HOTEL MANAGEMENT SYSTEM

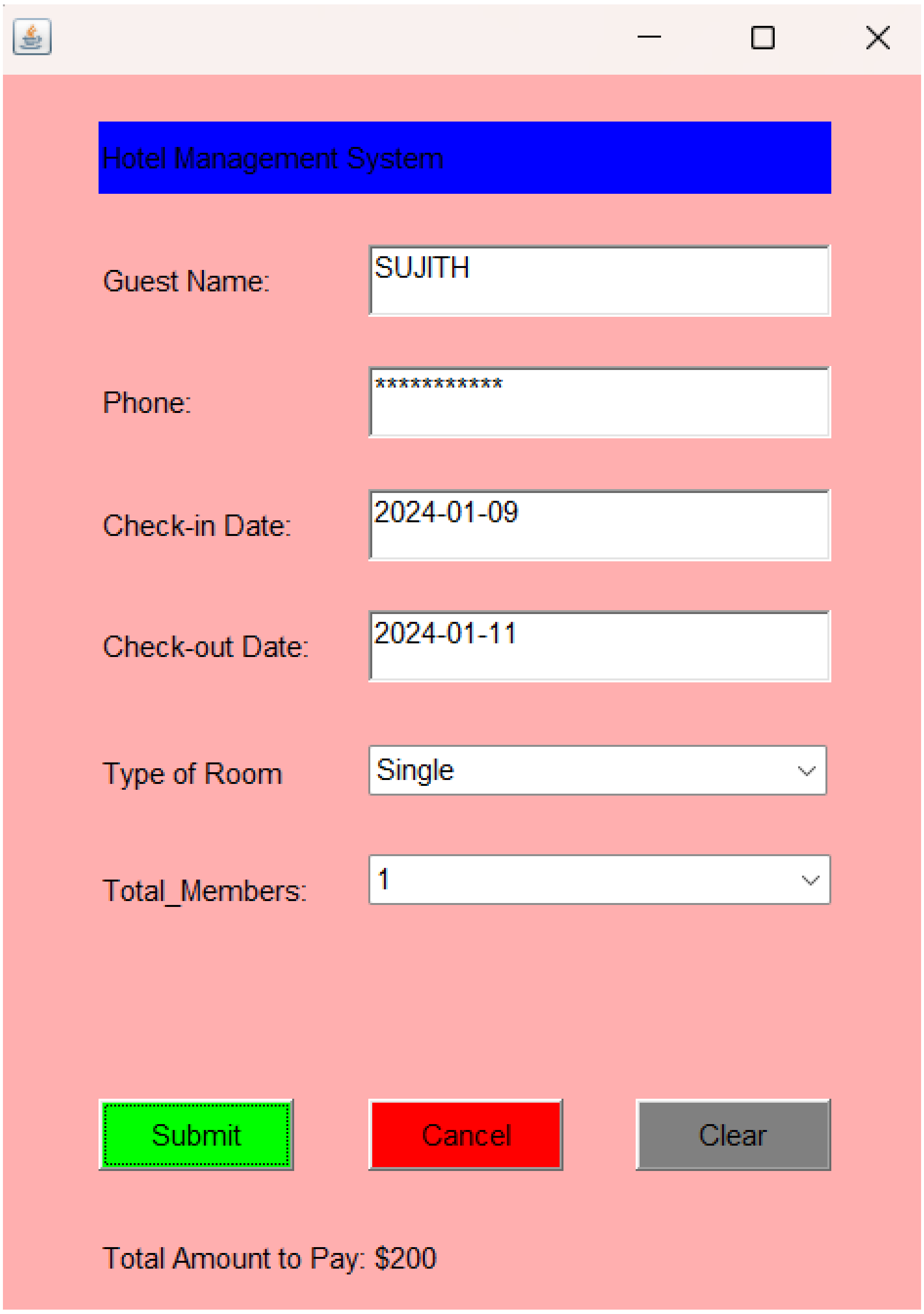
```
import java.awt.*;

import java.awt.event.*;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.concurrent.TimeUnit;


class HotelManagementSystem extends Frame implements ActionListener {

  Label l1, l2, l3, l4, l5, l6,l7,emailLabel, amountLabel;

  TextField tf1, tf2, tf3, tf4, emailField;

  Button b1, b2, clearButton;

  Choice c,room;


  HotelManagementSystem() {

    l1 = new Label("Hotel Management System");

    l1.setBounds(50, 50, 300, 30);

    l1.setBackground(Color.BLUE);


    l2 = new Label("Guest Name:");

    l2.setBounds(50, 100, 100, 30);

    tf1 = new TextField();

    tf1.setBounds(160, 100, 190, 30);


    l3 = new Label("Phone:");

    l3.setBounds(50, 150, 100, 30);

    tf2 = new TextField();

    tf2.setBounds(160, 150, 190, 30);



    l4 = new Label("Check-in Date:");

    l4.setBounds(50, 200, 100, 30);

    tf3 = new TextField();
```

```
tf3.setBounds(160, 200, 190, 30);

l5 = new Label("Check-out Date:");
l5.setBounds(50, 250, 100, 30);
tf4 = new TextField();
tf4.setBounds(160, 250, 190, 30);

l6 = new Label("Type of Room");
l6.setBounds(50, 300, 100, 33);
room = new Choice();
room.setBounds(160, 305, 188,33);
room.add("Single");
room.add("Double");
room.add("Triple");
room.add("Deluxe");



l7 = new Label("Total_Members:");
l7.setBounds(50, 350, 100, 30);
c = new Choice();
c.setBounds(160, 350, 190, 30);
c.add("1");
c.add("2");
c.add("3");
c.add("4");
c.add("5");



b1 = new Button("Submit");
b1.setBounds(50, 450, 80, 30);
b1.addActionListener(this);
b1.setBackground(Color.GREEN);
```

```
b2 = new Button("Cancel");
b2.setBounds(160, 450, 80, 30);
b2.addActionListener(this);
b2.setBackground(Color.RED);

clearButton = new Button("Clear");
clearButton.setBounds(270, 450, 80, 30);
clearButton.addActionListener(this);
clearButton.setBackground(Color.GRAY);

amountLabel = new Label();
amountLabel.setBounds(50, 500, 300, 30);

add(l1);
add(l2);
add(tf1);
add(l3);
add(tf2);
add(l4);
add(tf3);
add(l5);
add(tf4);
add(l6);
add(room);

add(l7);
add(c);
add(b1);
add(b2);
add(clearButton);
add(amountLabel);

setBackground(Color.PINK);
```

```
        setSize(400, 550);
        setLayout(null);
        setVisible(true);
        setLocationRelativeTo(null);
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                dispose();
            }
        });
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == b1) {
            String guestName = tf1.getText();
            String roomNumber = tf2.getText();
            String checkInDate = tf3.getText();
            String checkOutDate = tf4.getText();



            System.out.println("Guest_Name: " + guestName);
            System.out.println("Phone: Invalid");
            System.out.println("CheckInDate: " + checkInDate);
            System.out.println("CheckOutDate: " + checkOutDate);
           if(roomNumber.length()>10){
System.out.println("INvalid");
}
            if (guestName.isEmpty() || roomNumber.isEmpty() || checkInDate.isEmpty() || checkOutDate.isEmpty()) {
                showMessageDialog("Please fill in all fields.");
            } else {
                boolean roomAvailable = checkRoomAvailability(checkInDate, checkOutDate);
                if (roomAvailable) {
                    int totalAmount = calculateAmount(checkInDate, checkOutDate);
                    showMessageDialog("Room is available. Booking confirmed! Total Amount: $" + totalAmount);
                    amountLabel.setText("Total Amount to Pay: $" + totalAmount);
                } else {
                    showMessageDialog("Selected room is not available for the given dates.");
                }
            }
        } else if (e.getSource() == b2) {
            System.exit(0);
        } else if (e.getSource() == clearButton) {
            clearFields();
        }
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e1) {
                dispose();
            }
        });

    }

    private void clearFields() {
        tf1.setText("");
        tf2.setText("");
        tf3.setText("");
        tf4.setText("");
        emailField.setText("");
        amountLabel.setText("");
    }

    private boolean checkRoomAvailability(String checkInDate, String checkOutDate) {

        return true;
    }

    private void showMessageDialog(String message) {
        Dialog d = new Dialog(this, "Message", true);
        d.setLayout(new FlowLayout());
```

```java
  Label l = new Label(message);
  Button okButton = new Button("OK");
  okButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
      d.setVisible(false);
      d.dispose();
    }
  });
  d.add(l);
  d.add(okButton);
  d.setSize(300, 100);
  d.setLocationRelativeTo(this);
  d.setVisible(true);
}

private int calculateAmount(String checkInDate, String checkOutDate) {
  SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
  try {
    Date checkIn = sdf.parse(checkInDate);
    Date checkOut = sdf.parse(checkOutDate);
    long diffInMillies = Math.abs(checkOut.getTime() - checkIn.getTime());
    long diff = TimeUnit.DAYS.convert(diffInMillies, TimeUnit.MILLISECONDS);
    int ratePerDay = 100; // Example rate per day
    return (int) diff * ratePerDay;
  } catch (ParseException e) {
    e.printStackTrace();
    return 0;
  }
}

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


} }
```



Hotel Management System

Guest Name: SUJITH

Phone: ************

Check-in Date: 2024-01-09

Check-out Date: 2024-01-11

Type of Room: Single

Total_Members: 1

Total Amount to Pay: $200