A
PROJECT REPORT
ON

Hospital Management System

Submitted in Partial Fulfillment of the
Requirements for the award of the degree
MASTER OF COMPUTER APPLICATIONS
By

Name:- Asif
Roll no:- 2306950140027

Under The supervision of

Internal Guide:- Mr. Vishwas Sharma
MCA Dept. SRGC

Dep’t. of Computer Application
Shri Ram Group of College,
Muzaffarnagar.

DR. APJ ABDUL KALAM TECHNICAL UNIVERSITY,
LUCKNOW

DECLARATION

I **Mohd Harish** hereby declare that the project report title **“Hospital management system”** is an original work carried out by me under the supervision of **Mr. Vishwas Sharma** I further declare that this work has not been submitted to any other Institute/University for the requirement of third semester of master of computer applications.

Student Name:

Mohd Harish

(2306950140062)

ACKNOWLEDGEMENT

Apart from the efforts of team, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. The completion of any inter-disciplinary project depends upon cooperation, co-ordination and combined efforts of several sources of knowledge. We are eternally grateful to our teacher **Mr. Vishwas Sharma** for her even willingness to give us valuable advice and direction under which we executed this project. Her constant guidance and willingness to share her vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks.

Mohd Harish

(2306950140062)

Table of Content

TITLE OF THE PROJECT

INTRODUCTION

This is a Project work undertaken in context of partial fulfillment of MCA.I have tried my best to make the complicated process of Online **Hospital Management System** as simple as possible using Structured & Modular technique & Menu oriented interface. I have tried to design the software in such a way that user may not have any difficulty in using this package & further expansion is possible without much effort. Even though I cannot claim that this work to be entirely exhaustive, the main purpose of my exercise is perform each Patient’s activity in computerized way rather than manually which is time consuming.

I am confident that this software package can be readily used by non-programming personal avoiding human handled chance of error. This project is used by

1. Administrator (Management of the Hospital).

- Administrator can maintain daily updates in the hotel records.

Administrator is must be an authorized user. He can further change the password. There is the facility for password recovery, logout etc.

The main aim of the entire activity is to automate the process of day to day activities of Hospital like Room activities, Admission of a New Customer, Assign a room according to customer’s demand, checkout of a computer and releasing the room and finally compute the bill etc.

The limited time and resources have restricted us to incorporate, in this project, only a main activities that are performed in a Hospital Management System, but utmost care has been taken to make the system efficient and user friendly. “Hospital Management System” has been designed to computerized the following functions that are performed by the system:

- Room Detail Functions Opening a New Room
- Modification to room assigned Check-in and check-out Detail Functions
- Admission of New Patient Check-out of Patient
- Room assigning related to Patient’s need.

1) Statement of
    - Patient Details
    - Check-in Patient
    - Check-out Patient
    - Room Details
2) Total number of
    - Patient in the Hospital
    - Individual Patient Report

OBJECTIVE

The objective of these systems was to streamline admin-type work like scheduling appointments, managing pharmacies, and medical records, among others. A hospital management system contains all data and operations in one platform.
The entire information has maintained in the database or Files and whoever wants to retrieve can't retrieve, only authorization user can retrieve the necessary information which can be easily be accessible from the file.

- The patients receive better care because they have faster access to medical records and coordination.
- Improved management of administration through automated appointment scheduling, medical billing, reporting
- Data security through encryption and enhanced access controls based on secured patient information
- Resource management in terms of inventory, staff, and equipment to maintain its operational efficiency.
- Accurate reporting as well as analytics for improving decisions made by hospitals for better healthcare.
- Efficient coordination among departments so that the workflow does not get disturbed.
- Low operating costs because of reduced paper works, minimal manual errors, and efficient use of resources.
- Compliance with regulations to ensure accurate registration of patients and adherence to health-related standards.
- Claims processing accelerates when totally integrated medical billing and insurance management systems are applied.
- Patient satisfaction levels are high since less time is spent within the healthcare sector awaiting its services.

PROJECT CATEGORY

This Project is coupled with material on how to use the various tool, sub sets available in JAVA, and MY SQL.

This is a consol based project which comes under the category of JAVA JDK 64 BIT consol application.

The need of today’s software development is competence in a GUI (Graphical User Interface) based front-end tools, which can connect to Relational Database engines.

HARDWARE AND SOFTWARE REQUIREMENTS

HARDWARE REQUIREMENT:

- **MINIMUM:**

Processor	: intel i5
Clock Speed	: 2.30 GHz
RAM	: 4 GB
Hard disk Capacity	: 10 GB

- **RECOMMENDED:**

Processor	: intel i5
Clock Speed	: 2 GHz
RAM	: 8 GB
Hard disk Capacity	: 80 GB

SOFTWARE REQUIREMENT:

FRONTEND: JAVA swing,AWT.

BECKEND: My SQL .

OPERATING SYSTEM: WINDOW 10.

IDE: INTELI J IDE.

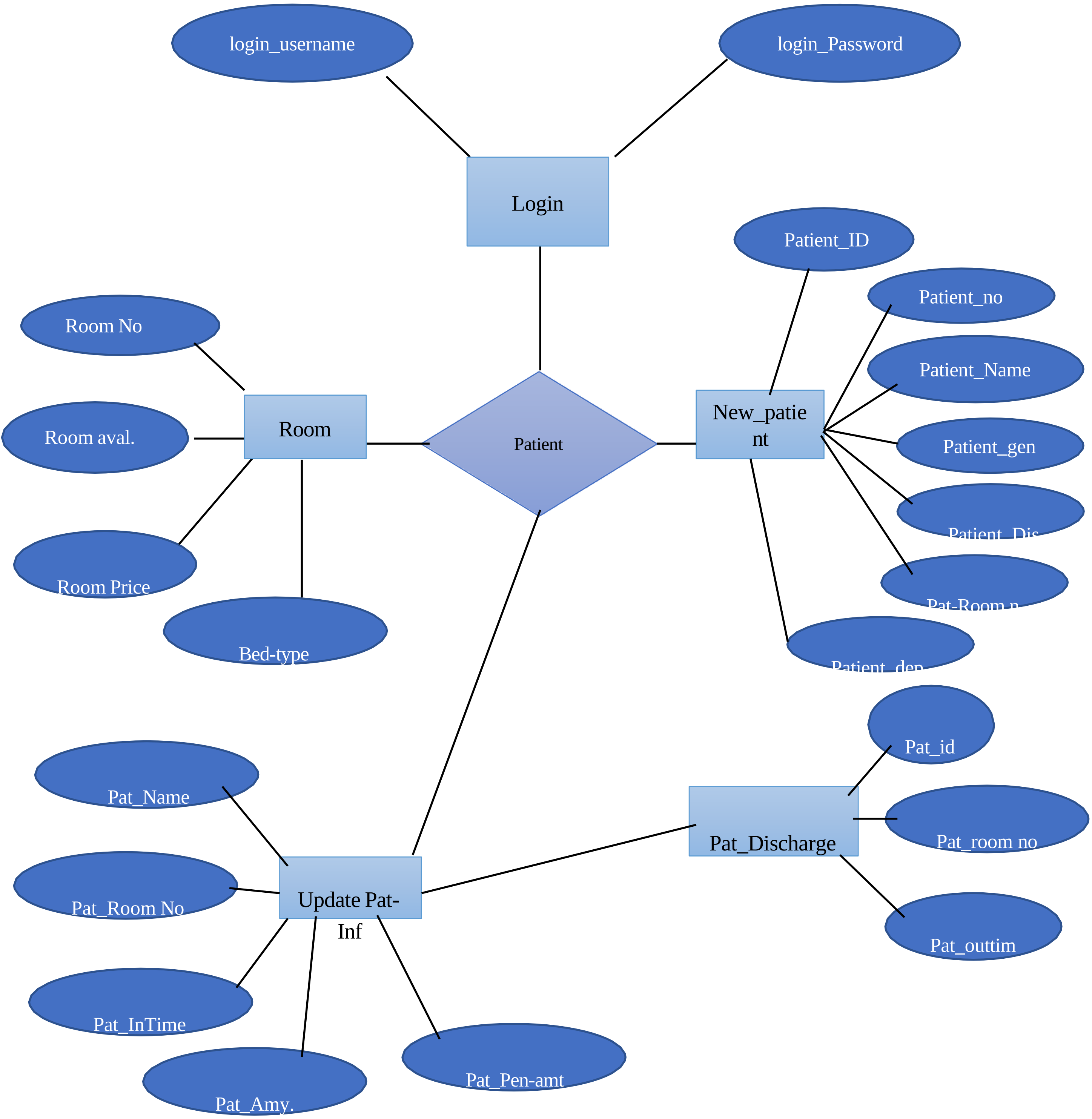
ANALYSIS

The “HOSPITAL **Management system**” has been developed to the problem in the practicing manual system. This software is supported to eliminate and in same causes the problems faced by this existing system(manual).

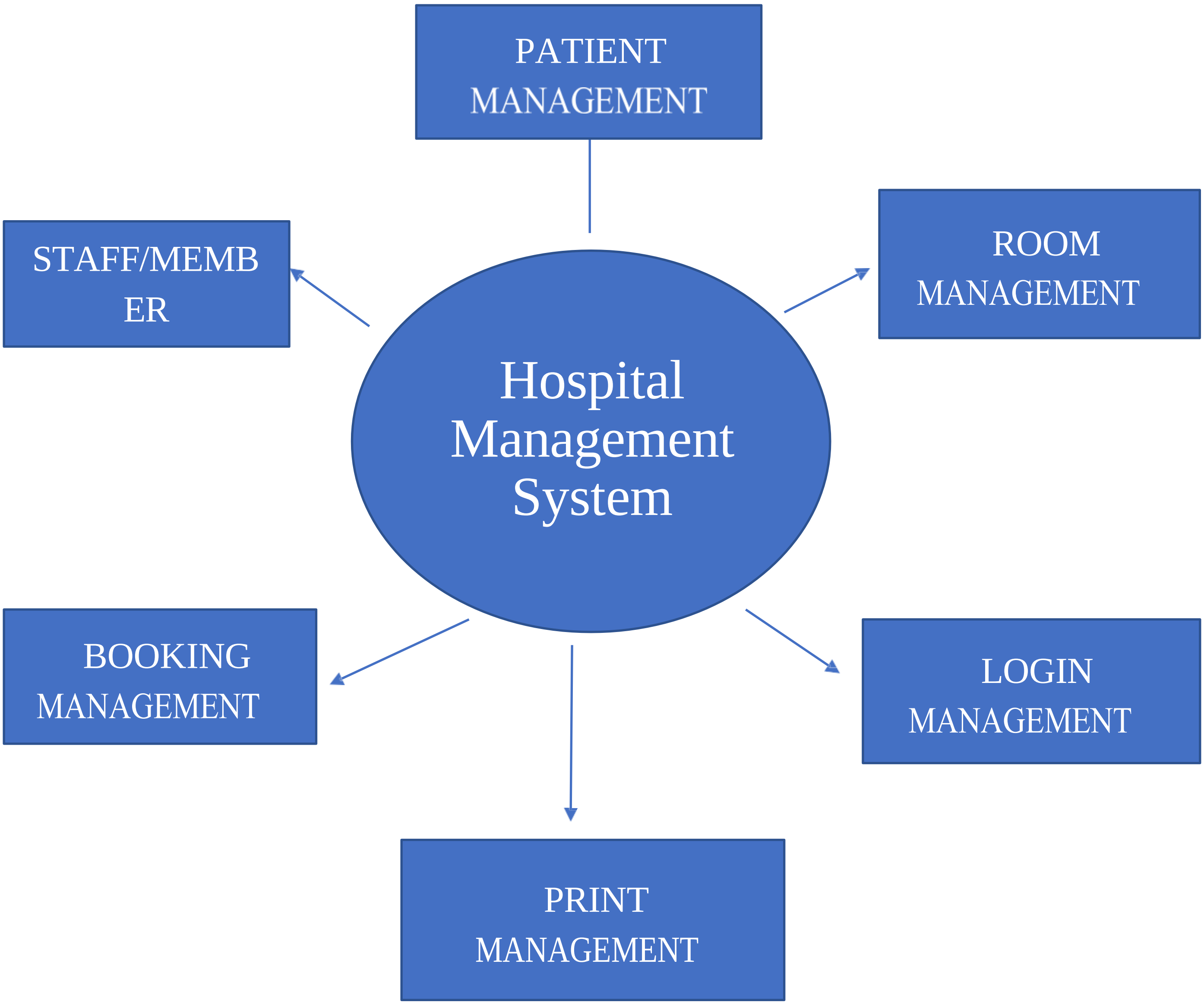
This application is helpful to reduce or to avoid the errors while entering the data. It also provides error message if user enter invalid data. No formal knowledge is required for the user to use this system.

Every organization whether big or small has challenges to overcome and managing the information of students details etc. The management system is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals.

ER Diagram For Hospital Management System



CONTEXT LEVEL DFD HOTEL MANAGEMENT SYSTEM



DATA MODELLING

Customer Details Table

Fields	Type	Size	Constraint Name
Name	VARCHAR	50	PRIMARY KEY

Items Details Table

Fields	Type	Size	Constraint Name
Item_Name	VARCHAR	50	PRIMARY KEY
Quantity	INT	10000000	NOT NULL

Feasibility study

A feasibility study is a type of analysis used in measuring the ability and likelihood to successfully complete a project including all relevant factors. It must account for factors that affect it such as economic, managers use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it. Instead of diving into a project and hoping for the best, a feasibility study acts a precursor for mangers to investigate the possible negative and positive outcomes of a project investing too much time and money.

- **Importance of feasibility studies**

Feasibility studies allow companies to determine and organize all of the necessary details to make a business work.The feasibility study will evaluate the technical, marketing, financial, management, and socio-economic aspects of starting this laundry business. It will determine if the business is viable and how it could benefit customers, employees, factories, and the researchers conducting the study.

- **Components of a feasibility study**

Consequently, a clear and detail explanation of the feasibility analysis including four main elements, which are product/service feasibility, industry and market feasibility, organizational feasibility and financial feasibility, is offered in chapter 2 as a guidance for the implementation part in chapter 5.

There are several components of a feasibility study:

Description:- A layout of the business, the products and or services to be offered and how they will be delivered.

Market feasibility:- Describes the industry, the current and future market potential, competition, sales estimations and prospective buyers.

Technical feasibility:- Lays out details on how a good or service will be delivered, which includes transportation, business location , technology needed , materials and labour.

Financial feasibility:- A projection of the amount of funding or start Up capital needed , what sources of capital can and will be used ,and what kind of return can be expected on the investment.

FUNCTION DETAILS AND MODULES

The basic objective of **Hospital MANAGEMENT SYSTEM** is to generalize and simplify the monthly or day to day activities of Hospital like item activities, Check in of New patient, Check out of patient, Assigning a doctor according to patient requirement etc. which has to be performed repeatedly on regular basis. To provide efficient, fast, reliable and user-friendly system is the basic motto behind this exercise.

Let us now discuss how different functions handle the structure and data files:

- **Patient Module**

This is main module in the project. The user can and write information about any member. The user can also Updates, Add, Pending, Completed, and Delete records of items details.

Allow the user to register new member and update details of items.

SCOPE OF FUTURE APPLICATION

This project can be used in the Hospital after adding some more useful modules in the project for which hotel are providing services.

Utmost care and back-up procedures must be established to ensure 100% successful implementation of the computerized hotel system. In case of system failure, the organization should be in a position to process the transaction with another organization or if the worst comes to the worst, it should be in a position to complete it manually.

SCOPE OF IMPROVEMENT

Now a day’s hotel is providing many other facilities, this project can also be improved with the improvement in the Hospital.

Utmost care and back-up procedures must be established to ensure 100% successful implementation of the computerized banking system. In case of system failure, the organization should be in a position to process the transaction with another organization or if the worst comes to the worst, it should be in a position to complete it manually**.**

Limitations of project

This project deals with the information criteria of the customers information management system basically the project describe how to manage records of patient, and patient details.

In this project we can store the information of customers, we can feed information of details, we can access information of a particular member we can retrieves the information of patient.

In this project user can view the reports but cannot take the prints of reports.

CONCLUSION

This project is designed to meet the requirements of Online Hospital Management. It has been developed in JAVA, MY SQL keeping in mind the specifications of the system.

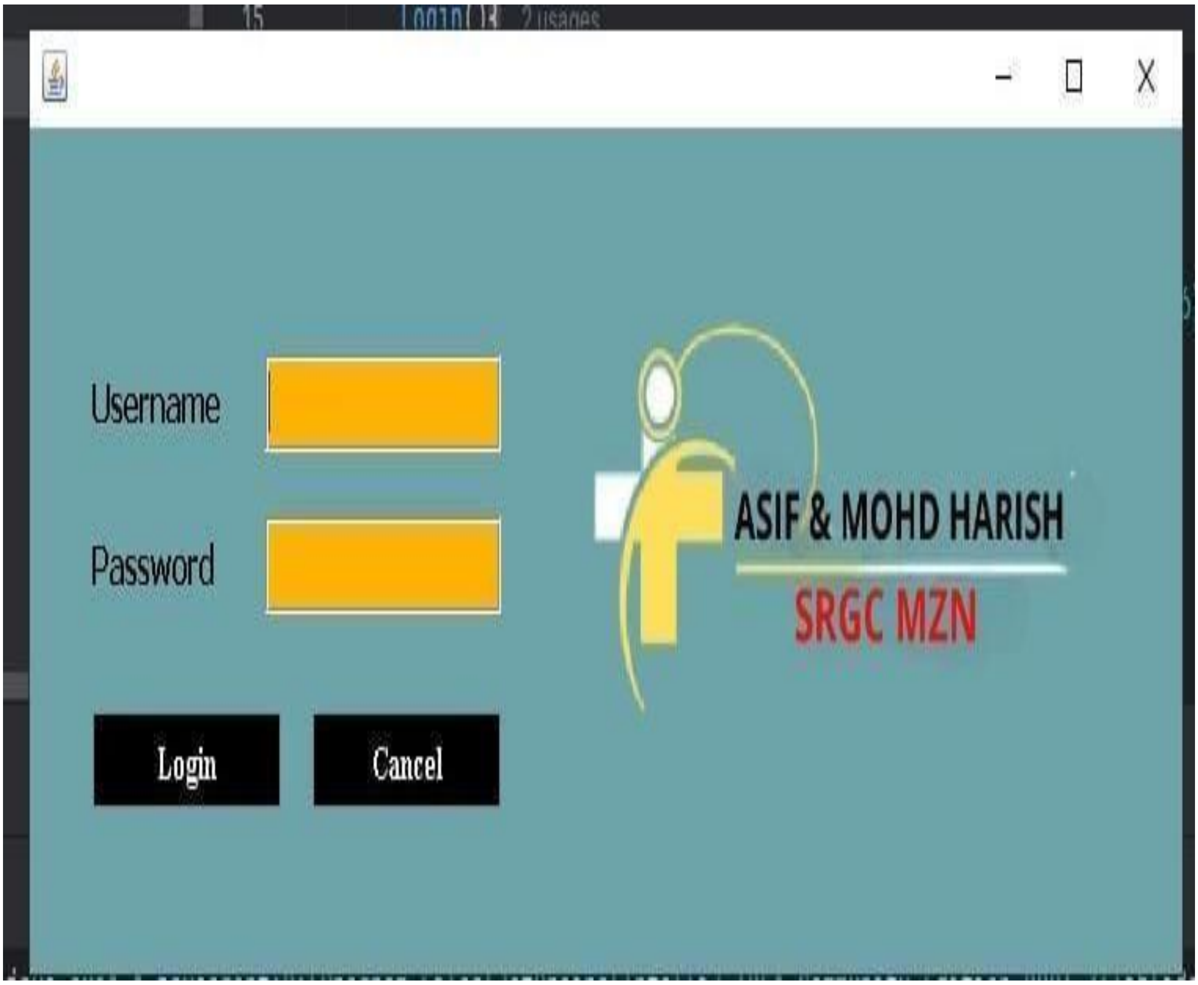
For designing the system, we have used simple data flow diagrams. Over all the project teaches us the essential skills like:

Using system analysis and design techniques like data flow diagram in designing the system.

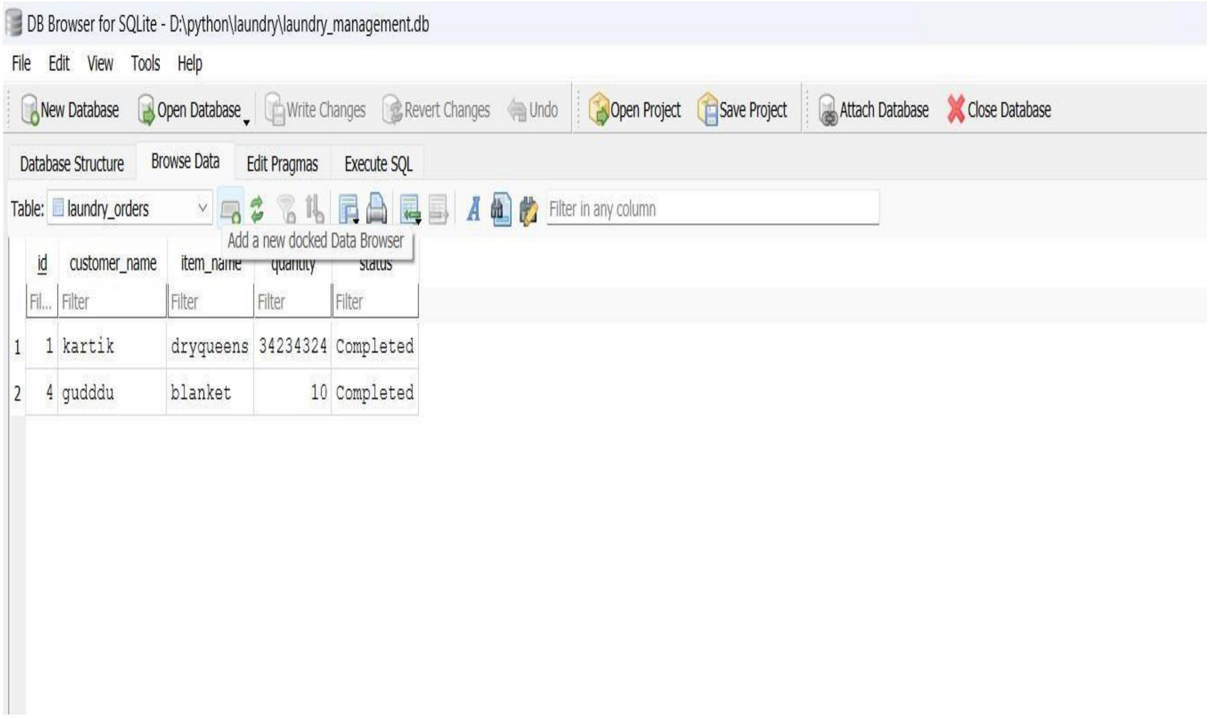
Understanding the database handling and query processing.

Project Images

1. Login page:-

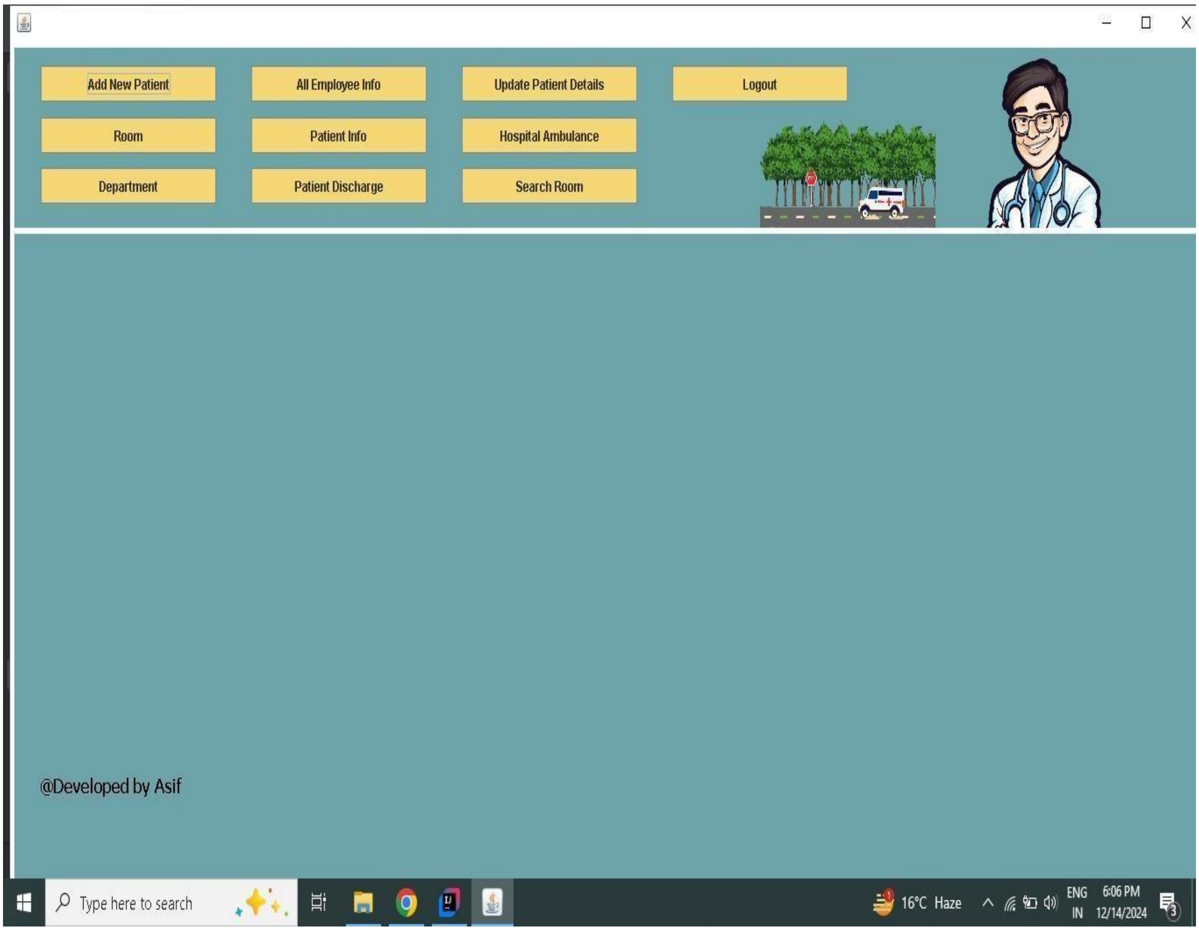


2. Database Page:-

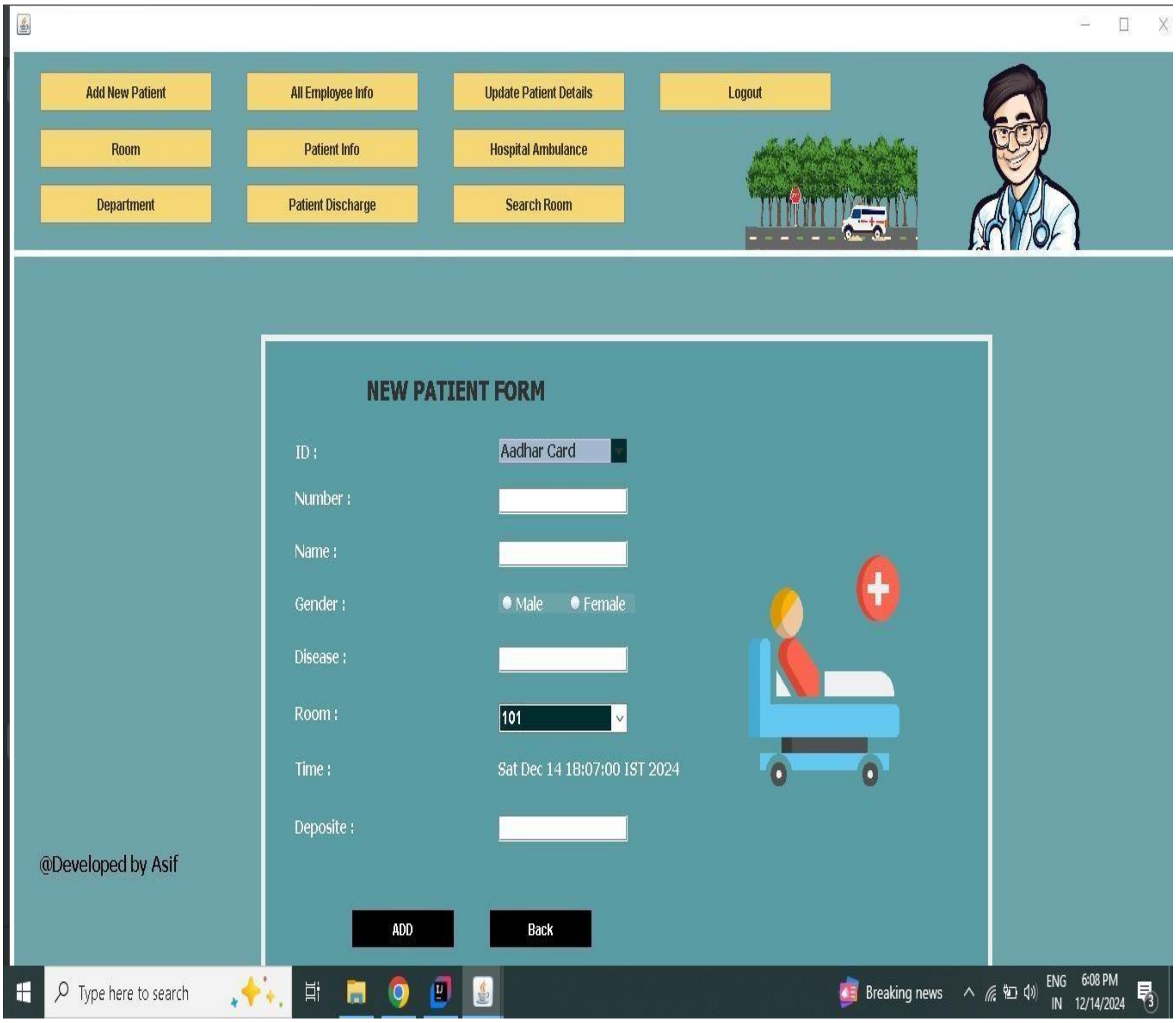


	id	customer_name	item_name	quantity	status
	Fil...	Filter	Filter	Filter	Filter
1	1	kartik	dryqueens	34234324	Completed
2	4	gudddu	blanket	10	Completed

3. **Reception class:-**



4. ADD Patient:-



NEW PATIENT FORM

ID : Aadhar Card

Number :

Name :

Gender : ○ Male ○ Female

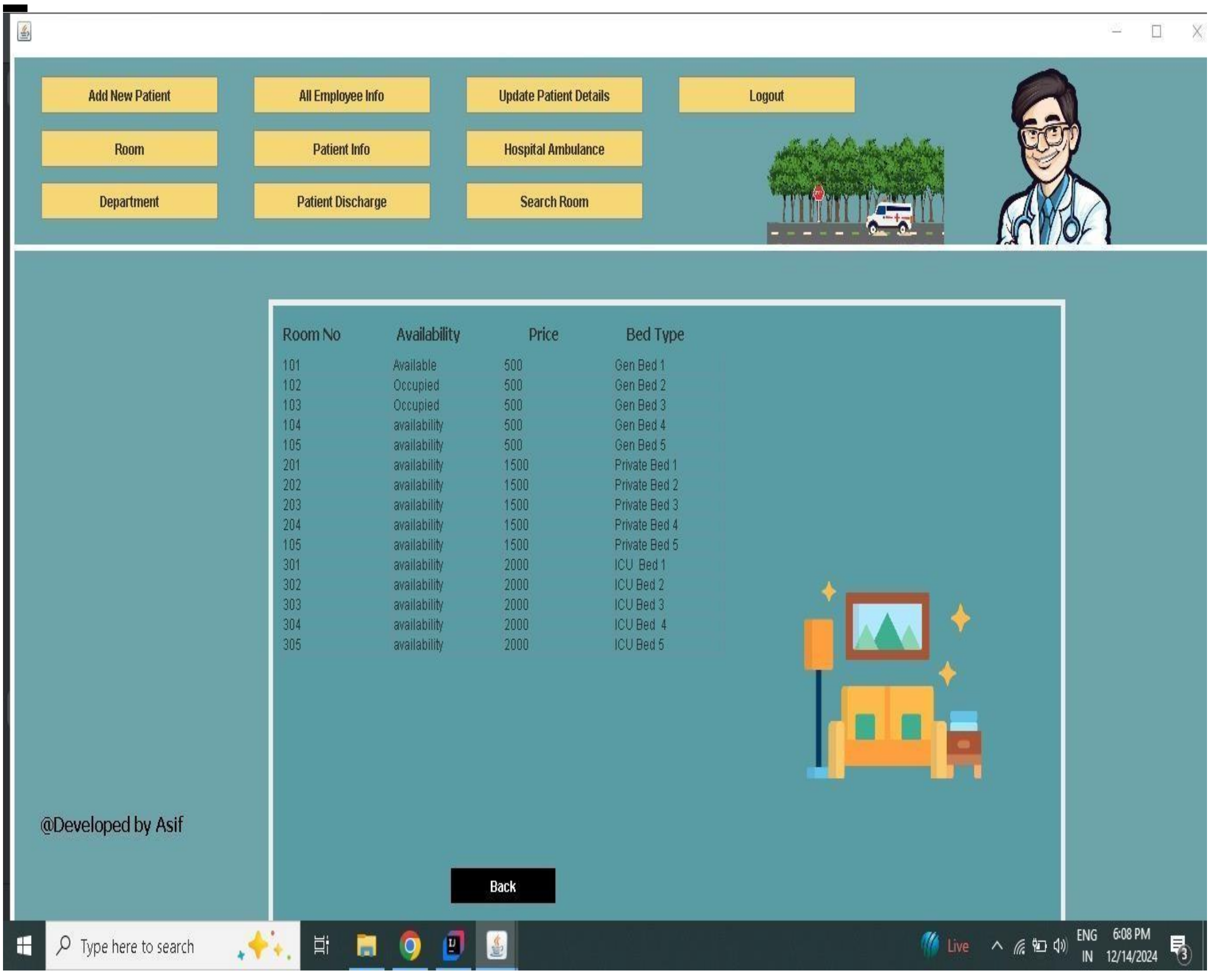
Disease :

Room : 101

Time : Sat Dec 14 18:07:00 IST 2024

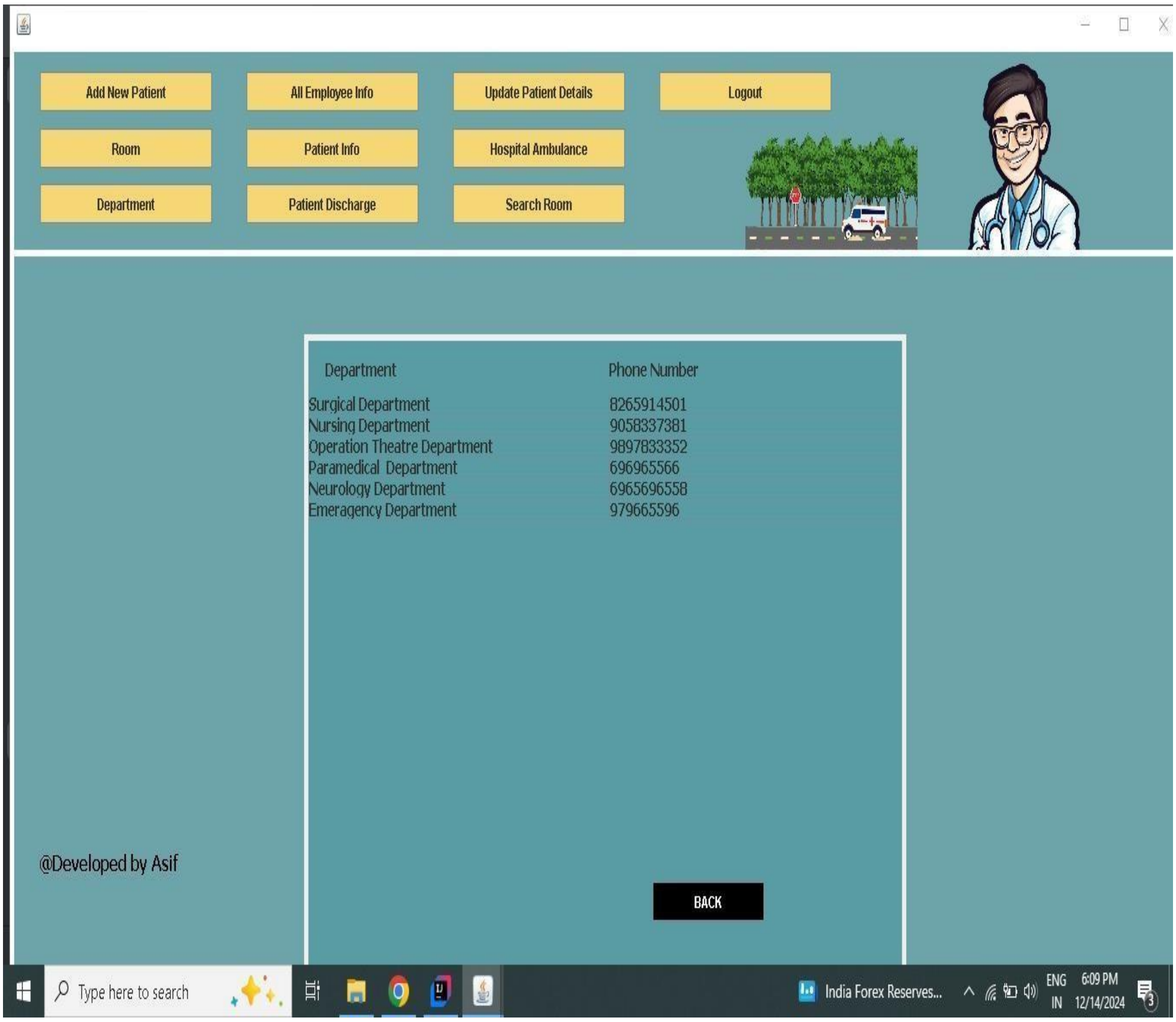
Deposite :

5. Room Class:-



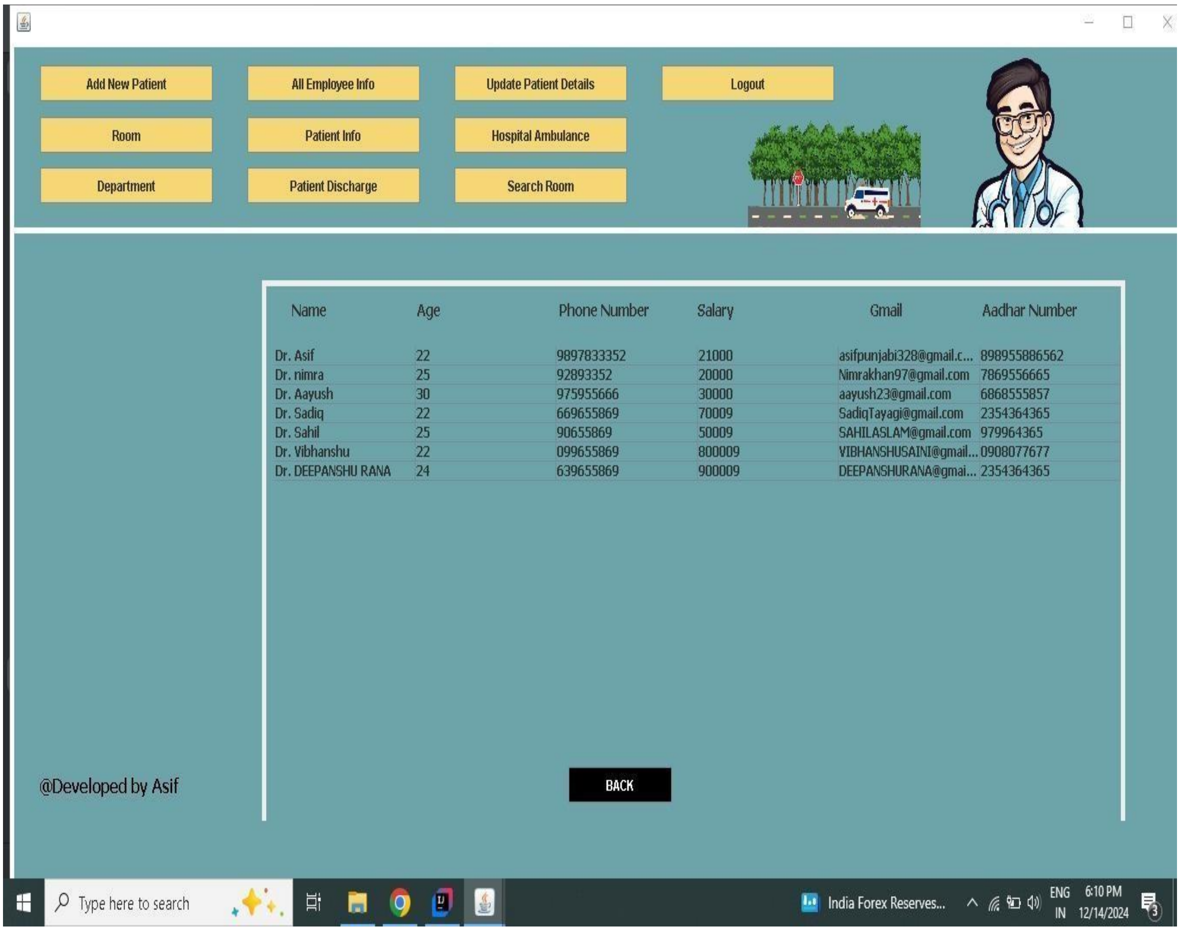
Room No	Availability	Price	Bed Type
101	Available	500	Gen Bed 1
102	Occupied	500	Gen Bed 2
103	Occupied	500	Gen Bed 3
104	availability	500	Gen Bed 4
105	availability	500	Gen Bed 5
201	availability	1500	Private Bed 1
202	availability	1500	Private Bed 2
203	availability	1500	Private Bed 3
204	availability	1500	Private Bed 4
105	availability	1500	Private Bed 5
301	availability	2000	ICU Bed 1
302	availability	2000	ICU Bed 2
303	availability	2000	ICU Bed 3
304	availability	2000	ICU Bed 4
305	availability	2000	ICU Bed 5

6. Department Class:-



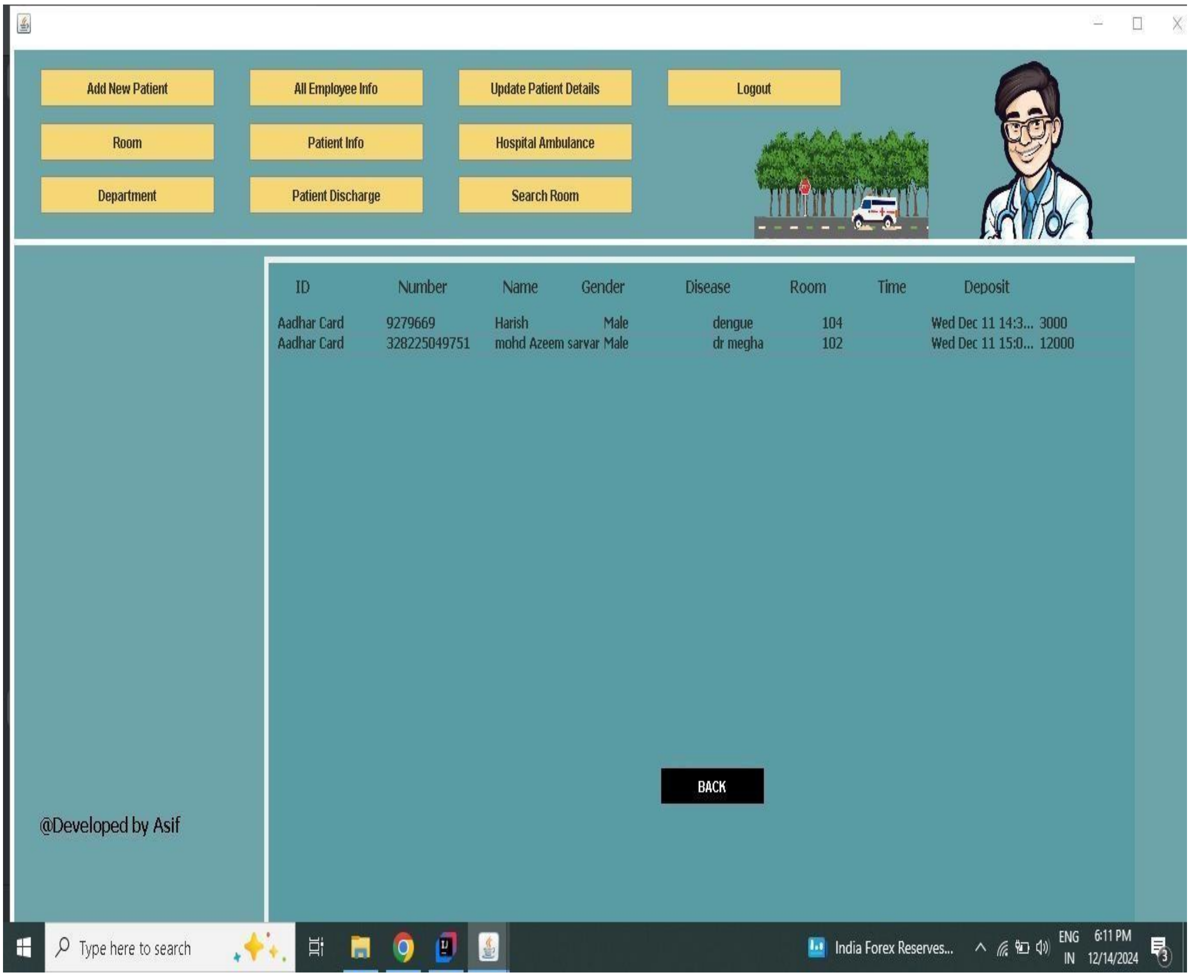
Department	Phone Number
Surgical Department	8265914501
Nursing Department	9058337381
Operation Theatre Department	9897833352
Paramedical Department	696965566
Neurology Department	6965696558
Emeragency Department	979665596

7. Employes Class:-



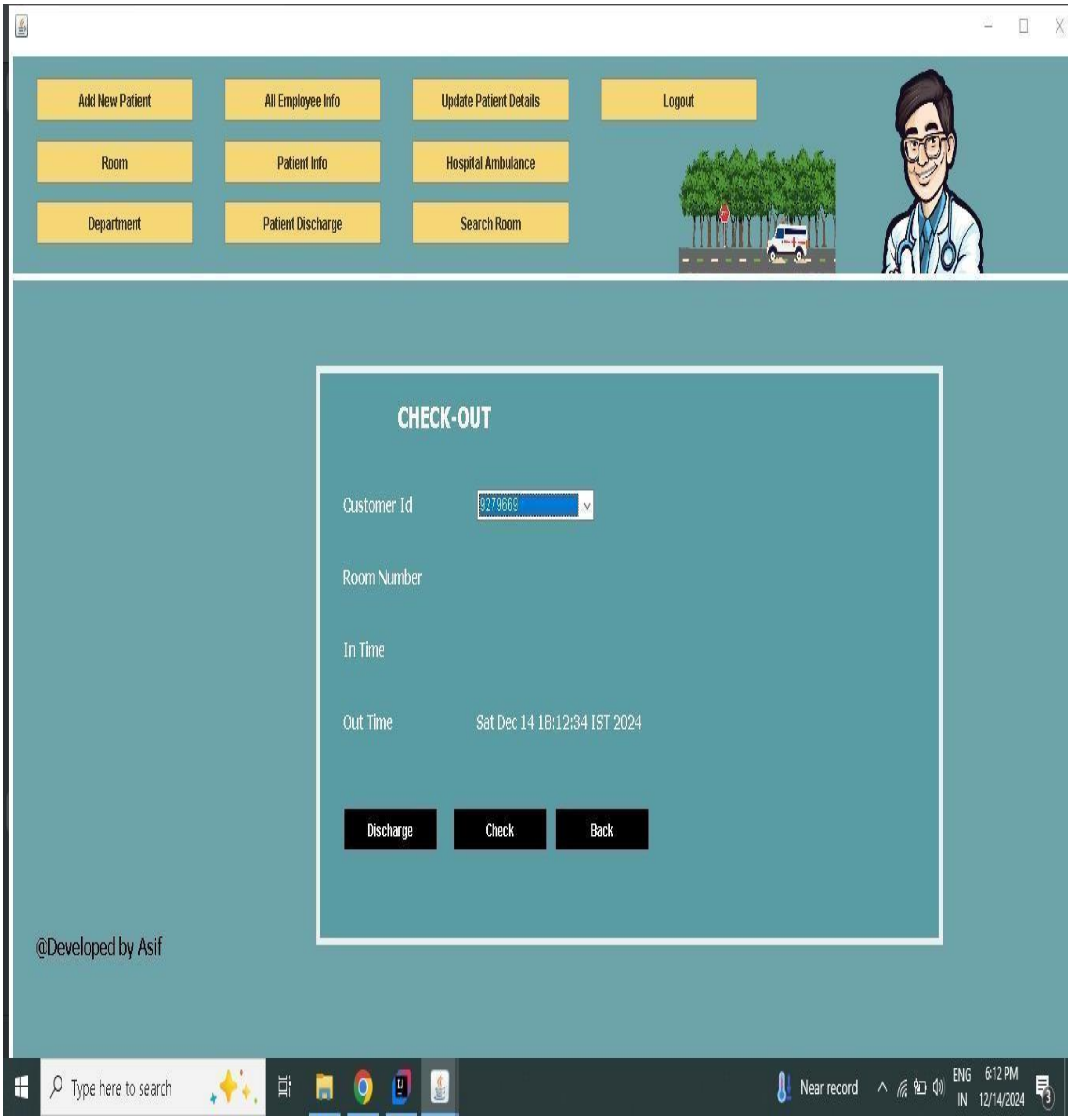
Name	Age	Phone Number	Salary	Gmail	Aadhar Number
Dr. Asif	22	9897833352	21000	asifpunjabi328@gmail.c...	898955886562
Dr. nimra	25	92893352	20000	Nimrakhan97@gmail.com	7869556665
Dr. Aayush	30	975955666	30000	aayush23@gmail.com	6868555857
Dr. Sadiq	22	669655869	70009	SadiqTayagi@gmail.com	2354364365
Dr. Sahil	25	90655869	50009	SAHILASLAM@gmail.com	979964365
Dr. Vibhanshu	22	099655869	800009	VIBHANSHUSAINI@gmail...	0908077677
Dr. DEEPANSHU RANA	24	639655869	900009	DEEPANSHURANA@gmai...	2354364365

8. Patient Class:-



ID	Number	Name	Gender	Disease	Room	Time	Deposit
Aadhar Card	9279669	Harish	Male	dengue	104	Wed Dec 11 14:3...	3000
Aadhar Card	328225049751	mohd Azeem sarvar	Male	dr megha	102	Wed Dec 11 15:0...	12000

9. Patient Discharge:-



CHECK-OUT

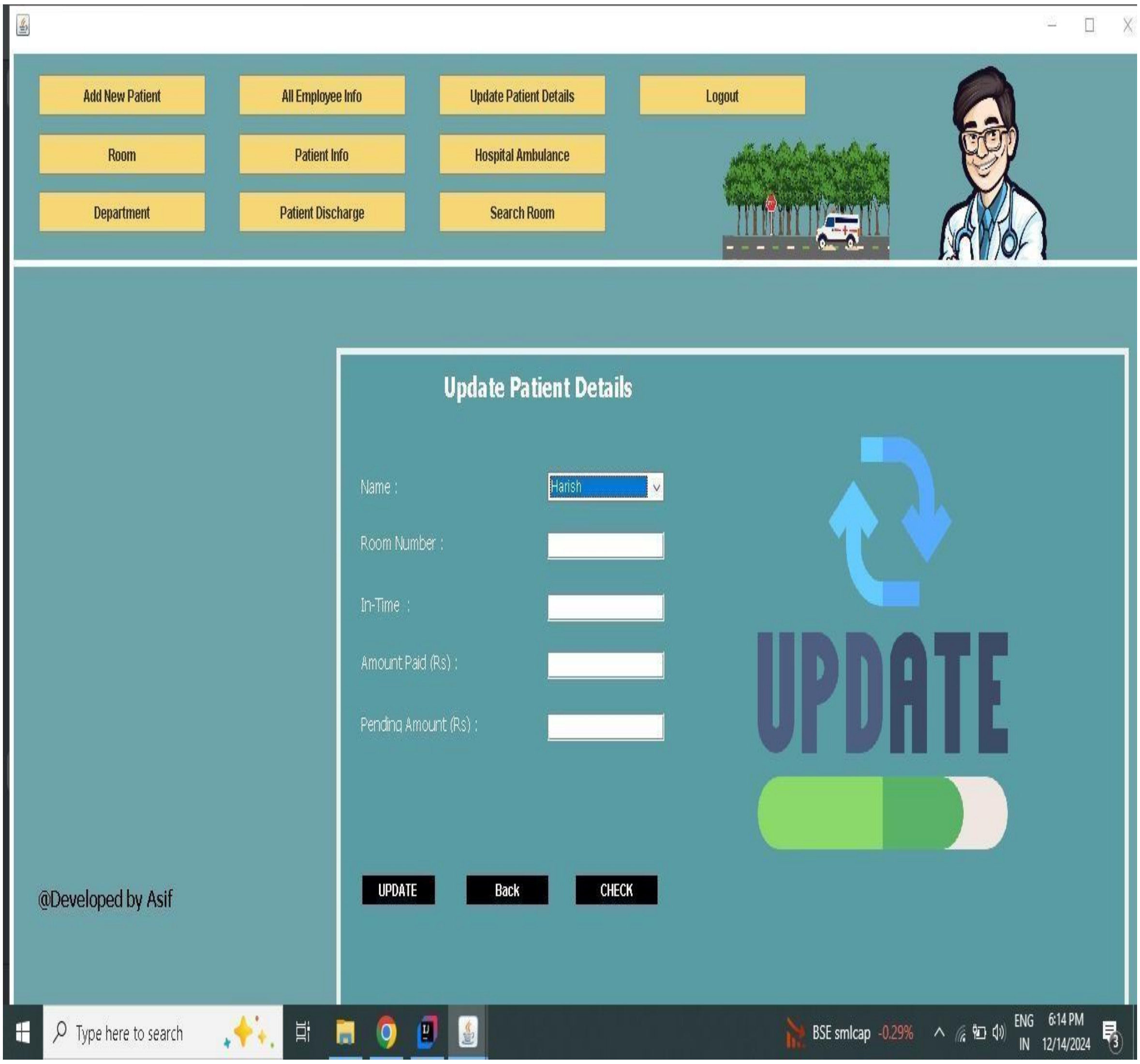
Customer Id 9279669

Room Number

In Time

Out Time Sat Dec 14 18:12:34 IST 2024

10. Update Patient Details Class:-



Update Patient Details

Name : Harish

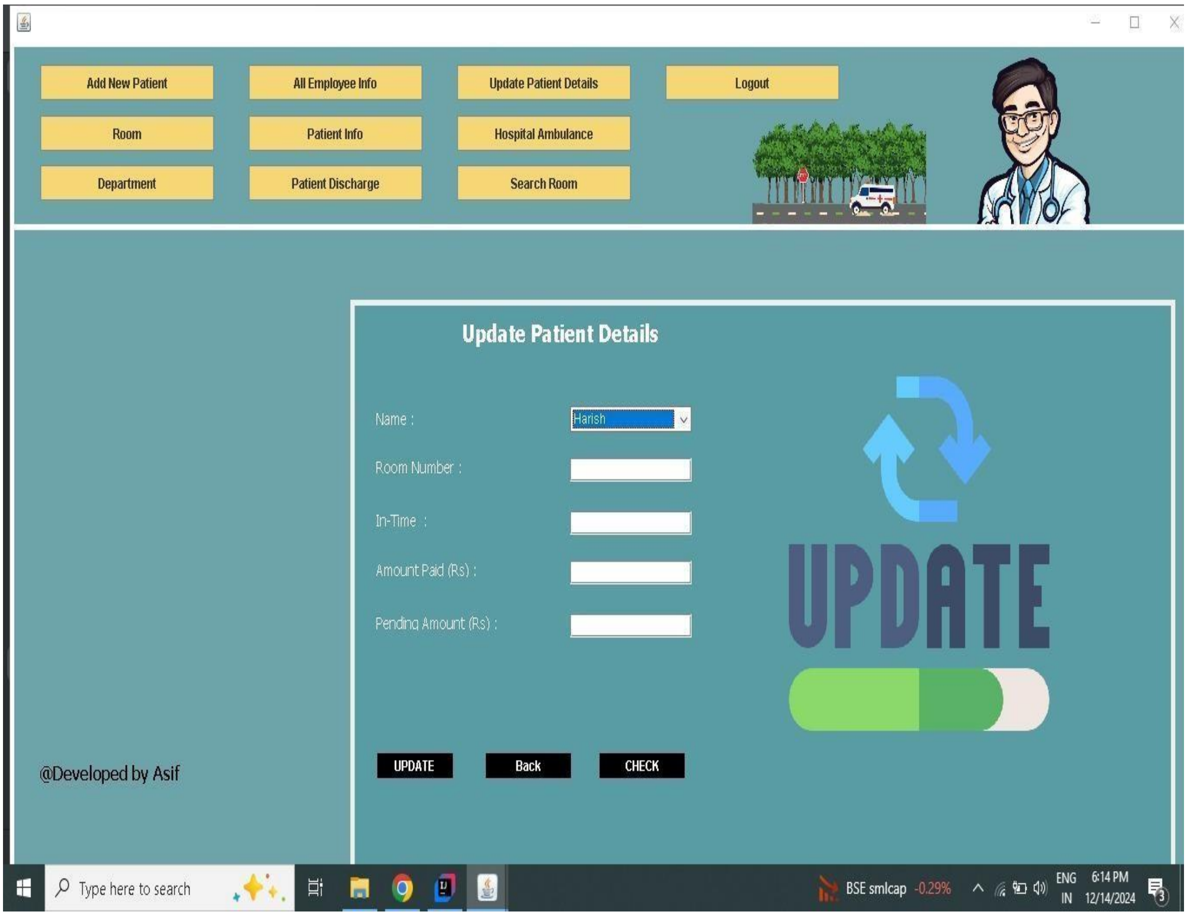
Room Number :

In-Time :

Amount Paid (Rs) :

Pending Amount (Rs) :

11. Ambulance Class:-



Update Patient Details

Name : Harish

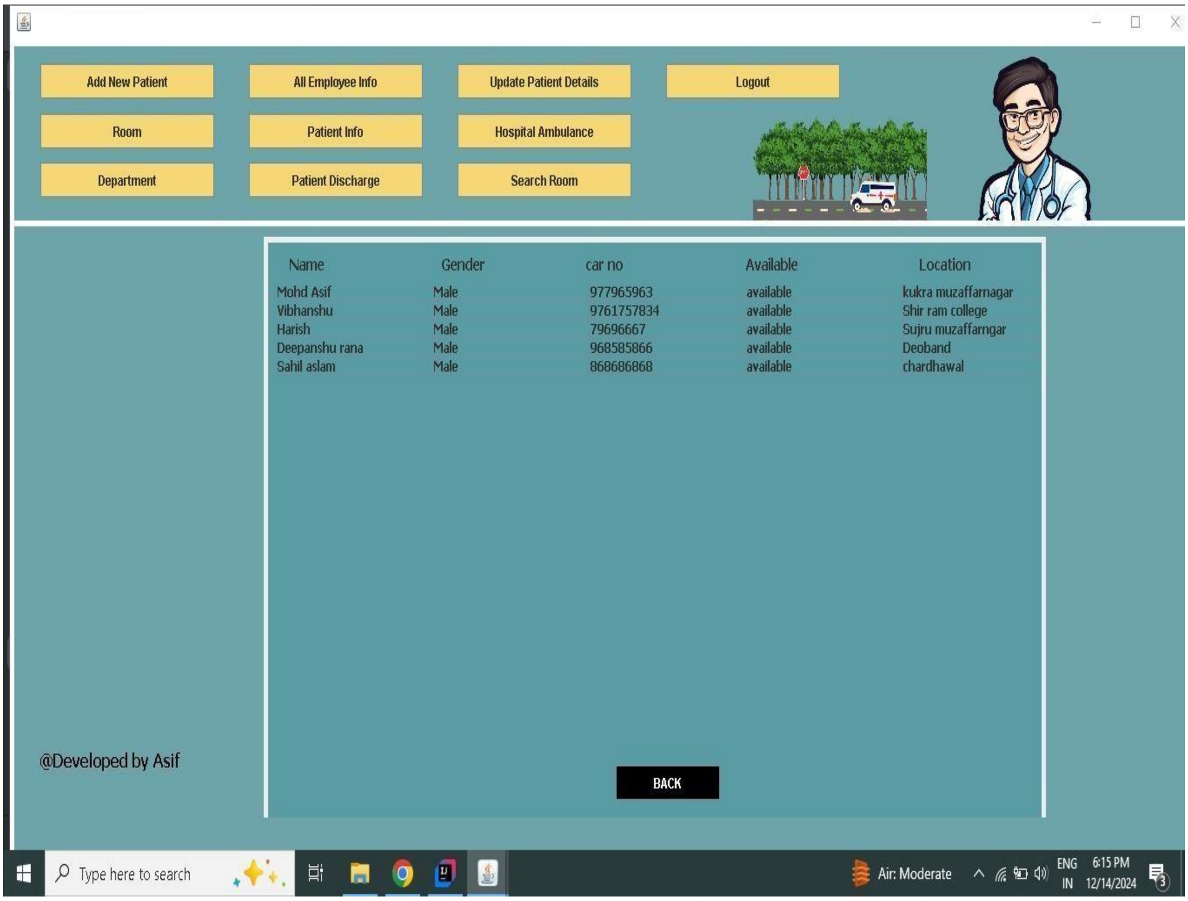
Room Number :

In-Time :

Amount Paid (Rs) :

Pending Amount (Rs) :

12. Search Room Class:-



Name	Gender	car no	Available	Location
Mohd Asif	Male	977965963	available	kukra muzaffarnagar
Vibhanshu	Male	9761757834	available	Shir ram college
Harish	Male	79696667	available	Sujru muzaffarngar
Deepanshu rana	Male	968585866	available	Deoband
Sahil aslam	Male	868686868	available	chardhawal

Source Code

```
package hospital.management.system;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;

public class conn {
    public Connection connection;
    public Statement statement;

    public conn() {
        try {
            // Adjust these parameters as needed
            String url = 
"jdbc:mysql://localhost:3306/hospital_management_system";
            String user = "root";
            String password = "asif@9897";

            // Initialize the connection
            connection = DriverManager.getConnection(url, user, password);
            System.out.println("Asif your are connected with database :");

            // Initialize the statement
            statement = connection.createStatement();
        } catch (Exception e) {
            e.printStackTrace(); // Debugging purposes
        }
    }
}
```

Login Code:-

```
package hospital.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Login extends JFrame implements ActionListener {

    JTextField textField;
    JPasswordField jPasswordField;
    JButton b1,b2;

    Login(){

        JLabel namelabel = new JLabel("Username");
        namelabel.setBounds(40,70,100,30);
        namelabel.setFont(new Font("Tahoma",Font.BOLD,16));
        namelabel.setForeground(Color.BLACK);
        add(namelabel);

        JLabel password = new JLabel("Password");
        password.setBounds(40,120,100,30);
        password.setFont(new Font("Tahoma",Font.BOLD,16));
        password.setForeground(Color.BLACK);
        add(password);

        textField = new JTextField();
        textField.setBounds(150,70,150,30);
        textField.setFont(new Font("Tahoma",Font.PLAIN,15));
        textField.setBackground(new Color(255,179,0));
        add(textField);

        jPasswordField = new JPasswordField();
        jPasswordField.setBounds(150,120, 150,30);
        jPasswordField.setFont(new Font("Tahoma",Font.PLAIN,15));
        jPasswordField.setBackground(new Color(255,179,0));
        add(jPasswordField);
        ImageIcon imageIcon = new
ImageIcon(ClassLoader.getSystemResource("icon/login.jpeg"));
        Image i1 =
imageIcon.getImage().getScaledInstance(500,500,Image.SCALE_DEFAULT);
        ImageIcon imageIcon1 = new ImageIcon(i1);
        JLabel label = new JLabel(imageIcon1);
        label.setBounds(320,-30,400,300);
        add(label);

        b1 = new JButton("Login");
        b1.setBounds(40,180,120,30);
        b1.setFont(new Font("serif",Font.BOLD,15));
        b1.setBackground(Color.BLACK);
        b1.setForeground(Color.white);
        b1.addActionListener(this);
        add(b1);


        b2 = new JButton("Cancel");
        b2.setBounds(180,180,120,30);
        b2.setFont(new Font("serif",Font.BOLD,15));
        b2.setBackground(Color.BLACK);
        b2.setForeground(Color.white);
        b2.addActionListener(this);
        add(b2);

        getContentPane().setBackground(new Color(109,164,170));
        setSize(750,300);
        setLocation(370,250);
        setLayout(null);
        setVisible(true);

    }
    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == b1){
            try{
                conn c= new conn();
                String user = textField.getText();
                String Pass = jPasswordField.getText();

                String q = "select * from login where ID = '"+user+"' and  PW =
'"+Pass+"'";
            ResultSet resultSet = c.statement.executeQuery(q);

            if (resultSet.next()){
               new  Reception();
               setVisible(false);
            }else {
               JOptionPane.showMessageDialog(null,"Asif your password or Id
invalid please check ?? ");
            }

         }catch (Exception E){
            E.printStackTrace();
         }

      }
else
{
         System.exit(10);
      }




   }


   public static void main(String[] args)
{
      new Login();
   }


}
```

RECEPTION CLASS CODE

```
package hospital.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class Reception extends JFrame
{

   Reception()

{

       JPanel panel = new JPanel();
       panel.setLayout(null);
       panel.setBounds(5,160,1525,670);
                             panel.setBackground(new
Color(109,164,170));
       add(panel);

 JLabel namelabel = new JLabel("@Developed by Asif ");
      namelabel.setBounds(25,450,400,20);
      namelabel.setFont(new Font("Tahoma",Font.BOLD,16));
      namelabel.setForeground(Color.BLACK);
      panel.add(namelabel);


       JPanel panel1 = new JPanel();
       panel1.setLayout(null);
       panel1.setBounds(5,5,1525,150);
       panel1.setBackground(new Color(109,164,170));
       add(panel1);

       ImageIcon i1 = new
ImageIcon(ClassLoader.getSystemResource("icon/dr.png"));
       Image image =
i1.getImage().getScaledInstance(250,250,Image.SCALE_DEFAULT);
       ImageIcon i2 = new ImageIcon(image);
       JLabel label = new JLabel(i2);
```

```
        label.setBounds(1050,0,250,250);
        panel1.add(label);



        ImageIcon i11 = new
ImageIcon(ClassLoader.getSystemResource("icon/amb.png"));
        Image image1 =
i11.getImage().getScaledInstance(300,100,Image.SCALE_DEFAULT);
        ImageIcon i22 = new ImageIcon(image1);
        JLabel label1 = new JLabel(i22);
        label1.setBounds(850,50,300,100);
        panel1.add(label1);

        JButton btn1 = new JButton("Add New Patient");
        btn1.setBounds(30,15,200,30);
        btn1.setBackground(new Color(246,215,118));
        panel1.add(btn1);
        btn1.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e)

{
              new  NEW_PATIENT();
            }
        });



        JButton btn2 = new JButton("Room");
        btn2.setBounds(30,58,200,30);
        btn2.setBackground(new Color(246,215,118));
        panel1.add(btn2);
        btn2.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {

                new Room();
            }
        });

        JButton btn3 = new JButton("Department");
        btn3.setBounds(30,100,200,30);
        btn3.setBackground(new Color(246,215,118));
```

```
panel1.add(btn3);
btn3.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {

        new Department();
    }
});

JButton btn4 = new JButton("All Employee Info");
btn4.setBounds(270,15,200,30);
btn4.setBackground(new Color(246,215,118));
panel1.add(btn4);
btn4.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        new Employee_info();
    }
});

JButton btn5 = new JButton("Patient Info");
btn5.setBounds(270,58,200,30);
btn5.setBackground(new Color(246,215,118));
panel1.add(btn5);
btn5.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        new All_patient_info();
    }
});

JButton btn6= new JButton("Patient Discharge");
btn6.setBounds(270,100,200,30);
btn6.setBackground(new Color(246,215,118));
panel1.add(btn6);
btn6.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {

        new patient_discharge();
    }
});
```

```
        JButton btn7= new JButton("Update Patient Details");
        btn7.setBounds(510,15,200,30);
        btn7.setBackground(new Color(246,215,118));
        panel1.add(btn7);
        btn7.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {

                new Update_patient_details();
            }
        });

        JButton btn8= new JButton("Hospital Ambulance");
        btn8.setBounds(510,58,200,30);
        btn8.setBackground(new Color(246,215,118));
        panel1.add(btn8);
        btn8.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e)
{

                new Ambulance();
            }
        });

        JButton btn9= new JButton("Search Room");
        btn9.setBounds(510,100,200,30);
        btn9.setBackground(new Color(246,215,118));
        panel1.add(btn9);
        btn9.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e)
 {

                new Search_room();
            }
        });

        JButton btn10= new JButton("Logout");
        btn10.setBounds(750,15,200,30);
        btn10.setBackground(new Color(246,215,118));
        panel1.add(btn10);
        btn10.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e)
    {
                new Login();
                setVisible(false);

            }
        });


        setSize(1900,1090);
        setLayout(null);
        getContentPane().setBackground(Color.WHITE);
        setVisible(true);


    }

    public static void main(String[] args)
    {
        new Reception();
    }
}
```

ADD PATIENT CODE

```
package hospital.management.system;

import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;


public class All_patient_info extends JFrame{

    All_patient_info(){
        JPanel panel = new JPanel();
        panel.setBounds(5,5,1000,550);
        panel.setBackground(new Color(90, 156, 163));
        panel.setLayout(null);
        add(panel);

        JTable table = new JTable();
        table.setBounds(10,40,1000,350);
        table.setBackground(new Color(90, 156, 163));
        table.setFont(new Font("Tahoma",Font.BOLD,12));
        panel.add(table);

        try{
            conn c = new conn();
            String q = "select * from Patient_Info";
            ResultSet resultSet = c.statement.executeQuery(q);
            table.setModel(DbUtils.resultSetToTableModel(resultSet));

        }catch (Exception e){
            e.printStackTrace();
        }

        JLabel label1 = new JLabel("ID");
        label1.setBounds(31,11,100,14);
        label1.setFont(new Font("Tahoma",Font.BOLD,14));
        panel.add(label1);
```

```
JLabel label2 = new JLabel("Number");
label2.setBounds(150,11,100,14);
label2.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label2);

JLabel label3 = new JLabel("Name");
label3.setBounds(270,11,100,14);
label3.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label3);

JLabel label4 = new JLabel("Gender");
label4.setBounds(360,11,100,14);
label4.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label4);

JLabel label5 = new JLabel("Disease");
label5.setBounds(480,11,100,14);
label5.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label5);

JLabel label6 = new JLabel("Room");
label6.setBounds(600,11,100,14);
label6.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label6);

JLabel label7 = new JLabel("Time");
label7.setBounds(700,11,100,14);
label7.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label7);

JLabel label8 = new JLabel("Deposit");
label8.setBounds(800,11,100,14);
label8.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label8);

JButton button = new JButton("BACK");
button.setBounds(450,400,120,30);
button.setBackground(Color.black);
button.setForeground(Color.white);
panel.add(button);
button.addActionListener(new ActionListener() {
   @Override
```

```
                public void actionPerformed(ActionEvent e) {
                    setVisible(false);
                }
            });




            setUndecorated(true);
            setSize(1000,550);
            setLayout(null);
            setLocation(300,200);
            setVisible(true);

        }


        public static void main(String[] args) {

            new All_patient_info();
        }

    }
```

ROOM CLASS

```
package hospital.management.system;

import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Room extends JFrame {

    JTable table;

    Room(){

        JPanel panel = new JPanel();
        panel.setBounds(5,5,890,590);
        panel.setBackground(new Color(90, 156, 163));
        panel.setLayout(null);
        add(panel);

        ImageIcon imageIcon = new
ImageIcon(ClassLoader.getSystemResource("icon/roomm.png"));
        Image image =
imageIcon.getImage().getScaledInstance(200,200,Image.SCALE_DEFAULT);
        ImageIcon imageIcon1 = new ImageIcon(image);
        JLabel label = new JLabel(imageIcon1);
        label.setBounds(600,200,200,200);
        panel.add(label);

        table = new JTable();
        table.setBounds(10,40,500,400);
        table.setBackground(new Color(90, 156, 163));
        panel.add(table);

        try{

            conn c = new conn();
            String q = "select * from room";
            ResultSet resultSet = c.statement.executeQuery(q);
    table.setModel(DbUtils.resultSetToTableModel(resultSet));
}catch (Exception e){
    e.printStackTrace();
}

JLabel label1 = new JLabel("Room No");
label1.setBounds(12,15,80,15);
label1.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label1);

JLabel label2 = new JLabel("Availability");
label2.setBounds(140,15,80,15);
label2.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label2);

JLabel label3 = new JLabel("Price");
label3.setBounds(290,15,80,15);
label3.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label3);

JLabel label4 = new JLabel("Bed Type");
label4.setBounds(400,15,80,15);
label4.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label4);

JButton back = new JButton("Back");
back.setBounds(200,450,120,30);
back.setBackground(Color.BLACK);
back.setForeground(Color.white);
panel.add(back);
back.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        setVisible(false);
    }
});


setUndecorated(true);
setSize(900,600);
setLayout(null);
setLocation(300,230);
setVisible(true);
```

```
    }
    public static void main(String[] args) {
        new Room();
    }
}
```

DEPARTMENT CLASS

```
package hospital.management.system;

import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Department extends JFrame {



    Department(){

        JPanel panel = new JPanel();
        panel.setBounds(5,5,690,490);
        panel.setLayout(null);
        panel.setBackground(new Color(90, 156, 163));
        add(panel);

        JTable table = new JTable();
        table.setBounds(0,40,700,350);
        table.setBackground(new Color(90, 156, 163));
        table.setFont(new Font("Tahoma",Font.BOLD,14));
        panel.add(table);

        try{
            conn c = new conn();
            String q = "select * from department";
            ResultSet resultSet = c.statement.executeQuery(q);
            table.setModel(DbUtils.resultSetToTableModel(resultSet));
        }catch (Exception e){
            e.printStackTrace();
        }

        JLabel label1 = new JLabel("Department");
        label1.setBounds(20,11,105,20);
        label1.setFont(new Font("Tahoma",Font.BOLD,14));
        panel.add(label1);

        JLabel label2 = new JLabel("Phone Number");
        label2.setBounds(350,11,150,20);
        label2.setFont(new Font("Tahoma",Font.BOLD,14));
        panel.add(label2);

        JButton b1 = new JButton("BACK");
        b1.setBounds(400,410,130,30);
        b1.setBackground(Color.black);
        b1.setForeground(Color.white);
        panel.add(b1);
        b1.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                setVisible(false);
            }
        });

        setUndecorated(true);
        setSize(700,500);
        setLayout(null);
        setLocation(350,250);
        setVisible(true);

    }

    public static void main(String[] args) {
        new Department();
    }

}
```

EMPLOYEE CLASS

```
package hospital.management.system;

import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;


public class Employee_info  extends  JFrame{

    Employee_info(){

        JPanel panel = new JPanel();
        panel.setBounds(5,5,990,590);
        panel.setBackground(new Color(109, 164, 170));
        panel.setLayout(null);
        add(panel);

        JTable table = new JTable();
        table.setBounds(10,50,980,300);
        table.setBackground(new Color(109, 164, 170));
        table.setFont(new Font("Tahoma",Font.BOLD,12));
        panel.add(table);



        try{
            conn c = new conn();
            String q = "select * from EMP_INFO";
            ResultSet resultSet = c.statement.executeQuery(q);
            table.setModel(DbUtils.resultSetToTableModel(resultSet));
        }catch (Exception e){
            e.printStackTrace();
        }

        JLabel label1 = new JLabel("Name");
        label1.setBounds(30,9,70,20);
```

```
label1.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label1);


JLabel label2 = new JLabel("Age");
label2.setBounds(175,9,70,20);
label2.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label2);

JLabel label3 = new JLabel("Phone Number");
label3.setBounds(340,9,150,20);
label3.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label3);

JLabel label4 = new JLabel("Salary");
label4.setBounds(500,9,150,20);
label4.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label4);

JLabel label5 = new JLabel("Gmail");
label5.setBounds(700,9,150,20);
label5.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label5);

JLabel label6 = new JLabel("Aadhar Number");
label6.setBounds(830,9,150,20);
label6.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label6);

JButton button = new JButton("BACK");
button.setBounds(350,400,120,30);
button.setBackground(Color.BLACK);
button.setForeground(Color.white);
panel.add(button);
button.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        setVisible(false);
    }
});
```

```
        setUndecorated(true);
        setSize(1000,450);
        setLocation(300,230);
        setLayout(null);
        setVisible(true);


    }


    public static void main(String[] args) {
        new Employee_info();
    }

}
```

PATIENT CLASS

```
package hospital.management.system;

import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;


public class All_patient_info extends  JFrame{

    All_patient_info(){
        JPanel panel = new JPanel();
        panel.setBounds(5,5,1000,550);
        panel.setBackground(new Color(90, 156, 163));
        panel.setLayout(null);
        add(panel);

        JTable table = new JTable();
        table.setBounds(10,40,1000,350);
        table.setBackground(new Color(90, 156, 163));
        table.setFont(new Font("Tahoma",Font.BOLD,12));
        panel.add(table);

        try{
            conn c = new conn();
            String q = "select * from Patient_Info";
            ResultSet resultSet = c.statement.executeQuery(q);
            table.setModel(DbUtils.resultSetToTableModel(resultSet));

        }catch (Exception e){
            e.printStackTrace();
        }

        JLabel label1 = new JLabel("ID");
        label1.setBounds(31,11,100,14);
        label1.setFont(new Font("Tahoma",Font.BOLD,14));
```

```
panel.add(label1);

JLabel label2 = new JLabel("Number");
label2.setBounds(150,11,100,14);
label2.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label2);

JLabel label3 = new JLabel("Name");
label3.setBounds(270,11,100,14);
label3.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label3);

JLabel label4 = new JLabel("Gender");
label4.setBounds(360,11,100,14);
label4.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label4);

JLabel label5 = new JLabel("Disease");
label5.setBounds(480,11,100,14);
label5.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label5);

JLabel label6 = new JLabel("Room");
label6.setBounds(600,11,100,14);
label6.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label6);

JLabel label7 = new JLabel("Time");
label7.setBounds(700,11,100,14);
label7.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label7);

JLabel label8 = new JLabel("Deposit");
label8.setBounds(800,11,100,14);
label8.setFont(new Font("Tahoma",Font.BOLD,14));
panel.add(label8);

JButton button = new JButton("BACK");
button.setBounds(450,400,120,30);
button.setBackground(Color.black);
button.setForeground(Color.white);
panel.add(button);
button.addActionListener(new ActionListener() {
```

```
            @Override
            public void actionPerformed(ActionEvent e) {
                setVisible(false);
            }
        });




        setUndecorated(true);
        setSize(1000,550);
        setLayout(null);
        setLocation(300,200);
        setVisible(true);

    }


    public static void main(String[] args) {

        new All_patient_info();
    }

}
```

PATIENT DISCHARGE

```
package hospital.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.util.Date;

public class patient_discharge extends JFrame{


    patient_discharge(){


        JPanel panel = new JPanel();
        panel.setBounds(5,5,790,390);
        panel.setBackground(new Color(90, 156, 163));
        panel.setLayout(null);
        add(panel);

        JLabel label = new JLabel("CHECK-OUT");
        label.setBounds(100,20,150,20);
        label.setFont(new Font("Tahoma",Font.BOLD,20));
        label.setForeground(Color.white);
        panel.add(label);

        JLabel label2 = new JLabel("Customer Id");
        label2.setBounds(30,80,150,20);
        label2.setFont(new Font("Tahoma",Font.BOLD,14));
        label2.setForeground(Color.white);
        panel.add(label2);

        Choice choice = new Choice();
        choice.setBounds(200,80,150,25);
        panel.add(choice);


        try{
            conn c = new conn();
            ResultSet resultSet = c.statement.executeQuery("select * from
Patient_Info");
            while (resultSet.next()){
                choice.add(resultSet.getString("number"));
            }

        }catch (Exception e){
            e.printStackTrace();
        }


        JLabel label3 = new JLabel("Room Number");
        label3.setBounds(30,130,150,20);
        label3.setFont(new Font("Tahoma",Font.BOLD,14));
        label3.setForeground(Color.white);
        panel.add(label3);

        JLabel RNo = new JLabel();
        RNo.setBounds(200,130,150,20);
        RNo.setFont(new Font("Tahoma",Font.BOLD,14));
        RNo.setForeground(Color.white);
        panel.add(RNo);

        JLabel label4 = new JLabel("In Time");
        label4.setBounds(30,180,150,20);
        label4.setFont(new Font("Tahoma",Font.BOLD,14));
        label4.setForeground(Color.white);
        panel.add(label4);

        JLabel INTime = new JLabel();
        INTime.setBounds(200,180,250,20);
        INTime.setFont(new Font("Tahoma",Font.BOLD,14));
        INTime.setForeground(Color.white);
        panel.add(INTime);

        JLabel label5 = new JLabel("Out Time");
        label5.setBounds(30,230,150,20);
        label5.setFont(new Font("Tahoma",Font.BOLD,14));
        label5.setForeground(Color.white);
        panel.add(label5);

        Date date = new Date();
        JLabel OUTTime = new JLabel(""+date);
        OUTTime .setBounds(200,230,250,20);
        OUTTime .setFont(new Font("Tahoma",Font.BOLD,14));
        OUTTime .setForeground(Color.white);
        panel.add(OUTTime );



        JButton discharge = new JButton("Discharge");
        discharge.setBounds(30,300,120,30);
        discharge.setBackground(Color.black);
        discharge.setForeground(Color.white);
        panel.add(discharge);
        discharge.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {

                conn c = new conn();
                try {
                    c.statement.executeUpdate("delete from Patient_Info where number
= '"+choice.getSelectedItem()+"'");
                    c.statement.executeUpdate("update room set Availability =
'Available' where room_no = '"+RNo.getText()+"'");
                    JOptionPane.showMessageDialog(null,"Done");
                    setVisible(false);
                }catch (Exception E){
                    E.printStackTrace();
                }



            }
        });


        JButton Check = new JButton("Check");
        Check.setBounds(170,300,120,30);
        Check.setBackground(Color.black);
        Check.setForeground(Color.white);
        panel.add(Check);

        Check.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                conn c = new conn();
                try{
                    ResultSet resultSet = c.statement.executeQuery("select * from
Patient_Info where number = '"+choice.getSelectedItem()+"'");
                    while (resultSet.next()){
                        RNo.setText(resultSet.getString("room_no"));
                        INTime.setText(resultSet.getString("time"));
                    }
                }catch (Exception E){
                    E.printStackTrace();
                }

            }
        });



        JButton Back = new JButton("Back");
        Back.setBounds(300,300,120,30);
        Back.setBackground(Color.black);
        Back.setForeground(Color.white);
        panel.add(Back);
        Back.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                setVisible(false);
            }
        });




        setUndecorated(true);
        setSize(800,400);
        setLayout(null);
        setLocation(400,250);
        setVisible(true);
```

```
    }

    public static void main(String[] args) {
        new patient_discharge();
    }


}
```

Update Patient Details Class

```
package hospital.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Update_patient_details extends JFrame  {

    Update_patient_details(){

        JPanel panel = new JPanel();
        panel.setBounds(5,5,940,490);
        panel.setBackground(new Color(90, 156, 163));
        panel.setLayout(null);
        add(panel);

        ImageIcon imageIcon = new
ImageIcon(ClassLoader.getSystemResource("icon/updated.png"));
        Image image =
imageIcon.getImage().getScaledInstance(300,300,Image.SCALE_DEFAULT);
        ImageIcon imageIcon1 = new ImageIcon(image);
        JLabel label = new JLabel(imageIcon1);
        label.setBounds(500,60,300,300);
        panel.add(label);

        JLabel label1 = new JLabel("Update Patient Details");
        label1.setBounds(124,11,260,25);
        label1.setFont(new Font("Tahoma",Font.BOLD,20));
        label1.setForeground(Color.white);
        panel.add(label1);


        JLabel label2 = new JLabel("Name :");
        label2.setBounds(25,88,100,14);
        label2.setFont(new Font("Tahoma",Font.PLAIN,14));
        label2.setForeground(Color.white);
        panel.add(label2);
        Choice choice = new Choice();
        choice.setBounds(248,85,140,25);
        panel.add(choice);

        try {
            conn c= new conn();
            ResultSet resultSet = c.statement.executeQuery("select * from
Patient_Info");
            while (resultSet.next()){
                choice.add(resultSet.getString("Name"));
            }

        }catch (Exception e){
            e.printStackTrace();
        }

        JLabel label3 = new JLabel("Room Number :");
        label3.setBounds(25,129,100,14);
        label3.setFont(new Font("Tahoma",Font.PLAIN,14));
        label3.setForeground(Color.white);
        panel.add(label3);

        JTextField textFieldR = new JTextField();
        textFieldR.setBounds(248,129,140,20);
        panel.add(textFieldR);

        JLabel label4 = new JLabel("In-Time  :");
        label4.setBounds(25,174,100,14);
        label4.setFont(new Font("Tahoma",Font.PLAIN,14));
        label4.setForeground(Color.white);
        panel.add(label4);


        JTextField textFieldINTIme = new JTextField();
        textFieldINTIme.setBounds(248,174,140,20);
        panel.add(textFieldINTIme);

        JLabel label5 = new JLabel("Amount Paid (Rs) :");
        label5.setBounds(25,216,150,14);
        label5.setFont(new Font("Tahoma",Font.PLAIN,14));
        label5.setForeground(Color.white);
        panel.add(label5);
        JTextField textFieldAmount = new JTextField();
        textFieldAmount.setBounds(248,216,140,20);
        panel.add(textFieldAmount);

        JLabel label6 = new JLabel("Pending Amount (Rs) :");
        label6.setBounds(25,261,150,14);
        label6.setFont(new Font("Tahoma",Font.PLAIN,14));
        label6.setForeground(Color.white);
        panel.add(label6);

        JTextField textFieldPending = new JTextField();
        textFieldPending.setBounds(248,261,140,20);
        panel.add(textFieldPending);

        JButton check = new JButton("CHECK");
        check.setBounds(281,378,100,23);
        check.setBackground(Color.black);
        check.setForeground(Color.white);
        panel.add(check);
        check.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String id = choice.getSelectedItem();
                String q = "select * from Patient_Info where Name = '"+id+"'";
                try{
                    conn c = new conn();
                    ResultSet resultSet = c.statement.executeQuery(q);
                    while (resultSet.next()){
                        textFieldR.setText(resultSet.getString("room_no"));
                        textFieldINTIme.setText(resultSet.getString("Time"));
                        textFieldAmount.setText(resultSet.getString("Deposite"));
                    }

                    ResultSet resultSet1 = c.statement.executeQuery("select * from
room where room_no = '"+textFieldR.getText()+"'");
                    while (resultSet1.next()){
                        String price = resultSet1.getString("Price");
                        int amountPaid = Integer.parseInt(price) -
Integer.parseInt(textFieldAmount.getText());
                        textFieldPending.setText(""+amountPaid);
                    }

                }catch (Exception E){
                    E.printStackTrace();
                }
            }
        });



        JButton back = new JButton("Back");
        back.setBounds(150,378,100,23);
        back.setBackground(Color.black);
        back.setForeground(Color.white);
        panel.add(back);
        back.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                setVisible(false);
            }
        });


        JButton update = new JButton("UPDATE");
        update.setBounds(25,378,89,23);
        update.setBackground(Color.black);
        update.setForeground(Color.white);
        panel.add(update);
        update.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                try {
                    conn c = new conn();
                    String q = choice.getSelectedItem();
                    String room = textFieldR.getText();
                    String time = textFieldINTIme.getText();
                    String amount = textFieldAmount.getText();
                    c.statement.executeUpdate("update Patient_Info set room_no = 
'"+room+"', Time = '"+time+"', Deposite = '"+amount+"' where name = '"+q+"'"
);
                    JOptionPane.showMessageDialog(null,"Updated Successfully");
                    setVisible(false);
                }catch (Exception E){
                    E.printStackTrace();
                }
            }
        });
        setUndecorated(true);
        setSize(950,500);
        setLayout(null);
        setLocation(400,250);
        setVisible(true);


    }

    public static void main(String[] args) {
        new Update_patient_details();
    }


}
```

Ambulance Class Code

```
package hospital.management.system;

import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Ambulance extends JFrame {


    Ambulance() {
        JPanel panel = new JPanel();
        panel.setBounds(5, 5, 890, 590);
        panel.setBackground(new Color(90, 156, 163));
        panel.setLayout(null);
        add(panel);

        JTable table = new JTable();
        table.setBounds(10, 35, 900, 400);
        table.setBackground(new Color(90, 156, 163));
        table.setFont(new Font("Tahoma", Font.BOLD, 12));
        panel.add(table);

        try{
            conn c = new conn();
            String q = "select * from Ambulance";
            ResultSet resultSet = c.statement.executeQuery(q);
            table.setModel(DbUtils.resultSetToTableModel(resultSet));

        }catch (Exception e){
            e.printStackTrace();
        }



        JLabel label1 = new JLabel("Name");
        label1.setBounds(25, 11, 100, 14);
        label1.setFont(new Font("Tahoma", Font.BOLD, 14));
    panel.add(label1);

    JLabel label2 = new JLabel("Gender");
    label2.setBounds(200, 11, 100, 14);
    label2.setFont(new Font("Tahoma", Font.BOLD, 14));
    panel.add(label2);

    JLabel label3 = new JLabel("car no ");
    label3.setBounds(366, 11, 100, 14);
    label3.setFont(new Font("Tahoma", Font.BOLD, 14));
    panel.add(label3);

    JLabel label4 = new JLabel("Available");
    label4.setBounds(550, 11, 100, 14);
    label4.setFont(new Font("Tahoma", Font.BOLD, 14));
    panel.add(label4);

    JLabel label5 = new JLabel("Location");
    label5.setBounds(750, 11, 100, 14);
    label5.setFont(new Font("Tahoma", Font.BOLD, 14));
    panel.add(label5);


    JButton button = new JButton("BACK");
    button.setBounds(400, 450, 120, 30);
    button.setBackground(Color.black);
    button.setForeground(Color.white);
    panel.add(button);
    button.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            setVisible(false);
        }
    });

    setUndecorated(true);
    setSize(900, 500);
    setLayout(null);
    setLocation(300, 200);
    setVisible(true);


}
```

```
    public static void main(String[] args) {
    new Ambulance();

    }

}
```

Search Room Class

```
package hospital.management.system;
import net.proteanit.sql.DbUtils;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Search_room extends  JFrame {
   Choice choice;
   JTable table;

   Search_room() {
      JPanel panel = new JPanel();
      panel.setBounds(5, 5, 690, 490);
      panel.setBackground(new Color(90, 156, 163));
      panel.setLayout(null);
      add(panel);

      JLabel For = new JLabel("Search For Room");
      For.setBounds(250, 11, 186, 31);
      For.setForeground(Color.white);
      For.setFont(new Font("Tahoma", Font.BOLD, 20));
      panel.add(For);

      JLabel status = new JLabel("Status :");
      status.setBounds(70, 70, 80, 20);
      status.setForeground(Color.white);
      status.setFont(new Font("Tahoma", Font.BOLD, 14));
      panel.add(status);

      choice = new Choice();
      choice.setBounds(170, 70, 120, 20);
      choice.add("Available");
      choice.add("Occupied");
      panel.add(choice);

      table = new JTable();
      table.setBounds(0, 187, 700, 210);
table.setBackground(new Color(90, 156, 163));
table.setForeground(Color.white);
panel.add(table);

try {
   conn c = new conn();
   String q = "select * from room";
   ResultSet resultSet = c.statement.executeQuery(q);
   table.setModel(DbUtils.resultSetToTableModel(resultSet));

} catch (Exception e) {
   e.printStackTrace();
}

JLabel Roomno = new JLabel("Room Number");
Roomno.setBounds(23, 162, 150, 20);
Roomno.setForeground(Color.white);
Roomno.setFont(new Font("Tahoma", Font.BOLD, 14));
panel.add(Roomno);

JLabel available = new JLabel("Availability");
available.setBounds(175, 162, 150, 20);
available.setForeground(Color.white);
available.setFont(new Font("Tahoma", Font.BOLD, 14));
panel.add(available);

JLabel price = new JLabel("Price");
price.setBounds(458, 162, 150, 20);
price.setForeground(Color.white);
price.setFont(new Font("Tahoma", Font.BOLD, 14));
panel.add(price);

JLabel Bed = new JLabel("Bed Type");
Bed.setBounds(580, 162, 150, 20);
Bed.setForeground(Color.white);
Bed.setFont(new Font("Tahoma", Font.BOLD, 14));
panel.add(Bed);

JButton Search = new JButton("Search");
Search.setBounds(200, 420, 120, 25);
Search.setBackground(Color.black);
Search.setForeground(Color.white);
panel.add(Search);
```

```
        Search.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String q = "select * from Room where Availability = '" +
choice.getSelectedItem() + "'";
                try {
                    conn c = new conn();
                    ResultSet resultSet = c.statement.executeQuery(q);
                    table.setModel(DbUtils.resultSetToTableModel(resultSet));
                } catch (Exception E) {
                    E.printStackTrace();
                }
            }
        });

        JButton Back = new JButton("Back");
        Back.setBounds(380, 420, 120, 25);
        Back.setBackground(Color.black);
        Back.setForeground(Color.white);
        panel.add(Back);
        Back.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                setVisible(false);
            }
        });


        setUndecorated(true);
        setSize(700, 500);
        setLayout(null);
        setLocation(450, 250);
        setVisible(true);

    }

    public static void main(String[] args) {
        new Search_room();
    }
}
```

```
import sqlite3
import tkinter as tk
from tkinter import messagebox

# Function to connect to the SQLite database
def connect_db():
    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute('''
    CREATE TABLE IF NOT EXISTS laundry_orders (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        customer_name TEXT,
        item_name TEXT,
        quantity INTEGER,
        status TEXT
    )''')
    conn.commit()
    conn.close()

# Function to add a laundry order
def add_order():
    customer_name = customer_name_entry.get()
    item_name = item_name_entry.get()
    quantity = quantity_entry.get()

    if not customer_name or not item_name or not quantity:
        messagebox.showwarning("Input Error", "All fields must be filled!")
        return

    quantity = int(quantity)
    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute('''
    INSERT INTO laundry_orders (customer_name, item_name, quantity, status)
    VALUES (?, ?, ?, ?)''', (customer_name, item_name, quantity, 'Pending'))
    conn.commit()
    conn.close()

    # Clear input fields
    customer_name_entry.delete(0, tk.END)
    item_name_entry.delete(0, tk.END)
    quantity_entry.delete(0, tk.END)
    messagebox.showinfo("Success", "Order added successfully!")
    refresh_orders()

# Function to mark an order as completed
def mark_completed():
    selected_order_index = order_listbox.curselection()
    if not selected_order_index:
        messagebox.showwarning("Selection Error", "Please select an order to
mark as completed.")
        return
    selected_order_id = order_listbox.get(selected_order_index)
    order_id = selected_order_id.split()[1].strip(',')
    status = selected_order_id.split()[9].strip(',')
    if status == "Completed":
        messagebox.showwarning("Selection Error", "Please select an pending
order only")
        return
    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute('''
    UPDATE laundry_orders
    SET status = ?
    WHERE id = ?''', ('Completed', order_id))
    conn.commit()
    conn.close()

    messagebox.showinfo("Success", "Order marked as completed!")
    refresh_orders()

# Function to refresh the order list from the database
def refresh_orders():
    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute('SELECT * FROM laundry_orders')
    orders = c.fetchall()
    conn.close()

    order_listbox.delete(0, tk.END)
    for order in orders:
        order_listbox.insert(tk.END, f"ID: {order[0]}, {order[1]} - {order[2]}
(Qty: {order[3]}) - Status: {order[4]}")

def delete_orders():
    selected_order_index = order_listbox.curselection()
    if not selected_order_index:
        messagebox.showwarning("Selection Error", "Please select an order to
mark as completed.")
        return
    selected_order_id = order_listbox.get(selected_order_index)
    order_id = selected_order_id.split()[1].strip(',')
    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute('''
    DELETE FROM laundry_orders
    WHERE id = ?''', (order_id,))
    conn.commit()
    conn.close()

    messagebox.showinfo("Success", "Order deleted successfully")
    refresh_orders()

def load_order_data():
    selected_index = order_listbox.curselection()

    if not selected_index:
        messagebox.showwarning("Selection Error", "Please select an order to
edit.")
        return

    selected_order = order_listbox.get(selected_index)

    order_id = selected_order.split()[1].strip(',')

    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute("SELECT * FROM laundry_orders WHERE id = ?", (order_id,))
    order_details = c.fetchone()
    conn.close()

    if order_details:
        customer_name_entry.delete(0, tk.END)
        customer_name_entry.insert(0, order_details[1])

        item_name_entry.delete(0, tk.END)
        item_name_entry.insert(0, order_details[2])
        quantity_entry.delete(0, tk.END)
        quantity_entry.insert(0, order_details[3])

        update_button.grid(row=3, column=1, columnspan=2, pady=10)

        global current_order_id
        current_order_id = order_id
    else:
        messagebox.showwarning("Error", "Order not found.")

def update_order():
    customer_name = customer_name_entry.get()
    item_name = item_name_entry.get()
    quantity = quantity_entry.get()

    if not customer_name or not item_name or not quantity:
        messagebox.showwarning("Input Error", "Please fill all fields.")
        return

    conn = sqlite3.connect('laundry_management.db')
    c = conn.cursor()
    c.execute('''
    UPDATE laundry_orders
    SET customer_name = ?, item_name = ?, quantity = ?
    WHERE id = ?''', (customer_name, item_name, quantity, current_order_id))

    conn.commit()
    conn.close()

    messagebox.showinfo("Success", "Order updated successfully.")

    # Hide the update button after update
    update_button.grid_forget()
    customer_name_entry.delete(0, tk.END)
    item_name_entry.delete(0, tk.END)
    quantity_entry.delete(0, tk.END)

    refresh_orders()

# Setup the Tkinter window
window = tk.Tk()
window.title("Laundry Management System")
```

```
# Create UI elements
customer_name_label = tk.Label(window, text="Customer Name:")
customer_name_label.grid(row=0, column=0, padx=10, pady=10)
customer_name_entry = tk.Entry(window)
customer_name_entry.grid(row=0, column=1, padx=10, pady=10)

item_name_label = tk.Label(window, text="Item Name:")
item_name_label.grid(row=1, column=0, padx=10, pady=10)
item_name_entry = tk.Entry(window)
item_name_entry.grid(row=1, column=1, padx=10, pady=10)

quantity_label = tk.Label(window, text="Quantity:")
quantity_label.grid(row=2, column=0, padx=10, pady=10)
quantity_entry = tk.Entry(window)
quantity_entry.grid(row=2, column=1, padx=10, pady=10)

add_order_button = tk.Button(window, text="Add Order",
command=add_order)
add_order_button.grid(row=3, column=0, columnspan=2, pady=10)

update_button = tk.Button(window, text="Update", command=update_order)
update_button.grid(row=3, column=1, columnspan=2, pady=10)
update_button.grid_forget()

# Order list display
order_listbox_label = tk.Label(window, text="Order list:")
order_listbox_label.grid(row=4, column=0, columnspan=2, pady=10)

order_listbox = tk.Listbox(window, height=10, width=70)
order_listbox.grid(row=5, column=0, columnspan=2, padx=10, pady=10)

window.grid_columnconfigure(0, weight=0)
window.grid_columnconfigure(1, weight=0)
window.grid_columnconfigure(2, weight=0)

# Define buttons with consistent width
button_width = 15

mark_completed_button = tk.Button(window, text="Mark Completed",
command=mark_completed, width=button_width)
mark_completed_button.grid(row=6, column=0)

delete_button = tk.Button(window, text="Delete", command=delete_orders,
width=button_width)
delete_button.grid(row=6, column=1, padx=2, pady=2)

edit_button = tk.Button(window, text="Edit", command=load_order_data,
width=button_width)
edit_button.grid(row=6, column=2, padx=2, pady=2)


# Initialize the database and UI
connect_db()
refresh_orders()

# Run the Tkinter event loop
window.mainloop()
```

BIBLIOGRAPHY

The following are the books references that have been studied in the duration of making of this project.

1. Youtube

   Code with harry

2. Java books

Head First Java
Kathy Sierra, 2003

Effective Java (3rd Edition)
Joshua Bloch, 2008
Effective Java (3rd Edition)