EXAM MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted By

HARSH
(University Roll No. 2100290140067)

SIDDHARTH
(University Roll No. 2100290140134)

HIMANSHU PUNDIR
(University Roll No. 2100290140070)

SHIVAM SHARMA
(University Roll No. 2100290140129)

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of
Mr. Ankit Verma
Assistant Professor
KIET Group of Institutions Delhi-NCR, Ghaziabad

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206
(FEBRUARY 2023)

CERTIFICATE

Certified that **Harsh, Siddharth, Himanshu Pundir and Shivam Sharma** have carried out the project work having title "**Exam Management System**" for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself / herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date: 14 Feb 2023

Harsh (2100290140067)
Siddharth (2100290140134)
Himanshu Pundir (2100290140070)
Shivam Sharma (2100290140129)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 14 Feb 2023

Mr. Ankit Verma
Assistant Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Signature of Internal Examiner **Signature of External Examiner**

Dr. Arun Kumar Tripathi
Head, Department of Computer Applications
KIET Group of Institutions, Ghaziabad

ACKNOWLEDGEMENTS

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, Mr. Ankit Verma for his guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Arun Kumar Tripathi Sir, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Harsh

Siddharth

Himanshu Pundir

Shivam Sharma

ABSTRACT

The project Exam Management System allows a particular institute or company to arrange, conduct and manage examinations in a computerized and automated environment. Although such a project has a very wide scope, this system fulfils the most important features i.e. management of exam details and user information. The system has the facility of admin login through which the admin can monitor and manage the information regarding the candidates and also manage the exam details. The candidate can login through the candidate login and attempt the exam as well as see the results.

Exam Management System, as described above, can lead to an error-free, secure, reliable, and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on record keeping. The organization can maintain computerized records without redundant entries.

The aim is to automate its existing manual system with the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy access and manipulation of the same.

LIST OF CHAPTERS

CHAPTER – 1

INTRODUCTION

1.1 Project Description

The "Exam Management System" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the company/institute to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides an error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user-friendly. Exam Management System, as described above, can lead to an error-free, secure, reliable, and fast management system. It can assist the user to concentrate on their other activities rather than concentrate on record keeping. Thus, it will help organizations in better utilization of resources.

Exam Management System is a software developed for the students who give exams in colleges and institutes. It facilitates to access the examination information of a particular student in a particular class. The information is sorted by the operators, which will be provided by the teacher for a particular class. This system will also help in evaluating examination eligibility criteria of a student.

Also, for those busy executives who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

1.2 Project Objective

The main objective of the project 'Exam Management System' is to manage the details of candidate, exam, score and levels. The project is totally at administrative end and thus only the administrator is granted the access to management modules. The purpose of the project is to build an application to reduce the manual work for managing the MCQ exam and we will follow to achieve these objectives in this project. –

- To create an appropriate platform for best managing of MCQ questions.
- To overcome the time-consuming issues and taking MCQ exam.
- To release the marks of the exam as soon as possible.
- To manage the information of candidates.
- To reduce paper work.

1.3 Project Scope

The proposed system is automated that is faster than the existing manually maintained system and can handle data easily. Computerized of the details of manually maintained examination operations. The system allow administrator to control all the activities hence identifying the roles and accessibility of other users. Accurate information can be generated easily and quickly at different levels. Report can be generating easily and quickly.

1. Candidate Registration facility
2. Candidate Login
3. View Candidate Profile
4. Exam Attempt Page
5. View Results Page
6. View List of Candidates
7. Administrator Control Panel
8. Updates by the Administrator
9. View Updates by the Candidates

1.4 Module Description

1.4.1 Login and Registration Module

- In this module, the admin as well as the candidate can login into the system.
- The admin can login to the system using credentials saved into the database and addition of new admin is possible only at the database level.
- The candidate can login using the roll no and date of birth as username and password.
- Unregistered candidates can register through the registration page giving all the required details and later login using the credentials entered.

1.4.2 Admin Module

- The system is completely at the administrative end.
- Once the admin logins, they'll be able to view and delete the registered candidates.
- They update the list of questions and also update the details of the candidates.
- Admin will be able to see the details of the exam attempts by the candidates along with their results.
- The admin maintains all the candidates who registered for the exam.

1.4.3 Candidate Module

- The candidate can login to the system or register by entering the details if not already registered.
- They can access the exam module to appear for the exam.
- Results can be viewed by the candidates along with their details.
- In case of any incorrect data entry in the database, admin must be requested to update the data.

1.4.4 Exam Module

- The exam module provides the core examination functionality to the system where a candidate can access the exam module to attempt the MCQ based questions.
- Questions from the database are fetched to the interface one by one.

- The candidate can see the results at the instant and which are also stored into the database in the candidate's profile.

1.4.5 Report Module

- In this module following types of reports will be generated:
- List of all levels
- List of all registered candidates along with their results
- List of all questions

1.5 Hardware Software Used

1.5.1 Software Tools Used:

- Operating System- Windows 10
- Code Editor- NetBeans IDE 12.0
- Front End- Java Swing
- Back End- Java, MySQL

1.5.2 Hardware Tools Used:

- Intel i3 or Above
- 4GB RAM or Above
- Hard-Drive Capacity 64GB or Above

CHAPTER-2

FEASIBILITY STUDY

After doing the project Exam Management System, study and analyze all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible - given unlimited resources and infinite time.

The feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

2.1 Economic Feasibility

Besides being technically feasible, developing this system is economically feasible as well. The development of the system does not require the developers to spend a lot of money. The tools used to develop the system are not expensive and the software's are open source. All that is needed is time. Even the maintenance of the system will not be expensive. The system is indeed economically feasible.

This is a very important aspect to be considered while developing a project. We decided on the technology based on the minimum possible cost factor.

- All hardware and software costs must be borne by the organization.
- Overall, we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later-on running cost for the system.

2.2 Technical Feasibility

Building this system is technically feasible. The hardware and software needed are all available, it not difficult to get them. Brief it can be said that the necessary resources needed for the development and maintenance of the system are available. We are going to use java programming language and MySQL database.

This included the study of function, performance, and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied the complete functionality to be provided in the system, as described in the System Requirement

Specification (SRS) and checked if everything was possible using a different type of frontend and backend platforms.

2.3 Operational Feasibility

No doubt the proposed system is fully GUI based very user-friendly and all inputs to be taken all self-explanatory even to a layman. Besides, proper training has been conducted to let them know the essence of the system to the users so that they feel comfortable with the new system. As far as our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing.

CHAPTER – 3

DATABASEDESIGN

3.1 Database Tables

3.1.1 Admin Table

#	Name	Type	Collation	Attributes	Null	Default
1	username	varchar(20)	latin1_swedish_ci		No	*None*
2	password	varchar(20)	latin1_swedish_ci		No	*None*

3.1.2 Candidate Table

#	Name	Type
1	roll_no	varchar(10)
2	name	varchar(40)
3	gender	varchar(10)
4	dob	varchar(20)
5	email	varchar(40)
6	contact_no	varchar(10)

3.1.3 Question Table

#	Name	Type
1	qid	int(10)
2	ques	varchar(100)
3	opt1	varchar(50)
4	opt2	varchar(50)
5	opt3	varchar(50)
6	opt4	varchar(50)
7	answer	varchar(50)

3.1.4 Result Table

#	Name	Type
1	roll_no	varchar(10)
2	name	varchar(40)
3	date	varchar(20)
4	grade	int(10)

3.2 ER Model/Diagrams

The entity-relationship (ER) data model allows us to describe the data involved in a real-world enterprise term of objects and their relationships and is widely used to develop an initial database.

The ER model is important primarily for its role in database design. It provides useful concepts that allow us to move from an informal description of what users want from their database to a more detailed and precise, description that can be implemented in a DBMS. We note that many variations of ER diagrams are in use, and no widely accepted standards prevail.

The database design process can be divided into six steps. The ER model is most relevant to the first three steps:

- **Requirements Analysis:**

The very first step in designing a database application is to understand what data is to be stored in the database, what applications must be built on top of it, and what operations are most frequent and subject to performance.

(1) Conceptual Database Design:

The information gathered in the requirements analysis step is used to develop a high-level description of the data to be stored in the database, along with the constraints that are known to hold over this data. This step is often carried out using the ER model, or a similar high-level data model, and is discussed in the rest of this chapter.

(2) Logical Database Design:

We must choose a DBMS to implement our database design, and convert the conceptual database design into a database schema in the data model of the chosen DBMS. We will only consider relational DBMS, and therefore, the task in the logical design step is to convert an ER schema into a relational database schema. The result is a conceptual schema, sometimes called the **logical schema,** in the relational data model.

- **Beyond the ER model:**

ER model is sometimes regarded as a complete approach to designing a logical database schema. This is incorrect because the ER diagram is just an approximate description of the data, constructed through a very subjective evaluation of the information collected during requirements analysis.

(3) Schema Refinement:

The fourth step in database design is to analyse the collection of relations in our relational database schema to identify potential problems, and to refine it.

(4) Physical Database Design:

In this step we must consider typical expected workloads that our database must support and further refine the database design to ensure that it meets desired.

(5) Security Design

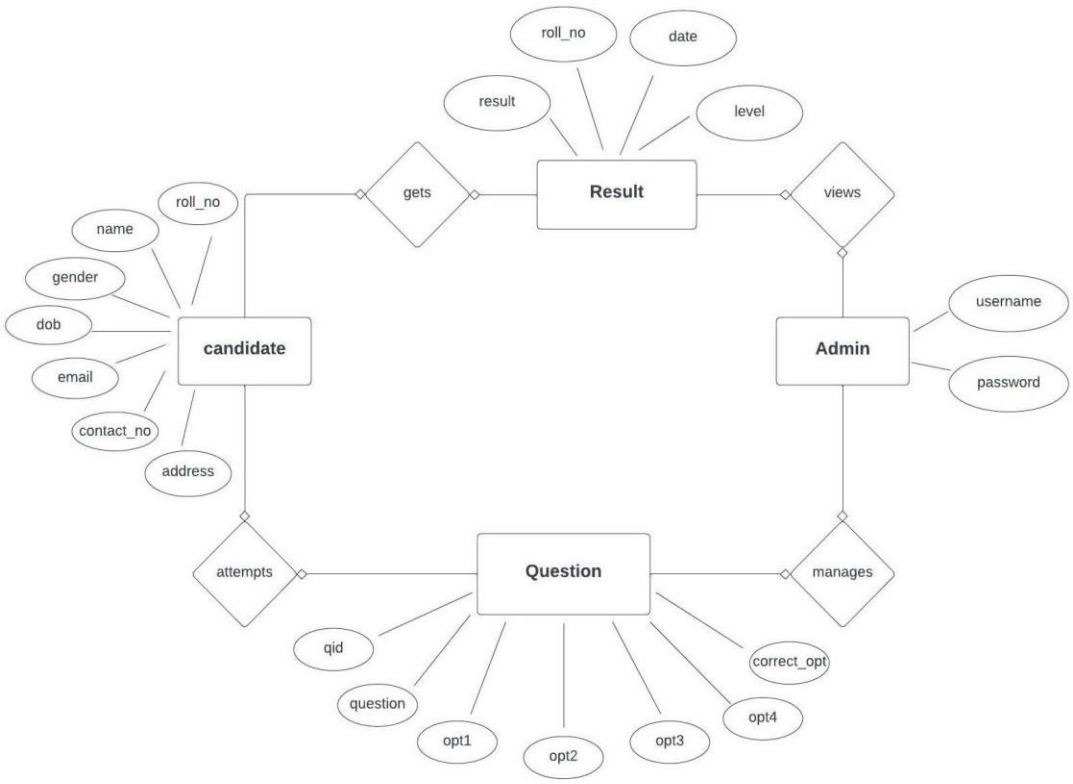
In this step, we identify different user groups and different **roles** played by various users (e.g., the development for a product, the customer support representatives, the product manager).

- **Entities, Attributes and Entity sets:**

An **entity** is an object in the real world that is distinguishable from other objects manager of the toy department, the home address of the manager of the toy department. It is often useful to identify a collection of similar entities. Such a collection is called an **entity set**. Examples include the following: the Green Dragonwort toy, the toy department, the performance criteria.

An entity is described by set of **attributes**. All entities in a given entity set have the same attribute; this is essentially what u has seen by similar. For each attribute associated with an entity set, we must identify a **domain** of possible values. A **key** is a minimal set of attributes whose values uniquely identify an entity in the set. There could be more than one **candidate** key.

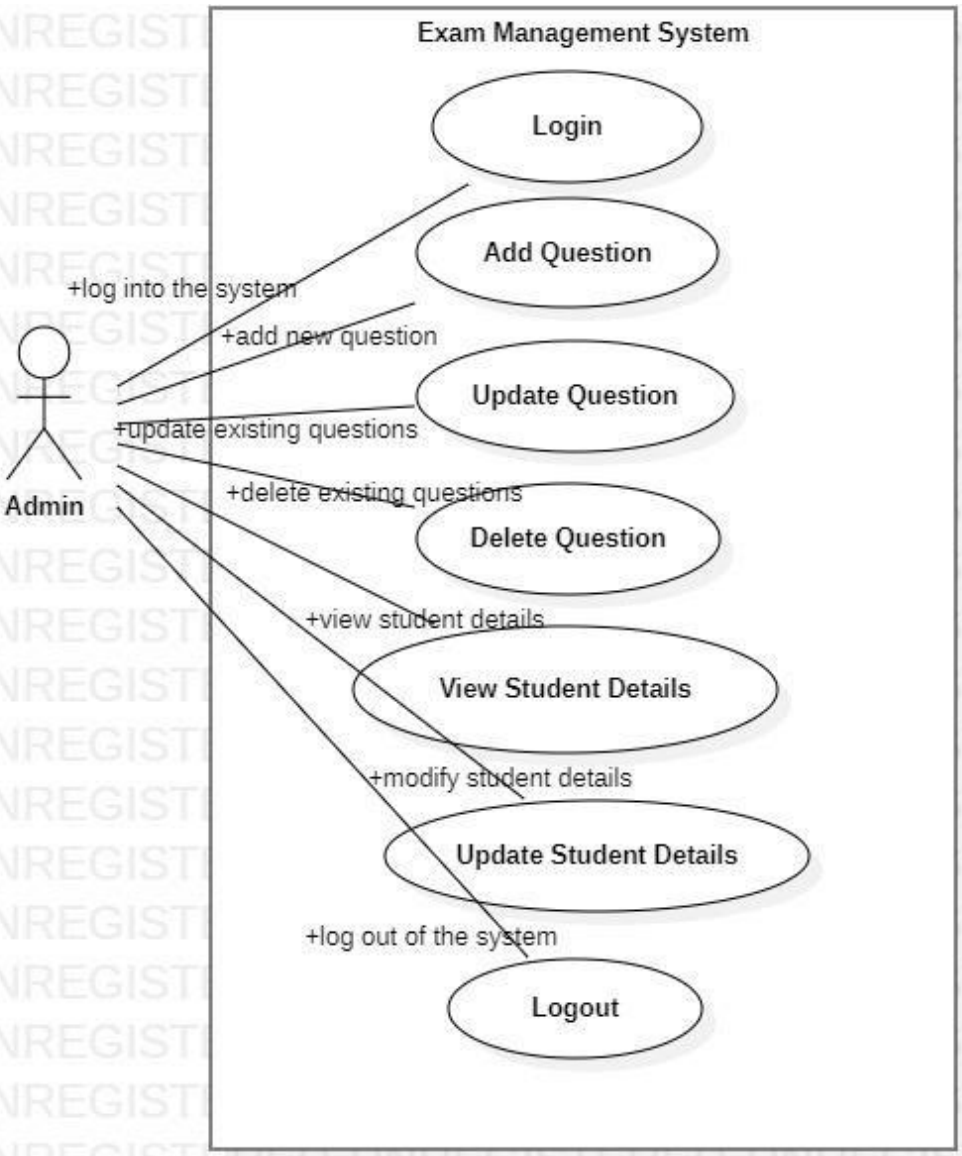
- ## ER Diagram



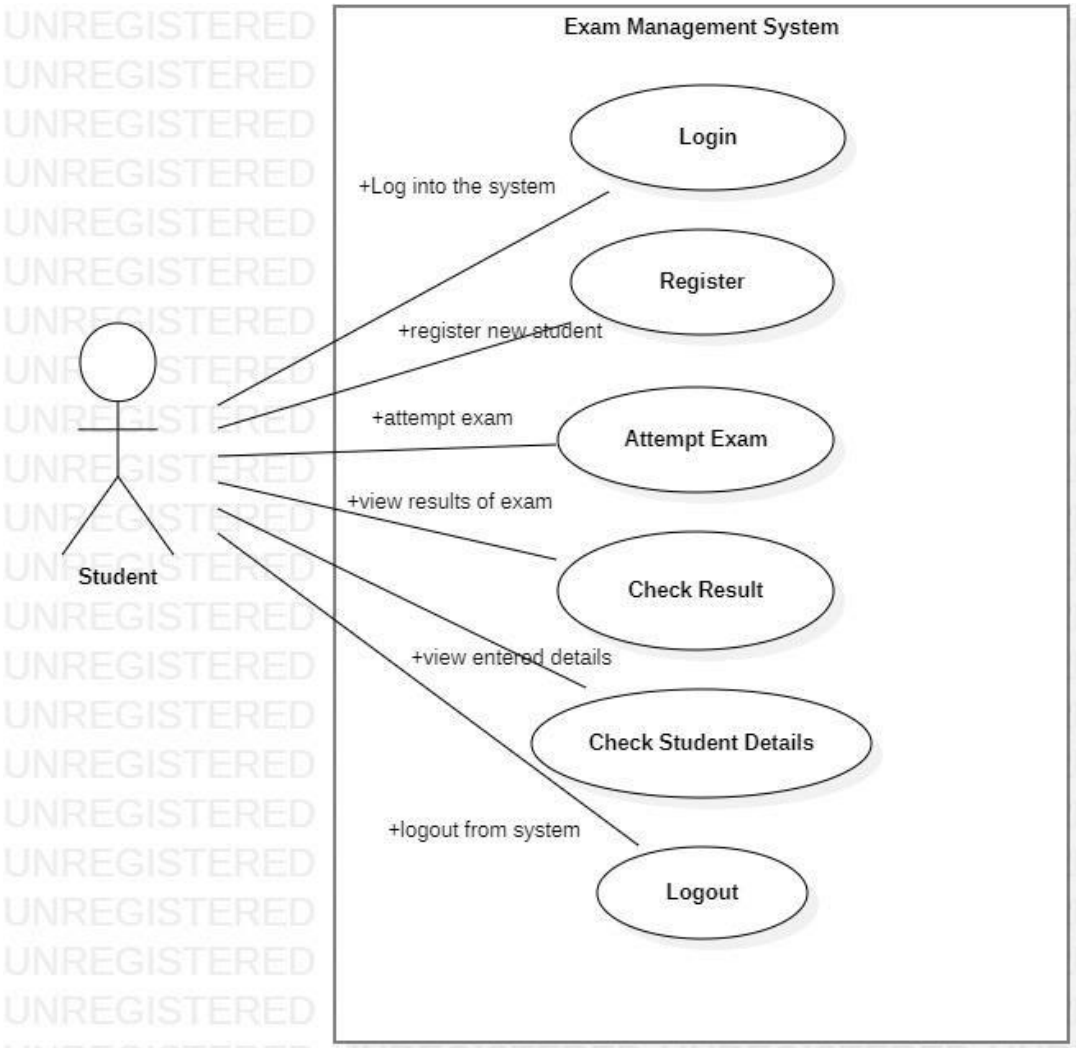
3.3 Use Case Diagram

A use case diagram shows a set of use cases and Actors (a special kind of class) and their relationships. Use case diagrams address the static use case view of a system. These diagrams are especially important in organizing and modelling the behaviours of a system.

- **Admin**

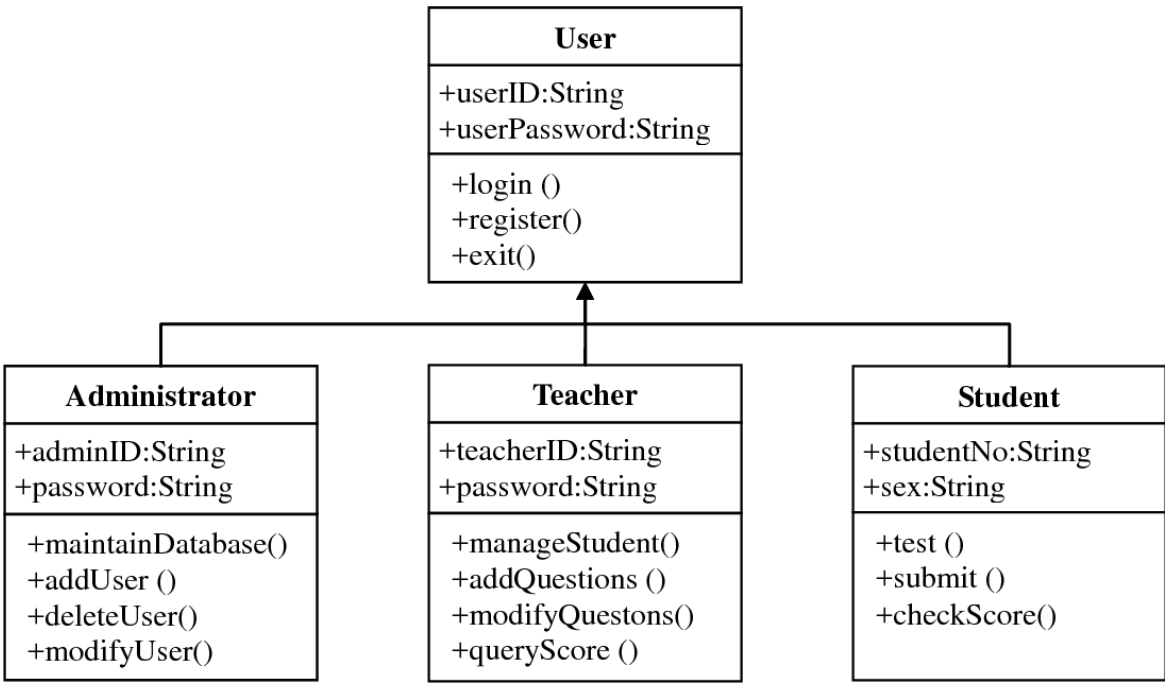


- **Student**



3.4 Class Diagram

A class diagram shows a set of classes, interfaces, and collaborations and their relationships. Class diagrams address the static design view of a system. Class diagrams that include active classes address the static process view of a system. A class is a description of a set of objects that share the same attributes, operations, relationships, and semantics. A class implements on or more interfaces.

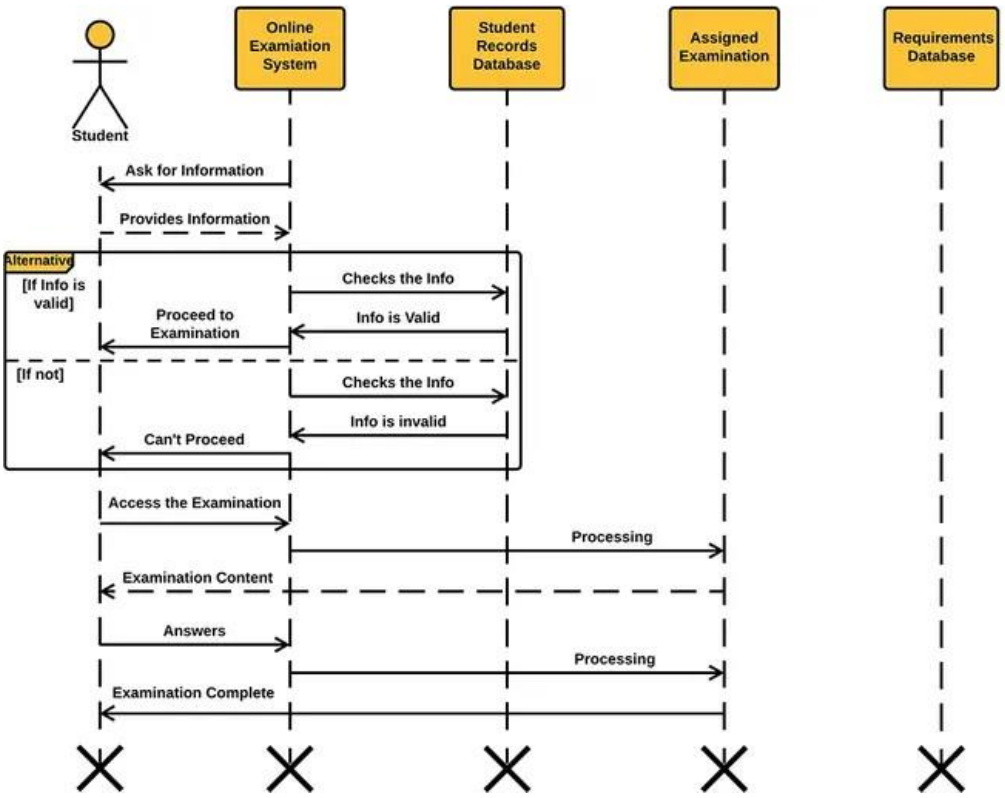


3.5 Sequence Diagram

The sequence diagram is an interaction diagram that emphasizes the time ordering of messages for modelling a real time system. Graphically, a sequence diagram is a table that shows objects arranged along the X-axis and messages, ordered in increasing time, along the Y-axis. Sequence Diagram consists of objects, links, lifeline, focus of control, and messages.

It has two features they are:

- This is the object life time
- There is the focus of control


Student
Online Examiation System
Student Records Database
Assigned Examination
Requirements Database
Ask for Information
Provides Information
Alternative
[If Info is valid]
Checks the Info
Info is Valid
Proceed to Examination
[If not]
Checks the Info
Info is invalid
Can't Proceed
Access the Examination
Processing
Examination Content
Answers
Processing
Examination Complete

CHAPTER – 4

SYSTEM DESIGN

In this phase, a logical system is built which fulfills the given requirements. The design phase of software development deals with transforming the clients' requirements into a logically working system. Normally, design is performed in the following two steps:

4.1 Primary Design Phase

In this phase, the system is designed at the block level. The blocks are created based on analysis done in the problem identification phase. Different blocks are created for different functions emphasis is put on minimizing the information flow between blocks. Thus, all activities which require more interaction are kept in one block.

4.2 Secondary Design Phase:

In the second phase, the detailed design of every block is performed.

The general tasks involved in the design process are the following:

1. Design various blocks for overall system processes.
2. Design smaller, compact and workable modules in each block
3. Design various database structures
4. Specify details of programs to achieve the desired functionality
5. Design the form of inputs, and outputs of the system.
6. Perform documentation of the design.
7. System reviews.

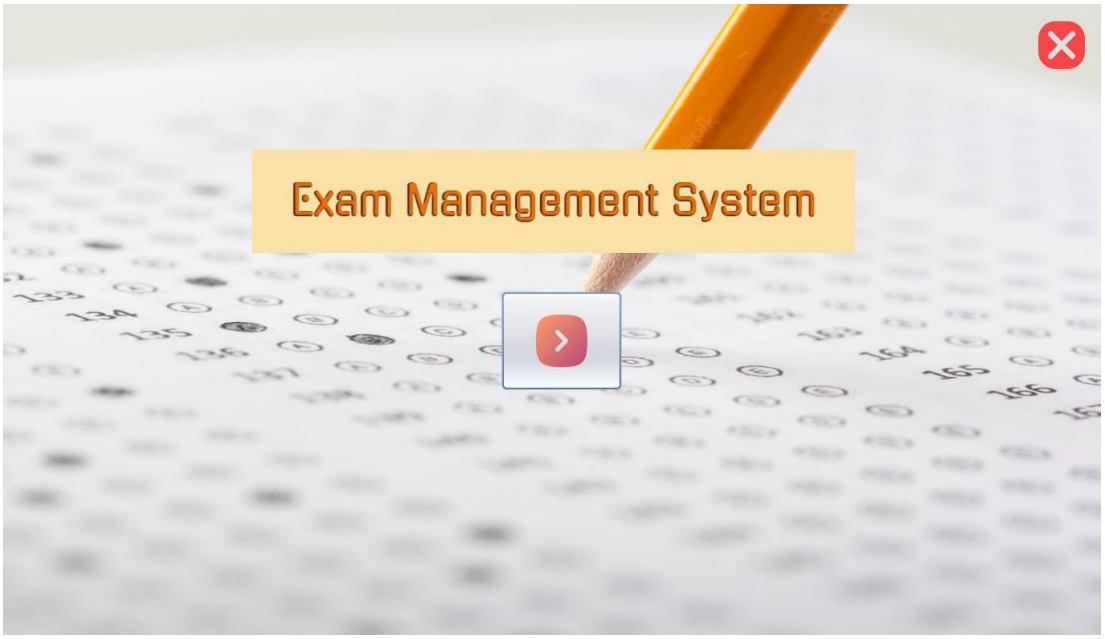
CHAPTER – 5

USER INTERFACE DESIGN

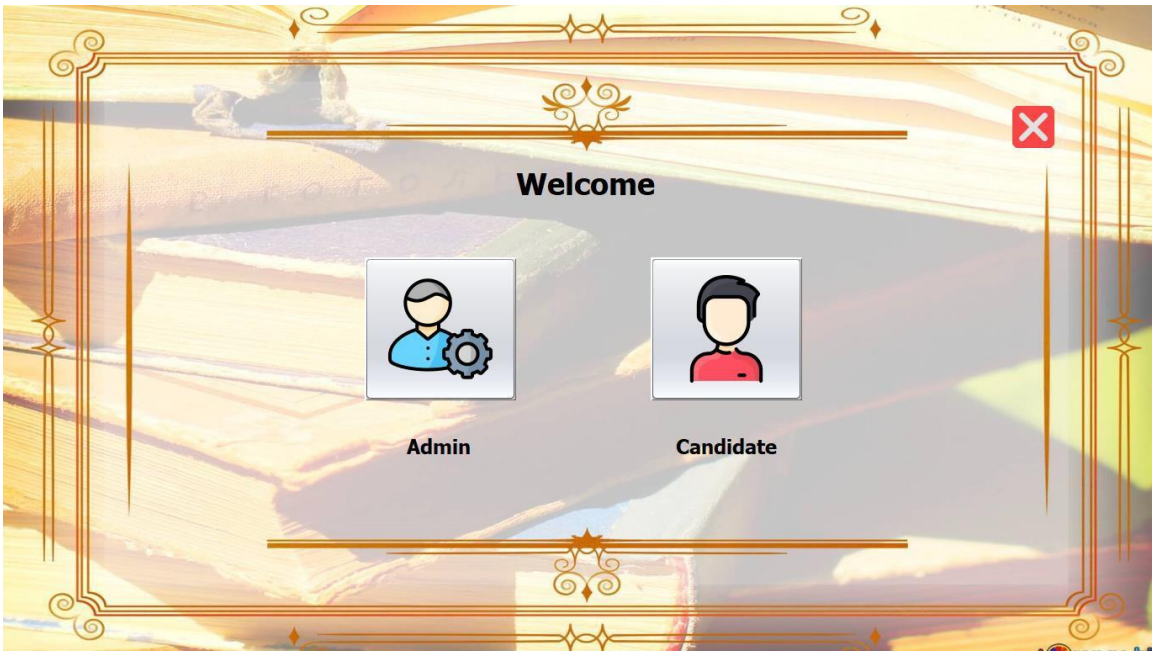
User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventual presentation of desired inputs and outputs. The overall flow of screens and messages is called dialogue.

The following steps are various guidelines for User Interface Design:

1. **Welcome Page**

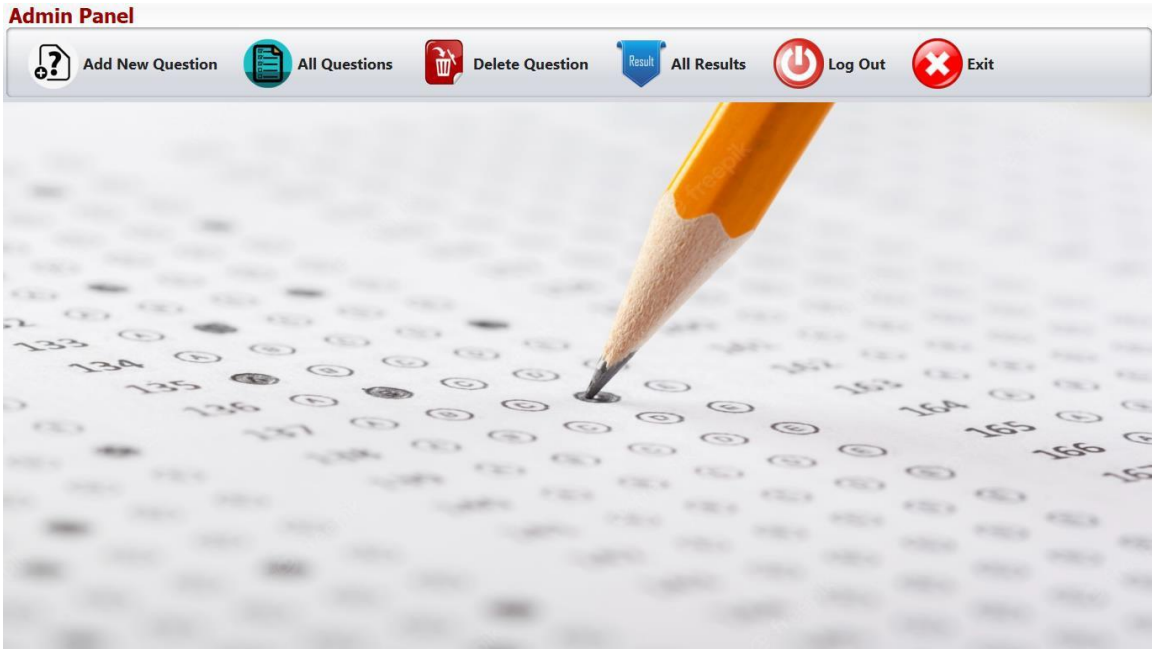


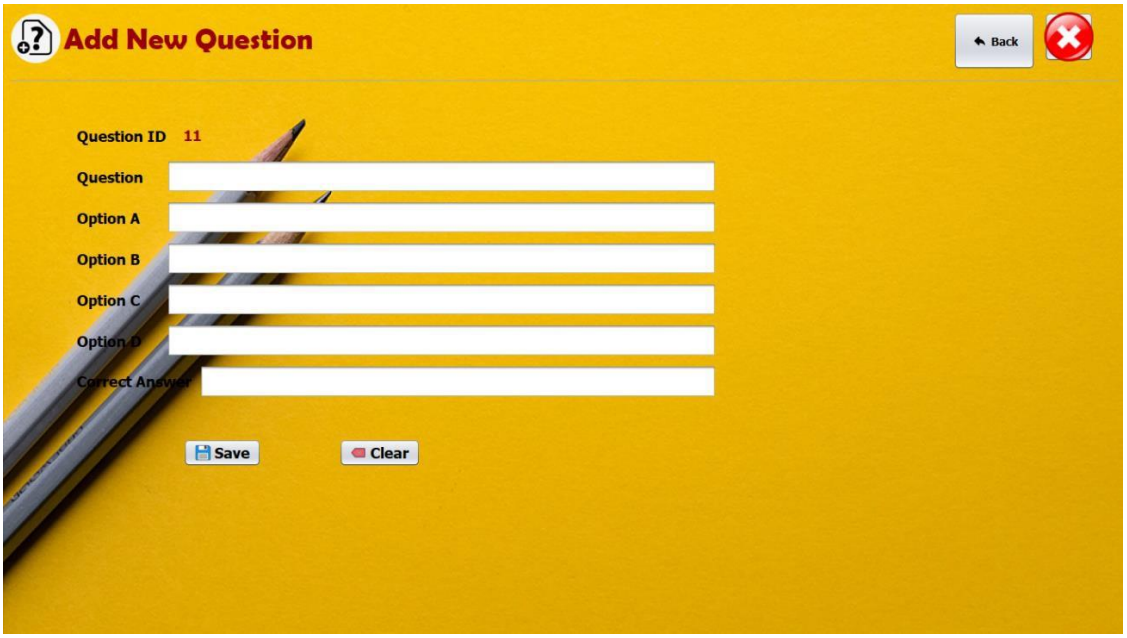
2. Index Page



3. Admin Login

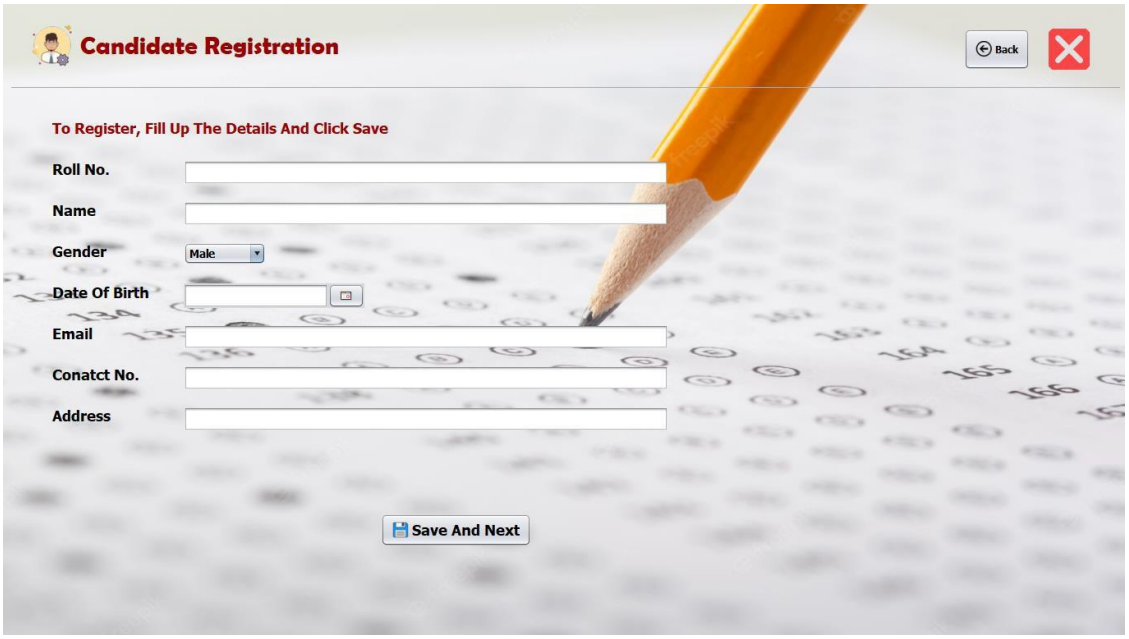
4. Admin Panel



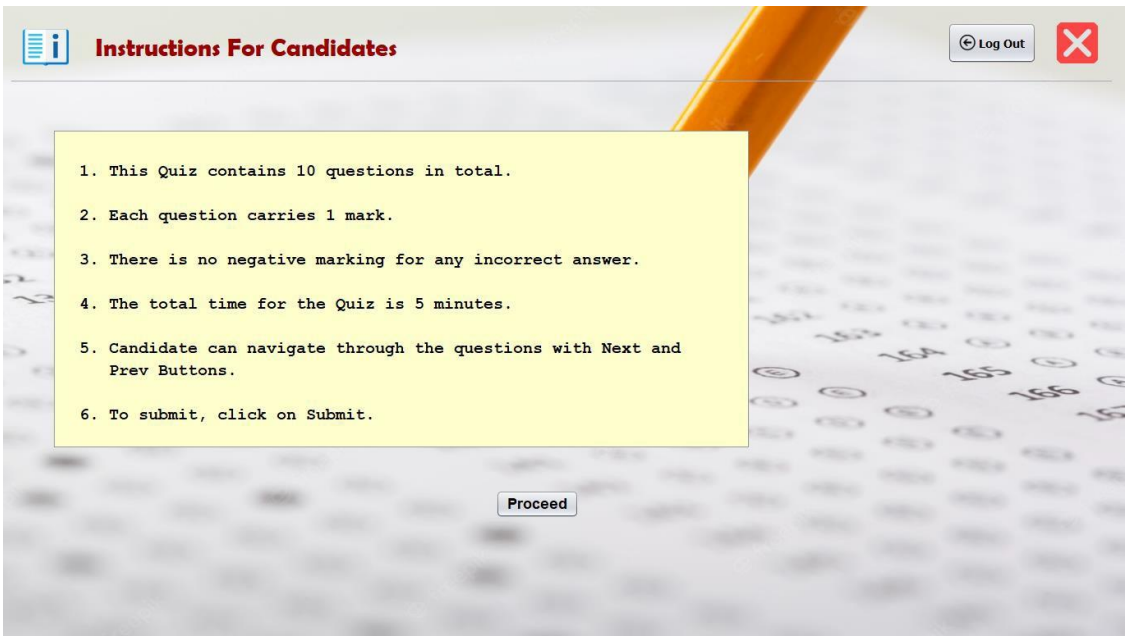


5. Candidate Login

6. Candidate Register

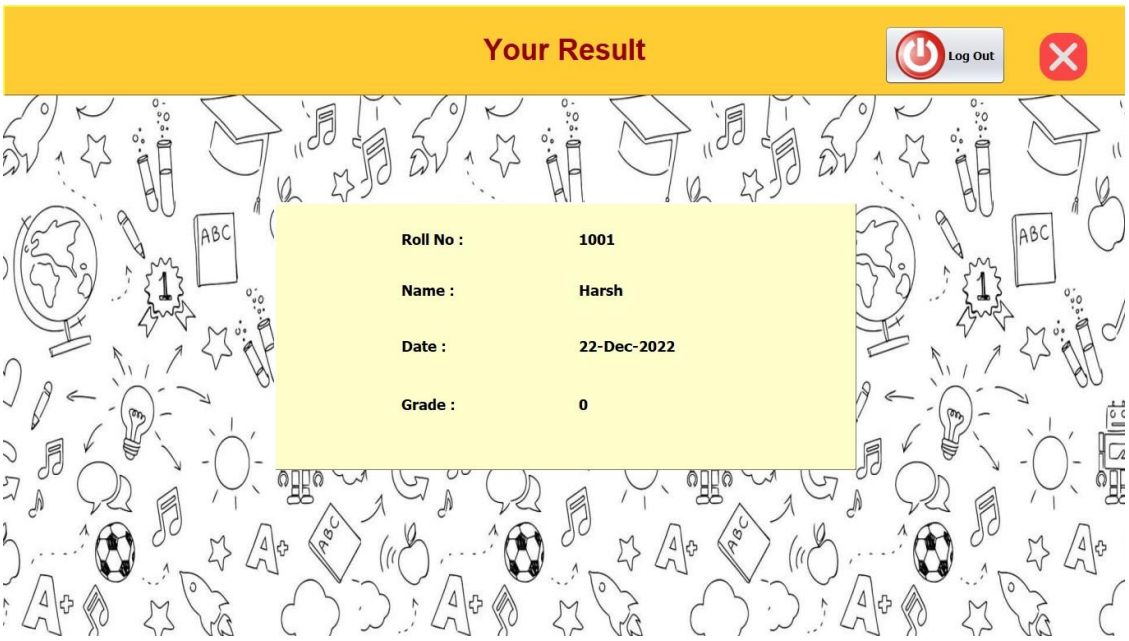


7. Instruction Page



8. Attempt Exam

9. Result page



CHAPTER – 6

CODING

Introduction of Coding

At the most basic level, programming is a broader discipline whereas coding is a narrower one. Coding involves writing many lines of code in order to create a software program Programming involves not only coding but also other tasks. such as analytical and implementing algorithms, understanding data structures, solving problems, and more. Programmers are typically technically-minded and have strong analytical skills. To put it simply, all programmers are coders but not all coders are programmers. Some experienced programmers use the word "coder" as jargon that refers to a beginner (junior) software developer.

- **Approaches Used**

Top-down and bottom-up approach are strategies of information processing and knowledge ordering. A top-down approach is essentially breaking down a system to gain insight into its subsystems. In a top-down approach an overview of the system is first formulated, specifying but not detailing any first-level subsystems. Each subsystem is then refined in yet greater detail, sometimes in many additional subsystem levels, until the entire specification is reduced to base elements.

A bottom-up approach is essentially piecing together systems to give rise to grander systems, thus making the original systems sub-systems of the emergent system. In a bottom-up approach the individual base elements of the system are first specified in great detail. These elements are then linked together to form larger subsystems, which then in turn are linked sometimes in many levels, until a complete top-level system is formed.

6.1 Index Page

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

class index implements ActionListener
{
JButton ba,bp;
JFrame f;
```

```
index()
{
f=new JFrame("EMS");
f.setSize(1400,750);
f.setResizable(false);
f.setLocationRelativeTo(null);
f.setLayout(null);

ImageIcon img=new ImageIcon("adminlogo.png");
ba=new JButton(img);
ba.setBounds(0,0,700,750);
ba.addActionListener(this);
f.add(ba);

ImageIcon img1=new ImageIcon("candidate.png");
bp=new JButton(img1);
bp.setBounds(700,0,700,750);
bp.addActionListener(this);
f.add(bp);
f.setVisible(true);
}

public void actionPerformed(ActionEvent bb)
{
if(bb.getSource()==ba)
{
adminlogin adl=new adminlogin();
f.dispose();
}
else if(bb.getSource()==bp)
{
playerlogin pl=new playerlogin();
f.dispose();
}
}
}
class index1
{
public static void main(String[] s)
{
index id=new index();
}
}
```

6.2 Admin Module

- **Admin Login**

```
package ems;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

class adminlogin extends JFrame implements ActionListener,FocusListener
{
JFrame f;
JPanel p;
JLabel lmain,l2,imglabel,imglabel1;
JButton b1,b2;
JTextField t1;
JPasswordField t2;
Font f1,f2;
String user,pwd;
Connection con;
PreparedStatement st;
ResultSet rs;

adminlogin()
{
f=new JFrame("Login...");
f.setSize(1400,750);
f.setResizable(false);
f.setLayout(null);
f.setLocationRelativeTo(null);

Font f1=new Font("Algerian",Font.BOLD,30);

ImageIcon img1 = new ImageIcon("adminlogin.png");
imglabel1=new JLabel("",img1,JLabel.CENTER);
imglabel1.setBounds(0,0,1400,750);
f.add(imglabel1);

p=new JPanel();
p.setBounds(900,120,400,300);
p.setLayout(null);
p.setBackground(Color.white);
imglabel1.add(p);

ImageIcon img = new ImageIcon("bb2.png");
```

```
imglabel=new JLabel(img);
imglabel.setBounds(80,5,200,80);
p.add(imglabel);

f1=new Font("Algerian",Font.BOLD,30);
f2=new Font("arial",Font.BOLD,15);

lmain=new JLabel("Admin Login");
lmain.setBounds(400,10,250,30);
lmain.setFont(f1);
lmain.setForeground(Color.white);
imglabel1.add(lmain);

t1=new JTextField("Enter User Name :");
t1.setBounds(35,100,320,45);
t1.addFocusListener(this);
p.add(t1);

t2=new JPasswordField("Enter Password :");
t2.setBounds(35,160,320,45);
t2.addFocusListener(this);
p.add(t2);

b1=new JButton("Login");
b1.setBounds(35,230,100,30);
b1.addActionListener(this);
p.add(b1);

b2=new JButton("Back");
b2.setBounds(255,230,100,30);
b2.addActionListener(this);
p.add(b2);

JSeparator sep=new JSeparator();
sep.setOrientation(SwingConstants.HORIZONTAL);
sep.setBounds(20,270,340,10);
p.add(sep);

try
{
Class.forName("com.mysql.cj.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/bbfinal","root","");
}
catch(Exception e)
{
System.out.print(e);
}
```

```
b1.requestFocus();
f.setVisible(true);
}

public void focusGained(FocusEvent fe)
{
if(fe.getSource()==t1)
{
t1.selectAll();
t1.setFont(f2);
}
else if(fe.getSource()==t2)
{
t2.selectAll();
t2.setFont(f2);
}
}

public void focusLost(FocusEvent fe)
{
}

public void actionPerformed(ActionEvent ae)
{
if(ae.getSource()==b2)
{
index id=new index();
f.dispose();
}
else if(ae.getSource()==b1)
{

user=t1.getText();
pwd=t2.getText();
try
{
st=con.prepareStatement("select * from admin where user_name=? and password=?");

st.setString(1,user);
st.setString(2,pwd);

rs=st.executeQuery();
if(rs.next())
{
admin a=new admin();
f.dispose();
}
```

```
else
{
JOptionPane.showMessageDialog(f,"Invalid User Name or Password");
}
}
catch(Exception e)
{
System.out.print(e);
}
}
}
}

class adminlogin1
{
public static void main(String[] s)
{
adminlogin adl=new adminlogin();
}
}
```

- **Add New Question**

```
import java.awt.*;
import javax.swing.*;
import java.sql.*;
import java.awt.event.*;

class addnewques extends JFrame implements ActionListener
{

JDialog d;
JLabel lmain,l1,l2,l3,l4,l5,l6,l7,lvl,imglabel;
JButton b1,b2;
JTextField t1,t3,t4,t5,t6,t7;
JTextArea tques;
JComboBox cb;
PreparedStatement st;
Connection con;
ResultSet rs;
int q_num,nq;
addnewques(JFrame f)
{
 d=new JDialog(f,"Add New Question",true);
 d.setLayout(null);
 d.setBounds(40,110,1300,600);
```

```
 d.setResizable(false);
ImageIcon img = new ImageIcon("bg1.png");
imglabel=new JLabel("",img,JLabel.CENTER);
imglabel.setBounds(0,0,1300,600);
d.add(imglabel);
Font f1=new Font("Algerian",Font.BOLD,30);
Font f2=new Font("arial",Font.BOLD,15);
lmain=new JLabel("Add New Question");
lmain.setBounds(450,10,300,30);
lmain.setFont(f1);
lmain.setForeground(Color.white);
imglabel.add(lmain);
l1=new JLabel("Question Number :");
l1.setFont(f2);
l1.setBounds(30,90,150,30);
l1.setForeground(Color.white);
imglabel.add(l1);

t1=new JTextField();
t1.setEditable(false);
t1.setBounds(230,90,200,30);
t1.setFont(f2);
imglabel.add(t1);

l2=new JLabel("Question :");
l2.setFont(f2);
l2.setBounds(30,140,100,30);
l2.setForeground(Color.white);

imglabel.add(l2);
tques=new JTextArea();
tques.setFont(f2);
tques.setBounds(230,140,650,60);
imglabel.add(tques);

l3=new JLabel("Option 1 :");
l3.setFont(f2);
l3.setBounds(30,220,100,30);
l3.setForeground(Color.white);
imglabel.add(l3);

t3=new JTextField();
t3.setBounds(230,220,250,30);
t3.setFont(f2);
imglabel.add(t3);

l4=new JLabel("Option 2 :");
```

```
l4.setFont(f2);
l4.setBounds(510,220,100,30);
l4.setForeground(Color.white);
imglabel.add(l4);

t4=new JTextField();
t4.setBounds(630,220,250,30);
t4.setFont(f2);
imglabel.add(t4);

l5=new JLabel("Option 3 :");
l5.setFont(f2);
l5.setBounds(30,270,100,30);
l5.setForeground(Color.white);
imglabel.add(l5);

t5=new JTextField();
t5.setBounds(230,270,250,30);
t5.setFont(f2);
imglabel.add(t5);

l6=new JLabel("Option 4 :");
l6.setFont(f2);
l6.setBounds(510,270,100,30);
l6.setForeground(Color.white);
imglabel.add(l6);

t6=new JTextField();
t6.setBounds(630,270,250,30);
t6.setFont(f2);
imglabel.add(t6);

l7=new JLabel("Correct Option :");
l7.setFont(f2);
l7.setBounds(30,320,150,30);
l7.setForeground(Color.white);
imglabel.add(l7);


b1=new JButton("Save");
b1.setBounds(320,420,90,30);
b1.setFont(f2);
imglabel.add(b1);
b1.addActionListener(this);

b2=new JButton("Clear");
b2.setBounds(690,420,90,30);
```

```
b2.setFont(f2);
imglabel.add(b2);
b2.addActionListener(this);

try
{
Class.forName("com.mysql.cj.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost/test ","root","");
st=con.prepareStatement("select max(q_num) from question");
rs=st.executeQuery();
rs.next();
q_num=rs.getInt(1);
if(q_num==0)
q_num=1;
else
q_num++;
t1.setText(String.valueOf(q_num));

st=con.prepareStatement("select lvl_name from exam_level");
rs=st.executeQuery();
cb.addItem("Select level");
while(rs.next())
{
cb.addItem(rs.getString("lvl_name"));
}
}
catch(Exception e)
{
System.out.print(e);
}
d.setVisible(true);
tques.requestFocus();
}

public void actionPerformed(ActionEvent rm)
{
if(rm.getSource()==b2)
{
tques.setText("");
t3.setText("");
t4.setText("");
t5.setText("");
t6.setText("");
t7.setText("");

tques.requestFocus();
}
```

```
else if(rm.getSource()==b1)
{
String question,opt1,opt2,opt3,opt4,rt_opt,lvl_name;

q_num=Integer.parseInt(t1.getText());
question=tques.getText();
opt1=t3.getText();
opt2=t4.getText();
opt3=t5.getText();
opt4=t6.getText();
rt_opt=t7.getText();
lvl_name=(String) cb.getSelectedItem();

try
{
st=con.prepareStatement("insert into question values(?,?,?,?,?,?,?,?)");
st.setInt(1,q_num);
st.setString(2,question);
st.setString(3,opt1);
st.setString(4,opt2);
st.setString(5,opt3);
st.setString(6,opt4);
st.setString(7,rt_opt);
st.setString(8,lvl_name);
st.executeUpdate();

JOptionPane.showMessageDialog(this," Question Added  Successfully");

st=con.prepareStatement("select NOQ from exam_level where lvl_name=?");
st.setString(1,lvl_name);
rs=st.executeQuery();
if(rs.next())
{
nq=rs.getInt(1);
nq++;

st=con.prepareStatement("update exam_level set NOQ=? where lvl_name=?");
st.setInt(1,nq);
st.setString(2,lvl_name);
st.executeUpdate();
}

tques.setText("");
t3.setText("");
t4.setText("");
t5.setText("");
```

```
t6.setText("");
t7.setText("");
q_num++;
t1.setText(String.valueOf(q_num));

tques.requestFocus();
}
catch(Exception e)
{
System.out.print(e);
}
}
}
}
```

- **Delete Question**

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

class modques extends JFrame implements ActionListener
{
JDialog d;
JLabel lmain,l1,l2,l3,l4,l5,l6,l7,lvl,imglabel;
JButton b1,b2;
JTextField t3,t4,t5,t6,t7;
JTextArea tques;
JComboBox cb;
JRadioButton r1,r2;
ButtonGroup grp;
Connection con;
ResultSet rs;
PreparedStatement st;
boolean flag=false;
int q_num, nq,nqp;
String question,opt1,opt2,opt3,opt4,rt_opt,lvl_name,lvlnm;

modques(JFrame f)
{
d=new JDialog(f,"Modify Question",true);
d.setBounds(60,110,1200,600);
d.setResizable(false);
d.setLayout(null);
```

```
ImageIcon img = new ImageIcon("bg1.png");
imglabel=new JLabel("",img,JLabel.CENTER);
imglabel.setBounds(0,0,1200,600);
d.add(imglabel);

Font f1=new Font("Algerian",Font.BOLD,30);
Font f2=new Font("arial",Font.BOLD,15);

lmain=new JLabel("Modify Question");
lmain.setBounds(400,10,300,30);
lmain.setFont(f1);
lmain.setForeground(Color.white);
imglabel.add(lmain);

l1=new JLabel("Question Number :");
l1.setFont(f2);
l1.setBounds(30,90,150,30);
l1.setForeground(Color.white);
imglabel.add(l1);

cb=new JComboBox();
cb.setBounds(230,90,200,30);
cb.setFont(f2);
cb.addActionListener(this);
imglabel.add(cb);

l2=new JLabel("Question :");
l2.setFont(f2);
l2.setBounds(30,140,100,30);
l2.setForeground(Color.white);
imglabel.add(l2);

tques=new JTextArea();
tques.setFont(f2);
tques.setBounds(230,140,650,60);
tques.setLineWrap(true);
imglabel.add(tques);


t6=new JTextField();
t6.setBounds(630,270,250,30);
t6.setFont(f2);
imglabel.add(t6);

l7=new JLabel("Correct Option :");
l7.setFont(f2);
l7.setBounds(30,320,150,30);
```

```
l7.setForeground(Color.white);
imglabel.add(l7);

t7=new JTextField();
t7.setBounds(230,320,250,30);
t7.setFont(f2);
imglabel.add(t7);

lvl=new JLabel("Select Level :");
lvl.setFont(f2);
lvl.setBounds(510,90,150,30);
lvl.setForeground(Color.white);
imglabel.add(lvl);

ButtonGroup grp=new ButtonGroup();
grp.add(r1);
grp.add(r2);


b1=new JButton("Update");
b1.setBounds(320,420,90,30);
b1.setFont(f2);
b1.addActionListener(this);
imglabel.add(b1);

b2=new JButton("Delete");
b2.setBounds(690,420,90,30);
b2.setFont(f2);
b2.addActionListener(this);
imglabel.add(b2);

try
{
Class.forName("com.mysql.cj.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost/brain_baazi","root","");
st=con.prepareStatement("select q_num from question");
rs=st.executeQuery();
cb.addItem("Select Question ID");
while(rs.next())
{
cb.addItem(rs.getInt(1));
}
flag=true;
}
catch(Exception e)
{
System.out.print(e);
```

```
}

lvlnm=lvl_name;


d.setVisible(true);
tques.requestFocus();
}

public void actionPerformed(ActionEvent rm)
{
if((rm.getSource()==cb) && (flag==true))
{
q_num=(Integer) cb.getSelectedItem();

}
else
 JOptionPane.showMessageDialog(this,"Invalid Question Number");
}


catch(Exception e)
{
System.out.print(e);
}
}
else if(rm.getSource()==b2)
{
q_num=(Integer) cb.getSelectedItem();
try
{
st=con.prepareStatement("delete from question where q_num=?");
st.setInt(1,q_num);
st.executeUpdate();
JOptionPane.showMessageDialog(this,"Question Deleted Successfully");

st=con.prepareStatement("select NOQ from exam_level where lvl_name=?");
st.setString(1,lvl_name);
rs=st.executeQuery();
if(rs.next())
{
nq=rs.getInt(1);
nq--;

st=con.prepareStatement("update exam_level set NOQ=? where lvl_name=?");
st.setInt(1,nq);
st.setString(2,lvl_name);
```

```
st.executeUpdate();
}
cb.removeItem(cb.getSelectedItem());
tques.setText("");
t3.setText("");
t4.setText("");
t5.setText("");
t6.setText("");
t7.setText("");
cb.setSelectedIndex(0);
r1.setSelected(false);
r2.setSelected(false);
tques.requestFocus();
}
catch(Exception e)
{
System.out.print(e);
}
}
else if(rm.getSource()==b1)
{
question=tques.getText();
opt1=t3.getText();
opt2=t4.getText();
opt3=t5.getText();
opt4=t6.getText();
rt_opt=t7.getText();
if(r1.isSelected())
lvl_name="Beginner";
else if(r2.isSelected())
lvl_name="Pro";


try
{
st=con.prepareStatement("update question set
question=?,opt1=?,opt2=?,opt3=?,opt4=?,rt_opt=?,lvl_name=? where q_num=?");

st.setString(1,question);
st.setString(2,opt1);
st.setString(3,opt2);
st.setString(4,opt3);
st.setString(5,opt4);
st.setString(6,rt_opt);
st.setString(7,lvl_name);
st.setInt(8,q_num);
st.executeUpdate();
```

```
JOptionPane.showMessageDialog(this,"Question Updated Successfully");

st=con.prepareStatement("select NOQ from exam_level where lvl_name='Beginner'");
rs=st.executeQuery();
if(rs.next())
{
nq=rs.getInt(1);

st=con.prepareStatement("select NOQ from exam_level where lvl_name='Pro'");
rs=st.executeQuery();
if(rs.next())
{
nqp=rs.getInt(1);

if(lvl_name.equalsIgnoreCase(lvlnm))
{

}
else
{
if(lvl_name=="Beginner")
{
nq++;
nqp--;
}
else
{
nq--;
nqp++;
}
}
st=con.prepareStatement("update exam_level set NOQ=? where lvl_name='Beginner'");
st.setInt(1,nq);

st.executeUpdate();
st=con.prepareStatement("update exam_level set NOQ=? where lvl_name='Pro'");
st.setInt(1,nqp);

st.executeUpdate();
tques.setText("");
t3.setText("");
t4.setText("");
t5.setText("");
t6.setText("");
t7.setText("");
r1.setSelected(false);
```

```
r2.setSelected(false);
tques.requestFocus();
}
}

}
catch(Exception e)
{
System.out.print(e);
}
}

}

}
```

6.3 Candidate Module

- **Candidate Registration**

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;


class playerReg extends JFrame implements KeyListener, ActionListener
{
JDialog d;
JLabel lmain,l1,l2,l3,l4,l5,l6,l7,imglabel;
JTextField t1,t2,t3,t4,t5;
JTextArea ta1;
JRadioButton r1,r2,r3;
ButtonGroup grp;
JButton b1;

Connection con;
ResultSet rs;
PreparedStatement st;
String name,DOB,email,c_num,gender,address;
int reg_id, pid2;

playerReg(JFrame f, int pid1)
{
d=new JDialog(f,"Modify Registration",true);
```

```
d.setLayout(null);
d.setBounds(150,110,1000,600);
d.setResizable(false);

ImageIcon img = new ImageIcon("bg1.png");
imglabel=new JLabel(img);
imglabel.setBounds(0,0,1000,600);
d.add(imglabel);

Font f1=new Font("Algerian",Font.BOLD,30);
Font f2=new Font("arial",Font.BOLD,15);

pid2=pid1;

lmain=new JLabel("Player Modification Form");
lmain.setFont(f1);
lmain.setBounds(300,20,500,40);
lmain.setForeground(Color.white);
imglabel.add(lmain);

l1=new JLabel("Find by Registration ID :");
l1.setFont(f2);
l1.setBounds(40,130,200,20);
l1.setForeground(Color.white);
imglabel.add(l1);

t1=new JTextField("");
t1.setBounds(240,130,300,20);
t1.setFont(f2);
t1.setEditable(false);
imglabel.add(t1);

l2=new JLabel("Name of Player :");
l2.setFont(f2);
l2.setBounds(40,170,150,20);
l2.setForeground(Color.white);
imglabel.add(l2);

t2=new JTextField("");
t2.setFont(f2);
t2.setBounds(240,170,300,20);
t2.addKeyListener(this);
imglabel.add(t2);

l3=new JLabel("Date Of Birth :");
l3.setFont(f2);
l3.setBounds(40,210,150,20);
```

```
l3.setForeground(Color.white);
imglabel.add(l3);

t3=new JTextField("");
t3.setFont(f2);
t3.setBounds(240,210,300,20);
imglabel.add(t3);

l4=new JLabel("Contact Number :");
l4.setFont(f2);
l4.setBounds(40,250,150,20);
l4.setForeground(Color.white);
imglabel.add(l4);

t4=new JTextField("");
t4.setFont(f2);
t4.setBounds(240,250,300,20);
t4.addKeyListener(this);
imglabel.add(t4);

l5=new JLabel("E-mail ID :");
l5.setFont(f2);
l5.setBounds(40,290,150,20);
l5.setForeground(Color.white);
imglabel.add(l5);

t5=new JTextField("");
t5.setFont(f2);
t5.setBounds(240,290,300,20);
imglabel.add(t5);

l6=new JLabel("Address :");
l6.setFont(f2);
l6.setBounds(40,330,150,20);
l6.setForeground(Color.white);
imglabel.add(l6);

ta1=new JTextArea("");
ta1.setFont(f2);
ta1.setLineWrap(true);
ta1.setBounds(240,330,300,50);
imglabel.add(ta1);

l7=new JLabel("Gender :");
l7.setFont(f2);
l7.setBounds(40,400,100,20);
l7.setForeground(Color.white);
```

```
imglabel.add(l7);

r1=new JRadioButton("Male");
r1.setFont(f2);
r1.setBounds(240,400,60,20);
imglabel.add(r1);

r2=new JRadioButton("Female");
r2.setFont(f2);
r2.setBounds(320,400,80,20);
imglabel.add(r2);

r3=new JRadioButton("Others");
r3.setFont(f2);
r3.setBounds(420,400,80,20);
imglabel.add(r3);

ButtonGroup grp=new ButtonGroup();
grp.add(r1);
grp.add(r2);
grp.add(r3);

b1=new JButton("Update");
b1.setFont(f2);
b1.addActionListener(this);
b1.setBounds(260,490,200,30);
imglabel.add(b1);

try
{
Class.forName("com.mysql.cj.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost/bbfinal","root","");

st=con.prepareStatement("select * from player where reg_id=?");
st.setInt(1,pid2);
rs=st.executeQuery();
if(rs.next())
{
name=rs.getString(2);
DOB=rs.getString(3);
email=rs.getString(4);
address=rs.getString(5);
c_num=rs.getString(6);
gender=rs.getString(7);

t1.setText(String.valueOf(pid2));
t2.setText(name);
```

```
t3.setText(DOB);
t4.setText(c_num);
t5.setText(email);
ta1.setText(address);
if(gender.equals("Male"))
r1.setSelected(true);
else if(gender.equals("Female"))
r2.setSelected(true);
else if(gender.equals("Others"))
r3.setSelected(true);
}
}
catch(Exception e)
{
System.out.print(e);
}
d.setVisible(true);
t2.requestFocus();
}

public void keyPressed(KeyEvent ke)
{
char c=ke.getKeyChar();
if(ke.getSource()==t2)
{
if((c>='a' && c<='z') || (c>='A' && c<='Z') || (c==' ') || (c=='\b'))
{
t2.setEditable(true);
}
else
{
t2.setEditable(false);
}
}
else if(ke.getSource()==t4)
{
if((c>='0' && c<='9') || (c=='\b'))
{
t4.setEditable(true);
}
else
{
t4.setEditable(false);
}
}
}
```

```
public void keyReleased(KeyEvent ke)
{
}

public void keyTyped(KeyEvent ke)
{
}

public void actionPerformed(ActionEvent rm)
{
if(rm.getSource()==b1)
{
name=t2.getText();
DOB=t3.getText();
c_num=t4.getText();
email=t5.getText();
address=ta1.getText();
if(r1.isSelected())
gender="Male";
else if(r2.isSelected())
gender="Female";
else if(r3.isSelected())
gender="Others";

try
{
st=con.prepareStatement("update player set
name=?,DOB=?,email=?,address=?,c_num=?,gender=? where reg_id=?");
st.setString(1,name);
st.setString(2,DOB);
st.setString(3,email);
st.setString(4,address);
st.setString(5,c_num);
st.setString(6,gender);
st.setInt(7,pid2);
st.executeUpdate();

JOptionPane.showMessageDialog(this,"Player Registration Details Updated
Successfully");
d.setVisible(false);

t2.requestFocus();
}
catch(Exception e)
{
System.out.print(e);
}
```

```
}
}
}
```

6.4 Exam Module

- **Attempt Exam**

```
import java.awt.*;
import javax.swing.*;
import java.sql.*;
import java.awt.event.*;
import java.text.SimpleDateFormat;
import java.util.Date;
import javax.swing.Timer;

class exam implements ItemListener,ActionListener,MouseListener
{
JFrame f;
JLabel
lmain,imglabel,imglogo,lcd,crd,ltt,tt,ttk,lmin,lsec,tmin,lnm,lnm1,lpid,lpid1,ltq,ltq1,lindx,b
e;
JLabel lques,lans;
JRadioButton r1,r2,r3,r4;
ButtonGroup grp;
JTextField t1;
JButton bnxt,bsub;
int pid3;
String lvl1,qs,opa,opb,opc,opd;
ResultSet rs;
Connection con;
PreparedStatement st;

String nm,sop,cop,crdt;
Timer time;
int min=0,sec=0,scr=0,ind;


exam(int pid2, String lvl)
{
f= new JFrame("Exam");
f.setSize(1400,750);
f.setResizable(false);
f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
f.setLocationRelativeTo(null);
```

```
f.setLayout(null);

ImageIcon img = new ImageIcon("instructionBB.png");
imglabel=new JLabel(img);
imglabel.setBounds(0,0,1400,750);
f.add(imglabel);

ImageIcon img1 = new ImageIcon("candidatevector.png");
imglogo=new JLabel(img1);
imglogo.setBounds(10,0,68,92);
imglabel.add(imglogo);

pid3=pid2;
lvl1=lvl;

Font f1=new Font("Algerian",Font.BOLD,50);
Font f2=new Font("arial",Font.BOLD,18);
Font f3=new Font("arial",Font.BOLD,15);

lmain=new JLabel("WELCOME");
lmain.setBounds(80,35,300,45);
lmain.setForeground(Color.white);
lmain.setFont(f1);
imglabel.add(lmain);

lcd=new JLabel("Date :");
lcd.setBounds(500,40,180,30);
lcd.setForeground(Color.white);
lcd.setFont(f2);
imglabel.add(lcd);

crd=new JLabel();
crd.setBounds(580,40,200,30);
crd.setForeground(Color.white);
crd.setFont(f2);
imglabel.add(crd);

tt=new JLabel("Total Time :");
tt.setBounds(900,10,100,30);
tt.setForeground(Color.white);
tt.setFont(f3);
imglabel.add(tt);

tmin=new JLabel("5 Min");
tmin.setBounds(1002,10,100,30);
tmin.setForeground(Color.white);
tmin.setFont(f3);
```

```
imglabel.add(tmin);


ttk=new JLabel("Time Left :");
ttk.setBounds(900,45,100,30);
ttk.setForeground(Color.white);
ttk.setFont(f3);
imglabel.add(ttk);

lmin=new JLabel();
lmin.setBounds(1002,45,150,30);
lmin.setForeground(Color.red);
lmin.setFont(f2);
imglabel.add(lmin);

ImageIcon icon1=new ImageIcon("exitcirclelogo.png");
be=new JLabel(icon1);
be.setBounds(1270,0,100,100);
be.addMouseListener(this);
imglabel.add(be);

lsec=new JLabel();
lsec.setBounds(1016,45,150,30);
lsec.setForeground(Color.red);
lsec.setFont(f2);
imglabel.add(lsec);

time=new Timer(1000,new ActionListener() {

public void actionPerformed(ActionEvent ae)
{
lsec.setText(String.valueOf(sec));
lmin.setText(String.valueOf(min));

if(sec==60)
{
sec=00;
min++;
if(min==5)
{
time.stop();

try
{
st=con.prepareStatement("insert into score values(?,?,?,?)");
st.setInt(1,pid3);
st.setString(2,lvl1);
```

```
st.setString(3,crdt);
st.setInt(4,scr);
st.executeUpdate();

}
catch(Exception e)
{
System.out.print(e);
}
bnxt.setEnabled(false);
JOptionPane.showMessageDialog(null,"Your Time is Over\n Your  score is : "+ scr);
f.dispose();
player pl=new player(pid3);

}
}
sec++;
}
});
time.start();

JSeparator sep=new JSeparator();
sep.setOrientation(SwingConstants.HORIZONTAL);
sep.setBounds(0,90,1400,20);
sep.setForeground(Color.white);
imglabel.add(sep);

JSeparator sepv=new JSeparator();
sepv.setOrientation(SwingConstants.VERTICAL);
sepv.setBounds(340,96,20,652);
sepv.setForeground(Color.white);
imglabel.add(sepv);

lpid=new JLabel("Player ID :");
lpid.setBounds(5,200,100,20);
lpid.setFont(f3);
lpid.setForeground(Color.white);
imglabel.add(lpid);

lpid1=new JLabel();
lpid1.setBounds(160,200,100,20);
lpid1.setFont(f2);
lpid1.setForeground(Color.white);
imglabel.add(lpid1);

ButtonGroup grp=new ButtonGroup();
grp.add(r1);
```

```
grp.add(r2);
grp.add(r3);
grp.add(r4);

lans=new JLabel("Selected Answer");
lans.setBounds(400,530,150,30);
lans.setFont(f3);
lans.setForeground(Color.white);
imglabel.add(lans);

bnxt=new JButton("Next");
bnxt.setBounds(450,600,100,30);
bnxt.setFont(f2);
bnxt.addActionListener(this);
imglabel.add(bnxt);

bsub=new JButton("Submit");
bsub.setBounds(1200,600,100,30);
bsub.setFont(f2);
bsub.addActionListener(this);
imglabel.add(bsub);

Date date = new Date();
SimpleDateFormat fdt = new SimpleDateFormat("dd/MMM/yyyy");
crdt= fdt.format(date);
crd.setText(crdt);

try
{
Class.forName("com.mysql.cj.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost/brain_baazi","root","");

st=con.prepareStatement("select * from player where reg_id=?");
st.setInt(1,pid3);
rs=st.executeQuery();
if(rs.next())
{
lpid1.setText(String.valueOf(pid3));
nm=rs.getString(2);
lnm1.setText(nm);
}

st=con.prepareStatement("select * from question where lvl_name=? limit 10");
st.setString(1,lvl1);
rs=st.executeQuery();
while(rs.next())
{
```

```
qs=rs.getString(2);
opa=rs.getString(3);
opb=rs.getString(4);
opc=rs.getString(5);
opd=rs.getString(6);
cop=rs.getString(7);
lques.setText(qs);

ind++;
lindx.setText(String.valueOf(ind));
break;

}
}
catch(Exception e)
{
System.out.print(e);
}
f.setVisible(true);
}

public void actionPerformed(ActionEvent ae)
{

if(ae.getSource()==bnxt)
{
if(sop.equalsIgnoreCase(cop))
{
scr=scr+1;
}

ind++;
lindx.setText(String.valueOf(ind));

try
{
while(rs.next())
{
qs=rs.getString(2);
opa=rs.getString(3);
opb=rs.getString(4);
opc=rs.getString(5);
opd=rs.getString(6);
cop=rs.getString(7);

lques.setText(qs);
```

```
r1.setText(opa);
r2.setText(opb);
r3.setText(opc);
r4.setText(opd);
break;
}
}
catch(Exception e)
{
System.out.print(e);
}


if(ind==15)
{
if(sop.equalsIgnoreCase(cop))
{
scr++;
}
bnxt.setEnabled(false);
bsub.setEnabled(true);
}
else
{
bnxt.setEnabled(true);
bsub.setEnabled(false);
}
}

else if(ae.getSource()==bsub)
{
try
{
st=con.prepareStatement("insert into score values(?,?,?,?)");
st.setInt(1,pid3);
st.setString(2,lvl1);
st.setString(3,crdt);
st.setInt(4,scr);
st.executeUpdate();

}
catch(Exception e)
{
System.out.print(e);
}
JOptionPane.showMessageDialog(null,"Your Score is"+scr);
f.dispose();
```

```
player pl=new player(pid3);
}
else if(ae.getSource()==be)
{
f.dispose();
player pl=new player(pid3);
}
}


public void mouseClicked(MouseEvent me)
{
if(me.getSource()==be)
{
player pl1=new player(pid3);
f.dispose();
}

}
}
```

CHAPTER - 7

SOFTWARE TESTING

Software testing is a critical element of software quality assurance and represents the ultimate reuse of specification. Design and code testing represents interesting anomaly for the software during earlier definition and development phase, it was attempted to build software from an abstract concept to tangible implementation.

Testing plays a critical role in quality assurance for software Due to the limitation of the verification method for the previous phases, design and requirement fault also appear in the code. Testing is used to detect these errors, in addition to the error introduced during coding phase.

There are two method of testing: **functional** and **structural**. In functional testing, the internal logic of the system under testing is not considered and the test cases are decided from the specification or the requirements. It is often called "Black Box Testing". Equivalence class partitioning, boundary analysis, and cause effect graphing are examples of methods for selecting test cases for functional testing. In structural testing, the test cases are decided entirely on the internal logic of the program or module being tested.

The goal here is to test the system design. In system testing and acceptance testing, the entire System is tested. The goal here is to test the requirement themselves. Structural testing can be used for unit testing while at higher level mostly functional testing is used. System testing is a critical phase in systems implementation. Testing of a system hardware device testing and debugging of computer programs and testing information processing procedures. Testing can be done with test data, which attempts to simple all possible conditions that may arise during processing. The plane for testing are prepared and ben implemented.

7.1 Principles of Testing:

(i) All the test should meet party the customer requirements.
(ii) To make our software testing should be performed by third party
(iii) Exhaustive testing is not possible. As we need the optimal amount of testing based on the risk Assessment of the application.
(iv) All the test to be conducted should be planned before implementing it.
(v) It follows Pareto Rule (80/20 rule) which states that 80% of errors comes from 20% of components.
(vi) Start testing with small parts and extend it to large parts.

7.2 Types of Testing:

7.2.1 Unit Testing-

It focuses on smallest unit of software design. In this we test an individual unit or group of inters related units. It is often done by programmer by using sample input and observing its corresponding outputs.

Example:
a) In a program we are checking if loop, method or function is working fine
b) Misunderstood or incorrect, arithmetic precedence.
c) Incorrect initialization.

7.2.2 Integration Testing-

The objective is to take unit tested components and build a program structure that has been dictated by design. Integration testing is testing in which a group of components are combined to produce output.

7.2.3 Alpha Testing-

This is a type of validation testing. It is a type of acceptance testing which is done before the product is released to customers. It is typically done by QA people. Example: When software testing is performed internally within the organization.

7.2.4 Beta Testing-

The beta test is conducted at one or more customer sites by the end-user of the software. This version is released for the limited number of users for testing in real time environment.
Example: When software testing is performed for the limited number of people.

7.2.5 System Testing-

In this software is tested such that it works fine for different operating system .It is covered under the black box testing technique. In this we just focus on required input and output on internal. In this we have security testing, recovery testing stress testing and performance testing without focusing on internal working. In this we have security testing, recovery testing, stress testing and performance testing. Example: This include functional as well as non-functional Testing

7.2.6 Performance Testing-

It is designed to test the run-time performance of software within the context of an integrated system. It is used to test speed and effectiveness of program. Example: Checking number of processor cycle.

7.3 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-

7.3.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software i.e. the module Using the detailed design and the process specifications, testing is done to uncover errors within the boundary of the module. All modules must be successful in the unit test before the start of the integration testing begins.

In this project each service can be thought of a module. There are so many modules like Login and Registration, Admin, Candidate, Exam and Report Module. Each module has been tested by giving different sets of inputs. When developing the module as well as finishing the development, the module works without any error. The inputs are validated when accepting them from the user.

7.3.2 Integration Testing

After unit testing, we have to perform integration testing. The goal here is to see if modules can be integrated properly, the emphasis being on testing interfaces between modules. This testing activity can be considered as testing the design and hence the emphasis on testing module interactions.

In this project the main system is formed by integrating all the modules. When integrating all the modules I have checked whether the integration effects working of any of the services by giving different combinations of inputs with which the two services run perfectly before Integration.

7.3.3 System Testing

Here the entire software system is tested. The reference document for this process is the requirements document, and the goal is to see if software meets its requirements.

• Here entire 'HRRP has been tested against requirements of project and it is checked whether all requirements of project have been satisfied or not.

7.3.4 Acceptance Testing

Acceptance Testing is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here is focused on external behavior of the system: the internal logic of program is not emphasized.

• Test cases should be selected so that the largest number of attributes of an equivalence class is exercised at once. The testing phase is an important part of software development. It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

7.4 Test Cases

"EXAM MANAGEMENT SYSTEM" consisting of these test cases throughout its execution.

- Student registration
- Student login
- Administrator login
- Modifying student details
- Updating questions
- Submission of exam

Test Case No.	Input	Expected Behavior	Observed Behavior	Status P=Passed F=Failed
1.	Register as new Student	Registration Page should be displayed	-do-	P
2.	Register with empty fields	Error message should warn to fill required details	-do-	P
3.	Login as student with wrong credentials	Error message to for invalid details	-do-	P
4.	Login as admin with wrong credentials	Error message to for invalid details	-do-	P
5.	Add a new question	Question should be added to database	-do-	P
6.	Modify student details	Modification message should be displayed	-do-	P
7.	Submit exam automatically as time is up	Time up message and submitted message should be displayed	-do-	P

CHAPTER – 8

CONCLUSION

8.1 Conclusion and Discussion

The project was successfully completed within the time span allotted. All the modules are tested separately and put together to form the main system. Finally the system is tested with real data and everything worked successfully. Thus the system has fulfilled the entire objective identified.

The system had been developed in an attractive dialogs fashion. So user with minimum knowledge about computers can also operate the system easily. It will make easy interactions between users and store. The speed and accuracy are maintained in proper way. Overall the Online System procedure and the features and the customer to save the time effort and price. The system also produces brief result required by the management.

8.2 Future Scope of Project

The development of this project surely prompts many new areas of investigation. This project has wide scope to implement it in any University/Institution having multiple paper there. This project covers all functionalities related to online Examination Hall Hence it can be implemented anywhere else after minute organization level customization. First of all limitations of our project, which has been discussed in previous topic make place for future enhancements. Though that was not the part of objective of our project but it would have great to implement that provided we'd enough time.

BIBLIOGRAPHY

During course of this project, a number of books, projects and websites were referred to. Some of them are as listed as follows:

1. Software Engineering
Author - RS Pressman

2. Java- 5th Edition the Complete Reference
Author - Herbert Schildt
Publishing - TATA McGraw Hill

3. Unified Modelling Language
Author - Grady Booch, James Rumbaugh
Publishing - Pearson Education

4. Data Base Management System
Author - C.J. Date

Web References:

1) **J2EE-Overview:** http://java.sun.com/j2EE/overview.html, http://www.tizag.com
2) **J2EE-Component:** http://java.sun.com/j2ee/blueprints/platform_technologies/component/index.html
3) http://codeproject.com/tips.cs
4) http://www.sqlcommands.com
5) http://www.1000projects.com

Literature Review Chapter

Introduction

I. Brief explanation of what an exam management system is and its importance
II. Purpose of the literature review chapter

Overview of exam management systems

- Definition of exam management systems
- Brief history of exam management systems
- Overview of different types of exam management systems (e.g., paper-based, computer-based, online)

User experience and usability of exam management systems

- Factors that affect the usability of exam management systems (e.g., interface design, ease of use, accessibility)
- Best practices for designing effective exam management systems

Features and functionality of exam management systems

- Overview of common features found in exam management systems (e.g., exam creation, scheduling, grading, reporting)
- Discussion of more advanced features (e.g., analytics, plagiarism detection, adaptive testing)

Research and development of exam management systems

- Overview of current research and development in the field of exam management systems
- Discussion of emerging trends and technologies (e.g., artificial intelligence, blockchain)

Challenges and limitations of exam management systems

- Common challenges and limitations of exam management systems (e.g., technical issues, privacy concerns, cost)
- Strategies for overcoming these challenges

Comparison of different exam management systems

- Comparison of popular exam management systems available in the market (e.g., Blackboard, Moodle, ExamSoft)
- Advantages and disadvantages of each system

Conclusion

- Summary of key points covered in the literature review
- Future directions for research and development of exam management systems.