BLOOD BANK MANAGEMENT SYSTEM

GROUP MEMBERS	ROLL NO
SREESANKAR S	AM.EN.U4CSE20366
DEVESH KUMAR V V	AM.EN.U4CSE20321
VIGNESH A NAIR	AM.EN.U4CSE20374

COMPUTER SCIENCE DEPARTMENT
AMRITA VISHWA VIDYAPEETHAM
DECEMBER 2021

CHAPTER 1

ABSTRACT

INTRODUCTION

This Project 'Blood Bank Management System' is an automated digital system that stores and retrieves the information related to blood donation. The project aims to provide an efficient and convenient way to manage Blood Bank System.

This system allows the donor or receiver to enter the system and choose his or her purpose and store the details corresponding to their need. After that he or she gets a confirmation message regarding their purpose and their slot booking details.

This system also allows the donor or receiver to register to the system and give the information such as blood group, dob, appointment, etc.

CONCEPT

The project allows the donor to store the details regarding their donation. After that the receiver who needs the blood will check the availability of blood with the help of the system. After the availability is checked, corresponding message is shown regarding the receiving of the blood.

INDEX

1. **CHAPTER 1**
   - **ABSTRACT……2**
     - **INTRODUCTION**
     - **CONCEPT**
2. **CHAPTER 2**
   - **DESCRIPTION……..……4**
     - **ROLES OF DONOR AND RECEIVER**
     - **WORKING**
3. **CHAPTER 3**
   - **SYSTEM DESIGN……5**
     - **USECASE DIAGRAM**
     - **CLASS DIAGRAM**
     - **CLASS DESCRIPTION**
     - **JAVA CONCEPTS**
     - **TOOLS**
4. **CHAPTER 4**
   - **RESULTS……10**
     - **GRAPHICAL USER INTERFACE(GUI)**
     - **INPUTS AND OUTPUTS**
5. **CHAPTER 5**
   - **DATABASE DETAILS……...24**
     - **SCHEMA**
6. **CHAPTER 6**
   - **CONCLUSION……25**
     - **REFERENCES**
7. **CHAPTER 7**
   - **CODE……26**

CHAPTER 2

DESCRIPTION

ROLES OF DONOR AND RECEIVER

The Donor has access to following functions:

- He / She can register in to the system.
- He / She can log in to the system.
- He / She can log out from his current account.
- He / She can add Donation Details.
- He / She can view the slot confirmation message.

The Receiver has access to following functions:

- He / She can register in to the system.
- He / She can log in to the system.
- He / She can log out from his current account.
- He / She can add Receiver Details
- He / She can view the slot confirmation or disapproval message.

WORKING

1. The User can login to the system.
2. If he or she is not registered then they can register first and then login.
3. The registered details will be stored in the database.
4. Then the User is taken to the purpose window where he/she can choose their purpose whether to donate or to receive blood.
5. According to their Input they are redirected to the corresponding page whether to enter donation details or receiving details.
6. In donation details the donor has option to enter the details regarding the donation. After that a confirmation message is shown.
7. For receiver, the details given is checked and confirmed whether it is in the database, and corresponding message is shown.
8. If the receiving is confirmed, a confirmation window is shown and the slot details for receiving is displayed.
9. At last, their corresponding details are stored in the database and they can logout from the system.

Note:

- **He / She is first entering as a user and according to their need they will be changing to receiver or donor.**

CHAPTER 3

SYSTEM DESIGN

USECASE DIAGRAM

CLASS DIAGRAM

CLASS DESCRIPTION

Registration_1
This class is used for registration window where the user can enter the registration details and will be redirected to login page after successful submission.

LoginPage
This class is used for login window where the login credentials are entered and it is validated.

Purpose
In this class the purpose window is shown where the user can choose his/her purpose.

Donation_1
In this class the donation window is shown and is used to enter donation details.

Receiving_1
In this class the receiving window is shown and is used to enter receiving details.

ConfMsgDon
In this class the confirmation window is shown regarding the donation.

ConfMsgRec
In this class the details entered in receiving details are checked and corresponding message is shown regarding receiving.

JAVA CONCEPTS

Class
A class — in the context of Java — is a template used to create objects and to define object data types and methods.

Object
A Java object is a member (also called an instance) of a Java class. Each object has an identity, a behaviour and a state.

Encapsulation
It is achieved by default in Java, as the various classes wrap the respective data members and methods (attributes and behaviors) into a single unit

Custom Exceptions
Custom Exceptions have been created for the circumstances when there is no Data to be displayed.

Final keyword
Certain choices to be added in the drop-down menu are constant and should not be changed. So, we have labelled them as final to ensure they act as constant variables.

TOOLS

FRONT END

- JFrame

JFrame is used for a top-level container that provides a window on the screen

- JPanel

JPanel, a part of the Java Swing package, is a container that can store a group of components.

- JTextField

JTextField is used for the editing of a single line of text.

- JPasswordField

JPasswordField is used for attaining text in form of password

- JLabel

JLabel is used to display a short string or an image icon

- JButton

Used for adding buttons

- Image

Used for inserting image

- JDateChooser

JDateChooser containing a date editor and a button, that makes a JCalendar visible for choosing a date

- JComboBox

JComboBox is used as for showing drop down menu

BACK END

- ActionListener

ActionListener in Java is a class that is responsible for handling all action events such as when the user clicks on a component

- Connection

A Connection object controls the connection to the database.

- try- catch

The try statement allows you to define a block of code to be tested for errors while it is being executed. The catch statement allows you to define a block of code to be executed, if an error occurs in the try block.

- JOptionPane

JOptionPane makes it easy to pop up a standard dialog box that prompts users for a value or informs them of something.

- PreparedStatement

PreparedStatement allows us to execute dynamic queries with parameter inputs.

- ResultSet

A ResultSet is an object that contains the results of executing an SQL query.

- While loop

A "While" Loop is used to repeat a specific block of code an unknown number of times, until a condition is met.

- String

A Java string is a sequence of characters that exist as an object of the class java.

- SimpleDateFormat

SimpleDateFormat is a concrete class for formatting and parsing dates in a locale-sensitive manner

CHAPTER 4

RESULTS

GRAPHICAL USER INTERFACE (GUI)

LOGIN PAGE

REGISTRATION PAGE

PURPOSE PAGE

DONATION PAGE

RECEIVING PAGE

CONFIRMATION REGARDING DONATION PAGE

CONFIRMATION REGARDING RECEIVING PAGE

INPUTS AND OUTPUTS

- **Creating a new user**

Log-In

Login
Not a User? Register Here!
User ID :
Sree_10
Password :
Login
One Blood Donation can save upto Three Lives
After donating blood, the body works to replenish the blood loss. This stimulates the production of new blood cells and in turn, helps in maintaining good health.

Message
You have successfully logged in
OK

- **Choosing Purpose**

- **Entering Donation Details**

- **Slot Booking Confirmation**

Now we logout of the system and a new user in entered and checking for blood

- **Entering Receiving details**

- **Slot Booking Confirmation**

CHAPTER 5

DATABASE DETAILS

SCHEMA

Register (fName, lName, Mobile_no, User_id, Dob, Gender, Weight, Blood_grp, Pass_word, Conf_password, Address, Phy_dis)

fname	lname	mobile_no	user_id	dob	gender	weight	blood_grp	pass_word	conf_password	address	phy_dis
character varying (20)	character varying (20)	character varying (10)	[PK] character varying (25)	character varying (25	character varying (10)	integer	character varying (10)	character varying (30)	character varying (30)	character varying (1	character varying (5)

Login (User_id,)

user_id
character varying (25)

Donation_details (blood_grp, amt_blood, Branch, Date_don, Time_slot, User_id)

blood_grp	amt_blood	branch	date_don	time_slot	user_id
character varying (10)	integer	character varying (20)	character varying (20)	character varying (20)	character varying (25)

Receiving_details (blood_grp, amt_blood, Branch, Date_rec, Time_slot, User_id)

blood_grp	amt_blood	branch	date_rec	time_slot	user_id
character varying (10)	integer	character varying (20)	character varying (20)	character varying (20)	character varying (25)

CHAPTER 6

CONCLUSION

The project "Blood Bank Management System" aims to act as an important role in saving life of human beings and reduce the panic created in emergency situations. The project aims to provide an efficient and convenient way to manage Blood Bank System.

This system allows the donor or receiver to enter the system and choose his or her purpose and store the details corresponding to their need in an efficient way, Hence helping the people who is in need of blood.

"*Give blood, Save life*"

REFERENCES

- https://www.javatpoint.com/java-swing
- https://jdbc.postgresql.org/documentation/head/connect.html
- https://www.youtube.com/watch?v=OLmAZmBSwMo
- https://www.educba.com/swing-components-in-java/
- https://www.codejava.net/java-se/swing/how-to-use-jdatepicker-to-display-calendar-component

CHAPTER 7

CODE

Registration

```
package Default;
import javax.swing.*;



import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;

import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.text.SimpleDateFormat;
import java.awt.event.ActionEvent;
import com.toedter.calendar.JDateChooser;
import javax.swing.JComboBox;

import javax.swing.JOptionPane;


public class Registration_1 {

        private JFrame frame;
        private JTextField textField_2;
        private JTextField textField_3;
        private JTextField textField_4;
        private JPasswordField passwordField;
        private JPasswordField passwordField_1;
        private JTextField textField_5;
        private JTextField textField;
        private JTextField textField_1;
        private JTextField textField_6;
        private JTextField textField_7;
        private JTextField textField_8;
        private JPasswordField passwordField_2;
        private JPasswordField passwordField_3;
        private JTextField textField_9;



        /**
         * Launch the application.
         */
        public static void main(String[] args) {
                EventQueue.invokeLater(new Runnable() {
                        public void run() {

                                try {
                                        Registration_1 window = new Registration_1();
                                        window.frame.setVisible(true);
                                } catch (Exception e) {
                                        e.printStackTrace();
                                }
                        }
                });
```

```java
}

/**
 * Create the application.
 */
public Registration_1() {
        initialize();

}

/**
 * Initialize the contents of the frame.
 */
private void initialize() {
        frame = new JFrame();
        frame.setVisible(true);
        frame.setResizable(true);
        frame.setBounds(400, 100, 700, 899);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
        frame.setSize(899, 899);

        JPanel panel = new JPanel();
        panel.setBackground(new Color(204, 51, 51));
        panel.setBounds(0, 0, 899, 71);
        frame.getContentPane().add(panel);



        JLabel lblNewLabel = new JLabel("Registration");
        panel.add(lblNewLabel);
        lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 40));

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(new Color(153, 204, 204));
        panel_1.setBounds(0, 72, 899, 812);
        frame.getContentPane().add(panel_1);

        JLabel lblNewLabel_1 = new JLabel("First Name :");
        lblNewLabel_1.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_1.setBounds(173, 83, 99, 16);
        panel_1.add(lblNewLabel_1);

        JLabel lblNewLabel_2 = new JLabel("Last Name :");
        lblNewLabel_2.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_2.setBounds(173, 124, 99, 16);
        panel_1.add(lblNewLabel_2);

        JLabel lblNewLabel_i = new JLabel("");
        lblNewLabel_i.setBounds(6, 6, 60, 37);
        panel_1.add(lblNewLabel_i);

        JLabel lblNewLabel_3 = new JLabel("Mobile No :");
        lblNewLabel_3.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_3.setBounds(173, 167, 99, 16);
        panel_1.add(lblNewLabel_3);

        JLabel lblNewLabel_4 = new JLabel("     User ID :");
        lblNewLabel_4.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_4.setBounds(173, 209, 99, 16);
        panel_1.add(lblNewLabel_4);

        JLabel lblNewLabel_5 = new JLabel("DOB :");
        lblNewLabel_5.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_5.setBounds(221, 249, 51, 16);
        panel_1.add(lblNewLabel_5);

        JLabel lblNewLabel_6 = new JLabel("Gender :");
        lblNewLabel_6.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_6.setBounds(197, 289, 72, 16);
        panel_1.add(lblNewLabel_6);

        JLabel lblNewLabel_7 = new JLabel("Weight :");
        lblNewLabel_7.setFont(new Font("Lucida Grande", Font.BOLD, 15));
        lblNewLabel_7.setBounds(200, 329, 72, 16);
        panel_1.add(lblNewLabel_7);

        JLabel lblNewLabel_8 = new JLabel("Blood Group :");
```

```
lblNewLabel_8.setFont(new Font("Lucida Grande", Font.BOLD, 15));
lblNewLabel_8.setBounds(159, 372, 113, 16);
panel_1.add(lblNewLabel_8);

JLabel lblNewLabel_9 = new JLabel("Password :");
lblNewLabel_9.setFont(new Font("Lucida Grande", Font.BOLD, 15));
lblNewLabel_9.setBounds(184, 419, 89, 16);
panel_1.add(lblNewLabel_9);

JLabel lblNewLabel_10 = new JLabel("Confirm Password :");
lblNewLabel_10.setFont(new Font("Lucida Grande", Font.BOLD, 15));
lblNewLabel_10.setBounds(115, 457, 152, 16);
panel_1.add(lblNewLabel_10);

JLabel lblNewLabel_11 = new JLabel("Address :");
lblNewLabel_11.setFont(new Font("Lucida Grande", Font.BOLD, 15));
lblNewLabel_11.setBounds(197, 504, 113, 16);
panel_1.add(lblNewLabel_11);

JLabel lblNewLabel_12 = new JLabel("Physically Disabled :");
lblNewLabel_12.setFont(new Font("Lucida Grande", Font.BOLD, 15));
lblNewLabel_12.setBounds(106, 605, 166, 16);
panel_1.add(lblNewLabel_12);

textField = new JTextField();
textField.setColumns(10);
textField.setBounds(281, 71, 186, 34);
panel_1.add(textField);

textField_1 = new JTextField();
textField_1.setColumns(10);
textField_1.setBounds(281, 116, 186, 34);
panel_1.add(textField_1);

textField_2 = new JTextField();
textField_2.setColumns(10);
textField_2.setBounds(281, 159, 186, 34);
panel_1.add(textField_2);

textField_7 = new JTextField();
textField_7.setColumns(10);
textField_7.setBounds(281, 201, 186, 34);
panel_1.add(textField_7);

JDateChooser dateChooser = new JDateChooser();
dateChooser.setBounds(284, 241, 180, 34);
panel_1.add(dateChooser);

JComboBox comboBox = new JComboBox();
comboBox.addItem("Male");
comboBox.addItem("Female");
comboBox.addItem("Other");
comboBox.setSelectedItem(null);
comboBox.setBounds(283, 279, 184, 34);
panel_1.add(comboBox);

textField_8 = new JTextField();
textField_8.setColumns(10);
textField_8.setBounds(281, 325, 186, 34);
panel_1.add(textField_8);

passwordField_2 = new JPasswordField();
passwordField_2.setBounds(285, 407, 182, 34);
panel_1.add(passwordField_2);

passwordField_3 = new JPasswordField();
passwordField_3.setBounds(285, 453, 182, 34);
panel_1.add(passwordField_3);

textField_9 = new JTextField();
textField_9.setColumns(10);
textField_9.setBounds(281, 504, 186, 84);
panel_1.add(textField_9);

final JComboBox comboBox_1 = new JComboBox();
comboBox_1.addItem("A+");
comboBox_1.addItem("A-");
comboBox_1.addItem("B+");
comboBox_1.addItem("B-");
```

```
                comboBox_1.addItem("O+");
                comboBox_1.addItem("O-");
                comboBox_1.addItem("AB+");
                comboBox_1.addItem("AB-");

                comboBox_1.setSelectedItem(null);
                comboBox_1.setBounds(284, 369, 186, 27);
                panel_1.add(comboBox_1);

                JComboBox comboBox_2 = new JComboBox();
                comboBox_2.addItem("No");
                comboBox_2.addItem("Yes");
                comboBox_2.setBounds(284, 602, 183, 27);
                panel_1.add(comboBox_2);

                JButton btnNewButton = new JButton("Submit");


                btnNewButton.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e) {

                if(textField.getText().isEmpty() ) {
                                        JOptionPane.showMessageDialog(null, "First Name is Empty");

                                }
                                else if(textField_1.getText().isEmpty() ) {
                                        JOptionPane.showMessageDialog(null, "Last Name is Empty");


                                }

                                else if(textField_2.getText().isEmpty() ) {
                                        JOptionPane.showMessageDialog(null, "Moblie No is Empty");


                                }


                                else if(textField_7.getText().isEmpty()) {
                                        JOptionPane.showMessageDialog(null, "User ID is Empty");

                                }
                                else if(textField_8.getText().isEmpty()) {
                                        JOptionPane.showMessageDialog(null, "Weight is Empty");

                                }
                                else if(passwordField_2.getText().isEmpty() ) {
                                        JOptionPane.showMessageDialog(null, "Password is Empty");

                                }
                                else if(passwordField_3.getText().isEmpty()) {
                                        JOptionPane.showMessageDialog(null, "Password is Empty");

                                }

                                else if (!( new String (passwordField_2.getPassword() ).equals( new
String( passwordField_3.getPassword () ) ) ) ) {
                                        JOptionPane.showMessageDialog(null, "Your passwords do not
match!");
                                }

                                else if(textField_9.getText().isEmpty()) {
                                        JOptionPane.showMessageDialog(null, "Address is Empty");

                                }
                                else {
                                        JOptionPane.showMessageDialog(null, "you have successfully
registered");
                                        new LoginPage();
                                        frame.dispose();


                                }

                String fname = textField.getText();
                String lname = textField_1.getText();
                String mobno = textField_2.getText();
                String userid = textField_7.getText();
```

```
                SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
                String dob = sdf.format(dateChooser.getDate());
                String gender = comboBox.getSelectedItem().toString();
                int weight = Integer.parseInt(textField_8.getText());;
                String blood_grp = comboBox_1.getSelectedItem().toString();
                String pass = passwordField_2.getText();
                String con_pass = passwordField_3.getText();
                String address = textField_9.getText();
                String phy_dis = comboBox_2.getSelectedItem().toString();

                try {
                    Connection connection =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank", "postgres", "2580");

                    String query = "INSERT INTO register values('" + fname + "' ,'" + lname + "' ,
'" + mobno + "' , '" + userid + "' ,'" + dob + "' , '" + gender + "' , '" + weight + "' , '" +
blood_grp + "' , '" + pass + "' , '" + con_pass + "' , '" + address + "' , '" + phy_dis + "')";


                    PreparedStatement st= connection.prepareStatement("select
fName,lName,Mobile_no,User_id,Dob,Gender,Weight,Blood_grp,Pass_word,Conf_password,Address,Phy_dis
from register where fName=? and lName=? and Mobile_no=? and User_id=? and Dob=? and Gender=? and
Weight=? and Blood_grp=? and Pass_word=? and Conf_password=? and Address=? and Phy_dis=? ");


                    st.setString(1,fname);

                    st.setString(2,lname);
                    st.setString(3,mobno);
                    st.setString(4,userid);
                    st.setString(5,dob);
                    st.setString(6,gender);
                    st.setInt(7,weight);
                    st.setString(8,blood_grp);
                    st.setString(9,pass);
                    st.setString(10,con_pass);
                    st.setString(11,address);
                    st.setString(12,phy_dis);

                    ResultSet rs= st.executeQuery();
                    if (rs.next())
                    {
                        JOptionPane.showMessageDialog(btnNewButton, "already exist");
                    }
                    else
                    {
                        Statement sta = connection.createStatement();
                        int x = sta.executeUpdate(query);


                    }
                    connection.close();
                } catch (Exception exception) {
                    exception.printStackTrace();
                }
            }
        });
                btnNewButton.setFont(new Font("Lucida Grande", Font.BOLD, 14));
                btnNewButton.setBounds(319, 666, 117, 29);

                panel_1.add(btnNewButton);

                JLabel lblNewLabel_13 = new JLabel("");
                Image img2 = new ImageIcon(this.getClass().getResource("/RegImage.jpg")).getImage();
                lblNewLabel_13.setIcon(new ImageIcon(img2));

                lblNewLabel_13.setBounds(504, 224, 389, 264);
                panel_1.add(lblNewLabel_13);

                JLabel lblNewLabel_14 = new JLabel("");
                Image img4 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
                lblNewLabel_i.setIcon(new ImageIcon(img4));
                lblNewLabel_14.setBounds(6, 6, 60, 37);
                panel_1.add(lblNewLabel_14);



        }
}
```

LoginPage

```java
package Default;
import javax.swing.*;



import java.sql.*;
import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;

import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class LoginPage {

	private JFrame frame;
	private JTextField textField;
	private JPasswordField passwordField;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {

		String jdbcURL = "jdbc:postgresql://localhost:5432/BloodBank";
		String user = "postgres";
		String pass = "2580";
		try {
			Connection connnection = DriverManager.getConnection(jdbcURL, user, pass);
			System.out.println("Connected");
			connnection.close();
		} catch (SQLException e1) {
			// TODO Auto-generated catch block
			e1.printStackTrace();
		}
		EventQueue.invokeLater(new Runnable() {
			public void run() {

				try {
					LoginPage window = new LoginPage();
					window.frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the application.
	 */
	public LoginPage() {
		initialize();

	}

	/**
	 * Initialize the contents of the frame.
	 */
	private void initialize() {
		frame = new JFrame();
		frame.setVisible(true);
		frame.setResizable(true);
		frame.setBounds(270, 100, 450, 300);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
                frame.getContentPane().setLayout(null);
                frame.setSize(899, 899);

                JPanel panel = new JPanel();
                panel.setBackground(new Color(204, 51, 51));
                panel.setBounds(0, 0, 899, 71);
                frame.getContentPane().add(panel);



                JLabel lblNewLabel = new JLabel("Login");
                panel.add(lblNewLabel);
                lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 40));

                JPanel panel_1 = new JPanel();
                panel_1.setForeground(new Color(0, 0, 0));
                panel_1.setBackground(new Color(153, 204, 204));
                panel_1.setBounds(0, 72, 899, 867);
                panel_1.setSize(899, 899);
                frame.getContentPane().add(panel_1);
                panel_1.setLayout(null);

                JLabel lblNewLabel_1 = new JLabel("User ID     :");
                lblNewLabel_1.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 35));
                lblNewLabel_1.setBounds(195, 142, 167, 45);
                panel_1.add(lblNewLabel_1);

                JLabel lblNewLabel_2 = new JLabel("Password :");
                lblNewLabel_2.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 35));
                lblNewLabel_2.setBounds(195, 210, 167, 51);
                panel_1.add(lblNewLabel_2);

                textField = new JTextField();
                textField.setBackground(new Color(153, 153, 153));
                textField.setBounds(391, 148, 222, 37);
                panel_1.add(textField);
                textField.setColumns(10);

                passwordField = new JPasswordField();
                passwordField.setBackground(new Color(153, 153, 153));
                passwordField.setBounds(391, 217, 222, 40);
                panel_1.add(passwordField);

                JButton btnNewButton = new JButton("Login");
                Image img = new ImageIcon(this.getClass().getResource("/Ok.png")).getImage();
                btnNewButton.setIcon(new ImageIcon(img));
                btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 15));
                btnNewButton.setBackground(new Color(51, 51, 204));
                btnNewButton.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                                String userName = textField.getText();
                String password = passwordField.getText();
                try {
                    Connection connection = (Connection)
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank",
                        "postgres", "2580");

                    String query = "INSERT INTO login values('" + userName + "')";

                    PreparedStatement st = (PreparedStatement) connection
                        .prepareStatement("Select User_id, Pass_word from register where User_id=?
and Pass_word=?");
                    PreparedStatement st1 = (PreparedStatement) connection
                            .prepareStatement("Select User_id from login where User_id=? ");

                    st.setString(1, userName);
                    st.setString(2, password);

                    st1.setString(1, userName);

                    ResultSet rs = st.executeQuery();

                    if (rs.next()) {

                        JOptionPane.showMessageDialog(btnNewButton, "You have successfully logged
in");
                        Statement sta = connection.createStatement();
                        int x = sta.executeUpdate(query);
                        new Purpose();
```

```
                                frame.dispose();
                        } else {
                            JOptionPane.showMessageDialog(btnNewButton, "Wrong Username & Password");
                        }
                    } catch (SQLException sqlException) {
                        sqlException.printStackTrace();
                    }


                    }
                });
                btnNewButton.setBounds(440, 285, 133, 45);
                panel_1.add(btnNewButton);

                JButton btnNewButton_1 = new JButton("Not a User? Register Here!");
                btnNewButton_1.setBackground(new Color(204, 51, 102));
                btnNewButton_1.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                                new Registration_1();
                frame.dispose();

                        }
                });

                btnNewButton_1.setBounds(703, 6, 190, 29);
                panel_1.add(btnNewButton_1);

                JLabel lblNewLabel_3 = new JLabel("");
                Image img1 = new ImageIcon(this.getClass().getResource("/LDescp.jpg")).getImage();
                lblNewLabel_3.setIcon(new ImageIcon(img1));
                lblNewLabel_3.setBounds(197, 362, 500, 274);
                panel_1.add(lblNewLabel_3);

                JLabel lblNewLabel_4 = new JLabel("");
                Image img2 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
                lblNewLabel_4.setIcon(new ImageIcon(img2));
                lblNewLabel_4.setBounds(6, 6, 60, 37);
                panel_1.add(lblNewLabel_4);

                        }
}
```

Purpose

```
package Default;
import javax.swing.*;



import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;

import javax.swing.JTextField;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.awt.event.ActionEvent;


public class Purpose {

        private JFrame frame;

        /**
```

```
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {

				try {
					Purpose window = new Purpose();
					window.frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the application.
	 */
	public Purpose() {
		initialize();

	}

	/**
	 * Initialize the contents of the frame.
	 */
	private void initialize() {
		frame = new JFrame();
		frame.setVisible(true);
		frame.setResizable(true);
		frame.setBounds(270, 100, 450, 300);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);
		frame.setSize(899, 899);

		JPanel panel = new JPanel();
		panel.setBackground(new Color(204, 51, 51));
		panel.setBounds(0, 0, 899, 71);
		frame.getContentPane().add(panel);


		JLabel lblNewLabel = new JLabel("Purpose");
		panel.add(lblNewLabel);
		lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 40));

		JPanel panel_1 = new JPanel();
		panel_1.setForeground(new Color(0, 0, 0));
		panel_1.setBackground(new Color(153, 204, 204));
		panel_1.setBounds(0, 72, 899, 867);
		panel_1.setSize(899, 899);
		frame.getContentPane().add(panel_1);
		panel_1.setLayout(null);


		JButton btnNewButton_1 = new JButton("Logout");
		btnNewButton_1.setBackground(new Color(204, 51, 102));
		btnNewButton_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		btnNewButton_1.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				try {
	    Connection connection = (Connection)
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank",
			"postgres", "2580");



	    PreparedStatement st = (PreparedStatement) connection
		.prepareStatement("delete from login;");
		int x = st.executeUpdate();
		new LoginPage();
		frame.dispose();
				}


				catch (Exception exception) {
	    exception.printStackTrace();
		}
			}
```

```
                });


                btnNewButton_1.setBounds(703, 6, 190, 29);
                panel_1.add(btnNewButton_1);

                JLabel lblNewLabel_4 = new JLabel("");
                Image img2 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
                lblNewLabel_4.setIcon(new ImageIcon(img2));
                lblNewLabel_4.setBounds(6, 6, 60, 37);
                panel_1.add(lblNewLabel_4);

                JButton btnNewButton = new JButton("Donate");
                btnNewButton.setForeground(new Color(0, 204, 102));
                btnNewButton.setFont(new Font("Lucida Grande", Font.BOLD, 20));
                btnNewButton.setBounds(363, 173, 200, 85);
                btnNewButton.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                                new Donation_1();
                frame.dispose();
                        }
                });

                panel_1.add(btnNewButton);

                JLabel lblNewLabel_1 = new JLabel("");
                Image img3 = new ImageIcon(this.getClass().getResource("/Purpose.jpg")).getImage();
                lblNewLabel_1.setIcon(new ImageIcon(img3));
                lblNewLabel_1.setBounds(285, 463, 402, 130);
                panel_1.add(lblNewLabel_1);

                JButton btnNewButton_2 = new JButton("Receive");
                btnNewButton_2.setFont(new Font("Lucida Grande", Font.BOLD, 20));
                btnNewButton_2.setForeground(new Color(204, 0, 51));
                btnNewButton_2.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                                new Receiving_1();
                                frame.dispose();
                        }
                });
                btnNewButton_2.setBounds(363, 303, 200, 85);
                panel_1.add(btnNewButton_2);
        }
}
```

Donation_1

```
package Default;
import javax.swing.*;



import java.awt.EventQueue;
import java.awt.Color;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.text.SimpleDateFormat;
import java.awt.event.ActionEvent;
import com.toedter.calendar.JDateChooser;

public class Donation_1 {

        private JFrame frame;
        private JTextField textField;
        private JTextField textField_1;
```

```java
	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {

				try {
					Donation_1 window = new Donation_1();
					window.frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the application.
	 */
	public Donation_1() {
		initialize();

	}

	/**
	 * Initialize the contents of the frame.
	 */
	private void initialize() {
		frame = new JFrame();
		frame.setVisible(true);
		frame.setResizable(true);
		frame.setBounds(270, 100, 899, 899);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);
		frame.setSize(899, 899);

		JPanel panel = new JPanel();
		panel.setBackground(new Color(204, 51, 51));
		panel.setBounds(0, 0, 899, 71);
		frame.getContentPane().add(panel);

		JList list = new JList();
		panel.add(list);

		JLabel lblNewLabel = new JLabel("Donation");
		panel.add(lblNewLabel);
		lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 40));

		JPanel panel_1 = new JPanel();
		panel_1.setForeground(new Color(0, 0, 0));
		panel_1.setBackground(new Color(153, 204, 204));
		panel_1.setBounds(0, 72, 899, 867);
		panel_1.setSize(899, 899);
		frame.getContentPane().add(panel_1);
		panel_1.setLayout(null);


		JButton btnNewButton_1 = new JButton("Logout");
		btnNewButton_1.setBackground(new Color(204, 51, 102));
		btnNewButton_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		btnNewButton_1.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				try {
	    Connection connection = (Connection)
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank",
	    "postgres", "2580");



	    PreparedStatement st = (PreparedStatement) connection
	        .prepareStatement("delete from login;");
	        int x = st.executeUpdate();
	        new LoginPage();
	        frame.dispose();

				}
```

```
                catch (Exception exception) {
    exception.printStackTrace();
}


        }
});
btnNewButton_1.setBounds(703, 6, 190, 29);
panel_1.add(btnNewButton_1);


JLabel lblNewLabel_4 = new JLabel("");
Image img2 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
lblNewLabel_4.setIcon(new ImageIcon(img2));
lblNewLabel_4.setBounds(6, 6, 60, 37);
panel_1.add(lblNewLabel_4);

JLabel lblNewLabel_5 = new JLabel("Blood Group :");
lblNewLabel_5.setFont(new Font("Lucida Grande", Font.BOLD, 13));
lblNewLabel_5.setBounds(317, 172, 108, 16);
panel_1.add(lblNewLabel_5);

JComboBox comboBox_1 = new JComboBox();
comboBox_1.addItem("A+");
comboBox_1.addItem("A-");
comboBox_1.addItem("B+");
comboBox_1.addItem("B-");
comboBox_1.addItem("O+");
comboBox_1.addItem("O-");
comboBox_1.addItem("AB+");
comboBox_1.addItem("AB-");

comboBox_1.setSelectedItem(null);

comboBox_1.setBounds(432, 168, 165, 27);
panel_1.add(comboBox_1);

JLabel lblNewLabel_1_1 = new JLabel("Amount of Blood Donating(in ml) :");
lblNewLabel_1_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
lblNewLabel_1_1.setBounds(178, 196, 246, 35);
panel_1.add(lblNewLabel_1_1);

JLabel lblNewLabel_2 = new JLabel("Branch :");
lblNewLabel_2.setFont(new Font("Lucida Grande", Font.BOLD, 13));
lblNewLabel_2.setBounds(356, 250, 68, 16);
panel_1.add(lblNewLabel_2);

JLabel lblNewLabel_3 = new JLabel(" Date :");
lblNewLabel_3.setFont(new Font("Lucida Grande", Font.BOLD, 13));
lblNewLabel_3.setBounds(366, 299, 68, 16);
panel_1.add(lblNewLabel_3);

JLabel lblNewLabel_4_1 = new JLabel("Time slot :");
lblNewLabel_4_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
lblNewLabel_4_1.setBounds(341, 344, 73, 16);
panel_1.add(lblNewLabel_4_1);

textField = new JTextField();
textField.setColumns(10);
textField.setBounds(432, 201, 165, 30);
panel_1.add(textField);

textField_1 = new JTextField();
textField_1.setColumns(10);
textField_1.setBounds(432, 243, 165, 30);
panel_1.add(textField_1);

JDateChooser dateChooser = new JDateChooser();
dateChooser.setBounds(432, 289, 165, 26);
panel_1.add(dateChooser);

JComboBox comboBox = new JComboBox();
comboBox.addItem("9:00 - 10:00");
comboBox.addItem("10:00 - 11:00");
comboBox.addItem("11:00 - 12:00");
comboBox.addItem("3:00 - 4:00");
comboBox.setSelectedItem(null);
comboBox.setBounds(426, 340, 171, 27);
panel_1.add(comboBox);
```

```
JButton btnNewButton = new JButton("Submit");
btnNewButton.setFont(new Font("Lucida Grande", Font.BOLD, 13));
btnNewButton.setBounds(449, 369, 117, 29);
btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                if(textField.getText().isEmpty() ) {
                        JOptionPane.showMessageDialog(null, "Amount of Blood
Donating is Empty");

                }

                else if(textField_1.getText().isEmpty() ) {
                        JOptionPane.showMessageDialog(null, "Branch is Empty");

                }
                else {
                new ConfMsgDon();
frame.dispose();
                }

                int amt =Integer.parseInt(textField.getText());;
String branch = textField_1.getText();
SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
String date = sdf.format(dateChooser.getDate());
String time = comboBox.getSelectedItem().toString();
String blood_grp = comboBox_1.getSelectedItem().toString();
String userid = null;

try {

    Connection connection =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank", "postgres", "2580");

     PreparedStatement st1= connection.prepareStatement("select user_id from login
where user_id like '_%' ");


    ResultSet rs1= st1.executeQuery();
    while (rs1.next())
    {
        userid = rs1.getString("user_id");
    }


    String query = "INSERT INTO Donation_details values('" + blood_grp + "','" +
amt + "' ,'" + branch + "' , '" + date + "' , '" + time + "' ,'" + userid + "')";


    PreparedStatement st= connection.prepareStatement("select
blood_grp,Amt_blood,Branch,Date_Don,Time_slot,User_id from Donation_details where blood_grp=? and
Amt_blood=? and Branch=? and Date_Don=? and Time_slot=? and User_id=? ");

    st.setString(1,blood_grp);
    st.setInt(2,amt);

    st.setString(3,branch);
    st.setString(4,date);
    st.setString(5,time);
    st.setString(6,userid);

    ResultSet rs= st.executeQuery();
    if (rs.next())
    {
        JOptionPane.showMessageDialog(btnNewButton, "already exist");
    }
    else
    {
        Statement sta = connection.createStatement();
        int x = sta.executeUpdate(query);


    }
    connection.close();
} catch (Exception exception) {
    exception.printStackTrace();
}

        }
```

```
                });
                panel_1.add(btnNewButton);

                JLabel lblNewLabel_1 = new JLabel("");
                Image img3 = new ImageIcon(this.getClass().getResource("/Purpose.jpg")).getImage();
                lblNewLabel_1.setIcon(new ImageIcon(img3));

                lblNewLabel_1.setBounds(306, 463, 402, 130);
                panel_1.add(lblNewLabel_1);


        }
}
```

Receiving_1

```
package Default;
import javax.swing.*;


import java.awt.EventQueue;
import java.awt.Color;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.text.SimpleDateFormat;
import java.awt.event.ActionEvent;
import com.toedter.calendar.JDateChooser;

public class Receiving_1 {

        private JFrame frame;
        private JTextField textField;
        private JTextField textField_1;

        /**
         * Launch the application.
         */
        public static void main(String[] args) {

                String jdbcURL = "jdbc:postgresql://localhost:5432/BloodBank";
                String user = "postgres";
                String pass = "2580";
                try {
                        Connection connnection = DriverManager.getConnection(jdbcURL, user, pass);
                        System.out.println("Connected");
                        connnection.close();
                } catch (SQLException e1) {
                        // TODO Auto-generated catch block
                        e1.printStackTrace();
                }


                EventQueue.invokeLater(new Runnable() {
                        public void run() {

                                try {
                                        Receiving_1 window = new Receiving_1();
                                        window.frame.setVisible(true);
                                } catch (Exception e) {
                                        e.printStackTrace();
                                }
                        }
                });
        }

        /**
```

```
	 * Create the application.
	 */
	public Receiving_1() {
		initialize();

	}

	/**
	 * Initialize the contents of the frame.
	 */
	private void initialize() {
		frame = new JFrame();
		frame.setVisible(true);
		frame.setResizable(true);
		frame.setBounds(270, 100, 450, 300);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);
		frame.setSize(899, 899);

		JPanel panel = new JPanel();
		panel.setBackground(new Color(204, 51, 51));
		panel.setBounds(0, 0, 899, 71);
		frame.getContentPane().add(panel);

		JList list = new JList();
		panel.add(list);

		JLabel lblNewLabel = new JLabel("Receiving");
		panel.add(lblNewLabel);
		lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 40));

		JPanel panel_1 = new JPanel();
		panel_1.setForeground(new Color(0, 0, 0));
		panel_1.setBackground(new Color(153, 204, 204));
		panel_1.setBounds(0, 72, 899, 867);
		panel_1.setSize(899, 899);
		frame.getContentPane().add(panel_1);
		panel_1.setLayout(null);


		JButton btnNewButton_1 = new JButton("Logout");
		btnNewButton_1.setBackground(new Color(204, 51, 102));
		btnNewButton_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		btnNewButton_1.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				try {
	    Connection connection = (Connection)
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank",
	    "postgres", "2580");

	    PreparedStatement st = (PreparedStatement) connection .prepareStatement("delete
from login;");
	    int x = st.executeUpdate();
	    new LoginPage();
	    frame.dispose();
	    }


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

		btnNewButton_1.setBounds(703, 6, 190, 29);
		panel_1.add(btnNewButton_1);

		JLabel lblNewLabel_4 = new JLabel("");
		Image img2 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
		lblNewLabel_4.setIcon(new ImageIcon(img2));
		lblNewLabel_4.setBounds(6, 6, 60, 37);
		panel_1.add(lblNewLabel_4);

		JComboBox comboBox_1 = new JComboBox();
		comboBox_1.addItem("A+");
		comboBox_1.addItem("A-");
		comboBox_1.addItem("B+");
		comboBox_1.addItem("B-");
		comboBox_1.addItem("O+");
```

```
		comboBox_1.addItem("O-");
		comboBox_1.addItem("AB+");
		comboBox_1.addItem("AB-");

		comboBox_1.setSelectedItem(null);

		comboBox_1.setBounds(426, 148, 165, 27);
		panel_1.add(comboBox_1);

		JLabel lblNewLabel_1_1 = new JLabel("Amount of Blood Receiving(in ml) :");
		lblNewLabel_1_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		lblNewLabel_1_1.setBounds(168, 185, 246, 35);
		panel_1.add(lblNewLabel_1_1);

		JLabel lblNewLabel_2 = new JLabel("Branch :");
		lblNewLabel_2.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		lblNewLabel_2.setBounds(346, 236, 68, 16);
		panel_1.add(lblNewLabel_2);

		JLabel lblNewLabel_3 = new JLabel(" Date :");
		lblNewLabel_3.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		lblNewLabel_3.setBounds(360, 271, 68, 16);
		panel_1.add(lblNewLabel_3);

		JLabel lblNewLabel_4_1 = new JLabel("Time slot :");
		lblNewLabel_4_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		lblNewLabel_4_1.setBounds(334, 313, 73, 16);
		panel_1.add(lblNewLabel_4_1);

		textField = new JTextField();
		textField.setColumns(10);
		textField.setBounds(426, 187, 165, 30);
		panel_1.add(textField);

		textField_1 = new JTextField();
		textField_1.setColumns(10);
		textField_1.setBounds(426, 229, 165, 30);
		panel_1.add(textField_1);

		JDateChooser dateChooser = new JDateChooser();
		dateChooser.setBounds(426, 271, 165, 26);
		panel_1.add(dateChooser);

		JComboBox comboBox = new JComboBox();
		comboBox.addItem("9:00 - 10:00");
		comboBox.addItem("10:00 - 11:00");
		comboBox.addItem("11:00 - 12:00");
		comboBox.addItem("3:00 - 4:00");
		comboBox.setSelectedItem(null);
		comboBox.setBounds(420, 309, 171, 27);
		panel_1.add(comboBox);

		JButton btnNewButton = new JButton("Submit");
		btnNewButton.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		btnNewButton.setBounds(444, 342, 117, 29);
		btnNewButton.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				if(textField.getText().isEmpty() ) {
					JOptionPane.showMessageDialog(null, "Amount of Blood
Receiving is Empty");

				}

				else if(textField_1.getText().isEmpty() ) {
					JOptionPane.showMessageDialog(null, "Branch is Empty");

				}

				else {

				}

				int amt =Integer.parseInt(textField.getText());;
		String branch = textField_1.getText();
		SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
		String date = sdf.format(dateChooser.getDate());
		String time = comboBox.getSelectedItem().toString();
		String blood_grp = comboBox_1.getSelectedItem().toString();
		String userid = null;
```

```
                try {


                    Connection connection =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank", "postgres", "2580");

                     PreparedStatement st1= connection.prepareStatement("select user_id from login
where user_id like '_%' ");


                    ResultSet rs1= st1.executeQuery();
                    while (rs1.next())
                    {
                        userid = rs1.getString("user_id");
                    }

                    PreparedStatement st2 = (PreparedStatement) connection
                            .prepareStatement("Select blood_grp, branch from Donation_details where
blood_grp=? and branch=?");


                    st2.setString(1, blood_grp);
                    st2.setString(2, branch);

                    ResultSet rs2 = st2.executeQuery();

                    if (rs2.next()) {

                        JOptionPane.showMessageDialog(btnNewButton, "Required Blood is Available");

                        String query = "INSERT INTO Receiving_details values('" + blood_grp + "'
,'" + amt + "' ,'" + branch + "' , '" + date + "' , '" + time + "' ,'" + userid + "')";


                        PreparedStatement st= connection.prepareStatement("select
blood_grp,Amt_blood,blood_grp,Date_rec,Time_slot,User_id from Receiving_details where blood_grp=?
and Amt_blood=? and Branch=? and Date_rec=? and Time_slot=? and User_id=? ");

                        st.setString(1,blood_grp);
                        st.setInt(2,amt);

                        st.setString(3,branch);
                        st.setString(4,date);
                        st.setString(5,time);
                        st.setString(6,userid);

                        ResultSet rs= st.executeQuery();
                        if (rs.next())
                        {
                            JOptionPane.showMessageDialog(btnNewButton, "already exist");
                        }
                        else
                        {
                                Statement sta = connection.createStatement();
                            int x = sta.executeUpdate(query);


                        }

                        new ConfMsgRec();
                        frame.dispose();


                    } else {
                        JOptionPane.showMessageDialog(btnNewButton, "Required Blood is
Unavailable");
                    }


                    connection.close();
                } catch (Exception exception) {
                    exception.printStackTrace();
                }
                        }
                });

                panel_1.add(btnNewButton);
```

```
                JLabel lblNewLabel_1 = new JLabel("");
                Image img3 = new ImageIcon(this.getClass().getResource("/Purpose.jpg")).getImage();
                lblNewLabel_1.setIcon(new ImageIcon(img3));
                lblNewLabel_1.setBounds(306, 463, 402, 130);
                panel_1.add(lblNewLabel_1);



                JLabel lblNewLabel_5 = new JLabel("Blood Group :");
                lblNewLabel_5.setFont(new Font("Lucida Grande", Font.BOLD, 13));
                lblNewLabel_5.setBounds(311, 152, 117, 16);
                panel_1.add(lblNewLabel_5);
        }
}
```

ConfMsgDon

```
package Default;
import javax.swing.*;



import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;

import javax.swing.JTextField;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.awt.event.ActionEvent;

public class ConfMsgDon {

        private JFrame frame;
        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 {
                                        ConfMsgDon window = new ConfMsgDon();
                                        window.frame.setVisible(true);
                                } catch (Exception e) {
                                        e.printStackTrace();
                                }
                        }
                });
        }

        /**
```

```
	 * Create the application.
	 */
	public ConfMsgDon() {
		initialize();

	}

	/**
	 * Initialize the contents of the frame.
	 */
	private void initialize() {
		frame = new JFrame();
		frame.setVisible(true);
		frame.setResizable(true);
		frame.setBounds(270, 100, 450, 300);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);
		frame.setSize(899, 899);

		JPanel panel = new JPanel();
		panel.setBackground(new Color(204, 51, 51));
		panel.setBounds(0, 0, 899, 71);
		frame.getContentPane().add(panel);

		JList list = new JList();
		panel.add(list);

		JLabel lblNewLabel = new JLabel("Confirmation Window - Donation");
		panel.add(lblNewLabel);
		lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 17));

		JPanel panel_1 = new JPanel();
		panel_1.setForeground(new Color(0, 0, 0));
		panel_1.setBackground(new Color(153, 204, 204));
		panel_1.setBounds(0, 72, 899, 867);
		panel_1.setSize(899, 899);
		frame.getContentPane().add(panel_1);
		panel_1.setLayout(null);
		Image img = new ImageIcon(this.getClass().getResource("/Ok.png")).getImage();

		JButton btnNewButton_1 = new JButton("Logout");
		btnNewButton_1.setBackground(new Color(204, 51, 102));
		btnNewButton_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
		btnNewButton_1.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				try {
	    Connection connection = (Connection)
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank",
			"postgres", "2580");



	    PreparedStatement st = (PreparedStatement) connection
			.prepareStatement("delete from login;");
			int x = st.executeUpdate();
			new LoginPage();
			frame.dispose();
				}


				catch (Exception exception) {
	    exception.printStackTrace();
		}
			}
		});
		btnNewButton_1.setBounds(703, 6, 190, 29);
		panel_1.add(btnNewButton_1);

		JLabel lblNewLabel_3 = new JLabel("");
		Image img1 = new ImageIcon(this.getClass().getResource("/LDescp.jpg")).getImage();
		lblNewLabel_3.setIcon(new ImageIcon(img1));
		lblNewLabel_3.setBounds(225, 362, 500, 274);
		panel_1.add(lblNewLabel_3);

		JLabel lblNewLabel_4 = new JLabel("");
		Image img2 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
		lblNewLabel_4.setIcon(new ImageIcon(img2));
		lblNewLabel_4.setBounds(6, 6, 60, 37);
		panel_1.add(lblNewLabel_4);
```

```
            textField = new JTextField();
            textField.setFont(new Font("Lucida Grande", Font.BOLD, 13));
            textField.setText("Slot Booking For Donation Successfull");
            textField.setHorizontalAlignment(SwingConstants.CENTER);
            textField.setColumns(10);
            textField.setBackground(new Color(153, 153, 153));
            textField.setBounds(320, 100, 286, 77);
            panel_1.add(textField);

            Image img3 = new ImageIcon(this.getClass().getResource("/Ok.png")).getImage();

            JLabel lblNewLabel_2 = new JLabel("");
            lblNewLabel_2.setBounds(459, 189, 60, 37);
            lblNewLabel_2.setIcon(new ImageIcon(img3));
            panel_1.add(lblNewLabel_2);

            JLabel lblNewLabel_1 = new JLabel("Date :");
            lblNewLabel_1.setBounds(225, 276, 102, 29);
            panel_1.add(lblNewLabel_1);

            textField_1 = new JTextField();
            textField_1.setBounds(272, 277, 130, 26);
            panel_1.add(textField_1);
            textField_1.setColumns(10);

            JLabel lblNewLabel_5 = new JLabel("Time Slot :");
            lblNewLabel_5.setBounds(196, 248, 85, 16);
            panel_1.add(lblNewLabel_5);

            textField_2 = new JTextField();
            textField_2.setBounds(272, 243, 130, 26);
            panel_1.add(textField_2);
            textField_2.setColumns(10);

            JLabel lblNewLabel_6 = new JLabel("Branch :");
            lblNewLabel_6.setBounds(211, 317, 61, 16);
            panel_1.add(lblNewLabel_6);

            textField_3 = new JTextField();
            textField_3.setBounds(272, 312, 130, 26);
            panel_1.add(textField_3);
            textField_3.setColumns(10);

            JButton btnNewButton = new JButton("View Slot Details");
            btnNewButton.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            try {

                Connection connection =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank", "postgres", "2580");

                PreparedStatement st= connection.prepareStatement("select
d.Time_slot,d.Date_Don,d.branch from Donation_details d , login l where l.user_id = d.user_id");

                ResultSet rs = st.executeQuery();


                if(rs.next()) {


                    String time = rs.getString("Time_slot");
                    textField_2.setText(time);

                    String date = rs.getString("Date_Don");
                    textField_1.setText(date);

                    String branch = rs.getString("branch");
                    textField_3.setText(branch);

                }

            connection.close();
        } catch (Exception exception) {
            exception.printStackTrace();
        }
```

```
                        }
                });
                btnNewButton.setBounds(511, 277, 149, 29);
                panel_1.add(btnNewButton);

        }
}
```

ConfMsgRec

```
package Default;
import javax.swing.*;


import java.awt.EventQueue;

import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Frame;
import java.awt.Image;

import javax.swing.JTextField;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.awt.event.ActionEvent;

public class ConfMsgRec {

        private JFrame frame;
        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 {
                                        ConfMsgRec window = new ConfMsgRec();
                                        window.frame.setVisible(true);
                                } catch (Exception e) {
                                        e.printStackTrace();
                                }
                        }
                });
        }

        /**
         * Create the application.
         */
        public ConfMsgRec() {
                initialize();

        }

        /**
         * Initialize the contents of the frame.
         */
        private void initialize() {
                frame = new JFrame();
                frame.setVisible(true);
                frame.setResizable(true);
```

```
                frame.setBounds(270, 100, 450, 300);
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.getContentPane().setLayout(null);
                frame.setSize(899, 899);

                JPanel panel = new JPanel();
                panel.setBackground(new Color(204, 51, 51));
                panel.setBounds(0, 0, 899, 71);
                frame.getContentPane().add(panel);



                JLabel lblNewLabel = new JLabel("Confirmation Window - Receiving");
                panel.add(lblNewLabel);
                lblNewLabel.setFont(new Font("Devanagari Sangam MN", Font.BOLD, 17));

                JPanel panel_1 = new JPanel();
                panel_1.setForeground(new Color(0, 0, 0));
                panel_1.setBackground(new Color(153, 204, 204));
                panel_1.setBounds(0, 72, 899, 867);
                panel_1.setSize(899, 899);
                frame.getContentPane().add(panel_1);
                panel_1.setLayout(null);
                Image img = new ImageIcon(this.getClass().getResource("/Ok.png")).getImage();

                JButton btnNewButton_1 = new JButton("Logout");
                btnNewButton_1.setBackground(new Color(204, 51, 102));
                btnNewButton_1.setFont(new Font("Lucida Grande", Font.BOLD, 13));
                btnNewButton_1.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                                try {
                    Connection connection = (Connection)
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank",
                        "postgres", "2580");



                    PreparedStatement st = (PreparedStatement) connection
                        .prepareStatement("delete from login;");
                        int x = st.executeUpdate();
                        new LoginPage();
                        frame.dispose();
                                }


                                catch (Exception exception) {
                    exception.printStackTrace();
                }
                        }
                });
                btnNewButton_1.setBounds(703, 6, 190, 29);
                panel_1.add(btnNewButton_1);

                JLabel lblNewLabel_3 = new JLabel("");
                Image img1 = new ImageIcon(this.getClass().getResource("/LDescp.jpg")).getImage();
                lblNewLabel_3.setIcon(new ImageIcon(img1));
                lblNewLabel_3.setBounds(225, 362, 500, 274);
                panel_1.add(lblNewLabel_3);

                JLabel lblNewLabel_4 = new JLabel("");
                Image img2 = new ImageIcon(this.getClass().getResource("/NHM.jpg")).getImage();
                lblNewLabel_4.setIcon(new ImageIcon(img2));
                lblNewLabel_4.setBounds(6, 6, 60, 37);
                panel_1.add(lblNewLabel_4);

                textField = new JTextField();
                textField.setFont(new Font("Lucida Grande", Font.BOLD, 13));
                textField.setText("Slot Booking For Receiving Successfull");
                textField.setHorizontalAlignment(SwingConstants.CENTER);
                textField.setColumns(10);
                textField.setBackground(new Color(153, 153, 153));
                textField.setBounds(319, 100, 286, 77);
                panel_1.add(textField);

                Image img3 = new ImageIcon(this.getClass().getResource("/Ok.png")).getImage();

                JLabel lblNewLabel_2 = new JLabel("");
                lblNewLabel_2.setBounds(459, 189, 60, 37);
                lblNewLabel_2.setIcon(new ImageIcon(img3));
```

```
                panel_1.add(lblNewLabel_2);

                JLabel lblNewLabel_1 = new JLabel("Date :");
                lblNewLabel_1.setBounds(225, 276, 102, 29);
                panel_1.add(lblNewLabel_1);

                textField_1 = new JTextField();
                textField_1.setBounds(272, 277, 130, 26);
                panel_1.add(textField_1);
                textField_1.setColumns(10);

                JLabel lblNewLabel_5 = new JLabel("Time Slot :");
                lblNewLabel_5.setBounds(196, 248, 85, 16);
                panel_1.add(lblNewLabel_5);

                textField_2 = new JTextField();
                textField_2.setBounds(272, 243, 130, 26);
                panel_1.add(textField_2);
                textField_2.setColumns(10);

                JLabel lblNewLabel_6 = new JLabel("Branch :");
                lblNewLabel_6.setBounds(211, 317, 61, 16);
                panel_1.add(lblNewLabel_6);

                textField_3 = new JTextField();
                textField_3.setBounds(272, 312, 130, 26);
                panel_1.add(textField_3);
                textField_3.setColumns(10);

                JButton btnNewButton = new JButton("View Slot Details");
                btnNewButton.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {

try {

                    Connection connection =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/BloodBank", "postgres", "2580");

                    PreparedStatement st= connection.prepareStatement("select
r.Time_slot,r.Date_Rec,r.branch from Receiving_details r , login l where l.user_id = r.user_id");

                    ResultSet rs = st.executeQuery();



                    if(rs.next()) {


                        String time = rs.getString("Time_slot");
                        textField_2.setText(time);

                        String date = rs.getString("Date_Rec");
                        textField_1.setText(date);

                        String branch = rs.getString("branch");
                        textField_3.setText(branch);

                    }

                connection.close();
            } catch (Exception exception) {
                exception.printStackTrace();
            }


                        }
                });
                btnNewButton.setBounds(511, 277, 149, 29);
                panel_1.add(btnNewButton);

        }
}
```