

[Doctor's Clinic]

A Project Report for Industrial Internship

In the partial fulfillment for the award of the degree of

B.Tech/MCA/BCA

In the

Department OF

University Of Engineering & Management, Kolkata

At

Ardent Computech Pvt. Ltd.



CERTIFICATE FROM SUPERVISOR

This is to certify that, Students have successfully completed the project titled "Doctor's Clinic" under my supervision during the period from "03.06.2023" to "23.06.2023" which is in partial fulfilment of requirements for the award of the **B.Tech/MCA/BCA** degree and submitted to the Department of civil engineering.

STUDENT	SIGNATURE

Signature of the Supervisor

Date:

Name of the Project Supervisor: **Mr. Anindya Mukherjee**



ACKNOWLEDGEMENT

The achievement that is associated with the successful completion of any task would be incomplete without mentioning the names of those people whose endless cooperation made it possible. Their constant guidance and encouragement made all our efforts successful.

We take this opportunity to express our deep gratitude towards our project mentor, **[Mr. Anindya Mukherjee]** for giving such valuable suggestions, guidance and encouragement during the development of this project work.

Last but not the least we are grateful to all the faculty members of Ardent Computech Pvt. Ltd. for their support.

Doctor's Clinic

Minor Project Submitted in Partial Fulfilment of the
Requirements for the degree of

Bachelor of Computer
Application by

Pritikana Mondal(7029302254)

Under the guidance of
Mr. Anindya Mukherjee



University Of Engineering & Management

University Of Engineering & Management

Is a fully government approved and UGC recognized University located in
New Town, Kolkata

Faculty of B.Tech department

Certificate of Recommendation

This is to certify that all the group member have completed their project

work titled “Minor project on: Doctors Clinic”, under the direct supervision and guidance of Mr. Anindya Mukherjee. We are satisfied with her work, which is being presented for the partial fulfilment of the degree of Bachelor of Technology (B.Tech).

(Anindya Mukherjee)

Mentor

Date:

[SKG sir]

HOD of B.tech Department

University of Engineering & Management

Date:

University Of Engineering & Management

Is a fully government approved and UGC recognized University located in
New Town, Kolkata

Faculty of B.Tech department

Certificate of Approval

The foregoing Minor project is hereby approved as a creditable study of Bachelor of Technology (B.tech) and presented in a manner satisfactory to warrant its acceptance as a pre- requisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or any statement made, opinion expressed or conclusion therein but approve this Major project only for the purpose for which it is submitted.

(Signature of the examiner)

ACKNOWLEDGEMENT

Success of any project depends largely on the encouragement and guidelines of many others. We take this sincere opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project work.

We are highly indebted to for his guidance and constant supervision as well as for providing necessary information regarding the project & also for his support in completing the project.

We would like to express my gratitude towards our parents & our University Teachers for their kind co-operation and encouragement which help me in completion of this project.

Our thanks and appreciations also go to our friends in developing the project and people who have willingly helped us out with their abilities.

Table of Contents

1. Introduction

- I. Purpose
- II. Scope
- III. Technologies to be use

2. Software Requirement Specification

- I. Introduction
- II. Purpose
- III. External Interface requirement
- IV. Functional requirement
- V. Security requirement
- VI. Performance requirement
- VII. Maintainability

3. Data flow Diagram (DFD)

4. Entity relationship diagram (ERD)

5. Database Design

6. Screen design

7. Testing

8. Future Scope

9. Conclusion

INTRODUCTION

1.1 Purpose

Our software solves the complications coming from managing all the paper works of every receptionist and doctor with the various departments of the doctor's clinic. DC provides the ability to manage all the paperwork in one place, reducing the work of doctor arranging and analyzing the paperwork of the receptionist.

It maintains three levels of users-

- Admin Level
- Receptionist Level
- Doctor Level

The Application includes: -

- Maintaining Receptionist Details.
- Maintaining Doctor Details.
- Maintaining Appointments.

1.2 Scope

It can be used in any Hospital, Clinic or nursing home for maintaining patient or receptionist details, doctor details and appointment details .

1.3 Technologies to be used

This project will be an application to be developed in java having MySql as backend database.

- Database Design (MySql)
- Form Design (JAVA swing module)
- Coding (JAVA)

SOFTWARE REQUIREMENT SPECIFICATION

1. Introduction

The SRS is produced at the culmination of the analysis task. The function and performance allocated to software as part of the system engineering and refined by establishing a complete information description, a detailed functional description, a representation of system behaviour, indication of performance requirements and design constraints, appropriate validation criteria and the other information related to requirements.

The SRS is technical specification of requirement of Doctor's Clinic. This specification describes what the proposed system should do without describing how it will do it. It also describes complete external behaviour of proposed system.

2. Purpose

The main purpose of our system is to make Doctors management task easy and is to develop software that replaces the manual hospital system into automated Doctor's Clinic. This document serves as the unambiguous guide for the developers of this software system.

3. External Interface Requirements: -

3.1 User Interface: -

User interface is designed in a user friendly manner and the user, in another end he has to give the order, for that he will interface with keyboard and mouse.

3.2 Hardware Interface: -

- OS – windows XP
- Hard disk – 80 GB
- RAM – 1 GB
- Keyboard – Standard QWERTY keyboard for interface
- Mouse – Standard mouse with 2 buttons

3.3 Software Interface: -

- Front end – Java language
- OS – Windows
- Back end – MySQL Server
- IDE – intellij IDEA

4. Functional Requirements: -

4.1 Admin Module: -

4.1.1 Admin Manager: -

- Admin manager can add, delete and update details of receptionists.
- Admin manger can view all the receptionists along with its details in this tab.
- Admin manager can add, delete and update details of doctors.
- Admin manger can view all the doctor along with its details in this tab.

4.2 Front Desk Module: -

4.2.1. Admin Manager: -

- Admin manager can add, delete and update details of receptionists.
- Admin manger can view all the receptionists along with its details in this tab.
- Admin manager can add, delete and update details of doctors.
- Admin manger can view all the doctor along with its details in this tab.

4.2.2 Doctor Manager:-

- Doctor manager can add, delete and update details of doctors.
- Doctor manger can view all the doctor along with its details in this tab.

4.2.3 Receptionist Manager: -

- Receptionist manager can add, delete and update appointments.
- Receptionist manger can view all the receptionists along with its details in this tab.

4.3 Doctor Module: -

4.3.1 My Appointment

- Doctor can check details of the patient who have booked their appointments.

4.3.1 My Information

- Doctor can view and update his own details in the portal

4.3.1 View all Receptionist

- Doctor can view all the Patients admitted in the hospital.

5. Security Requirements:-

- Admins, front desk and doctors everyone has different user names and passwords so that they can login to the app securely.
- Any modification (add, delete or update) can only be done by the manager of that departments.

6. Performance Requirements: -

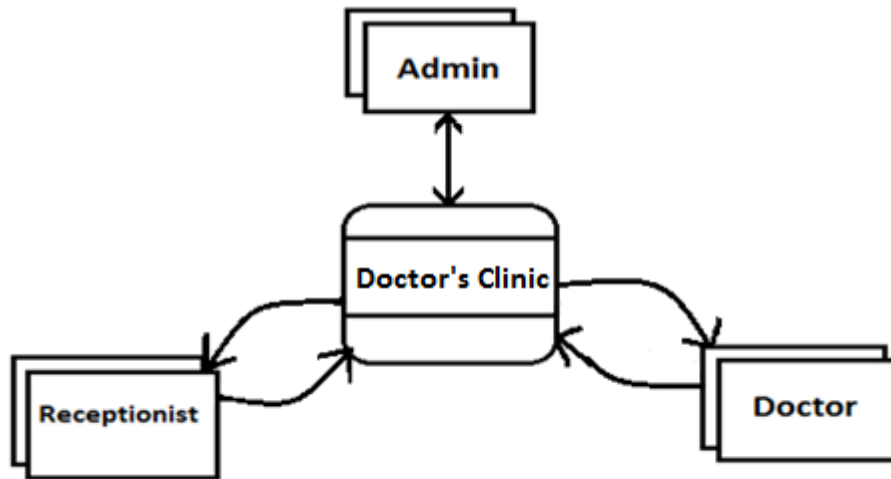
- The system shall give responses in 1 second
- The System must support 2 logins at a time in each department.
- The user-interface screen shall response within 5 seconds.
- The system must comfort to the Database guidelines.

7. Maintainability: -

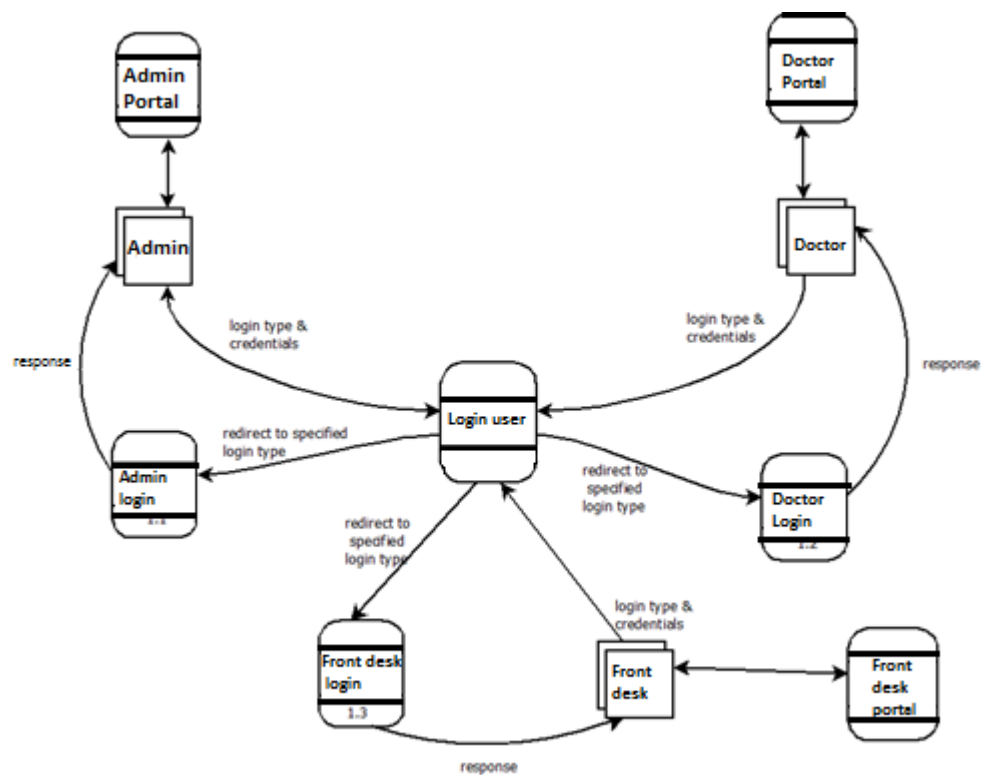
- The system shall provide the capability to back-up the Data.
- The system shall be available and accessible all the time.

Data Flow Diagram [DRD]

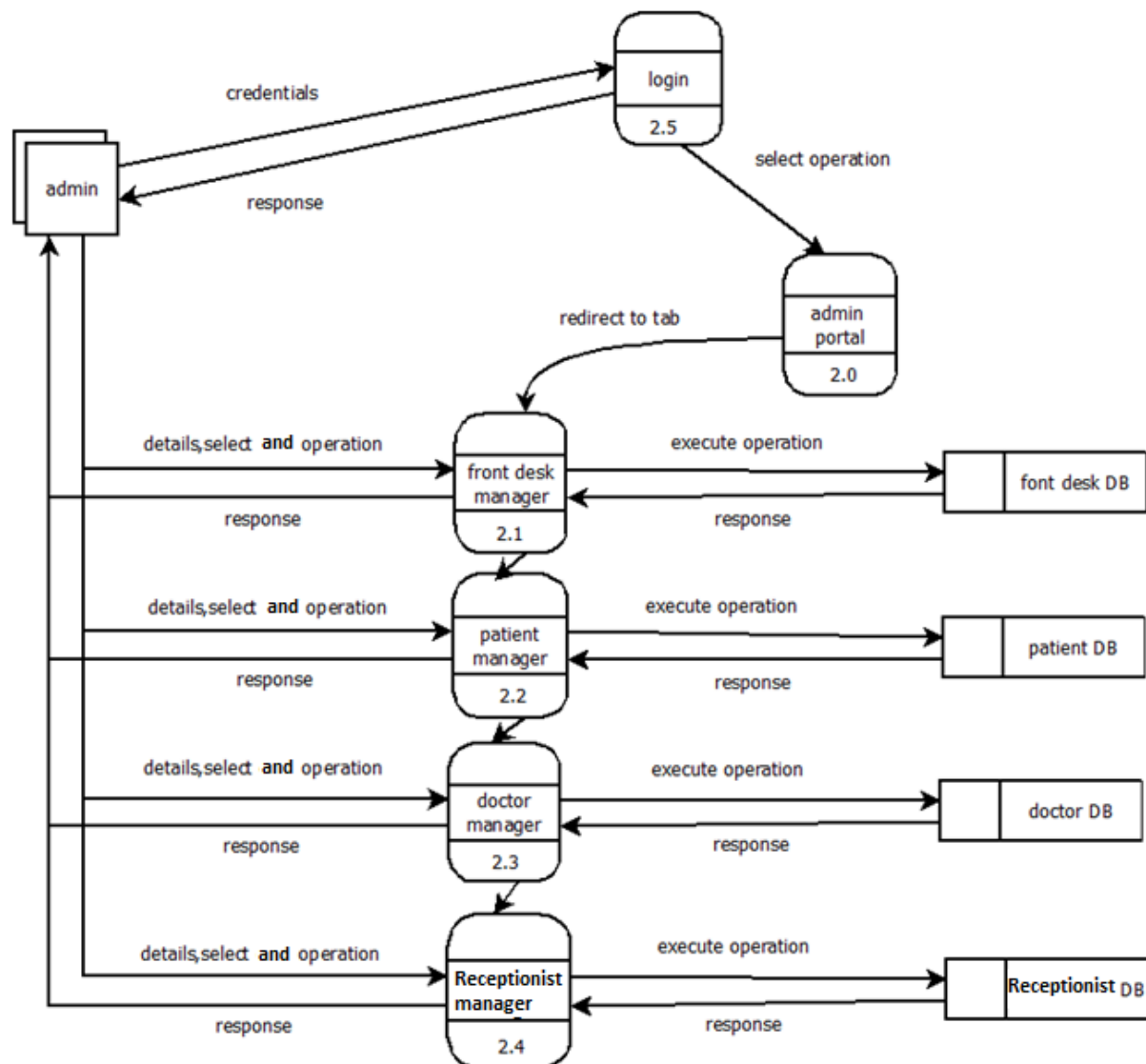
1. Context Level Diagram



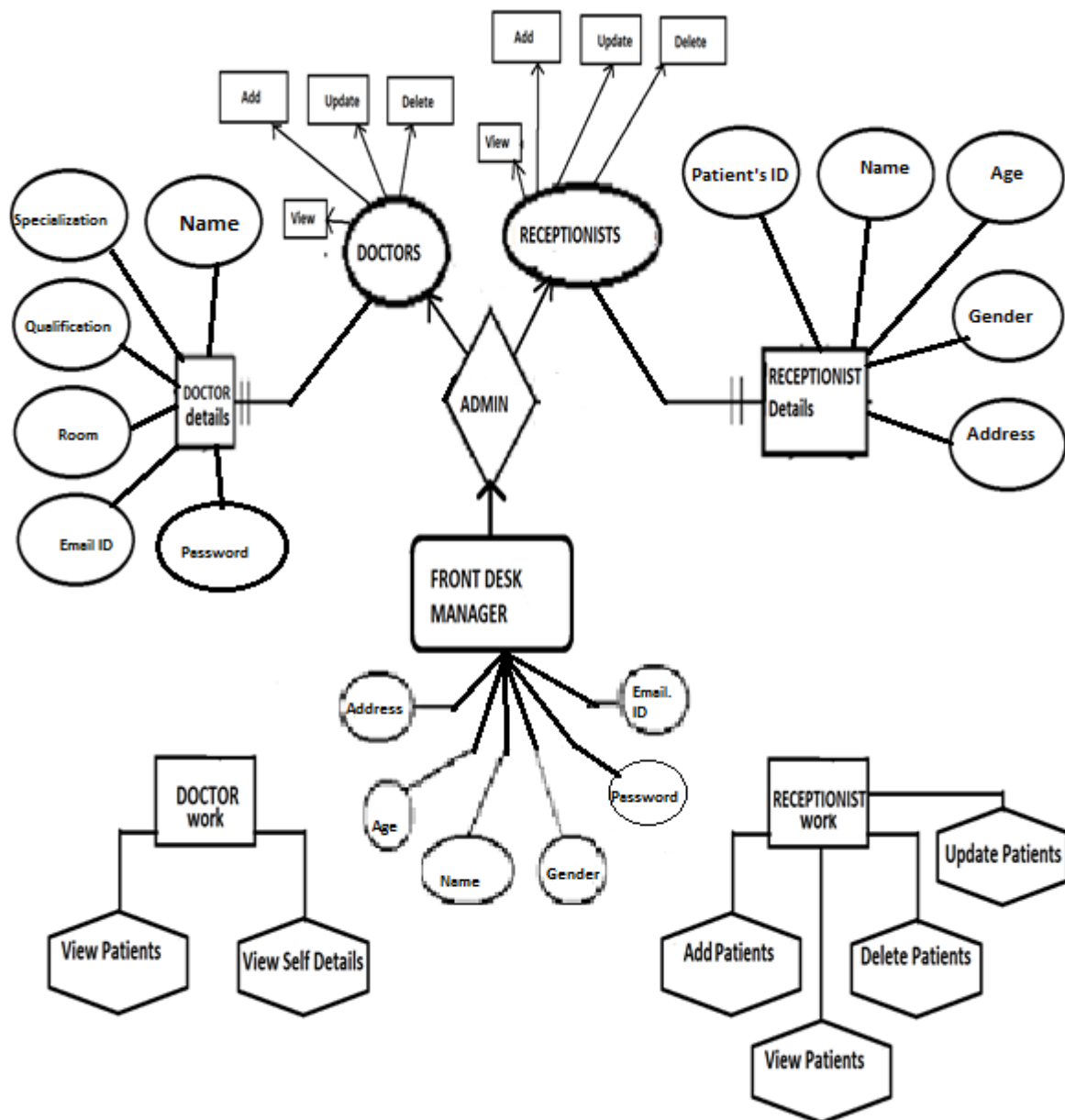
2. Level 1 DFD of Login system



3. Level 2 DFD of Admin Functionalities



Entity Relationship Diagram [ERD]



Database Design

1. Patient Database

patientid	name	age	gender	address
01	Tanmoy	18	Male	45 M.G Road

2. Doctor Database

Field	Type	Null	Key	Default	Extra
doc_id	varchar(10)	NO	PRI	NULL	
first_name	varchar(20)	YES		NULL	
last_name	varchar(20)	YES		NULL	
field	varchar(20)	YES		NULL	
age	varchar(5)	YES		NULL	
phn_no	varchar(20)	YES		NULL	

3. Receptionist Database

name	email	mobilen	password
Sayan Bhattacharya	sayan25@gmail.com	1234567890	sayan1234
Sanjay Dey	sanjay1234@gmail.com	9874561234	sanjay@2023

Screen Design

1. Home page



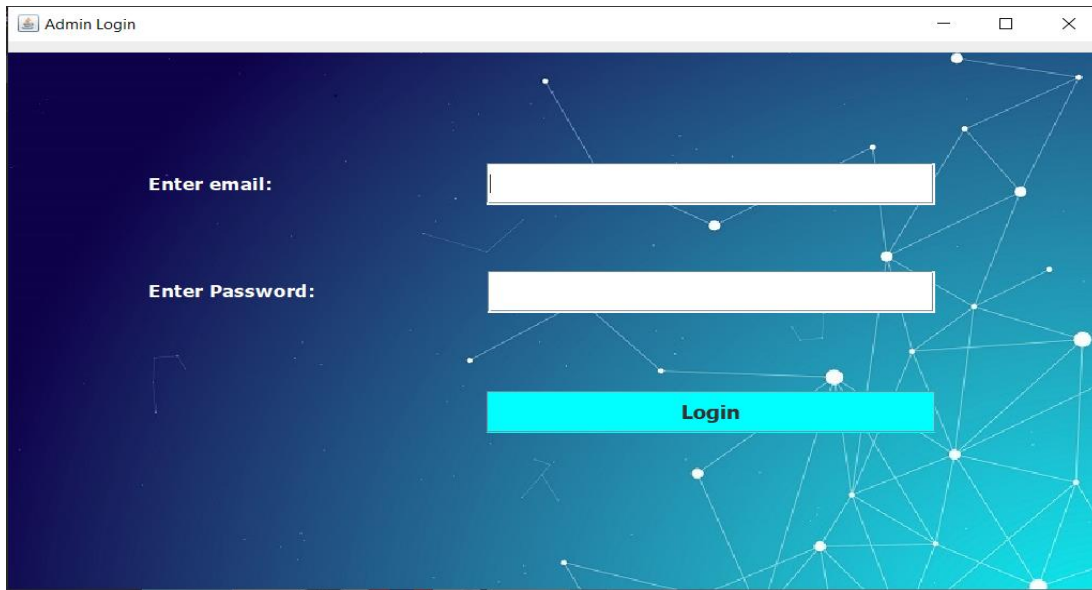
```
package org.Study;
```

```
import java.awt.EventQueue;
```

```
import javax.swing.JFrame;  
import javax.swing.JPanel;  
import javax.swing.border.EmptyBorder;  
import java.awt.Color;  
import java.awt.Button;  
import java.awt.SystemColor;  
import java.awt.Font;  
import java.awt.event.ActionListener;  
import java.awt.event.ActionEvent;  
import javax.swing.JLabel;  
import javax.swing.ImageIcon;
```

```
public class Home extends JFrame {  
  
    private JPanel contentPane;  
  
    /**  
     * Launch the application.  
     */  
    public static void main(String[] args) {  
        EventQueue.invokeLater(new Runnable() {  
            public void run() {  
                try {  
                    Home frame = new Home();  
                    frame.setVisible(true);  
                } catch (Exception e) {  
                    e.printStackTrace();  
                }  
            }  
        });  
    }  
}
```

2. Admin login



```
package org.Study;
import java.awt.event.*;
import javax.swing.*;
import java.awt.Color;
import java.awt.Font;
public class Admin {

    public static void main(String[] args) {
        JFrame frame=new JFrame("Admin Login");
        JLabel l1=new JLabel("Enter email:");
        l1.setForeground(Color.WHITE);
        l1.setFont(new Font("Tahoma", Font.BOLD, 15));
        l1.setBounds(105, 107, 100, 30);
        JTextField tf1=new JTextField();
        tf1.setBounds(358, 105, 335, 36);
        JLabel l2=new JLabel("Enter Password:");
        l2.setForeground(Color.WHITE);
        l2.setFont(new Font("Tahoma", Font.BOLD, 15));
        l2.setBounds(105, 199, 176, 30);
        JTextField tf2=new JTextField();
        tf2.setBounds(359, 198, 334, 36);
        JButton b1=new JButton("Login");
        b1.setFont(new Font("Tahoma", Font.BOLD, 16));
        b1.setBackground(Color.CYAN);
        b1.setBounds(358, 301, 335, 36);
        frame.getContentPane().setLayout(null);
```

```

        frame.getContentPane().add(l1);
        frame.getContentPane().add(tf1);
        frame.getContentPane().add(l2);
        frame.getContentPane().add(tf2);
        frame.getContentPane().add(b1);

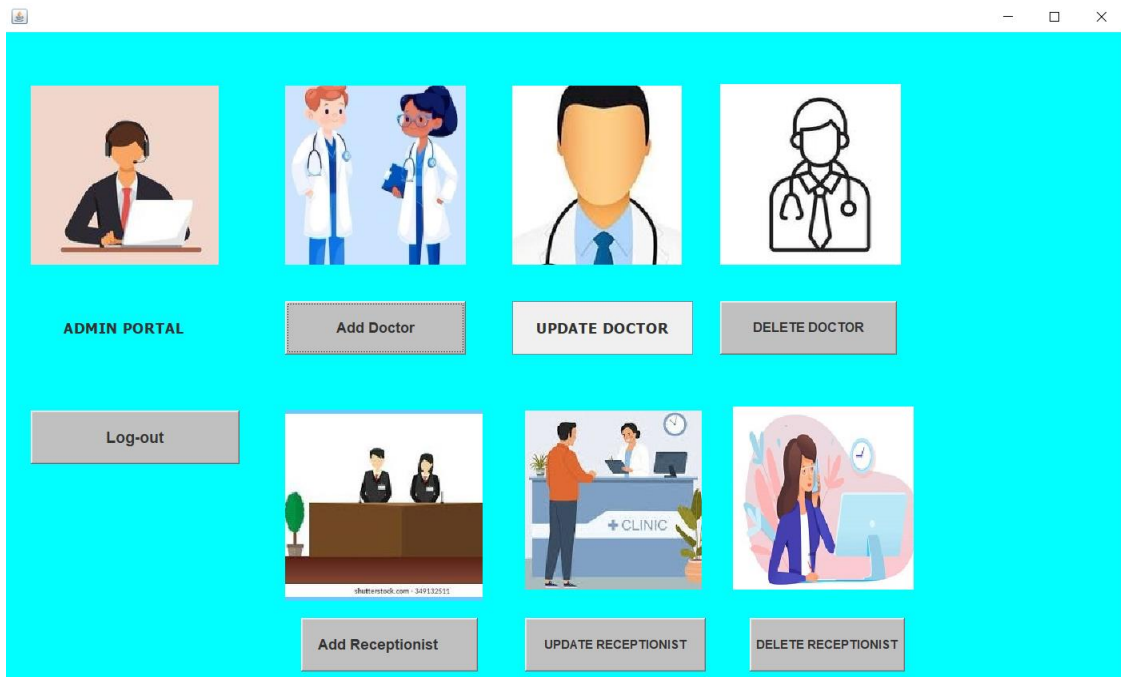
        JLabel lblNewLabel = new JLabel("New label");
        lblNewLabel.setIcon(new
ImageIcon(Admin.class.getResource("/Images/Login.jpg")));
        lblNewLabel.setBounds(0, 10, 817, 462);
        frame.getContentPane().add(lblNewLabel);
        frame.setSize(831,509);
        frame.setVisible(true);
        b1.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
        String Email=tf1.getText();
        String Pass=tf2.getText();
        if(Email.equals("admin@gmail.com") && Pass.equals("admin"))
        {
                AdminPanel.main(new String[] {});
                frame.dispose();
        }
        else
                JOptionPane.showMessageDialog(frame,"Email or Password not
Matched!");
        }}};

    }

}

```

3. Admin panel



```
package org.Study;
```

```
import java.awt.EventQueue;
import javax.swing.JOptionPane;
```

```
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import java.awt.Button;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.awt.Font;
import javax.swing.JLabel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.SystemColor;
```

```
public class AdminPanel extends JFrame {
```

```
    private JPanel contentPane;
    static String uemail;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    AdminPanel frame = new AdminPanel();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
    /**
```

```

* Create the frame.
*/
public AdminPanel() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 1136, 679);
    contentPane = new JPanel();
    contentPane.setBackground(Color.CYAN);
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    Button button = new Button("Add Doctor");
    button.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            AddDoc.main(new String[] {});
        }
    });

    button.setFont(new Font("Dialog", Font.BOLD, 14));
    button.setBackground(Color.LIGHT_GRAY);
    button.setBounds(281, 266, 181, 53);
    contentPane.add(button);

    Button button_2 = new Button("Add Receptionist\r\n");
    button_2.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            AddRecep.main(new String[] {});
        }
    });
    button_2.setFont(new Font("Dialog", Font.BOLD, 14));
    button_2.setBackground(Color.LIGHT_GRAY);
    button_2.setBounds(297, 579, 177, 53);
    contentPane.add(button_2);

    Button button_4 = new Button("Log-out");
    button_4.setFont(new Font("Dialog", Font.BOLD, 15));
    button_4.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            Home.main(new String[] {});
        }
    });
    button_4.setBackground(Color.LIGHT_GRAY);
    button_4.setBounds(27, 374, 209, 53);
    contentPane.add(button_4);

    JPanel panel = new JPanel();
    panel.setBackground(new Color(255, 255, 204));
    panel.setBounds(27, 53, 188, 177);
    contentPane.add(panel);
    panel.setLayout(null);

    JLabel lblNewLabel = new JLabel("New label");
    lblNewLabel.setIcon(new ImageIcon(AdminPanel.class.getResource("/Images/Admin
icon.jpg"))));
    lblNewLabel.setBounds(0, 0, 199, 192);
    panel.add(lblNewLabel);

    JPanel panel_1 = new JPanel();
    panel_1.setBackground(new Color(102, 204, 255));
    panel_1.setBounds(281, 53, 181, 177);
    contentPane.add(panel_1);

```

```

        panel_1.setLayout(null);

        JLabel lblNewLabel_1 = new JLabel("New label");
        lblNewLabel_1.setIcon(new
ImageIcon(AdminPanel.class.getResource("/Images/WhatsApp2.jpg")));
        lblNewLabel_1.setBounds(0, 0, 181, 177);
        panel_1.add(lblNewLabel_1);

        JPanel panel_2 = new JPanel();
        panel_2.setBackground(new Color(102, 204, 255));
        panel_2.setBounds(281, 373, 198, 189);
        contentPane.add(panel_2);
        panel_2.setLayout(null);

        JLabel lblNewLabel_2 = new JLabel("New label");
        lblNewLabel_2.setIcon(new
ImageIcon(AdminPanel.class.getResource("/Images/WhatsApp5.jpg")));
        lblNewLabel_2.setBounds(0, 0, 198, 189);
        panel_2.add(lblNewLabel_2);

        JButton btnNewButton = new JButton("UPDATE DOCTOR");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                //uemail=JOptionPane.showInputDialog(contentPane,"Enter Email to Update");
                //if()
                UpdateDoc.main(new String[]{});
                //frame.dispose();
            }
        });
        btnNewButton.setBackground(new Color(240, 240, 240));
        btnNewButton.setFont(new Font("Tahoma", Font.BOLD, 15));
        btnNewButton.setBounds(508, 266, 181, 53);
        contentPane.add(btnNewButton);

        JLabel lblNewLabel_3 = new JLabel("New label");
        lblNewLabel_3.setIcon(new
ImageIcon(AdminPanel.class.getResource("/Images/WhatsApp4.jpg")));
        lblNewLabel_3.setBounds(508, 53, 170, 177);
        contentPane.add(lblNewLabel_3);

        JLabel lblNewLabel_4 = new JLabel("New label");
        lblNewLabel_4.setIcon(new
ImageIcon(AdminPanel.class.getResource("/Images/WhatsApp3.jpg")));
        lblNewLabel_4.setBounds(521, 374, 177, 177);
        contentPane.add(lblNewLabel_4);

        Button button_1 = new Button("UPDATE RECEPTIONIST");
        button_1.setFont(new Font("Dialog", Font.BOLD, 12));
        button_1.setBackground(Color.LIGHT_GRAY);
        button_1.setBounds(521, 579, 181, 53);
        contentPane.add(button_1);

        JLabel lblNewLabel_5 = new JLabel("ADMIN PORTAL");
        lblNewLabel_5.setBackground(Color.WHITE);
        lblNewLabel_5.setFont(new Font("Tahoma", Font.BOLD, 15));
        lblNewLabel_5.setBounds(59, 277, 177, 31);
        contentPane.add(lblNewLabel_5);

        Button button_3 = new Button("DELETE DOCTOR");
        button_3.setFont(new Font("Dialog", Font.BOLD, 13));
        button_3.setBackground(Color.LIGHT_GRAY);
        button_3.setBounds(716, 266, 177, 53);
        contentPane.add(button_3);

        JLabel lblNewLabel_6 = new JLabel("New label");

```

```

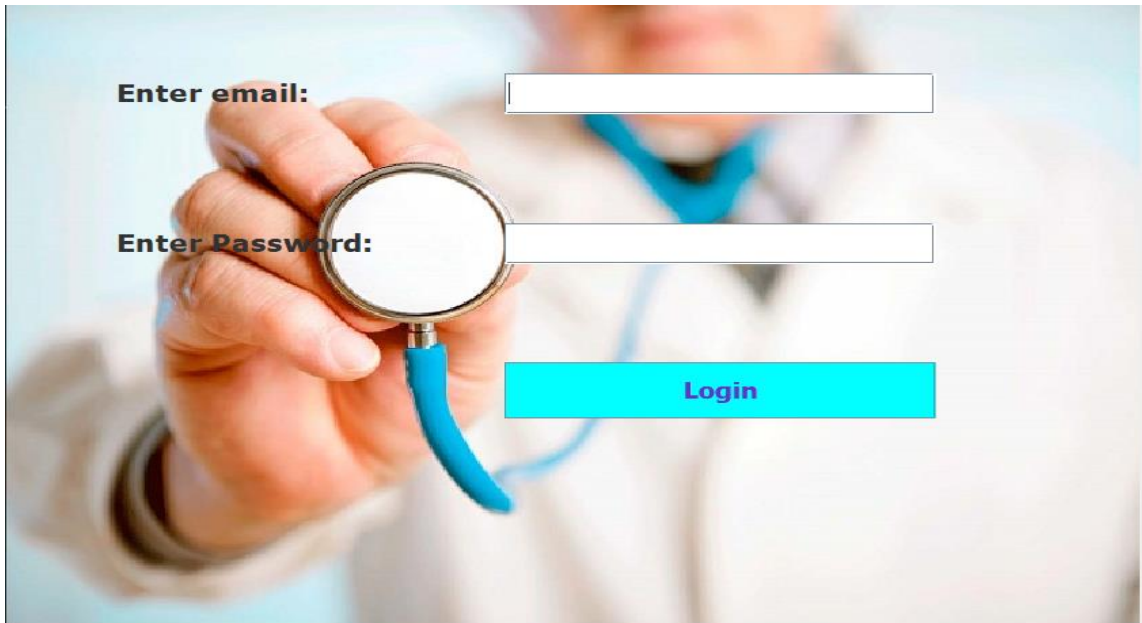
        lblNewLabel_6.setIcon(new
ImageIcon(AdminPanel.class.getResource("/Images/male-doctor-icon-free-vector.jpg")));
        lblNewLabel_6.setBounds(716, 52, 181, 178);
        contentPane.add(lblNewLabel_6);

        Button button_5 = new Button("DELETE RECEPTIONIST");
        button_5.setFont(new Font("Dialog", Font.BOLD, 12));
        button_5.setBackground(Color.LIGHT_GRAY);
        button_5.setBounds(746, 579, 155, 53);
        contentPane.add(button_5);

        JLabel lblNewLabel_7 = new JLabel("New label");
        lblNewLabel_7.setIcon(new
ImageIcon(AdminPanel.class.getResource("/Images/istockphoto-1208588832-612x612.jpg")));
        lblNewLabel_7.setBounds(729, 370, 181, 181);
        contentPane.add(lblNewLabel_7);
    }
}

```

4. Doctor Login



```
package org.Study;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.*;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import java.awt.Font;
import javax.swing.ImageIcon;
import java.awt.Color;

public class DocLogin {
    static String Email,Pass;
    public static void main(String[] args) {
        JFrame frame=new JFrame("Receptionist Login");
        JLabel l1=new JLabel("Enter email:");
        l1.setFont(new Font("Tahoma", Font.BOLD, 18));
        l1.setBounds(65, 60, 121, 30);
        JTextField tf1=new JTextField();
        tf1.setBounds(294, 61, 254, 30);
        JLabel l2=new JLabel("Enter Password:");
        l2.setFont(new Font("Tahoma", Font.BOLD, 18));
        l2.setBounds(65, 169, 185, 30);
        JTextField tf2=new JTextField();
        tf2.setBounds(294, 170, 254, 30);
        JButton b1=new JButton("Login");
        b1.setForeground(new Color(102, 51, 204));
        b1.setFont(new Font("Tahoma", Font.BOLD, 16));
        b1.setBackground(Color.CYAN);
        b1.setBounds(294, 271, 254, 41);
        frame.getContentPane().setLayout(null);
    }
}
```



```

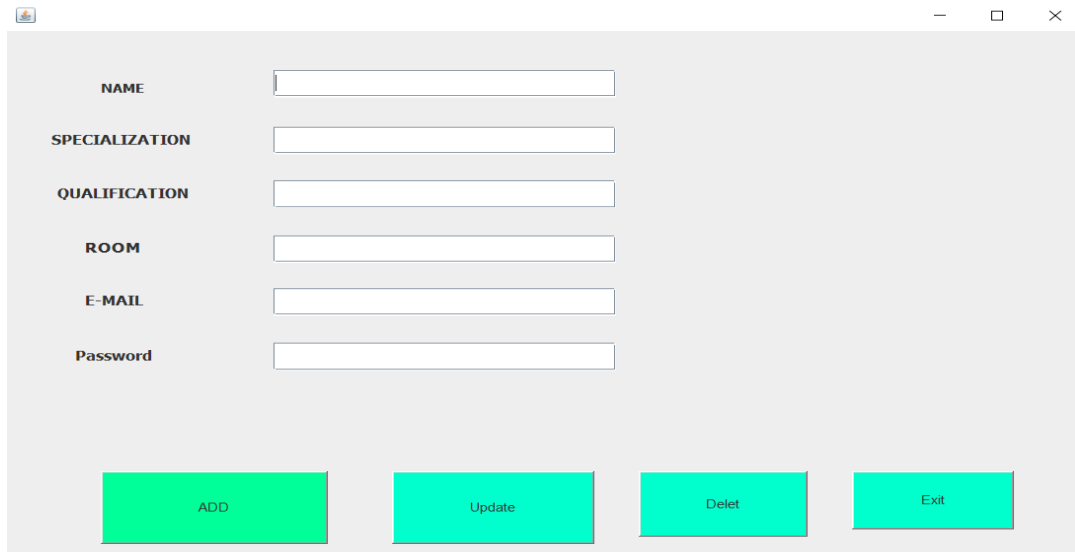
        frame.getContentPane().add(l1);
        frame.getContentPane().add(tf1);
        frame.getContentPane().add(l2);
        frame.getContentPane().add(tf2);
        frame.getContentPane().add(b1);

        JLabel lblNewLabel = new JLabel("New label");
        lblNewLabel.setIcon(new
ImageIcon(DocLogin.class.getResource("/Images/doctor-background.jpg")));
        lblNewLabel.setBounds(0, 0, 668, 472);
        frame.getContentPane().add(lblNewLabel);
        frame.setSize(692,509);
        frame.setVisible(true);
        b1.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
    Email=tf1.getText();
    Pass=tf2.getText();
    Connection con;
    PreparedStatement pstmt;
    ResultSet rs;
    boolean status=false;
    try
    {
        Class.forName("com.mysql.cj.jdbc.Driver");

        con=DriverManager.getConnection("jdbc:mysql://localhost:3306/doctor","root","say
andeep2023");
        pstmt=con.prepareStatement("select *from doc where email=? and
password=?");
        pstmt.setString(1, Email);
        pstmt.setString(2, Pass);
        rs=pstmt.executeQuery();
        status=rs.next();
        if(status)
        {
            Recep1.main(new String[]{});
            frame.dispose();
        }
        else {
            JOptionPane.showMessageDialog(frame,"Email or Password not Matched!");
        }
    }catch(Exception e1) {}
}});
    }
}

```

5. Add Doctor



```
package org.Study;
```

```
import java.awt.EventQueue;
```

```
import javax.swing.JFrame;  
import javax.swing.JPanel;  
import javax.swing.border.EmptyBorder;  
import java.awt.Button;  
import java.awt.Color;  
import javax.swing.JTextField;  
import javax.swing.JLabel;  
import javax.swing.JOptionPane;
```

```
import java.awt.Font;  
import java.awt.event.ActionListener;  
import java.awt.event.ActionEvent;  
import java.sql.*;  
public class AddDoc extends JFrame {
```

```
    private JPanel contentPane;  
    private JTextField textField;  
    private JTextField textField_1;  
    private JTextField textField_2;  
    private JTextField textField_3;  
    private JTextField textField_4;  
    private JTextField textField_5;
```

```
    /**  
     * Launch the application.  
     */  
    public static void main(String[] args) {  
        EventQueue.invokeLater(new Runnable() {  
            public void run() {  
                try {  
                    AddDoc frame = new AddDoc();  
                    frame.setVisible(true);  
                } catch (Exception e) {  
                    e.printStackTrace();  
                }  
            }  
        });  
    }
```

```

    }

    /**
     * Create the frame.
     */
    public AddDoc() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 888, 553);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

        setContentPane(contentPane);
        contentPane.setLayout(null);

        Button button = new Button("ADD");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String Name=textField.getText();
                String Specialization=textField_1.getText();
                String Qualification=textField_2.getText();
                String Room=textField_3.getText();
                String Email=textField_4.getText();
                String Password=textField_5.getText();

                Connection con;
                PreparedStatement pstmt;
                try
                {
                    Class.forName("com.mysql.cj.jdbc.Driver");

                    con=DriverManager.getConnection("jdbc:mysql://localhost:3306/doctor","root","sayandeep2
023");

                    pstmt=con.prepareStatement("insert into doc
values(?,?,?,?,?)");

                    pstmt.setString(1,Name);
                    pstmt.setString(2,Specialization);
                    pstmt.setString(3,Qualification);
                    pstmt.setString(4,Room);
                    pstmt.setString(5,Email);
                    pstmt.setString(6>Password);
                    pstmt.executeUpdate();
                    JOptionPane.showMessageDialog(contentPane,"Successfully
added");

                    con.close();
                }catch(Exception e1) {}

            }
        });
        button.setBackground(new Color(0, 255, 153));
        button.setBounds(76, 434, 184, 72);
        contentPane.add(button);

        textField = new JTextField();
        textField.setBounds(216, 39, 278, 27);
        contentPane.add(textField);
        textField.setColumns(10);

        textField_1 = new JTextField();
        textField_1.setBounds(216, 95, 278, 27);
        contentPane.add(textField_1);
        textField_1.setColumns(10);

        textField_2 = new JTextField();

```

```

textField_2.setBounds(216, 148, 278, 27);
contentPane.add(textField_2);
textField_2.setColumns(10);

textField_3 = new JTextField();
textField_3.setBounds(216, 202, 278, 27);
contentPane.add(textField_3);
textField_3.setColumns(10);

JLabel lblNewLabel = new JLabel("NAME");
lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 12));
lblNewLabel.setBounds(76, 50, 45, 13);
contentPane.add(lblNewLabel);

JLabel lblNewLabel_1 = new JLabel("SPECIALIZATION");
lblNewLabel_1.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_1.setBounds(36, 101, 137, 13);
contentPane.add(lblNewLabel_1);

JLabel lblNewLabel_2 = new JLabel("QUALIFICATION");
lblNewLabel_2.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_2.setBounds(41, 154, 122, 13);
contentPane.add(lblNewLabel_2);

JLabel lblNewLabel_3 = new JLabel("E-MAIL");
lblNewLabel_3.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_3.setBounds(39, 292, 45, -47);
contentPane.add(lblNewLabel_3);

JLabel lblNewLabel_4 = new JLabel("ROOM\r\n");
lblNewLabel_4.setFont(new Font("Tahoma", Font.BOLD, 14));
lblNewLabel_4.setBounds(63, 207, 72, 13);
contentPane.add(lblNewLabel_4);

textField_4 = new JTextField();
textField_4.setBounds(216, 254, 278, 27);
contentPane.add(textField_4);
textField_4.setColumns(10);

JLabel lblNewLabel_5 = new JLabel("E-MAIL");
lblNewLabel_5.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_5.setBounds(63, 260, 58, 13);
contentPane.add(lblNewLabel_5);

Button button_1 = new Button("Update");
button_1.setBackground(new Color(0, 255, 204));
button_1.setBounds(312, 434, 164, 72);
contentPane.add(button_1);

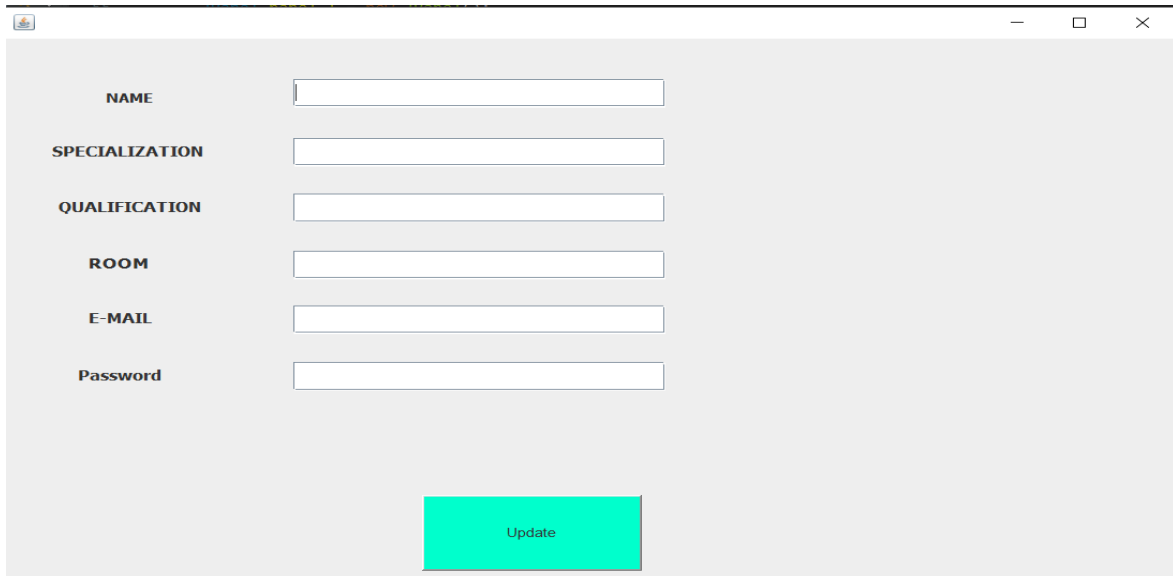
Button button_2 = new Button("Delet");
button_2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
    }
});
button_2.setBackground(new Color(0, 255, 204));
button_2.setBounds(512, 434, 137, 65);
contentPane.add(button_2);

Button button_3 = new Button("Exit");
button_3.setBackground(new Color(0, 255, 204));
button_3.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        AdminPanel.main(new String[] {});
    }
}

```

```
});  
button_3.setBounds(685, 434, 131, 58);  
contentPane.add(button_3);  
  
textField_5 = new JTextField();  
textField_5.setBounds(216, 308, 278, 27);  
contentPane.add(textField_5);  
textField_5.setColumns(10);  
  
JLabel lblNewLabel_6 = new JLabel("Password");  
lblNewLabel_6.setFont(new Font("Tahoma", Font.BOLD, 13));  
lblNewLabel_6.setBounds(55, 314, 80, 13);  
contentPane.add(lblNewLabel_6);  
}  
}
```

6. Update Doctor



```
package org.Study;
```

```
import java.awt.EventQueue;
```

```
import javax.swing.JFrame;  
import javax.swing.JPanel;  
import javax.swing.border.EmptyBorder;  
import java.awt.Button;  
import java.awt.Color;  
import javax.swing.JTextField;  
import javax.swing.JLabel;  
import javax.swing.JOptionPane;
```

```
import java.awt.Font;  
import java.awt.event.ActionListener;  
import java.awt.event.ActionEvent;  
import java.sql.*;  
public class UpdateDoc extends JFrame {
```

```
    private JPanel contentPane;  
    private JTextField textField;  
    private JTextField textField_1;  
    private JTextField textField_2;  
    private JTextField textField_3;  
    private JTextField textField_4;  
    private JTextField textField_5;
```

```
    /**
```

```
     * Launch the application.
```

```
     */
```

```
    public static void main(String[] args) {  
        EventQueue.invokeLater(new Runnable() {  
            public void run() {  
                try {  
                    UpdateDoc frame = new UpdateDoc();  
                    frame.setVisible(true);  
                } catch (Exception e) {  
                    e.printStackTrace();  
                }  
            }  
        })  
    }
```

```

    });
}

/**
 * Create the frame.
 */
public UpdateDoc() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 888, 553);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    textField = new JTextField();
    textField.setBounds(216, 39, 278, 27);
    contentPane.add(textField);
    textField.setColumns(10);

    textField_1 = new JTextField();
    textField_1.setBounds(216, 95, 278, 27);
    contentPane.add(textField_1);
    textField_1.setColumns(10);

    textField_2 = new JTextField();
    textField_2.setBounds(216, 148, 278, 27);
    contentPane.add(textField_2);
    textField_2.setColumns(10);

    textField_3 = new JTextField();
    textField_3.setBounds(216, 202, 278, 27);
    contentPane.add(textField_3);
    textField_3.setColumns(10);

    JLabel lblNewLabel = new JLabel("NAME");
    lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 12));
    lblNewLabel.setBounds(76, 50, 45, 13);
    contentPane.add(lblNewLabel);

    JLabel lblNewLabel_1 = new JLabel("SPECIALIZATION");
    lblNewLabel_1.setFont(new Font("Tahoma", Font.BOLD, 13));
    lblNewLabel_1.setBounds(36, 101, 137, 13);
    contentPane.add(lblNewLabel_1);

    JLabel lblNewLabel_2 = new JLabel("QUALIFICATION");
    lblNewLabel_2.setFont(new Font("Tahoma", Font.BOLD, 13));
    lblNewLabel_2.setBounds(41, 154, 122, 13);
    contentPane.add(lblNewLabel_2);

    JLabel lblNewLabel_3 = new JLabel("E-MAIL");
    lblNewLabel_3.setFont(new Font("Tahoma", Font.BOLD, 13));
    lblNewLabel_3.setBounds(39, 292, 45, -47);
    contentPane.add(lblNewLabel_3);

    JLabel lblNewLabel_4 = new JLabel("ROOM\r\n");
    lblNewLabel_4.setFont(new Font("Tahoma", Font.BOLD, 14));
    lblNewLabel_4.setBounds(63, 207, 72, 13);
    contentPane.add(lblNewLabel_4);

    textField_4 = new JTextField();
    textField_4.setBounds(216, 254, 278, 27);
    contentPane.add(textField_4);
    textField_4.setColumns(10);

```

```

JLabel lblNewLabel_5 = new JLabel("E-MAIL");
lblNewLabel_5.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_5.setBounds(63, 260, 58, 13);
contentPane.add(lblNewLabel_5);

Button button_1 = new Button("Update");
button_1.setBackground(new Color(0, 255, 204));
button_1.setBounds(312, 434, 164, 72);
contentPane.add(button_1);
button_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String name=textField.getText();
        String specialization=textField_1.getText();
        String qualification=textField_2.getText();
        String room=textField_3.getText();
        String email=textField_4.getText();
        String password=textField_5.getText();

        Connection con;
        PreparedStatement pstmt;
        try
        {
            Class.forName("com.mysql.cj.jdbc.Driver");

            con=DriverManager.getConnection("jdbc:mysql://localhost:3306/doctor","root","sayandeep2
023");

            pstmt=con.prepareStatement("update doc set
name='"+name+"',specialization='"+specialization+"',qualification='"+qualification+"',room='"+room
+'",email='"+email+"',password='"+password+"'");
            pstmt.executeUpdate();
            JOptionPane.showMessageDialog(contentPane,"Successfully
updated");

            //con.close();
        }catch(Exception e1) {}

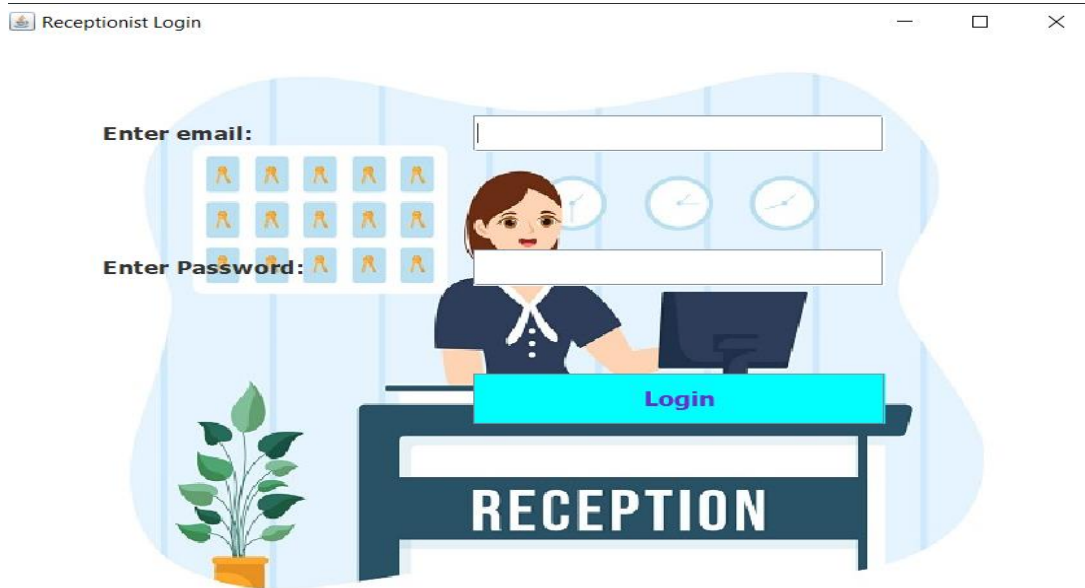
    }
});

textField_5 = new JTextField();
textField_5.setBounds(216, 308, 278, 27);
contentPane.add(textField_5);
textField_5.setColumns(10);

JLabel lblNewLabel_6 = new JLabel("Password");
lblNewLabel_6.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_6.setBounds(55, 314, 80, 13);
contentPane.add(lblNewLabel_6);
}
}

```


7. Receptionist Login



```
package org.Study;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.*;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import java.awt.Font;
import javax.swing.ImageIcon;
import java.awt.Color;

public class RecepTionist {
    static String Email,Pass;
    public static void main(String[] args) {
        JFrame frame=new JFrame("Receptionist Login");
        JLabel l1=new JLabel("Enter email:");
        l1.setFont(new Font("Tahoma", Font.BOLD, 15));
        l1.setBounds(65, 60, 121, 30);
        JTextField tf1=new JTextField();
        tf1.setBounds(294, 61, 254, 30);
        JLabel l2=new JLabel("Enter Password:");
        l2.setFont(new Font("Tahoma", Font.BOLD, 15));
        l2.setBounds(65, 169, 150, 30);
        JTextField tf2=new JTextField();
        tf2.setBounds(294, 170, 254, 30);
        JButton b1=new JButton("Login");
        b1.setForeground(new Color(102, 51, 204));
        b1.setFont(new Font("Tahoma", Font.BOLD, 16));
        b1.setBackground(Color.CYAN);
        b1.setBounds(294, 271, 254, 41);
        frame.getContentPane().setLayout(null);
    }
}
```

```

        frame.getContentPane().add(l1);
        frame.getContentPane().add(tf1);
        frame.getContentPane().add(l2);
        frame.getContentPane().add(tf2);
        frame.getContentPane().add(b1);

        JLabel lblNewLabel = new JLabel("New label");
        lblNewLabel.setIcon(new
ImageIcon(RecepTionist.class.getResource("/Images/REception.jpg")));
        lblNewLabel.setBounds(0, 0, 678, 472);
        frame.getContentPane().add(lblNewLabel);
        frame.setSize(692,509);
        frame.setVisible(true);
        b1.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent e) {
    Email=tf1.getText();
    Pass=tf2.getText();
    Connection con;
    PreparedStatement pstmt;
    ResultSet rs;
    boolean status=false;
    try
    {
        Class.forName("com.mysql.cj.jdbc.Driver");

        con=DriverManager.getConnection("jdbc:mysql://localhost:3306/addreceptionist","r
oot","sayandeep2023");
        pstmt=con.prepareStatement("select *from recep where email=? and
password=?");
        pstmt.setString(1, Email);
        pstmt.setString(2, Pass);
        rs=pstmt.executeQuery();
        status=rs.next();
        if(status)
        {
            Recep1.main(new String[]{});
            frame.dispose();
        }
        else {
            JOptionPane.showMessageDialog(frame,"Email or Password not Matched!");
        }
    }catch(Exception e1) {}
}});
    }
}

```

8. Add Receptionist

Enter details for adding new patient

Receptionist

View Patient

Update Patient

Delete Patient

Patient's ID

First Name

Age

Gender

Address

ADD

```
package org.Study;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import java.awt.Label;
import java.awt.Font;
import java.awt.Button;
import javax.swing.JList;
import java.awt.Panel;
import java.awt.TextField;
import javax.swing.JComboBox;
import javax.swing.JTextPane;
import java.awt.TextArea;
import java.awt.Color;

//For updating Patient details

public class Patient extends JFrame {

    private JPanel contentPane;

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Patient frame = new Patient();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
}
```

```

    });
}

public Patient() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 1135, 523);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(255, 128, 128));
    contentPane.setForeground(new Color(0, 0, 0));
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    Label label = new Label("Enter details to update patient info");
    label.setFont(new Font("Dialog", Font.BOLD, 21));
    label.setBounds(10, 0, 1111, 50);
    contentPane.add(label);

    Button button = new Button("Patient ID");
    button.setBounds(173, 93, 97, 21);
    contentPane.add(button);

    JList list = new JList();
    list.setBounds(192, 93, 1, 1);
    contentPane.add(list);

    Button button_1 = new Button("First Name");
    button_1.setBounds(173, 191, 97, 21);
    contentPane.add(button_1);

    Button button_2 = new Button("Last name");
    button_2.setBounds(173, 237, 97, 21);
    contentPane.add(button_2);

    Button button_3 = new Button("Age");
    button_3.setBounds(173, 282, 97, 26);
    contentPane.add(button_3);

    Button button_4 = new Button("Address");
    button_4.setBounds(173, 367, 97, 21);
    contentPane.add(button_4);

    Button button_5 = new Button("Gender");
    button_5.setBounds(173, 326, 97, 21);
    contentPane.add(button_5);

    TextField textField = new TextField();
    textField.setBounds(387, 191, 296, 32);
    contentPane.add(textField);

    TextField textField_1 = new TextField();
    textField_1.setBounds(387, 237, 296, 32);
    contentPane.add(textField_1);

    TextField textField_2 = new TextField();
    textField_2.setBounds(387, 282, 296, 32);
    contentPane.add(textField_2);

    TextField textField_4 = new TextField();
    textField_4.setBounds(309, 93, 165, 43);
    contentPane.add(textField_4);
}

```

```
TextField textField_5 = new TextField();
textField_5.setBounds(387, 367, 296, 32);
contentPane.add(textField_5);

JComboBox comboBox = new JComboBox();
comboBox.setBounds(387, 326, 296, 35);
contentPane.add(comboBox);

JTextPane textPane = new JTextPane();
textPane.setBounds(906, 193, -76, 30);
contentPane.add(textPane);

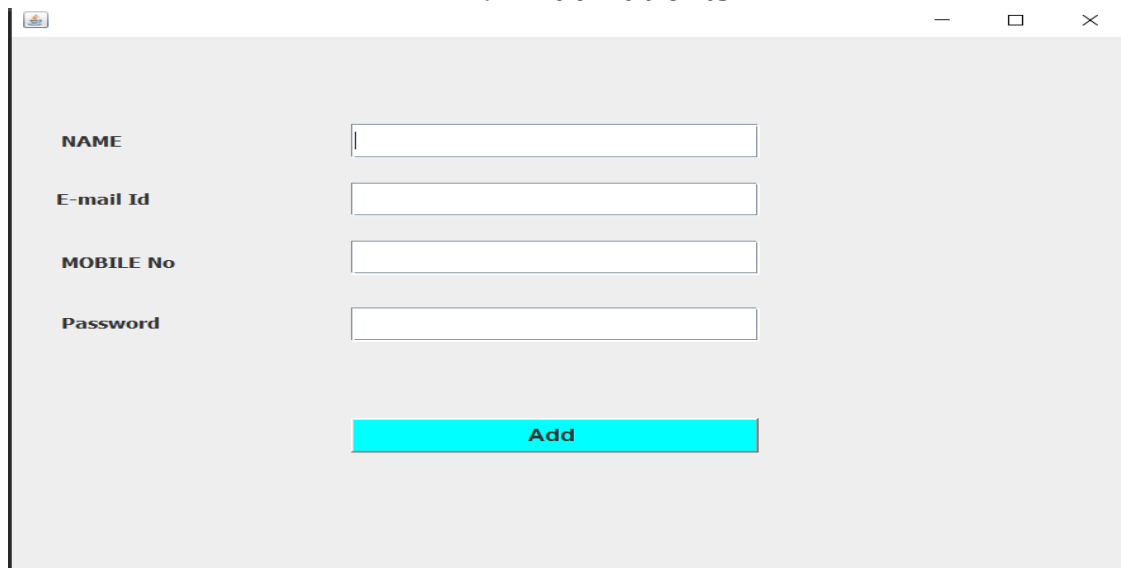
Button button_6 = new Button("Show Details");
button_6.setFont(new Font("Dialog", Font.BOLD, 12));
button_6.setBounds(108, 421, 132, 43);
contentPane.add(button_6);

Button button_7 = new Button("Update");
button_7.setBounds(747, 432, 107, 32);
contentPane.add(button_7);
```

```
}
```

```
}
```

9. Add Patients



The screenshot shows a Java Swing window titled "9. Add Patients". The window has a standard title bar with minimize, maximize, and close buttons. The main content area is light gray. It contains four labels on the left, each followed by a text input field on the right: "NAME", "E-mail Id", "MOBILE No", and "Password". At the bottom center, there is a cyan button with the text "Add".

```
package org.Study;

import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JTextField;
import javax.swing.JLabel;
import javax.swing.JOptionPane;

import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.awt.Button;
import java.awt.Color;

public class AddRecep extends JFrame {

    private JPanel contentPane;
    private JTextField textField;
    private JTextField textField_1;
    private JTextField textField_2;
    private JTextField textField_3;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    AddRecep frame = new AddRecep();
                    frame.setVisible(true);
                } catch (Exception e) {

```

```

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

/**
 * Create the frame.
 */
public AddRecep() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 720, 523);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

    setContentPane(contentPane);
    contentPane.setLayout(null);

    textField = new JTextField();
    textField.setBounds(215, 80, 258, 31);
    contentPane.add(textField);
    textField.setColumns(10);

    textField_1 = new JTextField();
    textField_1.setBounds(215, 133, 258, 31);
    contentPane.add(textField_1);
    textField_1.setColumns(10);

    textField_2 = new JTextField();
    textField_2.setBounds(215, 186, 258, 31);
    contentPane.add(textField_2);
    textField_2.setColumns(10);

    JLabel lblNewLabel = new JLabel("NAME");
    lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 13));
    lblNewLabel.setBounds(31, 80, 122, 31);
    contentPane.add(lblNewLabel);

    JLabel lblNewLabel_1 = new JLabel("E-mail Id");
    lblNewLabel_1.setFont(new Font("Tahoma", Font.BOLD, 13));
    lblNewLabel_1.setBounds(28, 133, 125, 31);
    contentPane.add(lblNewLabel_1);

    JLabel lblNewLabel_2 = new JLabel("MOBILE No");
    lblNewLabel_2.setFont(new Font("Tahoma", Font.BOLD, 13));
    lblNewLabel_2.setBounds(31, 195, 106, 22);
    contentPane.add(lblNewLabel_2);

    Button button = new Button("Add ");
    button.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            String Name=textField.getText();
            String Email=textField_1.getText();
            String MobileNo=textField_2.getText();
            String Password=textField_3.getText();
            //String Fee=textField_4.getText();

            Connection con;
            PreparedStatement pstmt;
            try
            {
                Class.forName("com.mysql.cj.jdbc.Driver");

```

```

con=DriverManager.getConnection("jdbc:mysql://localhost:3306/addreceptionist","root","sayandeep2023");
    pstmt=con.prepareStatement("insert into recep values(?,?,?,?)");
    pstmt.setString(1,Name);
    pstmt.setString(2,Email);
    pstmt.setString(3,MobileNo);
    pstmt.setString(4>Password);
    //pstmt.setString(5,Fee);
    pstmt.executeUpdate();
    JOptionPane.showMessageDialog(contentPane,"Successfully added");
    con.close();
}catch(Exception e1) {}
    }
});

button.setBackground(new Color(0, 255, 255));
button.setFont(new Font("Dialog", Font.BOLD, 15));
button.setBounds(215, 348, 258, 31);
contentPane.add(button);

textField_3 = new JTextField();
textField_3.setBounds(215, 247, 258, 31);
contentPane.add(textField_3);
textField_3.setColumns(10);

JLabel lblNewLabel_3 = new JLabel("Password");
lblNewLabel_3.setFont(new Font("Tahoma", Font.BOLD, 13));
lblNewLabel_3.setBounds(31, 250, 96, 22);
contentPane.add(lblNewLabel_3);
}
}

```


Testing

A) Unit testing —

Test individual unit of DC

Tested by - Developer

Methodology - Test individual unit of DC

B) Integration and System Testing –

Integration testing test two or more than two integrated models

System testing test functional and non-functional user requirement.

Tested by - Tester

Methodology - It is used for integrated model, functional and non-functional user requirement.

C) Performance Testing -

It measure performance / response under critical, load and stress condition.

Tested by - Tester

Methodology - Measure performance under critical, stress and load system.

D) Regression Testing -

Used to uncover new errors in existing functionality after changes have been made.

Tested by - Tester

Methodology - Test system response after functional enhancements in existing system.

FUTURE SCOPE

- The databases in the local machines can be replaced by a central web-based databases which will be used across all branches of the Doctor's List.
- More modules and functionality can be added to aid the management process.
- The security of the system can be greatly improved by using cryptography to store passwords and sensitive personal information
- A patient side client can also be introduced in later stages that might simplify tedious tasks for patients such as admission, appointments bill payments etc.

CONCLUSION

Doctor's Clinic is all about the modernizing a hospital through use of technology. Computers helps in taking over the manual system for quick and easy functioning. This Doctor's Clinic system is quite reliable and is proven on many stages. All the basic requirements of the hospital are provided in the system in order to manage it perfectly and large amount of data can also be stored. It gives many facilities like searching for the detail of patient, billing facilities as well as the creation of appointment receipts. Thus it's an important upgrade and a necessary step for hospital owners to elevate their business to the next level.

Bibliography

- **SWING DOCUMENTATION**
<https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html>
- **JDBC DOCUMENTATION**
<https://docs.oracle.com/javase/8/docs/technotes/guides/jdbc/>
- **ICONS**
<https://www.flaticon.com/>
- **MySQL DOCUMENTATION**
<https://dev.mysql.com/doc/>