style='width: 20%;
text-align: right;'><strong> (08) </strong></td></tr></tbody></table> ";
text += section24;


if (errorCount != 0)
{
    error.Visible = true;
    error.Text = errorText;
}
else
{
    string s = "select top 1 QId From QLog ORDER BY QId Desc";
    con.Open();
    string qid = "";
    SqlCommand cmd = new SqlCommand(s, con);
    object c = cmd.ExecuteScalar();
    if (c != null)
    {
        int i = Convert.ToInt32(c);
        i++;
        qid = i.ToString();
    }
    else
    {
        qid = "101";
    }
    con.Close();
    string ins = "insert into QLog (QId, QName, QCode, Date, Course, Subject) Values ('" + qid + "', '"
+ TextBox2.Text + "', '" + TextBox3.Text + "', '" + date + "', '" + cname + "', '" + subject.Text + "')";
    SqlCommand cm = new SqlCommand(ins, con);
    con.Open();
    cm.ExecuteNonQuery();
    con.Close();
    var strBody = new StringBuilder();
    AddFormatting(strBody, text);
    DownloadWord(strBody.ToString());

}


}
```

```
private void DownloadWord(string strBody)
{
    HttpContext.Current.Response.Clear();
    HttpContext.Current.Response.Charset = "";
    HttpContext.Current.Response.ContentType = "application/msword";
    string strFileName = TextBox2.Text + " " + TextBox3.Text + ".doc";
    HttpContext.Current.Response.AddHeader("Content-Disposition", "inline;filename=" + strFileName);
    HttpContext.Current.Response.Write(strBody);
    HttpContext.Current.Response.End();
    HttpContext.Current.Response.Flush();
}

private void AddFormatting(StringBuilder strBody, string yourHtmlContent)
{
    strBody.Append("<html " +
        "xmlns:o='urn:schemas-microsoft-com:office:office' " +
        "xmlns:w='urn:schemas-microsoft-com:office:word'" +
        "xmlns='http://www.w3.org/TR/REC-html40'>" +
        "<head><title>Time</title>");

    //The setting specifies document's view after it is downloaded as Print
    //instead of the default Web Layout
    strBody.Append("<!--[if gte mso 9]>" +
        "<xml>" +
        "<w:WordDocument>" +
        "<w:View>Print</w:View>" +
        "<w:Zoom>90</w:Zoom>" +
        "<w:DoNotOptimizeForBrowser/>" +
        "</w:WordDocument>" +
        "</xml>" +
        "<![endif]-->");

    strBody.Append("<style>" +
        "<!-- /* Style Definitions */" +
        "@page Section1" +
        "   {size:8.5in 11.0in; " +
        "   margin:1.0in 1.25in 1.0in 1.25in ; " +
        "   mso-header-margin:.5in; " +
        "   mso-footer-margin:.5in; mso-paper-source:0;}" +
        " div.Section1" +
        "   {page:Section1;}" +
        "-->" +
        "</style></head>");

    strBody.Append("<body lang=EN-US style='tab-interval:.5in'>" +
        "<div class=Section1>");
```

```csharp
            strBody.Append(yourHtmlContent);
            strBody.Append("</div></body></html>");
    }

    protected void mail_Click(object sender, EventArgs e)
    {
        try
        {
            MailMessage mail = new MailMessage();
            SmtpClient SmtpServer = new SmtpClient("smtp.gmail.com");
            mail.From = new MailAddress("parthshah111999@gmail.com");
            mail.To.Add(TextBox7.Text);
            mail.Subject = "New Question Paper";
            mail.Body = "Question Paper";

            System.Net.Mail.Attachment attachment;
            attachment = new System.Net.Mail.Attachment("C:\\Users\\parth\\Downloads\\" + TextBox2.Text
+ " " + TextBox3.Text + ".doc");
            mail.Attachments.Add(attachment);

            SmtpServer.Port = 587;
            SmtpServer.Credentials = new System.Net.NetworkCredential("parthshah111999@gmail.com",
"parth@1999");
            SmtpServer.EnableSsl = true;

            SmtpServer.Send(mail);
        }
        catch (Exception ex)
        {
            Console.WriteLine(ex.ToString());
        }
    }
}
```
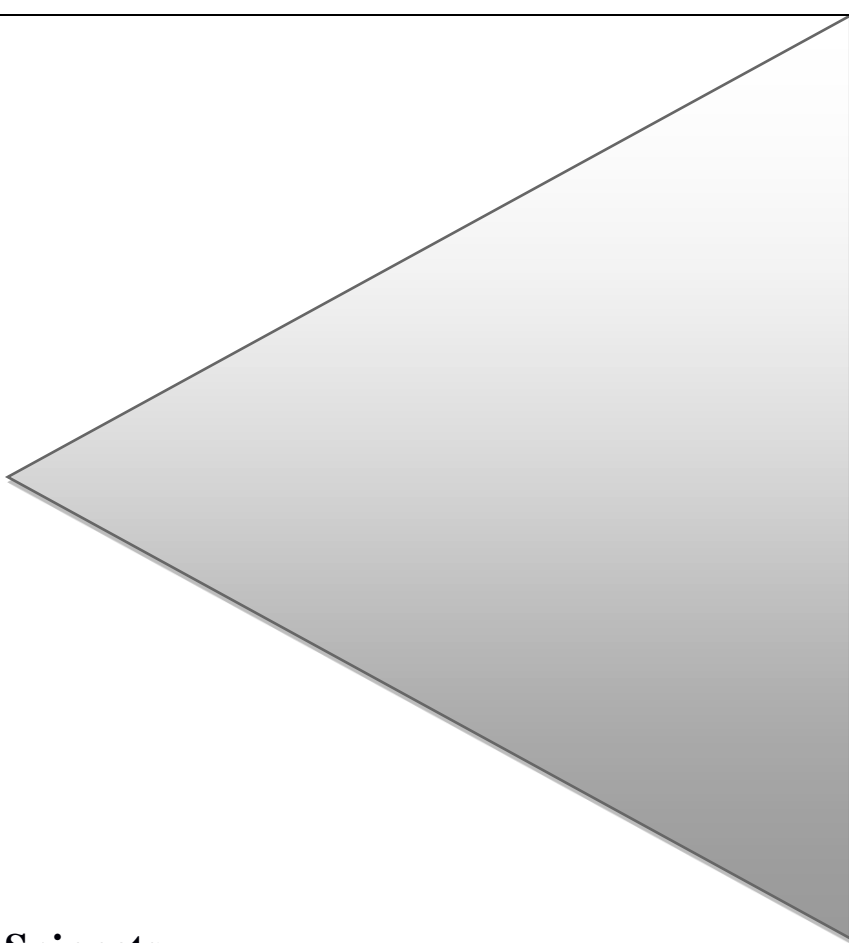
Chapter 6

Project Snippets

1. Main Login:

2. Admin Login:

3. Add Teacher:

4. Add Course & Subject:

5. Get Question Paper:

6. Get Question Paper with Attributes:

7. Downloaded Question Paper:

8. Generated Question Paper:

9. Question Paper:

TEST

Enrol No. [   ]

Course : SY_CE_8

Subject : Code: Advance DataBase[DB123]

Duration : 1

Total Marks: 30[Max.]   Date:2022-04-23

Q.1) Attempt any five of the following.   (10)

a) Give difference between Extended ROWID and Restricted ROWID?

b) Write Syntax to create new table?

c) List various methods used for storing data in Oracle?

d) Define hot and cold backup?

e) List any four features of RMAN?

f) State the use of System pre-defined tablespace and Undo Pre-defined tablespace?

Q.2) a) Explain Different Backup and Recovery Strategy?   (04)
     b) Attempt any one of the following.   (06)

   1) Explain different components of RMAN?

   2) Explain Open and Closed Backup?

Q.3) a) Describe different type of failures in Oracle Environment? (04)
     b) Attempt any one of the following.   (06)

   1) Explain create new tablespace , rename tablespace, change the size of tablespace and alse write sql query for following tablespace?

   2) Eplain different type of indexs? Hoe to Reorganize and drop the index?

10. Question Paper Master Database:

11. Generated MCQ Questions:

12. Adding MCQs:

13. Printing MCQs:

14. Adding Questions to Question Paper Master Database:

15. View Questions:

Chapter 7

Project Implementation

7.1 PROJECT IMPLEMENTATION

Project Implementation Technology:

The Project is loaded in Visual Studio 2010. We used Visual Studio for Design and coding of project. Created and maintained all databases into SQL Server 2008, in that we create tables, write query for store data or record of project.

Hardware Requirement:

- Minimum core i3 Processor Based Computer
- 1GB RAM
- 5 GB Hard Disk
- Internet Connection

Software Requirement:

- Windows 7or higher
- Visual studio 2010
- SQL Server 2008

7.2 PROJECT OVERVIEW

Front End:

Microsoft .NET Framework:

.NET Framework is a technology that supports building and running Windows apps and web services. .NET Framework is designed to fulfill the following objectives:

- Provide a consistent, object-oriented programming environment whether object code is stored and executed locally, executed locally but web-distributed, or executed remotely.

- Provide a code-execution environment that:
    1. Minimizes software deployment and versioning conflicts.
    2. Promotes safe execution of code, including code created by an unknown or semi-trusted third party.
    3. Eliminates the performance problems of scripted or interpreted environments.

- Make the developer experience consistent across widely varying types of apps, such as Windows-based apps and Web-based apps.

- Build all communication on industry standards to ensure that code based on .NET Framework integrates with any other code.

Features of the common language runtime

The common language runtime manages memory, thread execution, code execution, code safety verification, compilation, and other system services. These features are intrinsic to the managed code that runs on the common language runtime.

Regarding security, managed components are awarded varying degrees of trust, depending on a number of factors that include their origin (such as the Internet, enterprise network, or local computer). This means that a managed component might or might not be able to perform file-access operations, registry-access operations, or other sensitive functions, even if it's used in the same active app.

The runtime also enforces code robustness by implementing a strict type-and-code-verification infrastructure called the common type system (CTS). The CTS ensures that all managed code is self-describing. The various Microsoft and third-party language compilers generate managed code that conforms to the CTS. This means that managed code can consume other managed types and instances, while strictly enforcing type fidelity and type safety.

In addition, the managed environment of the runtime eliminates many common software issues. For example, the runtime automatically handles object layout and manages references to objects, releasing them when they are no longer being used. This automatic memory management resolves the two most common app errors, memory leaks and invalid memory references.

The runtime also accelerates developer productivity. For example, programmers write apps in their development language of choice yet take full advantage of the runtime, the class library, and components written in other languages by other developers. Any compiler vendor who chooses to target the runtime can do so. Language compilers that target the .NET Framework make the features of the .NET Framework available to existing code written in that language, greatly easing the migration process for existing apps.

What is ASP.NET Web Forms?

ASP.NET Web Forms is a part of the ASP.NET web application framework and is included with [Visual Studio](#). It is one of the four programming models you can use to create ASP.NET web applications, the others are ASP.NET MVC, ASP.NET Web Pages, and ASP.NET Single Page Applications.

Web Forms are pages that your users request using their browser. These pages can be written using a combination of HTML, client-script, server controls, and server code. When users request a page, it is compiled and executed on the server by the framework, and then the framework generates the HTML markup that the browser can render. An ASP.NET Web Forms page presents information to the user in any browser or client device.

Using Visual Studio, you can create ASP.NET Web Forms. The Visual Studio Integrated Development Environment (IDE) lets you drag and drop server controls to lay out your Web Forms page. You can then easily set properties, methods, and events for controls on the page or for the page itself. These properties, methods, and events are used to define the web page's behavior, look and feel, and so on. To write server code to handle the logic for the page, you can use a .NET language like Visual Basic or C#.

Back End:

Microsoft SQL Server:

Microsoft SQL Server is a Structured Query Language (SQL) based, client/server relational database. Each of these terms describes a fundamental part of the architecture of SQL Server.

Database

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

A database typically has two components: the files holding the physical database and the database management system (DBMS) software that applications use to access data. The DBMS is responsible for enforcing the database structure, including:

Maintaining the relationships between data in the database.

Ensuring that data is stored correctly and that the rules defining data relationships are not violated.

Recovering all data to a point of known consistency in case of system failures.

Relational Database:

There are different ways to organize data in a database but relational databases are one of the most effective. Relational database systems are an application of mathematical set theory to the problem of effectively organizing data. In a relational database, data is collected into tables (called relations in relational theory).

When organizing data into tables, you can usually find many different ways to define tables. Relational database theory defines a process, normalization, which ensures that the set of tables you define will organize your data effectively

Client/Server:

In a client/server system, the server is a relatively large computer in a central location that manages a resource used by many people. When individuals need to use the resource, they connect over the network from their computers, or clients, to the server.

Examples of servers are: In a client/server database architecture, the database files and DBMS software reside on a server. A communications component is provided so applications can run on separate clients and communicate to the database server over a network. The SQL Server communication component also allows communication between an application running on the server and SQL Server.

Server applications are usually capable of working with several clients at the same time. SQL Server can work with thousands of client applications simultaneously. The server has features to prevent the logical problems that occur if a user tries to read or modify data currently being used by others.

While SQL Server is designed to work as a server in a client/server network, it is also capable of working as a stand-alone database directly on the client. The scalability and ease-of- use features of SQL Server allow it to work efficiently on a client without consuming too many resources.

.

7.3 FEASIBILITY REPORT

Feasibility Study's a high-level capsule version of the entire process intended to answer a number of questions like: What is the problem? Is there any feasible solution to the given problem? Is the problem even worth solving? Feasibility study is conducted once the problem clearly understood. Feasibility study is necessary to determine that the proposed system is Feasible by considering the technical, Operational, and Economical factors. By having a detailed feasibility study the management will have a clear-cut view of the proposed system.

The following feasibilities are considered for the project in order to ensure that the project is variable and it does not have any major obstructions. Feasibility study encompasses the following things:

1. Technical Feasibility
2. Economic Feasibility
3. Operational Feasibility

Technical Feasibility:

In this step, we verify whether the proposed systems are technically feasible or not. i.e., all the technologies required to develop the system are available readily or not.

Technical Feasibility determines whether the organization has the technology and skills necessary to carry out the project and how this should be obtained. The system can be feasible because of the following grounds:

All necessary technology exists to develop the system.

This system is too flexible and it can be expanded further.

This system can give guarantees of accuracy, ease of use, reliability and the data security.

This system can give instant response to inquire.

Our project is technically feasible because, all the technology needed for our project is readily available.

Operating System   : Windows 7or higher

Languages   : Asp.Net with C# (.Net 2010)

Database System   : MS-SQL Server 2008

Documentation Tool: MS - Word 2013

Economic Feasibility:

Economically, this project is completely feasible because it requires no extra financial investment and with respect to time, it's completely possible to complete this project in 6 months.

In this step, we verify which proposal is more economical. We compare the financial benefits of the new system with the investment. The new system is economically feasible only when the financial benefits are more than the investments and expenditure. Economic Feasibility determines whether the project goal can be within the resource limits allocated to it or not. It must determine whether it is worthwhile to process with the entire project or whether the benefits obtained from the new system are not worth the costs. Financial benefits must be equal or exceed the costs. In this issue, we should consider:

- The cost to conduct a full system investigation.
- The cost of h/w and s/w for the class of application being considered.
- The development tools.
- The cost of maintenance etc...

Our project is economically feasible because the cost of development is very minimal when compared to financial benefits of the application.

Operational Feasibility:

In this step, we verify different operational factors of the proposed systems like man- power, time etc., whichever solution uses less operational resources, is the best operationally feasible solution. The solution should also be operationally possible to implement. Operational Feasibility determines if the proposed system satisfied user objectives could be fitted into the current system operation.

The methods of processing and presentation are completely accepted by the clients since they can meet all user requirements.

The clients have been involved in the planning and development of the system.

The proposed system will not cause any problem under any circumstances.
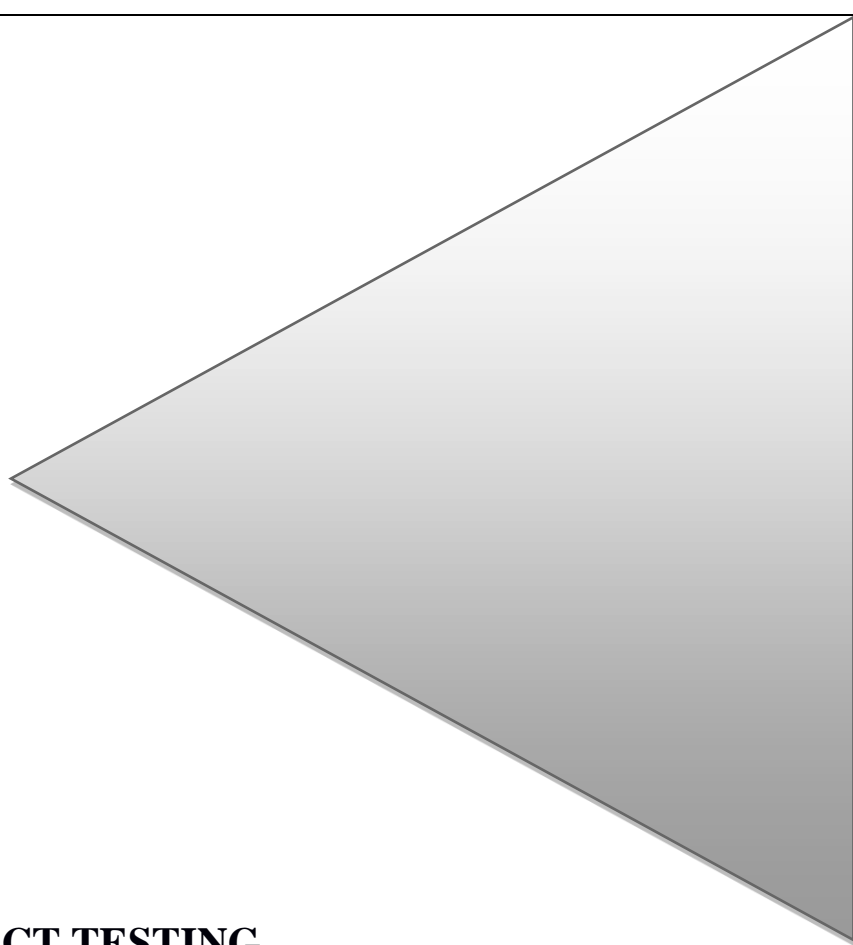
Chapter 8

PROJECT TESTING

8.1 TESTING:

As the project is on bit large scale, we always need testing to make it successful. If each components work properly in all respect and gives desired output for all kind of inputs then project is said to be successful. So the conclusion is-to make the project successful, it needs to be tested.

The testing done here was System Testing checking whether the user requirements were satisfied. The code for the new system has been written completely using ASP .NET with C# as the coding language, C# as the interface for front-end designing. The new system has been tested well with the help of the users and all the applications have been verified from every nook and corner of the user.

Although some applications were found to be erroneous these applications have been corrected before being implemented. The flow of the forms has been found to be very much in accordance with the actual flow of data.

8.2 LEVELS OF TESTING

In order to uncover the errors, present in different phases we have the concept of levels of testing. The basic levels of testing are:

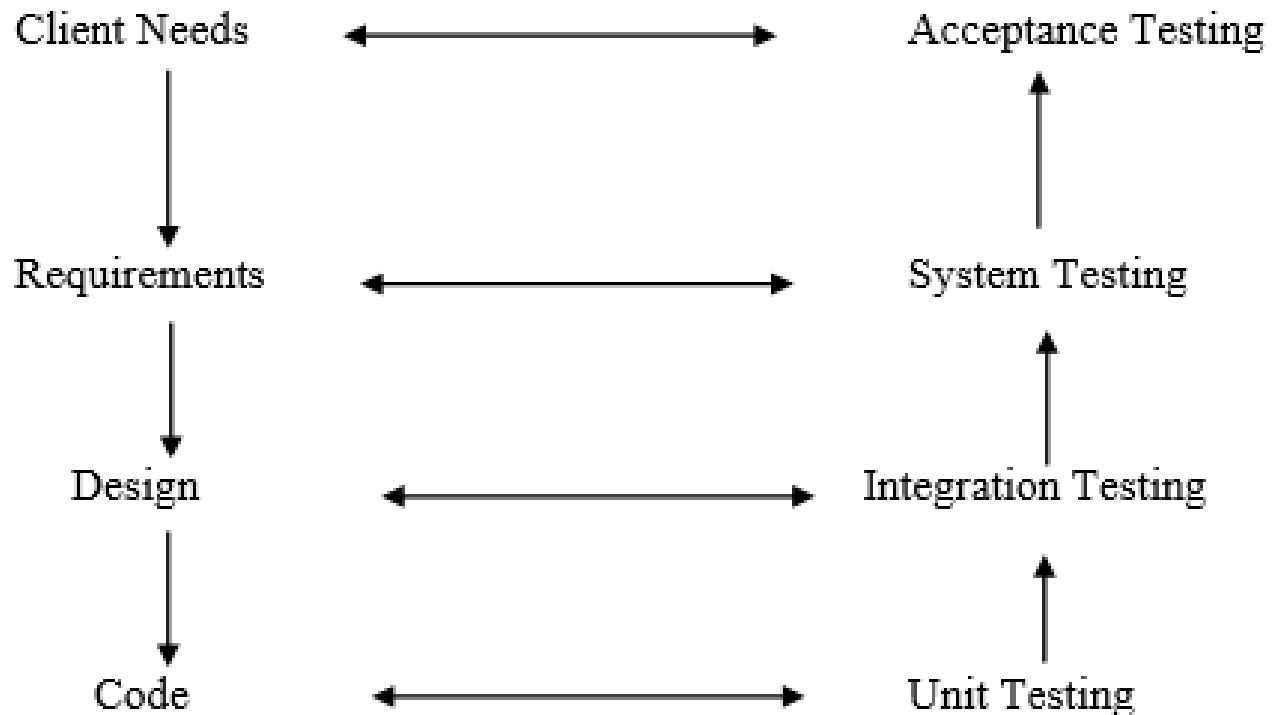| Client Needs	←———————→	Acceptance Testing
↓		↑
Requirements	←———————→	System Testing
↓		↑
Design	←———————→	Integration Testing
↓		↑
Code	←———————→	Unit Testing

Fig 8.1 Level of Testing

- Unit Testing

    Unit testing focuses verification efforts on the smallest unit of the software design, the module. This is also known as "Module Testing". The modules are tested separately. This testing carried out during programming stage itself. In this testing each module is found to be working satisfactorily as regards to the expected output from the module.

- Integration Testing

    Data can be grossed across an interface; one module can have adverse efforts on another. Integration testing is systematic testing for construction the program structure while at the same time conducting tests to uncover errors associated with in the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here correction is difficult because the isolation of cause is complicate by the vast expense of the entire program. Thus, in the integration testing stop, all the errors uncovered are corrected for the text testing steps.

- User Acceptance Testing

    User acceptance of a system is the key factor of the success of any system. The system under study is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required

8.3 TEST CASES

Admin Login: - Admin login id and password is kept compulsory fields, and if the admin id or password doesn't match then it will show an error message.

VALIDATION CRITERIA

1. In each form, no field which is not null able should be left blank.
2. All numeric fields should be checked for non-numeric values. Similarly, text fields like names should not contain any numeric characters.
3. All primary keys should be automatically generated to prevent the user from entering any existing key.
4. Use of error handling for each Save, Edit, delete and other important operations.
5. Whenever the user Tabs out or Enter from a text box, the data should be validated and if it is invalid, focus should again be sent to the text box with proper message
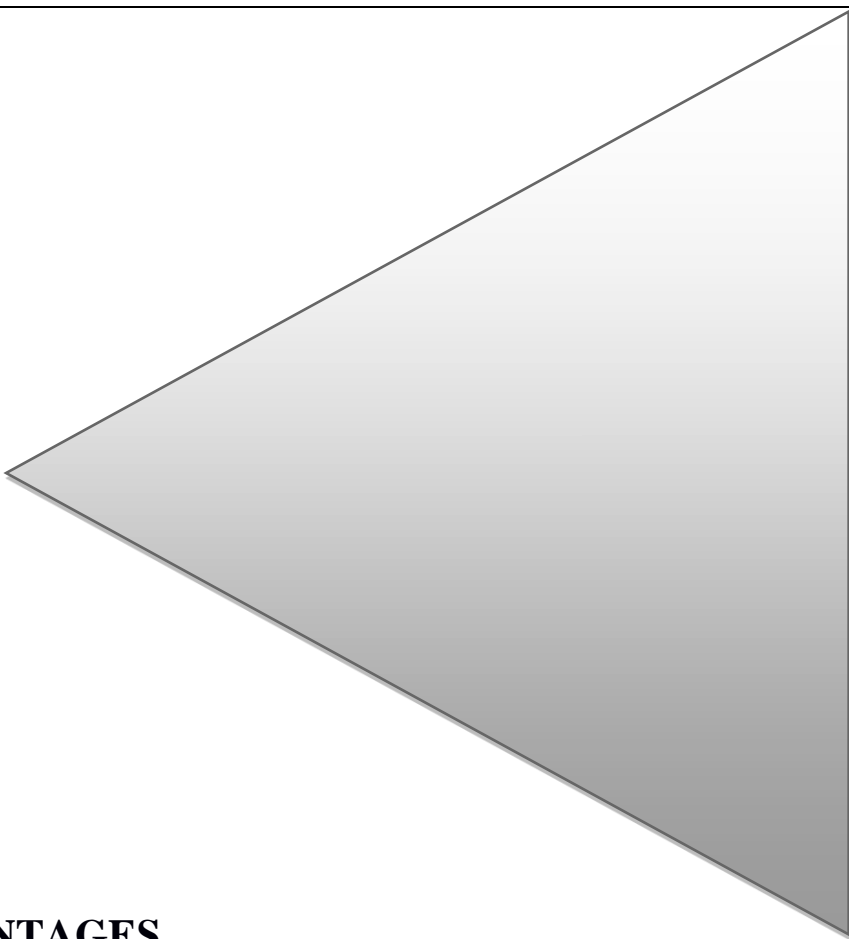
Chapter 9

**ADVANTAGES
& DISADVANTAGES**

9.1 ADVANTAGES OF PROJECT

1. Wide portion coverage and efficient question paper generation.
2. No chance of paper leaks.
3. No need of transporting papers through police/security vans to all colleges.
4. The system provides an unbiased result.
5. Thus, the system excludes human efforts and saves time and resources.

9.2 DISADVANTAGES OF PROJECT

The system must be given proper inputs otherwise system can produce wrong results.

APPLICATIONS:

1. This system can be used in universities for generating and distributing question papers.
2. The system can also be implemented in different organizations that conduct regular exams.

SYSTEM IS:

- Load Balancing:

Since the system will be available only the admin logs in the amount of load on server will be limited to time period of admin access.

- Easy Accessibility:

Records can be easily accessed and store and other information respectively.

- User Friendly:

The system will be giving a very user friendly approach for all user.

- Efficient and reliable:

Maintaining the all secured and database on the server which will be accessible according the user requirement without any maintenance cost will be a very efficient as compared to storing all the customer data on the spreadsheet or in physically in the record books.

- Easy maintenance:

Question Paper Generator systems design as easy way. So, maintenance is also easy.
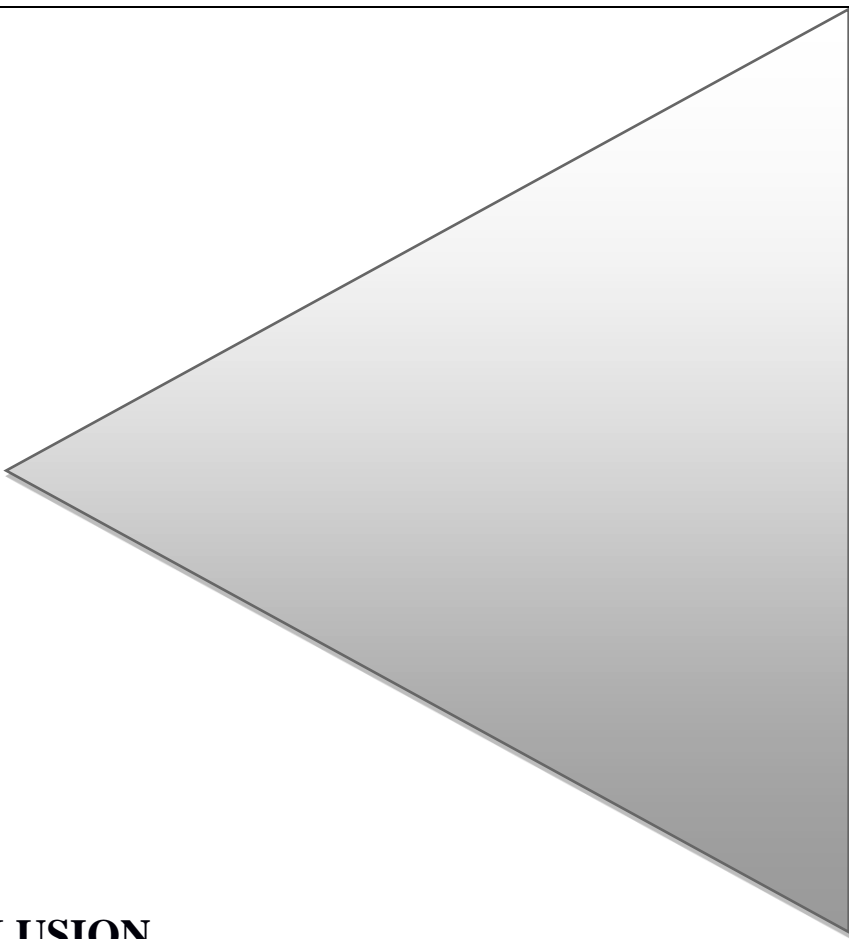
Chapter 10

CONCLUSION

10.1 CONCLUSION

In this paper, an automated model for Question Paper Generation is proposed which is implemented as a real-time application. The proposed work describes an automated system that progresses from the traditional method of paper generation to an automated process, by providing controlled access to the resources. This is achieved by comprehending users and their roles in the institute. We have also considered the importance of randomization in the task of paper generation. Our system has deployed an efficient algorithm which is totally randomized and avoids repetition of questions is consequent question papers, making it impossible to derive any pattern in the papers. We distinguish between administrators and subordinates by their tasks. Therefore, the resultant automated system for Question Paper Generation provides improvement in terms of controlled access to the resources, random generation of question papers and a secure platform.
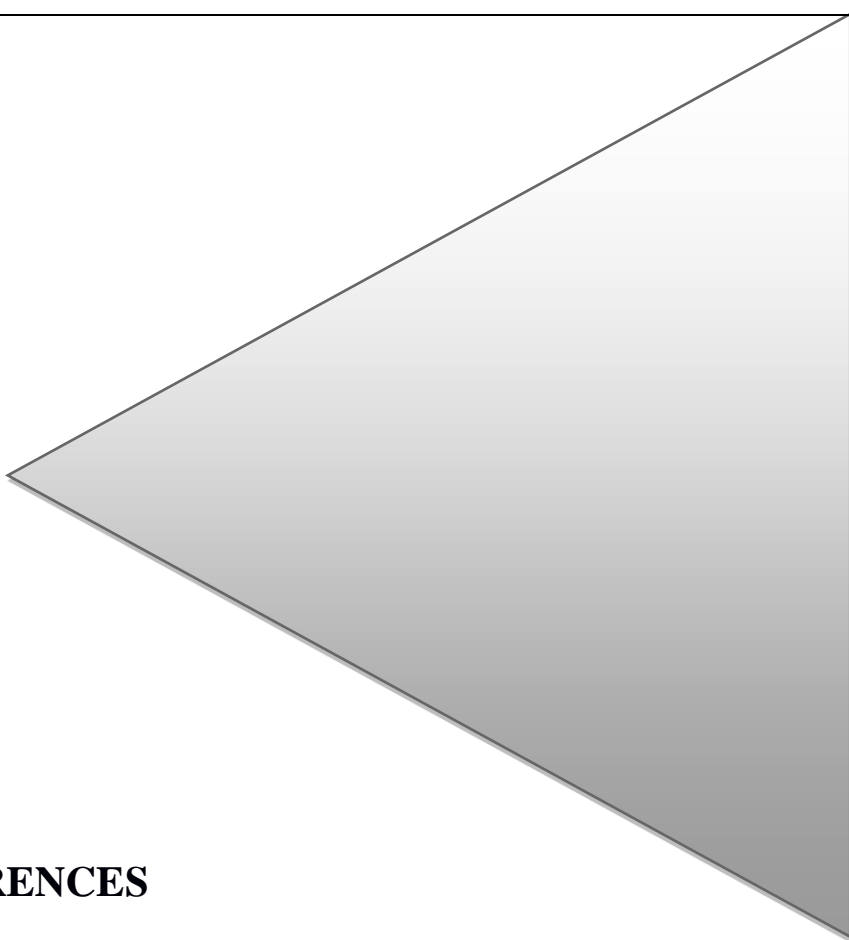
REFERENCES

REFERENCES

Websites:

- en.wikipedia.org

- Microsoft Developer Network (MSDN): http://msdn2.microsoft.com/en- us/default.aspx: This is a valuable online resource, and is a must for any developer using Microsoft tools.

- http://www.asp.net/: This is the official Microsoft ASP.NET web site. It has a lot of: tutorials, training videos, and sample projects

- http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6449828&queryText

- %3DQuestion+paper+generator+system

- http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6779494&queryText

- %3DQuestion+paper+generator

- http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4756830&queryText

- %3DQuestion+paper+generator+system
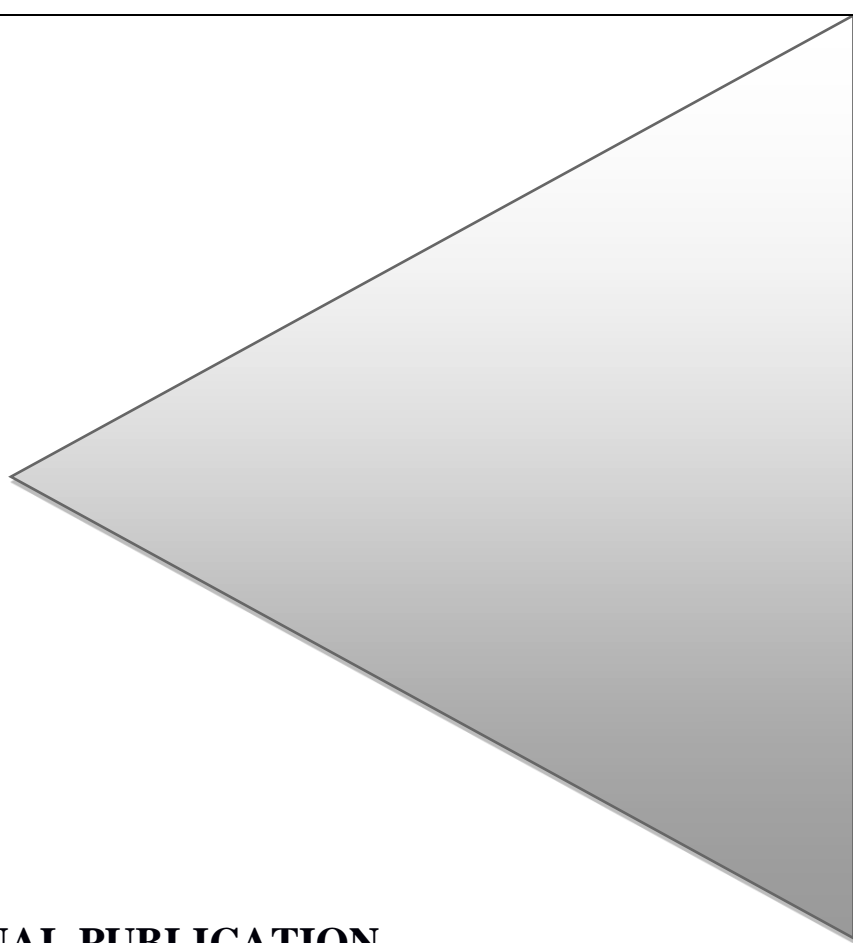
Chapter 11

JOURNAL PUBLICATION

An Intelligent Question Paper Generator using Randomized Algorithm

Parth Shah[1*], Uzair Faquih[1], Rutuja Devkar[1], Yogesh Shahare[1]
[1]MGM College of Engineering & Technology

Abstract:- This is a problematic era because of the boom in the subject of advanced technology and the demand we are dealing with nowadays. Therefore, examinations play a crucial role in checking out an individual's performance, and preparation of exam question papers has consistently been a matter of interest. And this is why it's far critical to have an intelligent system for the growth of students in addition to checking their learning skills, thereby retaining a check on student performance. Faculties generate various question papers in keeping with the university's assessment requirements. It's very challenging for the teachers to make question papers with multiple questions that meet the course's learning targets. We have proposed an automated process of question paper generation that is fast, streamlined, randomized, and secure. Every action executed task by the system is automated, so that storage space, bias, and security are not a challenge anymore. It automatically creates a variety of sets of questions now and then without worrying about replication and duplication from the initial exam at the same time as the question bank keeps growing.

Keywords: Paperless, randomization, automation, assessments, question taxonomy.

I. INTRODUCTION

In today's modern, ambitious world, an exam plays a vital role in checking the educational improvement of students, and the technology of the information era is now substituted through the productive application of the technology. Any product which can correctly reduce time and power consumption is accepted and preferred. So producing software from knowledge is a crucial task to do. In all the academic courses that reject a variety of tests, the instructor intends to create important documents in accordance with the guidelines and assessments of the autonomic university. It is much more challenging to deal with all the course features for teachers and avoid duplicating questions from subsequent estimates. There is no systematic process, and for this reason, this problem's paper quality is predicted for individual teachers and qualifications.

The truth that there is a shortage of experienced teachers makes the situation even worse for specifying courses, semesters, syllabus, and patterns. At times, all these factors might also deteriorate the quality of the question paper. The researcher says a good questionnaire is the right combination of subjects (questions) guided by various parameters: cognitive level, difficulty and distribution of scores on the questionnaire. Creating a good questionnaire that contains many questions related to gaining knowledge about the purpose of a course in terms of content and cognitive level is a difficult task. So, we are presenting an Automatic Question Paper Generator System which could reduce time intake by replacing the traditional approach of question paper generation. There are provisions to enter and edit data suitable to any educational organization with complete freed. Automatic test paper generation refers to questions selected from the question bank and automatically generates different kinds of papers that meet the requirements of teaching, so it is a typical solution process of the constraint satisfaction problem (CSP). We have implemented a role-based hierarchywhich will restricts access to the users. The system also deploys security mechanisms that prohibit duplication of question papers. This enables an educational institute to generate questions ensuring security and non-repetitiveness of question papers and is a boon for organizations with limited staff and resources. Our system aims to provide fast operations, data storage, and high security for all its tasks. Theevolution of traditional and existing Question Paper Generationsystems and the need for an automated system is unraveled. We have proposed our revised system of an Intelligent Automated Question Paper Generation.

Literature Survey is discussed in **section 2**. Methodology & Algorithm is available in **section 3**. **Section 4** consists of result analysis of the project. **Section 5** includes the conclusion.

II. LITERATURE SURVEY

For the automated generative system of examination papers, first, it needs to be designed by the exclusive varieties of type, the number of questions, the difficulty, and the score in order to establish the corresponding test database. Second, the papers is composed of random. Third, in a paper automatically generated, the knowledge keys concerned can now not arise. When the check paper is made, the questions are selected through the gadget within the questions database. They can meet the person's requirements, and the performance and probability of achievement are high. The user interface of this soft is friendly. The user's requirements can be set by way of human-computer interaction, such as: the scores for all kinds of questions in the test paper, the overall difficulty, thedistribution of knowledge points and the proportion of various types of questions, and so on [9].

A. *Problem with Current Scenario*:
- Traditionally, there was no such system that would easily generate a question paper by just inserting set of question papers to the system.
- In existing system, university use to generate question papers and distribute the question papers to respective

colleges manually.
- There might be chances of paper leaks due to the existing manual system.
- The system is relatively inefficient because question papers may not reach the respective colleges on time.

B. *Limitation of Paper-based Systems*: As with most human working processes, this system suffers due to bias. There might be some questions that are repeated in many question papers as the professor has a personal inclination towards them. So there is no guarantee of the pure randomly generated question paper. Other problems that may plague this system are the non- availability of staff and resources, natural calamities and accidents. Also, the security of the system can be easily compromised if leverage over the person responsible for generating question papers is obtained. Other limitations include: -

- Insufficient storage space
- Easy to Damage
- Inefficient document transportation
- Supply costs
- Poor environmental credentials
- Limited collaboration
- Editing problems

C. *Analysis of Paper-Based System*: From the above analysis, we know that we need an integrated Question Paper Generation System with improvements in speed, efficiency, controlled access to the resources, randomization of questions, and security. In addition, the system should perform tasks in the fastest way without violating the role-based hierarchy and their access rights policy, provide a central database for data storage, ensure security and optimize the system's overall performance.

III. METHODOLOGY & ALGORITHM

A) *Proposed System*: To overcome the existing system anomalies, this question paper generator system is developed.
- We present a smart question paper generating system for universities.
- It is made to permit universities to generate question papers with random but even questions to cover maximum chapters of subject with difficulty level within seconds and mail them to colleges immediately.
- In our system we allow administrators to enter a fixed of questions and respective answers for option ticking.
- We additionally permit admin to offer weightage & difficulty level for every questions.
- After this the questions are stored in database along with their weightage.
- While generating question the admin simply has to choose the level of difficulty.
- On this feature, the system selects questions randomly in a way that their weightage makes up for 100 marks and according to the difficulty the admin chooses thequestions are chosen based on their complexity level. Thequestions are also added for numerous difficulty levels sothat as soon admin selects the type of paper difficulty

(easy, medium, difficult) the system automatically generates paper, prepares document file as per selected paper format.
- We can also email it to other colleges. After this question paper is converted to pdf file and emailed to colleges on button click.

B) *Random Algorithm*: The random Algorithm has instances to generate exam papers. One is to randomly choose questions from the question bank and then choose whether they meet the paper's constraints. The alternative is to find out all the meet questions according to the given rules from the database, randomly selecting a number of them to constitute a take a look at the paper. Considering the low performance within the first case, in this paper, we adopt the second way toapply a random algorithm to generate test papers. In this paper, the overall parameters in the automatic test paper generation include paper title, examination time, the coefficient of difficulty, syllabus, question type and so on. Among the paper parameters, the coefficient of difficulty is one of the most important factors.
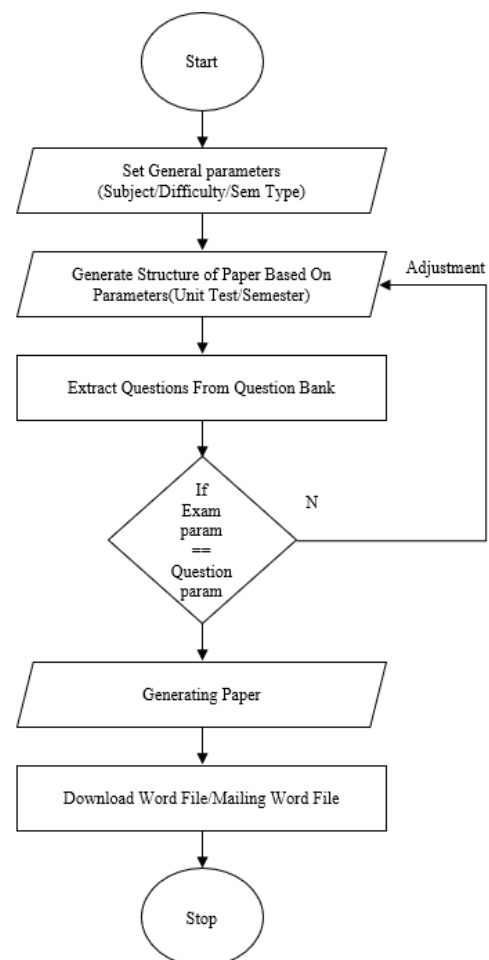

Fig 1. Random Algorithm

According to the analysis of the most exam results, the test score and the paper's coefficient of difficulty are in reverse proportion. In most cases, the test scores are required to be normally distributed [5], and the average score of all trainees should be consistent with the expected score of the paper.

Therefore, the number of questions with different coefficient of difficulty in the paper should be determined by the normal distribution function.

The normal density function can be described as follows:

$$f(x) = \frac{1}{\delta\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\delta^2}} \quad \text{...... Eqn 1}$$

Where x C (—œ, +œ), μ and ð are constants. μ is the average of normal random variable, and ð is variance. Firstly, the expected average test score and the distribution range of scores are set according to the range of average scores, which are determined by the paper's coefficient of difficulty, and then the average of normal random variable and variance can be figured out. Finally, the difficulty proportion of questions in the paper can be calculated according to the normal density function. So let us assume that the score range of lower and upper bounds from Easy to Difficult are $l_{i\,nin}$ and $l_{i\,nas}$, i = 1,2, … ,5 . And E_i are 5 probabilities followed by the coefficient of difficulty from easy to difficult.

The variable E_i can be calculated by the function described as follows:

$$E_i = \int_{l_{i\,min}}^{l_{i\,max}} \frac{1}{\delta\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\delta^2}} dx \quad \text{...... Eqn 2}$$

To improve the success rate of automatic test paper generation, the random algorithm will appropriately adjust the parameters in the above equation when the examination questions in the question bank do not meet the requirements in the parameter table. Thus change the proportion of different difficulty of the examination questions. There are two ways to adjust the parameters. One is the adjustment of μ and ð values. The other is to adjust the values of $l_{i\,nin}$ and $l_{i\,nas}$. If there is insufficient number of some coefficient of difficulty, then reduce the value of $l_{i\,nin}$ properly. Otherwise, increase the value of $l_{i\,nas}$. The goal of both adjustments is to try to increase the number of questions with a larger number in question bank in the coefficient of difficulty as much as possible, and reduce the number of questions with less in question bank in the coefficient of difficulty. Thus try to avoid causing the failure of the automatic test paper generation because of the insufficient questions. The automatic test paper generation system, whose Algorithm is designed according to what is designed above, can work very well and generate test papers by setting up the parameters.

2)

IV.   RESULT ANALYSIS

The Automated Question paper generator has been implemented in C sharp language, which is a general-purpose, multi-paradigm programming language. The fully working system stores courses, subjects, questions and

patterns of question papers. It then applies the Algorithm to the stored question set and prints the question paper in word format. This project is implemented as a web application using Visual Studio 2019 IDE. We used Visual Studio for the Design and coding of our project. We Created and maintained all databases into SQL Server 2018, in that we created tables wrote a query to store data or records of the project.

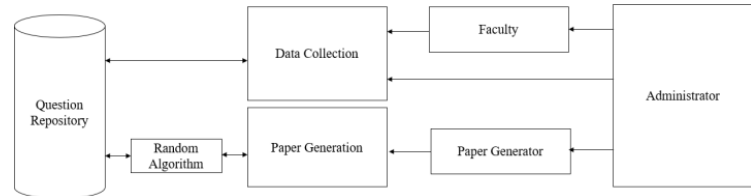System Architecture of implementation is given by :

Fig 2. System Architecture

A)   Login Form

The first web interface allows the user to select the role given while after that user needs to enter credentials to log in into the system.
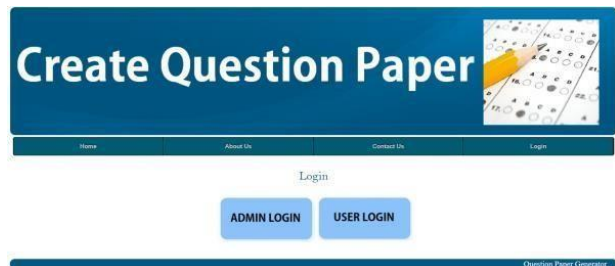

Fig 3. Login Form

1)   *Admin*: This Role has full access to the system, which includes
- Adding Teachers
- Adding Course & Subjects
- Generating Question Paper
- Directly Emailing the generated Question Paper
- View Question Paper Logs from Master Database
- Adding/View MCQ Question

Fig 4: Admin Role Menu

2)   *User:* Teacher/Faculty is the user role here. The main objective of the user is to add the question to the database of a particular course assigned by the admin.
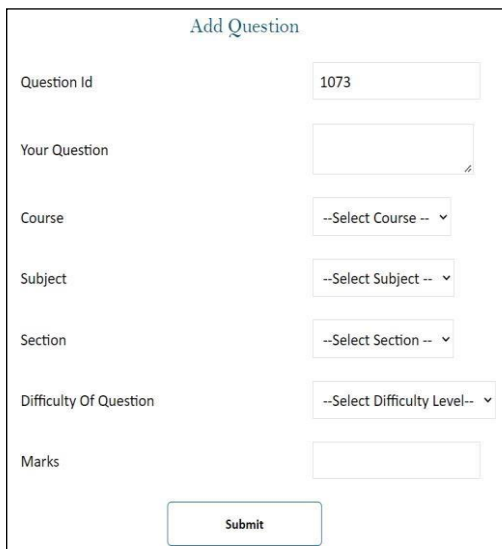
Fig 5. Question Insertion

B) *Question Paper Generation*

After the test subjects and questions are set, the parameters of the papers need to be checked in order to ensure the correct parameter settings. After the confirmation, the user can click the button to generate the papers. If not satisfied, the system will notify the admin regarding it.

The following process would combine the preamble information with question paper table contents to produce question paper in word format. Once all the info is passed to the system admin can move ahead and download the Question Paper.
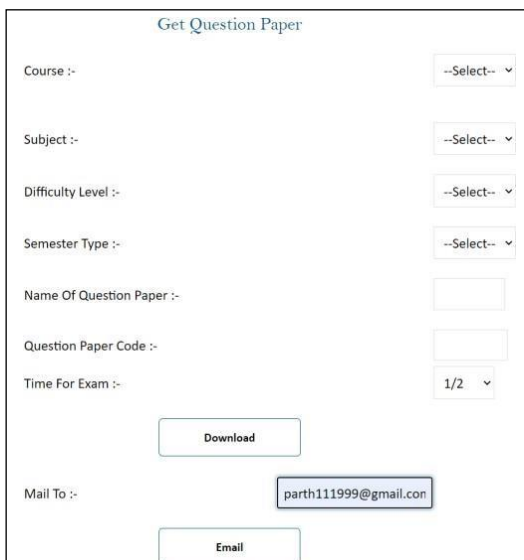
Fig 6: Question Paper Generation

V. CONCLUSION

In this research paper, an automated design model for Question Paper Generation has been proposed which is implemented as a real time application. The proposed work explains an automated system that shows progression from the traditional

method of paper generation to an automated process, by providing controlled access to their resources. This can be achieved by comprehension of users and their particular roles in the institute. We have considered the importance of randomization in the task of paper generation and has deployed an efficient algorithm that is completely randomized and also restricts repetition of questions in question papers. We can differentiate between administrators and subordinates by their respective tasks. Hence, the resultant automated system model for Question Paper Generation provides progression in terms of controlled access to the resources, random generation of question papers and an independent , fully secure platform.

Our system is a valuable resource for teachers in automatically generating question papers from the question repository. However, while the system designed by us stands out in all available systems, there's scope for extra enhancements to make it more useful. For example, depending on the kind of evaluation required, the system can be made to select specific question types.

For example, if the user wants an assessment for an online quiz, it could smartly include all MCQs. Or, if a user is choosing the term test assessment, more objective type and short answer questions should be preferred. Also, users would be overjoyed to have a feature to provide statistics for gaps in user given specifications and system- generated specs[10].Now the system is just up to generating question paper, but in the future, the system can even be implemented with separate student login for online test assessment with randomly generated questions at that moment, making it more efficient for Exam conduction.

ACKNOWLEDGEMENT

REFERENCES

[1] Vijay Krishan Purohit', Abhijeet Kumar', Asma Jabeen', Saurabh Srivastava', R H Goudar' ,Shivanagowda, "Design of Adaptive Question Bank Development and Management

[2] Kapil Naik, Shreyas Sule, Shruti Jadhav, Surya Pandey, "Automatic Question Paper Generation System Using Randomization Algorithm", (IJETR) ISSN: 2321 0869,Volume-2, Issue-12, December 2014

[3] W. Harlen and M. James, "Assessment and learning: differences and relationships between formative and summative assessment," Assessment in Education, 4(3), pp.365-379, 1997.

[4] J Biggs, "Aligning teaching and assessing to course objectives, "Teaching and Learning in Higher Education: New Trends and Innovations, 2, pp.13-17, 2003.

[5] Caminero, A.C. ; Dept. de Sist. de Comun. y Control, Univ. Nac. de Educ. a Distancia, Madrid, Spain ; Hernandez, R. ; Ros, S. ; RoblesGomez, A. , "Choosing the right LMS: A performance evaluation of three open-source LMS", Global Engineering Education Conference (EDUCON), 2013 IEEE, March 2013

[6] Ramesh, R. ; Mishra, S. ; Sasikumar, M. ; Iyer, S., "Semi-Automatic Generation of Metadata for Items in a Question Repository", Technology for Education (T4E), 2014 IEEE Sixth International Conference, Dec 2014

[7] Liancheng Guan , "Design of the Automatic Generative System of Examination Papers Based on ARM", International Conference on Computer Network, Electronic and Automation 2017

[8] Gauri Nalawade, and Rekha Ramesh," Automatic Generation of Question Paper from User Entered Specifications using a Semantically Tagged Question Repository", IEEE 8th International Conference on Technology for Education 2016

[9] Deena, G; Kothandaraman, Raja and Kaliyan, Dr. Kannan," An Automatic Question Generation System Using Rule-Based Approach in Bloom's Taxonomy", Recent Advances in Computer Science and Communications 2019

[10] Krishna R; Rahila Talupula and Jahnavi Thummala, "Question Paper Generator and Result Analyzer", Advances in Systems, Control and Automations (pp.315-323) 2021

CERTIFICATE
OF PUBLICATION

This is to certify that

Parth Shah

Has published a research paper entitled

An Intelligent Question Paper Generator using Randomized Algorithm

In IJERT, Volume 11, Issue 4, April - 2022

Chief Editor,IJERT

Date: 14-04-2022

Registration No: IJERTV11IS040041

CERTIFICATE
OF PUBLICATION

This is to certify that

Uzair Faquih

Has published a research paper entitled

An Intelligent Question Paper Generator using Randomized Algorithm

In IJERT, Volume 11, Issue 4 , April - 2022

Chief Editor,IJERT

Date: 14-04-2022

Registration No: IJERTV11IS040041

CERTIFICATE
OF PUBLICATION

This is to certify that

Rutuja Devkar

Has published a research paper entitled

An Intelligent Question Paper Generator using Randomized Algorithm

In IJERT, Volume 11, Issue 4 , April - 2022

Chief Editor,IJERT

Date: 14-04-2022

Registration No: IJERTV11IS040041

International Journal of
Engineering Research & Technology
ISSN : 2278 - 0181, www.ijert.org
(Published by : ESRSA Publications)

CERTIFICATE
OF PUBLICATION

This is to certify that

Yogesh Shahare

Has published a research paper entitled

An Intelligent Question Paper Generator using Randomized Algorithm

In IJERT, Volume 11, Issue 4, April - 2022

Chief Editor,IJERT

Date: 14-04-2022

Registration No: IJERTV11IS040041