EXAM HALL SEAT ALLOCATION SYSTEM

A PROJECT REPORT

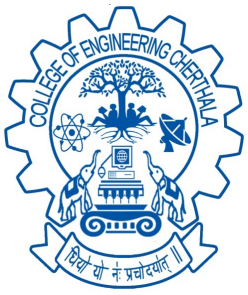
Submitted By

***AABDUL FARDEEN M N** (CEC19CS001)*
***APARNA V S** (CEC19CS013)*
***ARCHANA K PRASAD** (CEC19CS014)*
***ASWATHY B** (CEC19CS016)*

Under the esteemed guidance of

Mrs. GREESHMA N GOPAL

In partial fulfilment of the requirements for the award of the degree of
Bachelor of Technology
in
Computer Science and Engineering
of
A.P.J ABDUL KALAM Technological University



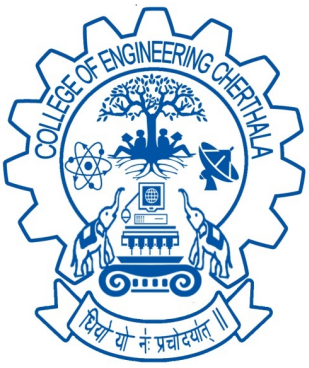
December 2021

Department of Computer Science and Engineering
College of Engineering Cherthala
Pallippuram P O,Alappuzha-688541
Phone: 0478 2553416,Fax: 0478 2552714
http://www.cectl.ac.in

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

COLLEGE OF ENGINEERING CHERTHALA

ALAPPUZHA-688541



C E R T I F I C A T E

This is to certify that, the project report titled ***EXAM HALL SEAT ALLOCATION SYSTEM*** is a bonafide record of the **CSL333 Database Management System Lab- Project** presented by **AABDUL FARDEEN M N** (REG NO: CEC19CS001), **APARNA V S** (REG NO: CEC19CS013), **ARCHANA K PRASAD** (REG NO: CEC19CS014) and **ASWATHY B** (REG NO: CEC19CS016), Fifth Semester B. Tech. Computer Science & Engineering student, under our guidance and supervision, in partial fulfillment of the requirements for the award of the degree, **B. Tech. Computer Science & Engineering** of **APJ Abdul Kalam Technological University**.

Internal Supervisor	**UG Coordinator & Head Of Dept.**	**External Supervisor**
Mrs. Greeshma N Gopal	**Dr. Priya S**	
Assistant Professor	Professor	
Computer Science & Engg.	Computer Science & Engg.	

ACKNOWLEDGEMENT

This work would not have been possible without the support of many people. First and the foremost, I give thanks to Almighty God who gave me the inner strength, resource and ability to complete my mini project successfully.

I would like to thank Dr. Vinu Thomas, our Principal, who has provided with the best facilities for the project completion and presentation. I would also like to thank our HOD Dr. Priya S (Professor, Computer Science and Engineering), our project coordinator Mrs.Greeshma N Gopal (Assistant Professor, Computer Science and Engineering) and Mrs Judy Ann Joy (Assistant Professor, Computer Science and Engineering) the help extended and also for the encouragement and support given to me while doing the project.

I would like to thank my dear friends for extending their cooperation and encouragement throughout the project work, without which I would never have completed the project this well. Thank you all for your love and also for being very understanding.

ABSTRACT

Exam hall seating allocation is one of the major concerns in quality education. With the increasing number of students, subjects, departments and rooms,exam seat management be comes complex.Maintaining a decent exam environment with the proper seating arrangement: is one of the difficult jobs for authority. The proposed system offers solution for seating arrangement problems that is achieved through the sequential execution. The main objective for ping this project is to managing all the requirement for the examinations in a single. platform.It will help in planning seating arrangements for all the students in the examination hall. This project provides a lot of features to manage the exam in very well manner.

Contents

1 INTRODUCTION **1**

2 DESIGN **2**

2.1 ER DIAGRAM 2

2.2 USECASE DIAGRAM 3

2.3 SEQUENTIAL DIAGRAM 4

3 IMPLEMENTATION **5**

3.1 LOGIN PAGE 5

3.2 STUDENT DETAILS 6

3.3 ROOM DETAILS 7

3.4 SELECTION OF STUDENTS AND ROOM 8

3.5 SEAT ALLOCATION 9

3.6 INSERT STUDENT DETAILS 10

3.7 LOGIN PAGE 11

3.8 INSERT STUDENT DETAILS 11

3.9 ROOM DETAILS 12

3.10 SEAT ARRANGEMENT 13

4 HARDWARE AND SOFTWARE REQUIREMENTS **19**

5 CONCLUSION **20**

6 REFERENCES **21**

List of Figures

2.1 ER Diagram 2
2.2 USECASE Diagram 3
2.3 Sequential Diagram 4

3.1 Login Page 5
3.2 Student Details Page 6
3.3 Uploading Room Details 7
3.4 Seating Rooms and Classes 8
3.5 Seat Allocation 9
3.6 Insert Student Details 10

Chapter 1

INTRODUCTION

EXAM HALL SEATING ALLOCATION is Java based software to arrange seats for an examination, Currently all seating arrangements are done manually. This software arranges the students in column wise order so that no two students of the same class appear side by side. Features include batch management, staff management, scheme management, hall management, subject management, department management, exam management, absentee management, seating management, user management, student management, seating arrangement printing, and batch student list printing Above aim can be established by implementing a lay out structure modules and each modules can have many sub modules Admin is responsible for allocating halls to the students Exam halls are allocated according to their exams schedule and date Admin enrolls the department details and staff details for the hall Student can view their seating arrangement through the list.

This management system consists of Admin module. Admin is responsible for

- Access to student details
- Subject and department management
- Allocation process
- View allocation

Chapter 2

DESIGN

2.1 ER DIAGRAM

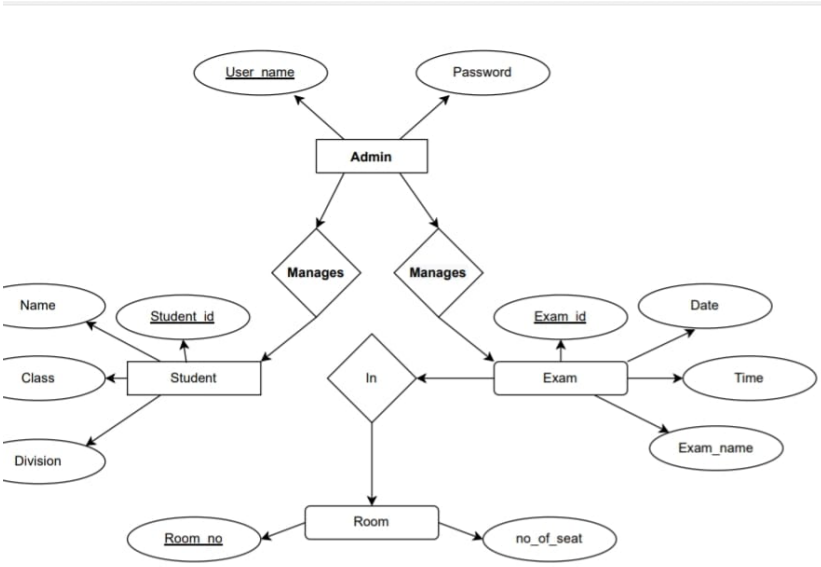


Fig. 2.1: ER Diagram

An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. In this diagram 3NF normalisation is used The diagram shows how the entities are connected Here there are four entities admin.student exam and room.

2.2 USECASE DIAGRAM

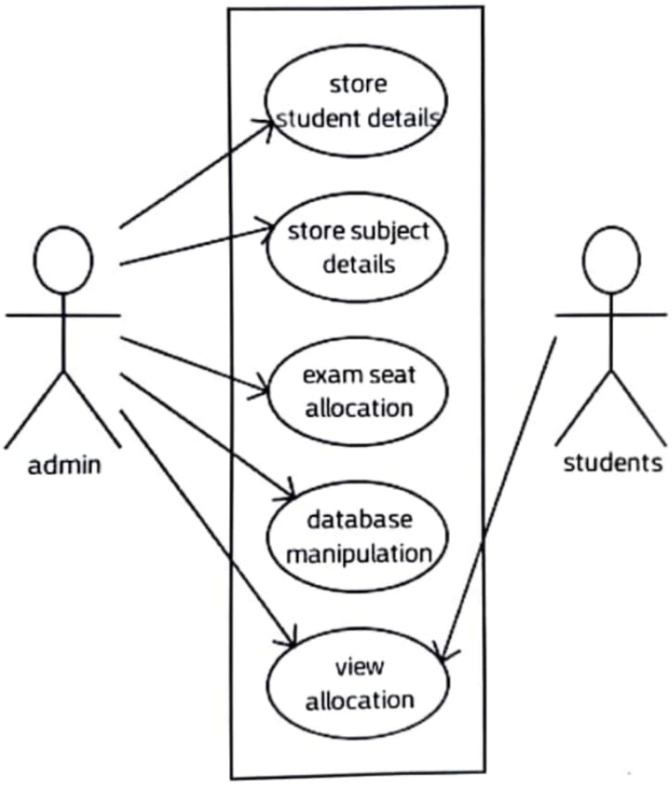


Fig. 2.2: USECASE Diagram

in the above shown Usecase Diagram there are two actors i.e Admin and student. Here the Admin can access to store student details,to store exam details allocate the seats for students have data manipulation of data and also view allocations of the seat after entering the details and student an only view their allocated seat details.allocate the seats for students have data manipulation of data and also view allocations of the seat after entering the details and student an only view their allocated seat

2.3 SEQUENTIAL DIAGRAM

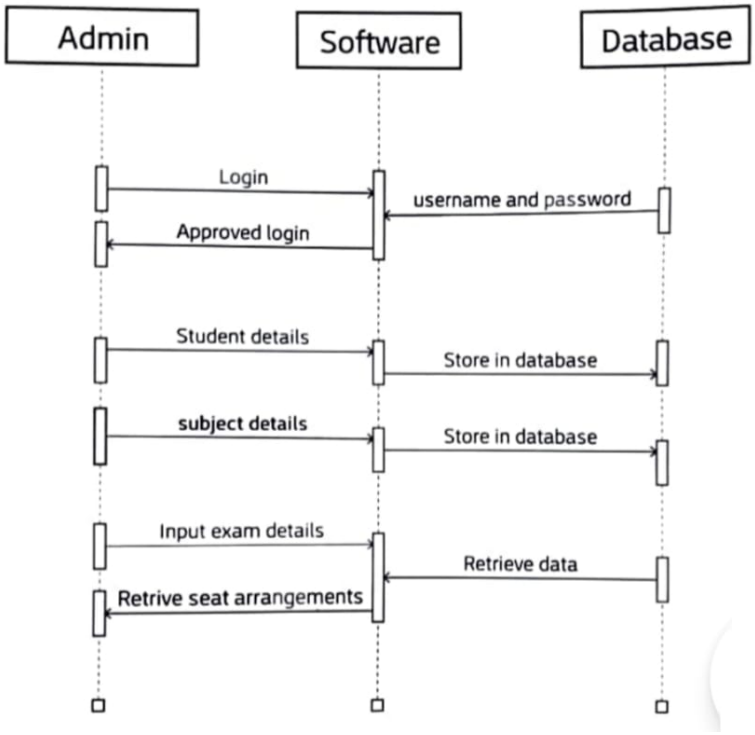


Fig. 2.3: Sequential Diagram

Here the Admin has an User ID and a password which when inserted are checked with the database and give access only if the match is found else access is denied. Here the admin after gaining the access can add the details of students and store them in the database. Here the admin can input the details that are required for allocating the seats as per criteria and hence after processing the data from database. The processed data are retrieved by the admin which can be viewed by the students after the allocation process.

Chapter 3

IMPLEMENTATION

3.1 LOGIN PAGE

The exam seating arrangement project is implemented on java where all the front-end works are done on java netbeans and back-end are done on mysql. The following GUI shows the implemented panels of this project.

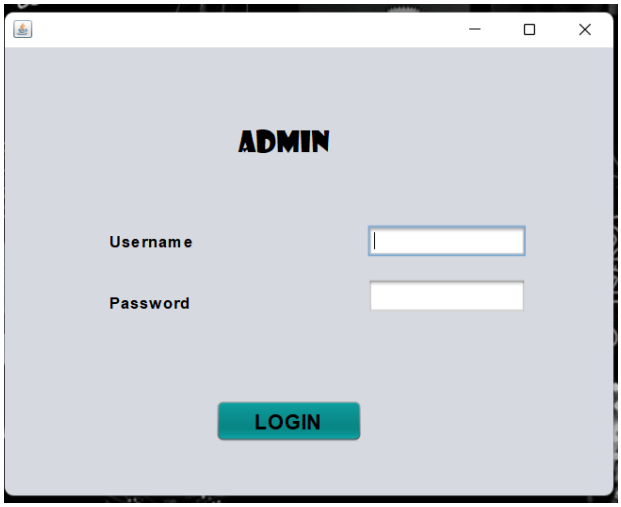


Fig. 3.1: Login Page

3.2 STUDENT DETAILS

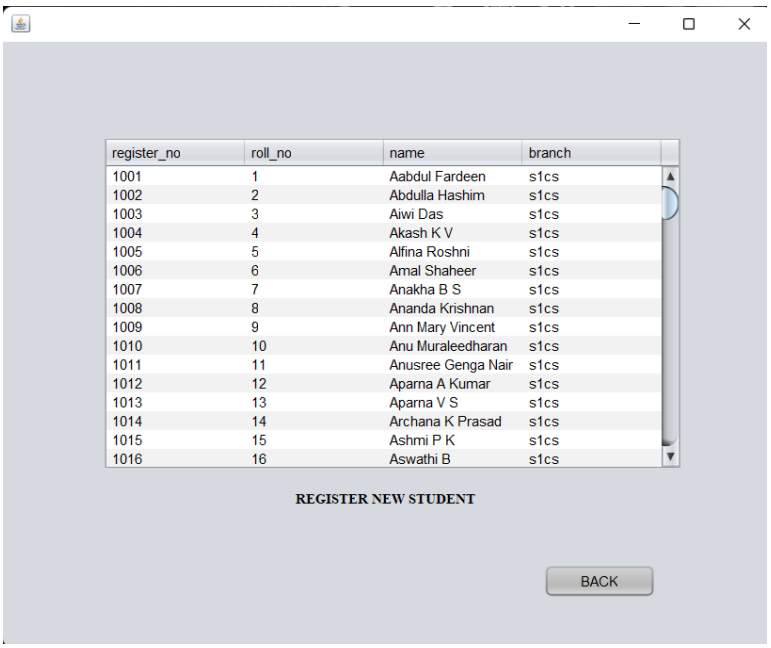


Fig. 3.2: Student Details Page

3.3 ROOM DETAILS

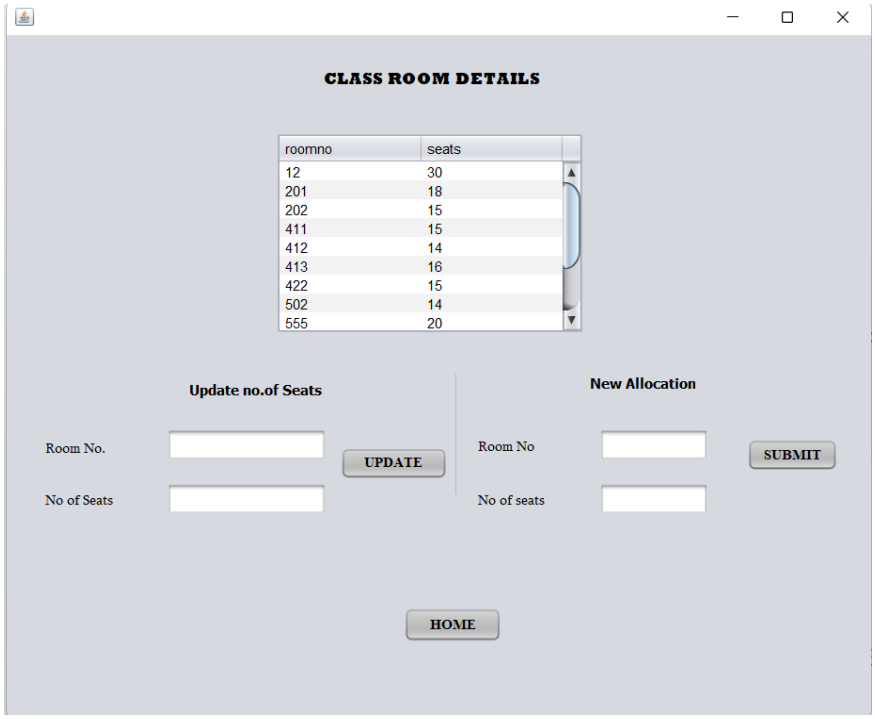


Fig. 3.3: Uploading Room Details

3.4 SELECTION OF STUDENTS AND ROOM

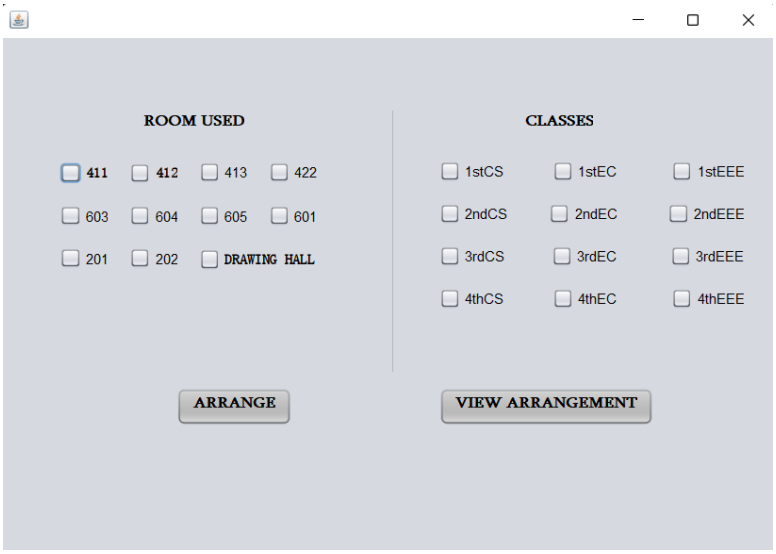
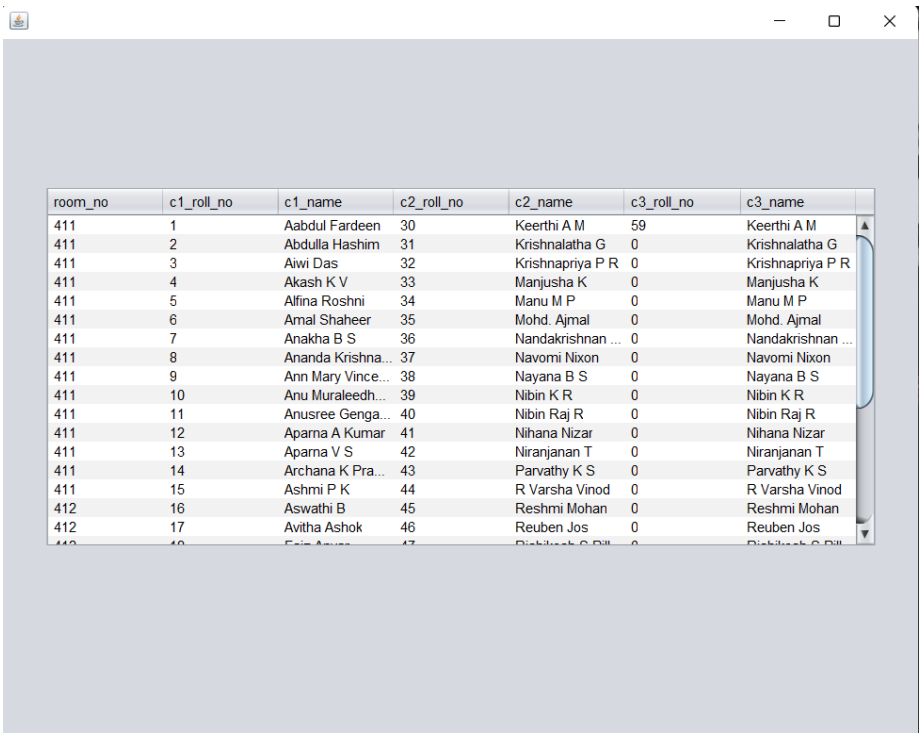


Fig. 3.4: Seating Rooms and Classes

3.5 SEAT ALLOCATION



room_no	c1_roll_no	c1_name	c2_roll_no	c2_name	c3_roll_no	c3_name
411	1	Aabdul Fardeen	30	Keerthi A M	59	Keerthi A M
411	2	Abdulla Hashim	31	Krishnalatha G	0	Krishnalatha G
411	3	Aiwi Das	32	Krishnapriya P R	0	Krishnapriya P R
411	4	Akash K V	33	Manjusha K	0	Manjusha K
411	5	Alfina Roshni	34	Manu M P	0	Manu M P
411	6	Amal Shaheer	35	Mohd. Ajmal	0	Mohd. Ajmal
411	7	Anakha B S	36	Nandakrishnan ...	0	Nandakrishnan ...
411	8	Ananda Krishna...	37	Navomi Nixon	0	Navomi Nixon
411	9	Ann Mary Vince...	38	Nayana B S	0	Nayana B S
411	10	Anu Muraleedh...	39	Nibin K R	0	Nibin K R
411	11	Anusree Genga...	40	Nibin Raj R	0	Nibin Raj R
411	12	Aparna A Kumar	41	Nihana Nizar	0	Nihana Nizar
411	13	Aparna V S	42	Niranjanan T	0	Niranjanan T
411	14	Archana K Pra...	43	Parvathy K S	0	Parvathy K S
411	15	Ashmi P K	44	R Varsha Vinod	0	R Varsha Vinod
412	16	Aswathi B	45	Reshmi Mohan	0	Reshmi Mohan
412	17	Avitha Ashok	46	Reuben Jos	0	Reuben Jos

Fig. 3.5: Seat Allocation

3.6 INSERT STUDENT DETAILS

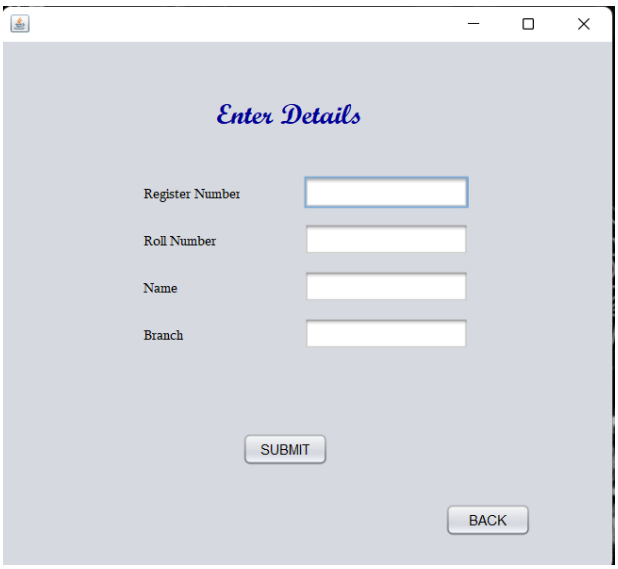


Fig. 3.6: Insert Student Details

PROGRAM CODE

3.7 LOGIN PAGE

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
// TODO add your handling code here:
String username=user.getText();
String password=pass.getText();
if(username.contain(”cs”) && password.contains(”cs”))
user.setText(””);
pass.setText(””);
student s=new student();
s.setVisible(true);
dispose();
else
JOptionPane.showMessageDialog(rootPane, ”The username or password is incorrect”);
```

3.8 INSERT STUDENT DETAILS

```
private void submitActionPerformed(java.awt.event.ActionEvent evt)
// TODO add your handling code here:
try
String url=”jdbc:mysql://localhost:3306/project”;
```

```
String uname=”root”;
String password=”archana@2002”;
String query=”INSERT INTO student values (?,?,?,?)”;
conn=DriverManager.getConnection(url,uname,password);
stmt=conn.prepareStatement(query);
stmt.setString(1,regno.getText());
stmt.setString(2,rollno.getText());
stmt.setString(3,name.getText());
stmt.setString(4,branch.getText());
stmt.executeUpdate();
JOptionPane.showMessageDialog(null,”registration successfull”);
catch(Exception e)
JOptionPane.showMessageDialog(null,e);
```

3.9 ROOM DETAILS

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
// TODO add your handling code here:
try
String url=”jdbc:mysql://localhost:3306/project”;
String uname=”root”;
String password=”archana@2002”;
String query=”UPDATE room SET seats=? WHERE roomid=?”;
conn=DriverManager.getConnection(url,uname,password);
stmt=conn.prepareStatement(query);
stmt.setString(1,seat.getText());
stmt.setString(2,roomno.getText());
stmt.executeUpdate();
JOptionPane.showMessageDialog(null,”Room details updated”);
catch(Exception e)
```

```
JOptionPane.showMessageDialog(null,e);
private void jButton3MouseClicked(java.awt.event.MouseEvent evt)
// TODO add your handling code here:
try
String url=”jdbc:mysql://localhost:3306/project”;
String uname=”root”;
String password=”archana@2002”;
String query=”INSERT INTO room values (?,?)”;
conn=DriverManager.getConnection(url,uname,password);
stmt=conn.prepareStatement(query);
stmt.setString(1,jTextField1.getText());
stmt.setString(2,jTextField2.getText());
stmt.executeUpdate();
JOptionPane.showMessageDialog(null,”Room details updated”);
catch(Exception e)
JOptionPane.showMessageDialog(null,e);
```

3.10 SEAT ARRANGEMENT

```
public class seatalloc extends javax.swing.JFrame {
Connection conn=null;
PreparedStatement stmt=null;
ResultSet rs=null;
Statement st=null;
String[] rooms=new String[20];
String[] dept=new String[15];
int[][] arrange=new int[500][10];
int n,m,flag;
int bench;
/**
```

```
* Creates new form seatalloc
*/
public seatalloc()
initComponents();
public void select()
int i=0;
if(c1.isSelected())
rooms[i]=”411”;
i++;
if(c2.isSelected())
rooms[i]=”412”;
i++;
if(c3.isSelected())
rooms[i]=”413”;
i++;
if(c4.isSelected())
rooms[i]=”422”;
i++;
if(c5.isSelected())
rooms[i]=”202”;
i++;
if(c6.isSelected())
rooms[i]=”201”;
i++;
if(c7.isSelected())
rooms[i]=”601”;
i++;
if(c8.isSelected())
rooms[i]=”603”;
i++;
```

```
if(c9.isSelected())
rooms[i]=”604”;
i++;
if(c10.isSelected())
rooms[i]=”605”;
i++;
if(c11.isSelected())
rooms[i]=”Drawing hall”;
i++;
n=i;
for(i=0;i¡n;i++)
System.out.println(rooms[i]);
i=0;
if(y1.isSelected())
dept[i]=”S1CS”;
i++;
if(y2.isSelected())
dept[i]=”S1EC”;
i++;
if(y3.isSelected())
dept[i]=”S1EEE”;
i++;
if(y4.isSelected())
dept[i]=”S3CS”;
i++;
if(y5.isSelected())
dept[i]=”S3EC”;
i++;
if(y6.isSelected())
dept[i]=”S3EEE”;
```

```
i++;
if(y7.isSelected())
dept[i]=”S5CSA”;
i++;
dept[i]=”S3CSB”;
i++;
if(y8.isSelected())
dept[i]=”S5EC”;
i++;
if(y9.isSelected())
dept[i]=”S5EEE”;
i++;
if(y10.isSelected())
dept[i]=”S7CS”;
i++;
if(y11.isSelected())
dept[i]=”S7EC”;
i++;
if(y12.isSelected())
dept[i]=”S7EEE”;
i++;
m=i;
for(i=0;i¡m;i++)
System.out.println(dept[i]);
public void NoOfBench()
String url =”jdbc:mysql://localhost:3306/project”;
bench=0;
try
String uname=”root”;
String password=”archana@2002”;
```

```
Connection conn = DriverManager.getConnection(url,uname,password);
Statement st = conn.createStatement();
String sql1=”SELECT * FROM room”;
PreparedStatement pst=conn.prepareStatement(sql1);
ResultSet rs=pst.executeQuery(sql1);
while(rs.next())
int i=0;
while(i¡n)
String s=rs.getString(1);
if(rooms[i].equals(s))
bench=bench+Integer.parseInt(rs.getString(2));
break;
i++;
System.out.println(”No Of Benches”+bench);
catch (Exception ex)
JOptionPane.showMessageDialog(null,ex);
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
// TODO add your handling code here:
select();
NoOfBench();
try
String url=”jdbc:mysql://localhost:3306/project”;
String uname=”root”;
String password=”archana@2002”;
Connection conn = DriverManager.getConnection(url,uname,password);
Statement st = conn.createStatement();
String sql1=”SELECT * FROM student”;
ResultSet rs=st.executeQuery(sql1);
int i=0;
int c=0;
```

```
do
for(int j=0;j¡bench;j++)
if(rs.next())
for(int d=0;d¡m;d++)
if (dept[d].equalsIgnoreCase(rs.getString(4)))
arrange[j][i]=rs.getInt(1);
break;
if(arrange[j][i]==0)
j–;
else
c=1;
break;
i++;
while(c==0);
flag=i;
if(i¿3)
JOptionPane.showMessageDialog(null, ”arrangement not possible”);
for(int k=0;k¡bench;k++)
for(int j=0;j¡flag;j++)
System.out.print(arrange[k][j]+” ”);
System.out.println(””);
viewalloc v=new viewalloc(arrange, rooms, bench);
catch (Exception ex)
JOptionPane.showMessageDialog(null,ex);
```

Chapter 4

HARDWARE AND SOFTWARE REQUIREMENTS

4.1 HARDWARE REQUIREMENTS

• Processor: Intel i3

• Ram: 4GB

4.2 SOFTWARE REQUIREMENTS

Operating System: Windows 10

Frontend :Netbeans

Backend: MySql

Chapter 5

CONCLUSION

On the basis of study, it is identified that there is a need to automate the exam seating allo cation.Currently this process is done manually which is time consuming and their is chance for wrong allocation Exam hall seating arrangement system is developed for the college to simplify examination hall allotment and seating arrangement. The software will keep records about stu dents and subjects in database. It will automatically allocate seats to students in given date and time. This is a user friendly platform for efficient and quick allocation of seat to students. The new software system will generate more accurate list of student with minimum manual work.

Chapter 6

REFERENCES

[1] Elmasri R. and S.Navathe ER Diagram: Database Systems: Models, Languages, Design and Application Programming, Pearson Education, 2013."

[2] Herbert Schildt "Use Case Diagram Java: The complete reference,8/e, Tata McGraw Hill,2011".