ONLINE MOVIE TICKET BOOKING SYSTEM

An

Object-Oriented Programming through Java Course Project Report

in partial fulfilment of the degree

Bachelor of Technology
in
Computer Science & Engineering

By

L. Abhijith	2103A53007
P. Nithin Reddy	2103A53009

Submitted to

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the **Object-Oriented Programming through Java - Course Project** Report entitled "**ONLINE MOVIE TICKET BOOKING SYSTEM**" is a record of bonafide work carried out by the student **L. ABHIJITH, P. NITHIN REDDY** bearing Roll No(s) **2103A53007, 2103A53009** during the academic year 2023-24 in partial fulfillment of the award of the degree of ***Bachelor of Technology*** in **Computer Science & Engineering** by the Sr University, Warangal.

Lab In-charge **Head of the Department**

Table of Contents

ABSTRACT

ONLINE MOVIE TICKET BOOKING SYSTEM

This project presents a Java-based Movie ticket booking system using the Eclipse Integrated Development Environment (IDE). The primary objective of the system is to provide a user-friendly platform for booking movie tickets for Customers. For registered customers, the system allows them to explore and book tickets for their movie they want to watch. Customers can create new accounts, view the available tickets in theatres, check movie details, and initiate bookings. The system offers an intuitive user interface for customers to seamlessly select their desired ticket and complete the reservation process. Registered customers also have access to their booking history and personal profile information for a personalized experience. On the other hand, administrators are equipped with the tools to manage the online movie ticket booking system efficiently. They can view the list of booked tickets, make modifications as needed, and oversee customer accounts for any potential issues or discrepancies. The Java-based Online Movie Ticket Booking System is built with a strong focus on usability, robustness, and data security. It leverages Java's Object-Oriented Programming Concepts (OOPC) principles for code organization. The Eclipse IDE provides a versatile development environment, making it easy to maintain and expand the system as required. This project represents a comprehensive solution for managing ticket booking, ensuring a seamless experience for both customers and administrators. It demonstrates the power of Java programming and the Eclipse IDE in developing practical, user-friendly software systems for the software industry.

CHAPTER 1

OBJECTIVE OF THE PROJECT

The primary objective of the Java-Based Online Movie Ticket Booking System project is to create a software solution that efficiently manages ticket bookings for customers. This project aims to:

1. User Registration and Authentication:
    - Users can create accounts securely and log in to access the system.
    - Password encryption and user authentication ensure data security.
2. Movie Listings:
    - The system displays a list of currently showing movies.
    - Users can view movie details, including title, synopsis, showtimes, and ratings.
3. Seat Selection:
    - Users can select their preferred seats from an interactive seating arrangement.
    - The system ensures that no two users can book the same seat simultaneously.
4. Booking and Payment:
    - You Can Hold your seats Such that you can pay the cash to the theater offline 15-20 minutes before the show time. But it will cost some charge for holding the seats. This charge will we credited back, if you come the theater, pay cash for your hold seats. Then money will be credited back to your account. Else that will charge you as penality.
    - Multiple payment options, such as credit card or online payment systems, are supported.
5. Booking Confirmation:
    - Users receive email or SMS confirmations of their bookings.

In summary, the project's objective is to develop a robust and user-centric ticket booking management system that simplifies the booking process for customers and streamlines administrative tasks for ticket booking systems.

CHAPTER 2

DEFINITIONS OF THE ELEMENTS USED IN THE PROJECT

1. **User Registration and Authentication**:
   - User registration forms for creating customer accounts.
   - Secure authentication mechanisms, including username and password, or social media login options.
2. **Movie Listings and Showtimes:**
   - A database of movies with details like title, genre, description, and poster images.
   - Showtime schedules for each movie at different theatres and locations.
3. **Seat Selection and Booking:**
   - Interactive seat maps for users to select their preferred seats in the cinema.
   - Booking forms for specifying the number of tickets and payment details.
4. **myPhpAdmin:**
   - Perform tasks like creating, altering, and dropping databases and tables.
   - Execute SQL queries and view the results in a user-friendly interface.
5. **Eclipse:**
   - Syntax highlighting, code completion, and code refactoring capabilities.
   - Integrated debugger for finding and fixing errors in code.
   - A wide range of plugins and extensions available for various programming languages and frameworks.
   - Tools for organizing and managing software projects.
   - Integration with version control systems (e.g., Git) for team collaboration.

In summary, these elements play integral roles in the Java-Based Online Movie Ticket Booking System project, contributing to the development, functionality, and security of the system to provide a user-friendly experience for customers.

CHAPTER 3

DESIGN

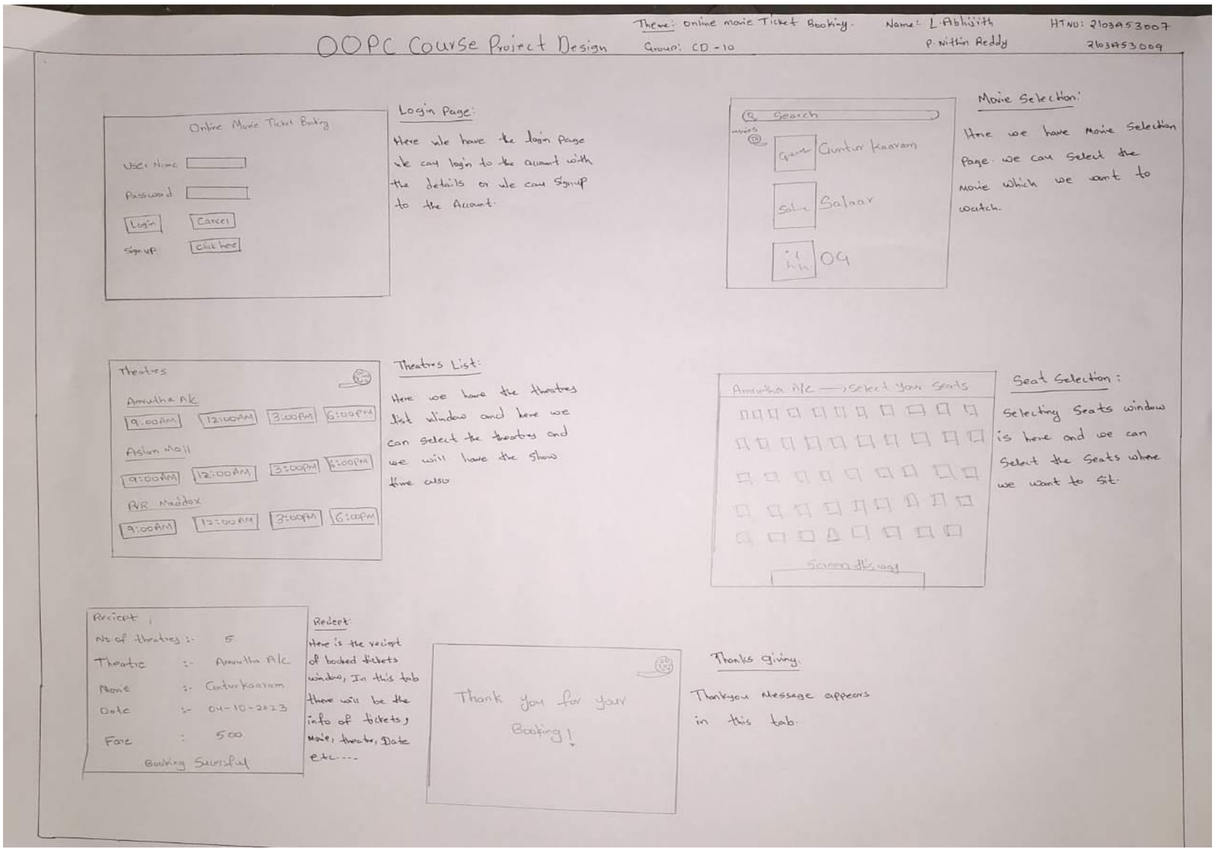
Creating an effective and user-friendly application for users to order movie tickets effortlessly over the internet is a multidimensional process when utilizing Java to design an Online Movie Ticket Booking System. This design plan begins with an introduction, which provides a brief description of the system's purpose, its importance to both customers and theatre operators, and the key goals it seeks to achieve.

User interface design is critical in providing clients with an interesting and user-friendly experience. This includes feature layout and design, such as movie listings, showtimes, seat selection, and payment. An admin interface is also included to allow for the easy management of movies, theatres, and bookings.

The implementation step entails coding the system in Java and associated technologies. Integration of third-party APIs for payment processing and email alerts, as well as the development of user authentication and permission methods, are all part of this.

The design concept finishes by summarizing the process's important aspects and emphasizing the benefits of the Online Movie Ticket Booking System for both customers and operators. It also explores potential future additions, such as addressing scalability and incorporating user input, in order to maintain the system adaptive and responsive to changing technology and consumer preferences.

This is the rough design of our project: -



OOPC Course Project Design

Theme: online movie Ticket Booking.
Group: CD - 10

Name: L.Abhijith, P. Nithin Reddy

HTNO: 2103A53007, 2103A53009

Login Page:

Here we have the login page we can login to the account with the details or we can Signup to the Account.

Movie Selection:

Here we have Movie Selection Page. we can Select the movie which we want to watch.

Theatres List:

Here we have the theatres list window and here we can select the theatres and we will have the show time also

Seat Selection:

Selecting Seats window is here and we can Select the Seats where we want to Sit.

Reciept

No of theatres :- 5
Theatre :- Amrutha A/C
Movie :- Gunturkaaram
Date :- 04-10-2023
Fare : 500
Booking Sucessful

Redeet:

Here is the reciept of booked tickets window, In this tab there will be the info of tickets, movie, theatre, Date etc....

Thank you for your Booking!

Thanks giving.

Thankyou Message appears in this tab.

3.1 SCREENS

The vehicle rental system contains 6 screens

Screen1: - **LOGIN PAGE** in this any user can login if they are already registered if not registered then they need to click on '**SIGN UP**' button then next window appears.

Screen 2: - **SIGN UP** page in this new customer or admin can register by filling all their personal details. These details will be stored in the database

Screen 3: - **MovieListPage,** in this the customer can view all the movies running in the theaters and can select their wish movie.

Screen 3: - **MovieListPage,** in this the customer can view all the theater that are running the above selected movie.

Screen 5: - **TimingPage,** this page will contain all the show dates and timings for the customer to book the ticket for the previous mentioned items (movie, theater).

Screen 6: - **SeatsBookingPage,** in this customer can select their choice seats. But before that he/she has to select the total number of seats to book first. Then he or she can select the seats. At the top of the bottom button, their will be the text that will be update cost you have to pay for the selected number of seats.

Screen 7: - **BillReportPage,** here the customer will get the details like movie name, theater name, date, time, number of seats booked and seats numbers he/she booked. Along with the final cost to pay. From here there is no way to go back.

Screen 8: - **ChoosePaymentPage,** here the customer has to choose the preferred way of payment process.

Screen 9: - **UPIPaymentPage,** here the customer has to pay money via UPI. If he/she selects UPI Payment or Cash Payment. Then, they will redirect to this page.

Screen 10: - **CardPaymentPage,** here the customer has to pay money via Debit or Credit card. If he/she selects Credit/Debit Payment. Then, they will redirect to the page.

Screen 11: - **EndingPage,** this is the ending page of the journey to book the tickets.

CHAPTER 4

IMPLEMENTATION

The Java-Based Vehicle Rental System project is implemented as a user-centric and efficient software solution for managing car rentals. It leverages Java programming language and is developed within the Eclipse IDE. The system incorporates two distinct user roles: customers and administrators.

For customers, the implementation allows user registration for personalized accounts, enabling them to browse car inventory, view vehicle details, and initiate bookings with ease. The booking process is seamless, with real-time availability checks and booking history tracking.

Administrators, on the other hand, have access to an administrative control panel for efficient management of car listings, customer accounts, and bookings. They can add new vehicles, update car information, and monitor bookings, ensuring smooth operations.

The system emphasizes data security to protect customer and car data. Its modular design ensures scalability, enabling future expansion and adaptation to changing industry needs. Overall, the project combines Java's versatility with Eclipse's development capabilities to create a robust car rental system, enhancing user experience and administrative control.

This implementation results in a user-friendly and secure car rental platform that simplifies the booking process for customers and streamlines administrative tasks for Vehicle rental service providers.

4.1 CODE

SCREEN 1: - LoginPage.java CODE

```
package Application;

import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import javax.swing.SwingConstants;
import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.JButton;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.awt.event.ActionEvent;
import java.awt.Toolkit;

public class LoginPage {

    String uname=null,password=null;
    int ok=0;
    private JFrame frame;
    private JTextField textField;
    private JPasswordField passwordField;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
            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();
		frame.setVisible(true);
	}

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

	frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Desktop\\movie-icon.jpg"));
		frame.getContentPane().setBackground(new Color(255, 0, 0));
		frame.setBounds(100, 100, 450, 300);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);

		JLabel lblNewLabel = new JLabel("Enjoy Your Booking");
		lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
		lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
		lblNewLabel.setBounds(10, 10, 416, 34);
		frame.getContentPane().add(lblNewLabel);

		JLabel lblNewLabel_1 = new JLabel("Username");
		lblNewLabel_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1.setBounds(77, 71, 86, 22);
		frame.getContentPane().add(lblNewLabel_1);

		JLabel lblNewLabel_1_1 = new JLabel("Password");
		lblNewLabel_1_1.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_1.setBounds(77, 103, 86, 22);
		frame.getContentPane().add(lblNewLabel_1_1);
```

```
        JLabel lblNewLabel_1_2 = new JLabel(":");
        lblNewLabel_1_2.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel_1_2.setFont(new Font("Times New Roman", Font.PLAIN, 16));
        lblNewLabel_1_2.setBounds(187, 71, 15, 22);
        frame.getContentPane().add(lblNewLabel_1_2);

        JLabel lblNewLabel_1_2_1 = new JLabel(":");
        lblNewLabel_1_2_1.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel_1_2_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
        lblNewLabel_1_2_1.setBounds(187, 103, 15, 22);
        frame.getContentPane().add(lblNewLabel_1_2_1);

        textField = new JTextField();
        textField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
        textField.setBounds(213, 71, 157, 22);
        frame.getContentPane().add(textField);
        textField.setColumns(10);

        passwordField = new JPasswordField();
        passwordField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
        passwordField.setBounds(212, 103, 158, 22);
        frame.getContentPane().add(passwordField);

        JButton btnNewButton = new JButton("Login");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                uname = textField.getText();
                password = String.valueOf(passwordField.getPassword());
                if(uname.equals("")) {
                    JOptionPane.showMessageDialog(frame, "Warning: 'Username' is Mandatory");
                }
                else if(password.equals("")) {
                    JOptionPane.showMessageDialog(frame, "Warning: 'Password'
is Mandatory");
                }
                else {
                    try {
                        Class.forName("com.mysql.cj.jdbc.Driver");
                        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/customermovieticketsbooking", "root",
"");
                        Statement stmt = con.createStatement();
                        String qry = "select * from customerlogin";
                        ResultSet rs = stmt.executeQuery(qry);
                        while(rs.next()) {
                            if(uname.equalsIgnoreCase(rs.getString(1)) &&
password.equals((rs.getString(2)))) {
                                ok=1;
                                break;
                            }
                        }
                        if(ok==1) {
                            stmt.close();
                            con.close();
                            JOptionPane.showMessageDialog(frame, "Login
Successfully");
                            MoviesListPage mlp = new MoviesListPage();
                            frame.dispose();
                        }
                        else {
                            JOptionPane.showMessageDialog(frame, "Login
Unsuccessfully");
                            textField.setText(null);
                            passwordField.setText(null);
                        }
                    }
```

```
                        catch(Exception ex) {
                            ex.printStackTrace();
                        }
                    }
                }
            });
            btnNewButton.setFont(new Font("Times New Roman", Font.ITALIC, 14));
            btnNewButton.setBounds(117, 155, 85, 21);
            frame.getContentPane().add(btnNewButton);

            JButton btnClear = new JButton("Clear");
            btnClear.setFont(new Font("Times New Roman", Font.ITALIC, 14));
            btnClear.setBounds(258, 155, 85, 21);
            frame.getContentPane().add(btnClear);

            JButton btnSignUp = new JButton("Sign Up");
            btnSignUp.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    SignUpPage sup = new SignUpPage();
                    frame.dispose();
                }
            });
            btnSignUp.setFont(new Font("Times New Roman", Font.ITALIC, 14));
            btnSignUp.setBounds(187, 218, 85, 21);
            frame.getContentPane().add(btnSignUp);

            JLabel lblNewLabel_1_3 = new JLabel("Didn't had Account? Create with below
Button");
            lblNewLabel_1_3.setHorizontalAlignment(SwingConstants.CENTER);
            lblNewLabel_1_3.setFont(new Font("Times New Roman", Font.PLAIN, 16));
            lblNewLabel_1_3.setBounds(10, 186, 416, 22);
            frame.getContentPane().add(lblNewLabel_1_3);
            //frame.setVisible(true);
        }
```

```
    }
```

SCREEN 2: - SignUpPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.SwingConstants;
import java.awt.Font;
import java.awt.Toolkit;
import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.JRadioButton;
import javax.swing.JButton;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.sql.*;
import java.awt.event.ActionEvent;

public class SignUpPage {

	String
uname=null,password=null,name=null,email=null,mobile=null,gender=null,warning=null;
	int ok=0,un=0,mob=0;
	private JFrame frame;
	private JTextField nameField;
	private JTextField emailField;
	private JTextField mobileField;
	private JTextField unameField;
	private JPasswordField passwordField;

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

	/**
	 * Create the application.
	 */
	public SignUpPage() {
		initialize();
		frame.setVisible(true);
	}

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

	frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
		frame.getContentPane().setBackground(new Color(255, 0, 0));
		frame.setBounds(100, 100, 900, 600);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);
```

```
JLabel lblNewLabel = new JLabel("Welcome to Online Booking System");
lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
lblNewLabel.setBounds(10, 10, 866, 46);
frame.getContentPane().add(lblNewLabel);

JLabel lblNewLabel_1 = new JLabel("Full Name");
lblNewLabel_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1.setHorizontalAlignment(SwingConstants.RIGHT);
lblNewLabel_1.setBounds(305, 98, 89, 23);
frame.getContentPane().add(lblNewLabel_1);

JLabel lblNewLabel_1_1 = new JLabel("Email ID");
lblNewLabel_1_1.setHorizontalAlignment(SwingConstants.RIGHT);
lblNewLabel_1_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_1.setBounds(305, 129, 89, 23);
frame.getContentPane().add(lblNewLabel_1_1);

JLabel lblNewLabel_1_2 = new JLabel("Mobile");
lblNewLabel_1_2.setHorizontalAlignment(SwingConstants.RIGHT);
lblNewLabel_1_2.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2.setBounds(305, 162, 89, 23);
frame.getContentPane().add(lblNewLabel_1_2);

JLabel lblNewLabel_1_3 = new JLabel("Gender");
lblNewLabel_1_3.setHorizontalAlignment(SwingConstants.RIGHT);
lblNewLabel_1_3.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_3.setBounds(305, 195, 89, 23);
frame.getContentPane().add(lblNewLabel_1_3);

JLabel lblNewLabel_1_4 = new JLabel("Username");
lblNewLabel_1_4.setHorizontalAlignment(SwingConstants.RIGHT);
lblNewLabel_1_4.setFont(new Font("Times New Roman", Font.PLAIN, 16));
```

```
lblNewLabel_1_4.setBounds(305, 228, 89, 23);
frame.getContentPane().add(lblNewLabel_1_4);

JLabel lblNewLabel_1_5 = new JLabel("Password");
lblNewLabel_1_5.setHorizontalAlignment(SwingConstants.RIGHT);
lblNewLabel_1_5.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_5.setBounds(305, 261, 89, 23);
frame.getContentPane().add(lblNewLabel_1_5);

JLabel lblNewLabel_1_2_1 = new JLabel(":");
lblNewLabel_1_2_1.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel_1_2_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2_1.setBounds(416, 98, 15, 22);
frame.getContentPane().add(lblNewLabel_1_2_1);

JLabel lblNewLabel_1_2_1_1 = new JLabel(":");
lblNewLabel_1_2_1_1.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel_1_2_1_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2_1_1.setBounds(416, 131, 15, 22);
frame.getContentPane().add(lblNewLabel_1_2_1_1);

JLabel lblNewLabel_1_2_1_2 = new JLabel(":");
lblNewLabel_1_2_1_2.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel_1_2_1_2.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2_1_2.setBounds(416, 164, 15, 22);
frame.getContentPane().add(lblNewLabel_1_2_1_2);

JLabel lblNewLabel_1_2_1_3 = new JLabel(":");
lblNewLabel_1_2_1_3.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel_1_2_1_3.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2_1_3.setBounds(416, 197, 15, 22);
frame.getContentPane().add(lblNewLabel_1_2_1_3);

JLabel lblNewLabel_1_2_1_4 = new JLabel(":");
```

```
lblNewLabel_1_2_1_4.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel_1_2_1_4.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2_1_4.setBounds(416, 230, 15, 22);
frame.getContentPane().add(lblNewLabel_1_2_1_4);

JLabel lblNewLabel_1_2_1_5 = new JLabel(":");
lblNewLabel_1_2_1_5.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel_1_2_1_5.setFont(new Font("Times New Roman", Font.PLAIN, 16));
lblNewLabel_1_2_1_5.setBounds(416, 262, 15, 22);
frame.getContentPane().add(lblNewLabel_1_2_1_5);

nameField = new JTextField();
nameField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
nameField.setColumns(10);
nameField.setBounds(461, 96, 157, 22);
frame.getContentPane().add(nameField);

emailField = new JTextField();
emailField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
emailField.setColumns(10);
emailField.setBounds(461, 129, 157, 22);
frame.getContentPane().add(emailField);

mobileField = new JTextField();
mobileField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
mobileField.setColumns(10);
mobileField.setBounds(461, 162, 157, 22);
frame.getContentPane().add(mobileField);

unameField = new JTextField();
unameField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
unameField.setColumns(10);
unameField.setBounds(461, 228, 157, 22);
frame.getContentPane().add(unameField);
```

```
passwordField = new JPasswordField();
passwordField.setFont(new Font("Times New Roman", Font.PLAIN, 16));
passwordField.setBounds(461, 261, 158, 22);
frame.getContentPane().add(passwordField);

JRadioButton maleRadioBtn = new JRadioButton("Male");
maleRadioBtn.setBackground(new Color(255, 0, 0));
maleRadioBtn.setFont(new Font("Times New Roman", Font.PLAIN, 16));
maleRadioBtn.setBounds(461, 197, 68, 21);
frame.getContentPane().add(maleRadioBtn);

JRadioButton femaleRadioBtn = new JRadioButton("Female");
femaleRadioBtn.setBackground(new Color(255, 0, 0));
femaleRadioBtn.setFont(new Font("Times New Roman", Font.PLAIN, 16));
femaleRadioBtn.setBounds(531, 197, 83, 21);
frame.getContentPane().add(femaleRadioBtn);

maleRadioBtn.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                gender = "Male";
                femaleRadioBtn.setSelected(false);
        }
});

femaleRadioBtn.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                gender = "Female";
                maleRadioBtn.setSelected(false);
        }
});

JButton btnSignUp = new JButton("Sign Up");
btnSignUp.setFont(new Font("Times New Roman", Font.ITALIC, 14));
btnSignUp.setBounds(328, 321, 85, 21);
frame.getContentPane().add(btnSignUp);

JButton btnClear = new JButton("Clear");
btnClear.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                unameField.setText(null);
                passwordField.setText(null);
                nameField.setText(null);
                mobileField.setText(null);
                emailField.setText(null);
                maleRadioBtn.setSelected(false);
                femaleRadioBtn.setSelected(false);
                gender = null;
        }
});
btnClear.setFont(new Font("Times New Roman", Font.ITALIC, 14));
btnClear.setBounds(511, 321, 85, 21);
frame.getContentPane().add(btnClear);

JButton btnNewButton = new JButton("Login");
btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                LoginPage lp = new LoginPage();
                frame.dispose();
        }
});
btnNewButton.setFont(new Font("Times New Roman", Font.ITALIC, 14));
btnNewButton.setBounds(398, 411, 85, 21);
frame.getContentPane().add(btnNewButton);

JLabel lblNewLabel_1_3_1 = new JLabel("Already had Account? Then Login");
lblNewLabel_1_3_1.setHorizontalAlignment(SwingConstants.CENTER);
lblNewLabel_1_3_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
```

```
		lblNewLabel_1_3_1.setBounds(10, 375, 866, 22);
		frame.getContentPane().add(lblNewLabel_1_3_1);

		warning = "Warning: All fields with '*' are mandatory";
		JLabel warningLabel = new JLabel(warning);
		warningLabel.setForeground(new Color(255, 255, 255));
		warningLabel.setHorizontalAlignment(SwingConstants.CENTER);
		warningLabel.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 14));
		warningLabel.setBounds(10, 56, 866, 23);
		frame.getContentPane().add(warningLabel);

		JLabel lblNewLabel_1_6 = new JLabel("*");
		lblNewLabel_1_6.setForeground(new Color(255, 255, 255));
		lblNewLabel_1_6.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_6.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_6.setBounds(391, 98, 15, 23);
		frame.getContentPane().add(lblNewLabel_1_6);

		JLabel lblNewLabel_1_6_1 = new JLabel("*");
		lblNewLabel_1_6_1.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_6_1.setForeground(Color.WHITE);
		lblNewLabel_1_6_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_6_1.setBounds(391, 129, 15, 23);
		frame.getContentPane().add(lblNewLabel_1_6_1);

		JLabel lblNewLabel_1_6_2 = new JLabel("*");
		lblNewLabel_1_6_2.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_6_2.setForeground(Color.WHITE);
		lblNewLabel_1_6_2.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_6_2.setBounds(391, 162, 15, 23);
		frame.getContentPane().add(lblNewLabel_1_6_2);

		JLabel lblNewLabel_1_6_3 = new JLabel("*");
		lblNewLabel_1_6_3.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_6_3.setForeground(Color.WHITE);
		lblNewLabel_1_6_3.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_6_3.setBounds(391, 195, 15, 23);
		frame.getContentPane().add(lblNewLabel_1_6_3);

		JLabel lblNewLabel_1_6_4 = new JLabel("*");
		lblNewLabel_1_6_4.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_6_4.setForeground(Color.WHITE);
		lblNewLabel_1_6_4.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_6_4.setBounds(391, 228, 15, 23);
		frame.getContentPane().add(lblNewLabel_1_6_4);

		JLabel lblNewLabel_1_6_5 = new JLabel("*");
		lblNewLabel_1_6_5.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1_6_5.setForeground(Color.WHITE);
		lblNewLabel_1_6_5.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1_6_5.setBounds(391, 261, 15, 23);
		frame.getContentPane().add(lblNewLabel_1_6_5);

		btnSignUp.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				try {
					Class.forName("com.mysql.cj.jdbc.Driver");
					Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/customermovieticketsbooking", "root", "");
					Statement stmt = con.createStatement();
					name = nameField.getText();
					email = emailField.getText();
					mobile = mobileField.getText();
					uname = unameField.getText();
					password = String.valueOf(passwordField.getPassword());
					if(name.equals("")) {
```

```
                        JOptionPane.showMessageDialog(frame,    "Warning:
'Name' is Mandatory");
                    }
                    else if(email.equals("")) {
                        JOptionPane.showMessageDialog(frame,    "Warning:
'Email ID' is Mandatory");
                    }
                    else if(mobile.equals("")) {
                        JOptionPane.showMessageDialog(frame,    "Warning:
'Mobile Number' is Mandatory");
                    }
                    else if(uname.equals("")) {
                        JOptionPane.showMessageDialog(frame,    "Warning:
'Username' is Mandatory");
                    }
                    else if(password.equals("")) {
                        JOptionPane.showMessageDialog(frame,    "Warning:
'Password' is Mandatory");
                    }
                    else if(gender.equals("")) {
                        JOptionPane.showMessageDialog(frame,    "Warning:
'Gender' is Mandatory");
                    }
                    else {
                        String qry = "select * from customerlogin";
                        ResultSet rs = stmt.executeQuery(qry);
                        while(rs.next()) {
                            un=0;
                            mob=0;
                            if(uname.equals(rs.getString(1))) {
                                un=1;
                                break;
                            }
                            if(mobile.equals(rs.getString(6))) {
                                mob=1;
                                break;
                            }
                        }
                        if(un==1) {
                            JOptionPane.showMessageDialog(frame,
"Warning: 'Username' is Already Taken");
                        }
                        else if(mob==1) {
                            JOptionPane.showMessageDialog(frame,
"Warning: 'Mobile Number' is Already Taken");
                        }
                        else {
                            ok=1;
                        }
                    }
                    if(ok==1) {
                        String sql = "insert into customerlogin values
('"+uname+"','"+password+"', '"+name+"', '"+gender+"','"+email+"', '"+mobile+"')";
                        stmt.executeUpdate(sql);
                        stmt.close();
                        con.close();
                        JOptionPane.showMessageDialog(frame, "Registration
Completed Successfully");
                        LoginPage lp = new LoginPage();
                        frame.dispose();
                    }
                }
                catch(Exception ex) {
                    ex.printStackTrace();
                }
            }
        });
    }
```

```
}
```

SCREEN 3: MoviesListPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.SwingConstants;
import java.awt.Font;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.JButton;
import javax.swing.ImageIcon;
import java.awt.Color;

public class MoviesListPage {

        int count=0;
        private JFrame frame;

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

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

    /**
     * Create the application.
     * @throws SQLException
     * @throws ClassNotFoundException
     */
    public MoviesListPage() throws ClassNotFoundException, SQLException {
        initialize();
        frame.setVisible(true);
    }

    /**
     * Initialize the contents of the frame.
     * @throws ClassNotFoundException
     * @throws SQLException
     */
    private void initialize() throws ClassNotFoundException, SQLException {
        frame = new JFrame();

    frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
        frame.setBounds(100, 100, 450, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);

        JLabel lblNewLabel = new JLabel("Movies List");
```

```
		lblNewLabel.setForeground(new Color(255, 255, 255));
		lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
		lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
		lblNewLabel.setBounds(10, 10, 416, 36);
		frame.getContentPane().add(lblNewLabel);

		JLabel movieSelection = new JLabel("Movie Name:- ");
		movieSelection.setForeground(new Color(255, 255, 255));
		movieSelection.setHorizontalAlignment(SwingConstants.CENTER);
		movieSelection.setFont(new Font("Times New Roman", Font.BOLD, 16));
		movieSelection.setBounds(10, 47, 416, 16);
		frame.getContentPane().add(movieSelection);

		JButton btnBack = new JButton("Back");
		btnBack.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				LoginPage lp = new LoginPage();
				frame.dispose();
			}
		});
		btnBack.setFont(new Font("Times New Roman", Font.ITALIC, 14));
		btnBack.setBounds(183, 232, 85, 21);
		frame.getContentPane().add(btnBack);

		try {
			Class.forName("com.mysql.cj.jdbc.Driver");
			Connection                    con                    =
DriverManager.getConnection("jdbc:mysql://localhost:3306/customermovieticketsbooking",   "root",
"");
			Statement stmt = con.createStatement();
			String qry = "select * from movies";
			ResultSet rs = stmt.executeQuery(qry);
			int y=70;
			while(rs.next()) {
```

```
                String movieName = rs.getString(1);
                JButton movie_1 = new JButton(movieName);
                movie_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
                movie_1.setBounds(10, y, 416, 27);
                frame.getContentPane().add(movie_1);
                y=y+30;
                movie_1.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                        String var = "Movie Name:- "+movieName;
                        movieSelection.setText(var);
                        JOptionPane.showMessageDialog(frame, var);
                        TheatersListPage tlp = new TheatersListPage(movieName);
                        frame.dispose();
                    }
                });
            }
        }
        catch(Exception ex) {
            ex.printStackTrace();
        }

        JLabel lblBackGround = new JLabel("");
        lblBackGround.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 1 (5) (2).png"));
        lblBackGround.setBounds(0, 0, 436, 263);
        frame.getContentPane().add(lblBackGround);
    }
}
```

SCREEN 4: TheatersListPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.SwingConstants;
import javax.swing.JButton;
import java.awt.Color;
import javax.swing.ImageIcon;

public class TheatersListPage {

	private JFrame frame;

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

```
                        }
                    }
                });
            }

            /**
             * Create the application.
             */
            public TheatersListPage(String movieName) {
                initialize(movieName);
                frame.setVisible(true);
            }

            /**
             * Initialize the contents of the frame.
             */




            private void initialize(String movieName) {
                frame = new JFrame();

            frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
    p\\movie-icon.jpg"));
                frame.setBounds(100, 100, 450, 600);
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.getContentPane().setLayout(null);

                JLabel lblNewLabel = new JLabel("Theaters List");
                lblNewLabel.setForeground(new Color(255, 255, 255));
                lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
                lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
                lblNewLabel.setBounds(10, 10, 416, 36);
                frame.getContentPane().add(lblNewLabel);
```

```
JLabel lblTheaterName = new JLabel("Theater Name:- ");
lblTheaterName.setForeground(new Color(255, 255, 255));
lblTheaterName.setHorizontalAlignment(SwingConstants.CENTER);
lblTheaterName.setFont(new Font("Times New Roman", Font.BOLD, 16));
lblTheaterName.setBounds(10, 49, 416, 19);
frame.getContentPane().add(lblTheaterName);

JButton btnNewButton = new JButton("Back");
btnNewButton.addActionListener(new ActionListener() {
	public void actionPerformed(ActionEvent e) {
		try {
			MoviesListPage mlp = new MoviesListPage();
		} catch (ClassNotFoundException e1) {
			// TODO Auto-generated catch block
			e1.printStackTrace();
		} catch (SQLException e1) {
			// TODO Auto-generated catch block
			e1.printStackTrace();
		}
		frame.dispose();
	}
});
btnNewButton.setFont(new Font("Times New Roman", Font.ITALIC, 14));
btnNewButton.setBounds(167, 532, 85, 21);
frame.getContentPane().add(btnNewButton);

/*JButton theaterName = new JButton("");
theaterName.setFont(new Font("Times New Roman", Font.BOLD, 16));
theaterName.setBounds(10, 78, 416, 28);
frame.getContentPane().add(theaterName);*/

try {
	Class.forName("com.mysql.cj.jdbc.Driver");
```

```
                        Connection                                  con                                  =
DriverManager.getConnection("jdbc:mysql://localhost:3306/customermovieticketsbooking",     "root",
"");
                        Statement stmt = con.createStatement();
                        String qry = "select * from movies";
                        ResultSet rs = stmt.executeQuery(qry);
                        String line = null;
                        while(rs.next()) {
                                if(movieName.equalsIgnoreCase(rs.getString(1))) {
                                        line = rs.getString(2);
                                        break;
                                }
                        }
                        String[] arr = line.split(", ");
                        //System.out.println(line);
                        int y=80;
                        for(int i=0;i<arr.length;i++) {
                                String var = arr[i];
                                JButton theaterName = new JButton(var);
                                theaterName.setFont(new Font("Times New Roman", Font.BOLD, 16));
                                theaterName.setBounds(10, y, 416, 28);
                                frame.getContentPane().add(theaterName);
                                theaterName.addActionListener(new ActionListener() {
                                        public void actionPerformed(ActionEvent e) {
                                                String label = "Theater Name:- "+var;
                                                lblTheaterName.setText(label);
                                                TimingsPage tp = new TimingsPage(movieName,var);
                                                frame.dispose();
                                        }
                                });
                                y+=30;
                        }
                }
                catch(Exception ex) {
                        ex.printStackTrace();
                }

                JLabel lblNewLabel_1 = new JLabel("");
                lblNewLabel_1.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 0
(2) (1).png"));
                lblNewLabel_1.setBounds(0, 0, 436, 563);
                frame.getContentPane().add(lblNewLabel_1);
          }

}
```

SCREEN 5: TimingsPage.java CODE

```
package Application;
import java.awt.EventQueue;
import java.text.*;
import java.time.*;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.text.SimpleDateFormat;
import java.util.Date;
import javax.swing.SwingConstants;
import javax.swing.JComboBox;
import javax.swing.JButton;
import java.awt.Color;
import javax.swing.ImageIcon;

public class TimingsPage {

	private JFrame frame;
	long daysDifference=0;
	String d1,line,bookingDate=null;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					TimingsPage window = new TimingsPage("Jawan","PVR cinema");
					window.frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the application.
	 */
	public TimingsPage(String movieName,String theaterName) {
		initialize(movieName,theaterName);
		frame.setVisible(true);
	}

	/**
```

```
	 * Initialize the contents of the frame.
	 */
	private void initialize(String movieName,String theaterName) {
		frame = new JFrame();

	frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
		frame.setBounds(100, 100, 450, 600);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);

		JLabel lblNewLabel = new JLabel("Mark your Date and Time");
		lblNewLabel.setForeground(new Color(255, 255, 255));
		lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
		lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
		lblNewLabel.setBounds(10, 10, 416, 36);
		frame.getContentPane().add(lblNewLabel);

		JLabel lblNewLabel_1 = new JLabel("Date:-");
		lblNewLabel_1.setForeground(new Color(255, 255, 255));
		lblNewLabel_1.setHorizontalAlignment(SwingConstants.RIGHT);
		lblNewLabel_1.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		lblNewLabel_1.setBounds(51, 83, 77, 22);
		frame.getContentPane().add(lblNewLabel_1);

		try {
			Class.forName("com.mysql.cj.jdbc.Driver");
			Connection                    con                    =
DriverManager.getConnection("jdbc:mysql://localhost:3306/customermovieticketsbooking",   "root",
"");
			Statement stmt = con.createStatement();
			String qry = "select * from theaters";
			ResultSet rs = stmt.executeQuery(qry);
			while(rs.next()) {
```

```
                              if(movieName.equalsIgnoreCase(rs.getString(2))                    &&
theaterName.equalsIgnoreCase(rs.getString(1))) {
                                        d1 = rs.getString(4);
                                        line = rs.getString(3);
                                        break;
                              }
                    }
                    //Converting String to date
                    SimpleDateFormat dateFormat = new SimpleDateFormat("dd-MM-yyyy");
                    Date date = null;
          try {
             date = dateFormat.parse(d1);
          } catch (ParseException e) {
             e.printStackTrace();
          }
                    //Finding Difference between the today and last date
                    Date today1 = new Date();
                    long difference = date.getTime() - today1.getTime();
     daysDifference = difference / (24 * 60 * 60 * 1000);
            }
            catch(Exception ex) {
                    ex.printStackTrace();
            }

            JComboBox<LocalDate> comboBox = new JComboBox<LocalDate>();
            comboBox.setBounds(153, 85, 149, 21);
            frame.getContentPane().add(comboBox);

            JLabel lblDateTime = new JLabel("Date & Time:- ");
            lblDateTime.setForeground(new Color(255, 255, 255));
            lblDateTime.setHorizontalAlignment(SwingConstants.CENTER);
            lblDateTime.setFont(new Font("Times New Roman", Font.PLAIN, 16));
            lblDateTime.setBounds(10, 51, 416, 22);
            frame.getContentPane().add(lblDateTime);
            JButton btnBack = new JButton("Back");
            btnBack.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            TheatersListPage tlp = new TheatersListPage(movieName);
                            frame.dispose();
                    }
            });
            btnBack.setFont(new Font("Times New Roman", Font.ITALIC, 14));
            btnBack.setBounds(181, 532, 85, 21);
            frame.getContentPane().add(btnBack);

            //Adding dates to comboBox
            LocalDate today = LocalDate.now();
            comboBox.addItem(null);
            comboBox.addItem(today);
for(int i=1;i<=daysDifference+1;i++) {
   LocalDate nextDays = today.plusDays(i);
   comboBox.addItem(nextDays);
}
            comboBox.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            bookingDate = String.valueOf(comboBox.getSelectedItem());
                    }
            });

            int y=130;
            String[] arr = line.split(", ");
            for(int i=0;i<arr.length;i++) {
                    String timing = arr[i];
                    JButton timingBtn = new JButton(timing);
                    timingBtn.addActionListener(new ActionListener() {
                            public void actionPerformed(ActionEvent e) {
                                    if(bookingDate==null || bookingDate.equals("null")) {
```

```
                        JOptionPane.showMessageDialog(frame, "Warning:
'Date' is Mandatory");
                    }
                    else {
                        String var = "Date & Time:- "+bookingDate+" &
"+timing;
                        lblDateTime.setText(var);
                        SeatsBookingPage sbp = new
SeatsBookingPage(movieName,theaterName,bookingDate);
                        frame.dispose();
                    }
                }
            });
            timingBtn.setFont(new Font("Times New Roman", Font.PLAIN, 14));
            timingBtn.setBounds(10, y, 416, 28);
            frame.getContentPane().add(timingBtn);
            y=y+30;
        }

        JLabel lblImage = new JLabel("");
        lblImage.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 1 (7)
(1).png"));
        lblImage.setBounds(0, 0, 436, 563);
        frame.getContentPane().add(lblImage);
    }
}
```

SCREEN 6: SeatsBookingPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.SwingConstants;
import java.awt.Font;
import javax.swing.JCheckBox;
import javax.swing.AbstractAction;
import java.awt.event.ActionEvent;
import javax.swing.Action;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Arrays;
import javax.swing.JComboBox;
import java.awt.Color;
import javax.swing.ImageIcon;

public class SeatsBookingPage {

	int totalNumberOfBookings=0,totalSelectedSeats=0,bookingInd=-1,totalAmount=0,removedInd=-1;
	int[] removedSeat = new int[4];
	String[] bookings = new String[4];
	private JFrame frame;
	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					SeatsBookingPage window = new SeatsBookingPage("Leo","Asian Sridevi Mall","29-10-2023");
					window.frame.setVisible(true);
				} catch (Exception e) {
```

```
                        e.printStackTrace();
                    }
                }
            });
        }

        /**
         * Create the application.
         */
        public SeatsBookingPage(String movieName,String theaterName,String bookingDate) {
            initialize(movieName,theaterName,bookingDate);
            frame.setVisible(true);
        }

        /**
         * Initialize the contents of the frame.
         */
        private void initialize(String movieName,String theaterName,String bookingDate) {
            frame = new JFrame();
            frame.getContentPane().setBackground(new Color(255, 0, 0));
            frame.setBounds(100, 100, 800, 700);
            frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            frame.getContentPane().setLayout(null);

            JLabel lblNewLabel = new JLabel("Enjoy With Your Seats");
            lblNewLabel.setForeground(new Color(255, 255, 255));
            lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
            lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
            lblNewLabel.setBounds(10, 10, 766, 39);
            frame.getContentPane().add(lblNewLabel);

            JLabel lblNewLabel_1 = new JLabel("Select Number of Seats to Book:- ");
            lblNewLabel_1.setForeground(new Color(255, 255, 255));
            lblNewLabel_1.setFont(new Font("Times New Roman", Font.PLAIN, 18));
            lblNewLabel_1.setHorizontalAlignment(SwingConstants.RIGHT);
            lblNewLabel_1.setBounds(208, 59, 272, 22);
            frame.getContentPane().add(lblNewLabel_1);

            JComboBox<String> comboBox = new JComboBox<String>();
            comboBox.addItem("0");
            comboBox.addItem("1");
            comboBox.addItem("2");
            comboBox.addItem("3");
            comboBox.addItem("4");
            comboBox.setBounds(485, 60, 54, 21);
            frame.getContentPane().add(comboBox);

            JButton btnNewButton = new JButton("Proceed to Payment");
            btnNewButton.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    if(totalSelectedSeats==0) {
                        JOptionPane.showMessageDialog(frame, "Warning: You Have
    to Select Seats First");
                    }
                    else if(totalSelectedSeats!=totalNumberOfBookings) {
                        String warn="Warning: Still You can Select
    "+(Integer.toString(totalNumberOfBookings-totalSelectedSeats))+" Seats";
                        JOptionPane.showMessageDialog(frame, warn);
                    }
                    else {
                        String[] finalBookings = new String[totalNumberOfBookings];
                        int index=0;
                        for(int i=0;i<bookings.length;i++) {
                            if(bookings[i] != null &&
    index<totalNumberOfBookings) {
                                finalBookings[index]=bookings[i];
                                System.out.println(finalBookings[index]+"
    "+index);
```

```
                            index++;
                        }
                    }
                    Arrays.sort(finalBookings);
                    String res="Your Booking Seats:- ";
                    for(int i=0;i<totalNumberOfBookings;i++) {
                        res+="\n"+finalBookings[i];
                    }
                    res+="\nTotol Amount Only for Seats Booking (Excluding Tax,
etc....):- "+totalAmount+" Rs.";
                    JOptionPane.showMessageDialog(frame, res);
                    BillReportPage brp = new
BillReportPage(movieName,theaterName,totalNumberOfBookings,finalBookings,totalAmount,booki
ngDate);
                    frame.dispose();
                }
            }
        });
        btnNewButton.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC,
16));
        btnNewButton.setBounds(283, 622, 222, 31);
        frame.getContentPane().add(btnNewButton);

        JLabel amountUpdate = new JLabel("Amount for the Select Number of Seats to Book:-
'0' rs.");
        amountUpdate.setForeground(new Color(255, 255, 255));
        amountUpdate.setHorizontalAlignment(SwingConstants.CENTER);
        amountUpdate.setFont(new Font("Times New Roman", Font.PLAIN, 18));
        amountUpdate.setBounds(10, 577, 766, 22);
        frame.getContentPane().add(amountUpdate);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                TimingsPage tp = new TimingsPage(movieName, theaterName);
                frame.dispose();
            }
        });
        btnBack.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        btnBack.setBounds(21, 622, 134, 31);
        frame.getContentPane().add(btnBack);

        comboBox.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                totalNumberOfBookings = Integer.parseInt((String)
comboBox.getSelectedItem());
            }
        });


        int seatX=45,seatY=90+30;
        char var='A';
        String rowName;
        for(int i=0;i<14;i++) {
            seatX=45;
            for(int j=0;j<24;j++) {
                rowName=Character.toString((char) (var+i))+" "+Integer.toString(j+1);
                JCheckBox chckbxNewCheckBox = new JCheckBox();
                chckbxNewCheckBox.setBounds(seatX, seatY, 21, 21);
                frame.getContentPane().add(chckbxNewCheckBox);
                chckbxNewCheckBox.setText(rowName);
                chckbxNewCheckBox.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                        try {
                            Class.forName("com.mysql.cj.jdbc.Driver");
                            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/customermovieticketsbooking", "root",
"");
```

```
                            Statement stmt = con.createStatement();
                            String qry = "select Amount from theaters where
movie_name='"+movieName+"' and theater_name='"+theaterName+"'";
                            ResultSet rs = stmt.executeQuery(qry);
                            rs.next();
                            if(chckbxNewCheckBox.isSelected()) {
                                if(totalNumberOfBookings==0) {

    JOptionPane.showMessageDialog(frame, "Warning: You Must Select Total Number of
Booking First.\n(\"Value Should Be Greater Than 0 and Less Than 5\")");

    chckbxNewCheckBox.setSelected(false);
                                }
                                else
if(totalNumberOfBookings==totalSelectedSeats) {

    chckbxNewCheckBox.setSelected(false);

    JOptionPane.showMessageDialog(frame, "Warning: You Have Reached Your Booking Limit");
                                }
                                else {
                                    totalSelectedSeats++;
                                    if(removedInd==-1) {
                                        bookingInd++;

    bookings[bookingInd]=chckbxNewCheckBox.getText();
                                    }
                                    else {

    bookings[removedSeat[removedInd]]=chckbxNewCheckBox.getText();
                                        removedInd--;
                                    }

    totalAmount+=Integer.parseInt((String)rs.getString(1));
```

```
                                    String text = "Amount for the
Select Number of Seats to Book:- '"+Integer.toString(totalAmount)+"' rs.";
                                    amountUpdate.setText(text);
                                }
                            }
                            else {
                                bookingInd--;
                                totalSelectedSeats--;
                                String
unWantSeat=chckbxNewCheckBox.getText();
                                for(int i=0;i<bookings.length;i++) {

    if(unWantSeat.equalsIgnoreCase(bookings[i])) {
                                        removedInd++;

    removedSeat[removedInd]=i;
                                        bookings[i]=null;
                                        break;
                                    }
                                }
                                totalAmount-
=Integer.parseInt((String)rs.getString(1));
                                String text = "Amount for the Select
Number of Seats to Book:- '"+Integer.toString(totalAmount)+"' rs.";
                                amountUpdate.setText(text);
                            }

                        }
                        catch(Exception ex) {
                            ex.printStackTrace();
                        }
                    }
                });
                seatX=seatX+30;
```

```
        }
        seatY=seatY+30;
}

/*JLabel lblNum = new JLabel("1");
lblNum.setHorizontalAlignment(SwingConstants.CENTER);
lblNum.setFont(new Font("Times New Roman", Font.BOLD, 14));
lblNum.setBounds(51, 59, 21, 29);
frame.getContentPane().add(lblNum);*/

int x=50;
for(int i=0;i<24;i++) {
        int num=i+1;
        JLabel lblNum = new JLabel(Integer.toString(num));
        lblNum.setForeground(new Color(255, 255, 255));
        lblNum.setHorizontalAlignment(SwingConstants.CENTER);
        lblNum.setFont(new Font("Times New Roman", Font.BOLD, 14));
        lblNum.setBounds(x, 90, 15, 20);
        frame.getContentPane().add(lblNum);
        x=x+30;
}

/*JLabel lblAlpha = new JLabel("A");
lblAlpha.setHorizontalAlignment(SwingConstants.CENTER);
lblAlpha.setFont(new Font("Times New Roman", Font.BOLD, 14));
lblAlpha.setBounds(10, 113, 21, 29);
frame.getContentPane().add(lblAlpha);*/

int y=90+30;
var='A';
for(int i=0;i<14;i++) {
        JLabel lblAlpha = new JLabel(Character.toString((char) (var+i)));
        lblAlpha.setForeground(new Color(255, 255, 255));
        lblAlpha.setHorizontalAlignment(SwingConstants.CENTER);
```

```
            lblAlpha.setFont(new Font("Times New Roman", Font.BOLD, 14));
            lblAlpha.setBounds(10, y, 15, 25);
            frame.getContentPane().add(lblAlpha);
            y=y+30;
        }
    }
}
```

SCREEN 7: BillReportPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.SwingConstants;
import java.awt.Font;
import java.awt.Toolkit;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.awt.Color;
import javax.swing.ImageIcon;

public class BillReportPage {

    private JFrame frame;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    String[] bookings = new String[] {"C 22","C 21","C 24"};
                    BillReportPage window = new BillReportPage("Leo","Asian
Sridevi Mall",3,bookings,900,"29-10-2023");
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the application.
     */
    public BillReportPage(String movieName,String theaterName,int
totalNumberOfBookings,String[] bookings,int totalAmount,String bookingDate) {

    initialize(movieName,theaterName,totalNumberOfBookings,bookings,totalAmount,bookingD
ate);
        frame.setVisible(true);
    }

    /**
     * Initialize the contents of the frame.
     */
    private void initialize(String movieName,String theaterName,int
totalNumberOfBookings,String[] bookings,int totalAmount,String bookingDate) {
        frame = new JFrame();
        frame.getContentPane().setBackground(new Color(255, 0, 0));

    frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
        frame.setBounds(100, 100, 450, 550);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
        JLabel lblNewLabel = new JLabel("Final Bill Report");
        lblNewLabel.setBackground(new Color(0, 0, 0));
        lblNewLabel.setForeground(new Color(255, 255, 255));
        lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
        lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
        lblNewLabel.setBounds(10, 10, 416, 35);
        frame.getContentPane().add(lblNewLabel);

        JLabel lblMovieName = new JLabel("Movie Name:- "+movieName);
        lblMovieName.setForeground(new Color(255, 255, 255));
        lblMovieName.setHorizontalAlignment(SwingConstants.CENTER);
        lblMovieName.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC,
16));
        lblMovieName.setBounds(10, 55, 416, 28);
        frame.getContentPane().add(lblMovieName);

        JLabel lblTheaterName = new JLabel("Theater Name:- "+theaterName);
        lblTheaterName.setForeground(new Color(255, 255, 255));
        lblTheaterName.setHorizontalAlignment(SwingConstants.CENTER);
        lblTheaterName.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC,
16));
        lblTheaterName.setBounds(10, 93, 416, 28);
        frame.getContentPane().add(lblTheaterName);

        JLabel lblSeatsYouHave = new JLabel("Seats You Have Booked are given below:- ");
        lblSeatsYouHave.setForeground(new Color(255, 255, 255));
        lblSeatsYouHave.setHorizontalAlignment(SwingConstants.CENTER);
        lblSeatsYouHave.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC,
16));
        lblSeatsYouHave.setBounds(10, 202, 416, 28);
        frame.getContentPane().add(lblSeatsYouHave);

        int y=240;
```

```
		for(int i=0;i<bookings.length;i++) {
			JLabel lblSeatingNumbers = new JLabel(bookings[i]);
			lblSeatingNumbers.setForeground(new Color(0, 0, 0));
			lblSeatingNumbers.setBackground(new Color(255,255,255));
			lblSeatingNumbers.setVerticalAlignment(SwingConstants.TOP);
			lblSeatingNumbers.setHorizontalAlignment(SwingConstants.CENTER);
			lblSeatingNumbers.setFont(new Font("Times New Roman", Font.BOLD | 
Font.ITALIC, 16));
			lblSeatingNumbers.setBounds(10, y, 416, 25);
			frame.getContentPane().add(lblSeatingNumbers);
			y=y+25;
		}

		JButton btnNewButton = new JButton("OK");
		btnNewButton.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				ChoosePayment cp = new ChoosePayment(totalAmount,bookingDate);
				frame.dispose();
			}
		});
		btnNewButton.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		btnNewButton.setBounds(286, 401, 101, 35);
		frame.getContentPane().add(btnNewButton);

		JButton btnBack = new JButton("Back");
		btnBack.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				SeatsBookingPage sbp = new 
SeatsBookingPage(movieName,theaterName,bookingDate);
				frame.dispose();
			}
		});
		btnBack.setFont(new Font("Times New Roman", Font.PLAIN, 16));
		btnBack.setBounds(50, 401, 101, 35);
```

```
        frame.getContentPane().add(btnBack);

        JLabel lblTotalSeats = new JLabel("Total Number Of Seats Booked:- "+totalNumberOfBookings);
        lblTotalSeats.setForeground(new Color(255, 255, 255));
        lblTotalSeats.setHorizontalAlignment(SwingConstants.CENTER);
        lblTotalSeats.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        lblTotalSeats.setBounds(10, 164, 416, 28);
        frame.getContentPane().add(lblTotalSeats);

        JLabel lblAmount = new JLabel("Total Amount To Be Paid:- "+totalAmount);
        lblAmount.setForeground(new Color(255, 255, 255));
        lblAmount.setHorizontalAlignment(SwingConstants.CENTER);
        lblAmount.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        lblAmount.setBounds(10, 363, 416, 28);
        frame.getContentPane().add(lblAmount);

        JLabel lblDate = new JLabel("Date:- "+bookingDate);
        lblDate.setForeground(new Color(255, 255, 255));
        lblDate.setHorizontalAlignment(SwingConstants.CENTER);
        lblDate.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        lblDate.setBounds(10, 131, 416, 28);
        frame.getContentPane().add(lblDate);
    }

}
```

SCREEN 8: ChoosePaymentPage.java CODE

```
package Application;
import java.awt.EventQueue;
```

```
}

import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.SwingConstants;
import javax.swing.UIClientPropertyKey;
import javax.swing.JRadioButton;
import javax.swing.JButton;
import java.awt.Color;
import javax.swing.ImageIcon;

public class ChoosePayment {

	String paymentMethod="null";
	private JFrame frame;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					ChoosePayment window = new ChoosePayment(400,"29-10-
2023");
					window.frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the application.
	 */
	public ChoosePayment(int totalAmount,String bookingDate) {
		initialize(totalAmount,bookingDate);
		frame.setVisible(true);
	}

	/**
	 * Initialize the contents of the frame.
	 */
	private void initialize(int totalAmount,String bookingDate) {
		frame = new JFrame();

	frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
		frame.setBounds(100, 100, 550, 400);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);

		JLabel lblNewLabel = new JLabel("Pay");
		lblNewLabel.setForeground(new Color(255, 255, 255));
		lblNewLabel.setVerticalAlignment(SwingConstants.TOP);
		lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
		lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD, 24));
		lblNewLabel.setBounds(10, 10, 516, 29);
		frame.getContentPane().add(lblNewLabel);

		JLabel lblNewLabel_1 = new JLabel("&");
		lblNewLabel_1.setForeground(new Color(255, 255, 255));
		lblNewLabel_1.setVerticalAlignment(SwingConstants.TOP);
		lblNewLabel_1.setHorizontalAlignment(SwingConstants.CENTER);
```

```
lblNewLabel_1.setFont(new Font("Times New Roman", Font.BOLD, 24));
lblNewLabel_1.setBounds(10, 40, 516, 29);
frame.getContentPane().add(lblNewLabel_1);

JLabel lblEnjoyYourDay = new JLabel("Enjoy Your Day");
lblEnjoyYourDay.setForeground(new Color(255, 255, 255));
lblEnjoyYourDay.setVerticalAlignment(SwingConstants.TOP);
lblEnjoyYourDay.setHorizontalAlignment(SwingConstants.CENTER);
lblEnjoyYourDay.setFont(new Font("Times New Roman", Font.BOLD, 24));
lblEnjoyYourDay.setBounds(10, 70, 516, 29);
frame.getContentPane().add(lblEnjoyYourDay);

JLabel lblDate = new JLabel(bookingDate);
lblDate.setForeground(new Color(255, 255, 255));
lblDate.setVerticalAlignment(SwingConstants.TOP);
lblDate.setHorizontalAlignment(SwingConstants.CENTER);
lblDate.setFont(new Font("Times New Roman", Font.BOLD, 24));
lblDate.setBounds(10, 100, 516, 29);
frame.getContentPane().add(lblDate);

JRadioButton upiRadioBtn = new JRadioButton("UPI Payment");
upiRadioBtn.setForeground(new Color(0, 0, 0));
upiRadioBtn.setFont(new Font("Times New Roman", Font.PLAIN, 16));
upiRadioBtn.setBounds(24, 197, 121, 34);
frame.getContentPane().add(upiRadioBtn);

JRadioButton bankRadioBtn = new JRadioButton("Cash Payment At Theater");
bankRadioBtn.setForeground(new Color(0, 0, 0));
bankRadioBtn.setFont(new Font("Times New Roman", Font.PLAIN, 16));
bankRadioBtn.setBounds(157, 197, 192, 34);
frame.getContentPane().add(bankRadioBtn);

JRadioButton cardRadioBtn = new JRadioButton("Debit or Credit Card");
cardRadioBtn.setForeground(new Color(0, 0, 0));
```

```
        cardRadioBtn.setFont(new Font("Times New Roman", Font.PLAIN, 16));
        cardRadioBtn.setBounds(365, 197, 161, 34);
        frame.getContentPane().add(cardRadioBtn);

        JButton btnNewButton = new JButton("Proceed");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                if(paymentMethod.equals("null")) {
                    JOptionPane.showMessageDialog(frame, "Warning:- Select a
Payment Method to Proceed further");
                }
                else {
                    JOptionPane.showMessageDialog(frame, "Your Payment
Method is "+paymentMethod);
                    if(paymentMethod.equalsIgnoreCase("UPI Payment")) {
                        UPIPaymentPage upi = new UPIPaymentPage("");
                        frame.dispose();
                    }
                    else if(paymentMethod.equalsIgnoreCase("Card Payment")) {
                        CardPaymentPage cpp = new CardPaymentPage();
                        frame.dispose();
                    }
                    else {
                        UPIPaymentPage cupi = new UPIPaymentPage("Pay
Some Fee To the Theater for Holding Your Seats");
                        frame.dispose();
                    }
                }
            }
        });
        btnNewButton.setFont(new Font("Times New Roman", Font.PLAIN, 16));
        btnNewButton.setBounds(186, 270, 161, 34);
        frame.getContentPane().add(btnNewButton);
```

```
upiRadioBtn.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                if(upiRadioBtn.isSelected()) {
                        paymentMethod = "UPI Payment";
                        bankRadioBtn.setSelected(false);
                        cardRadioBtn.setSelected(false);
                }
                else {
                        paymentMethod="null";
                }
        }
});
bankRadioBtn.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                if(bankRadioBtn.isSelected()) {
                        paymentMethod = "Cash Payment";
                        upiRadioBtn.setSelected(false);
                        cardRadioBtn.setSelected(false);
                }
                else {
                        paymentMethod="null";
                }
        }
});
cardRadioBtn.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                if(cardRadioBtn.isSelected()) {
                        paymentMethod = "Card Payment";
                        upiRadioBtn.setSelected(false);
                        bankRadioBtn.setSelected(false);
                }
                else {
                        paymentMethod="null";
                }
```

```
                    }
                });

                JLabel lblNewLabel_2 = new JLabel("New label");
                lblNewLabel_2.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 0 
(3) (2).png"));
                lblNewLabel_2.setBounds(0, 0, 536, 363);
                frame.getContentPane().add(lblNewLabel_2);
        }
}
```

SCREEN 9: UPIPaymentPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import java.awt.Toolkit;
import javax.swing.SwingConstants;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.awt.Color;
import javax.swing.ImageIcon;

public class UPIPaymentPage {

        private JFrame frame;
        private JTextField upiField;

        /**
         * Launch the application.
```

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

    /**
     * Create the application.
     */
    public UPIPaymentPage(String text) {
        initialize(text);
        frame.setVisible(true);
    }

    /**
     * Initialize the contents of the frame.
     */
    private void initialize(String text) {
        frame = new JFrame();

    frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Desktop\\movie-icon.jpg"));
        frame.setBounds(100, 100, 450, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
```

```
JLabel lblNewLabel = new JLabel("Welcome to UPI Payment Process");
lblNewLabel.setForeground(new Color(255, 255, 255));
lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 24));
lblNewLabel.setBounds(10, 10, 416, 50);
frame.getContentPane().add(lblNewLabel);

JLabel lblEnterYourUpi = new JLabel("Enter Your UPI ID:-");
lblEnterYourUpi.setForeground(new Color(255, 255, 255));
lblEnterYourUpi.setHorizontalAlignment(SwingConstants.RIGHT);
lblEnterYourUpi.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC,
16));
lblEnterYourUpi.setBounds(10, 125, 190, 29);
frame.getContentPane().add(lblEnterYourUpi);

upiField = new JTextField();
upiField.setBounds(210, 129, 190, 23);
frame.getContentPane().add(upiField);
upiField.setColumns(10);

JButton btnNewButton = new JButton("Proceed");
btnNewButton.addActionListener(new ActionListener() {
	public void actionPerformed(ActionEvent e) {
		EndingPage ep = new EndingPage();
		frame.dispose();
	}
});
btnNewButton.setFont(new Font("Times New Roman", Font.ITALIC, 16));
btnNewButton.setBounds(149, 192, 110, 29);
frame.getContentPane().add(btnNewButton);

JLabel lblMsg = new JLabel(text);
lblMsg.setForeground(new Color(255, 255, 255));
lblMsg.setHorizontalAlignment(SwingConstants.CENTER);
```

```
		lblMsg.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
		lblMsg.setBounds(10, 70, 416, 23);
		frame.getContentPane().add(lblMsg);

		JLabel lblMsg2 = new JLabel("via UPI Payment Only!");
		lblMsg2.setForeground(new Color(255, 255, 255));
		lblMsg2.setHorizontalAlignment(SwingConstants.CENTER);
		lblMsg2.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
		lblMsg2.setBounds(10, 103, 416, 23);
		frame.getContentPane().add(lblMsg2);

		JLabel lblNewLabel_1 = new JLabel("");
		lblNewLabel_1.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 1
(2) (1).png"));
		lblNewLabel_1.setBounds(0, 0, 436, 263);
		frame.getContentPane().add(lblNewLabel_1);
	}
}
```

SCREEN 10: CardPaymentPage.java CODE

```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Toolkit;
import javax.swing.SwingConstants;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.awt.Color;
import javax.swing.ImageIcon;
```

```
public class CardPaymentPage {

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

    /**
     * Create the application.
     */
    public CardPaymentPage() {
        initialize();
        frame.setVisible(true);
    }

    /**
     * Initialize the contents of the frame.
```

```
	 */
	private void initialize() {
		frame = new JFrame();

	frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
		frame.setBounds(100, 100, 450, 300);
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().setLayout(null);

		JLabel lblWelcomeToCard = new JLabel("Welcome to Card Payment Process");
		lblWelcomeToCard.setForeground(new Color(255, 255, 255));
		lblWelcomeToCard.setHorizontalAlignment(SwingConstants.CENTER);
		lblWelcomeToCard.setFont(new Font("Times New Roman", Font.BOLD |
Font.ITALIC, 24));
		lblWelcomeToCard.setBounds(10, 10, 416, 50);
		frame.getContentPane().add(lblWelcomeToCard);

		JLabel lblEnterYourCard = new JLabel("Enter Card Number:-");
		lblEnterYourCard.setForeground(new Color(255, 255, 255));
		lblEnterYourCard.setHorizontalAlignment(SwingConstants.RIGHT);
		lblEnterYourCard.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC,
16));
		lblEnterYourCard.setBounds(20, 109, 190, 29);
		frame.getContentPane().add(lblEnterYourCard);

		textField = new JTextField();
		textField.setColumns(10);
		textField.setBounds(220, 113, 190, 23);
		frame.getContentPane().add(textField);

		JLabel lblEnterCardExpiry = new JLabel("Enter Card Expiry Date:-");
		lblEnterCardExpiry.setForeground(new Color(255, 255, 255));
		lblEnterCardExpiry.setHorizontalAlignment(SwingConstants.RIGHT);
```

```
        lblEnterCardExpiry.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        lblEnterCardExpiry.setBounds(20, 148, 190, 29);
        frame.getContentPane().add(lblEnterCardExpiry);

        JLabel lblEnterCvvNumber = new JLabel("Enter CVV Number:-");
        lblEnterCvvNumber.setForeground(new Color(255, 255, 255));
        lblEnterCvvNumber.setHorizontalAlignment(SwingConstants.RIGHT);
        lblEnterCvvNumber.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        lblEnterCvvNumber.setBounds(20, 187, 190, 29);
        frame.getContentPane().add(lblEnterCvvNumber);

        JLabel lblEnterCardHolder = new JLabel("Enter Card Holder Name:-");
        lblEnterCardHolder.setForeground(new Color(255, 255, 255));
        lblEnterCardHolder.setHorizontalAlignment(SwingConstants.RIGHT);
        lblEnterCardHolder.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 16));
        lblEnterCardHolder.setBounds(20, 70, 190, 29);
        frame.getContentPane().add(lblEnterCardHolder);

        textField_1 = new JTextField();
        textField_1.setColumns(10);
        textField_1.setBounds(220, 152, 190, 23);
        frame.getContentPane().add(textField_1);

        textField_2 = new JTextField();
        textField_2.setColumns(10);
        textField_2.setBounds(220, 191, 190, 23);
        frame.getContentPane().add(textField_2);

        textField_3 = new JTextField();
        textField_3.setColumns(10);
        textField_3.setBounds(220, 70, 190, 23);
        frame.getContentPane().add(textField_3);

        JButton btnNewButton = new JButton("Proceed");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                EndingPage ep = new EndingPage();
                frame.dispose();
            }
        });
        btnNewButton.setFont(new Font("Times New Roman", Font.ITALIC, 16));
        btnNewButton.setBounds(159, 224, 110, 29);
        frame.getContentPane().add(btnNewButton);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 1
(2) (1).png"));
        lblNewLabel.setBounds(0, 0, 436, 263);
        frame.getContentPane().add(lblNewLabel);
    }
}
```

SCREEN 11: EndingPage.java CODE

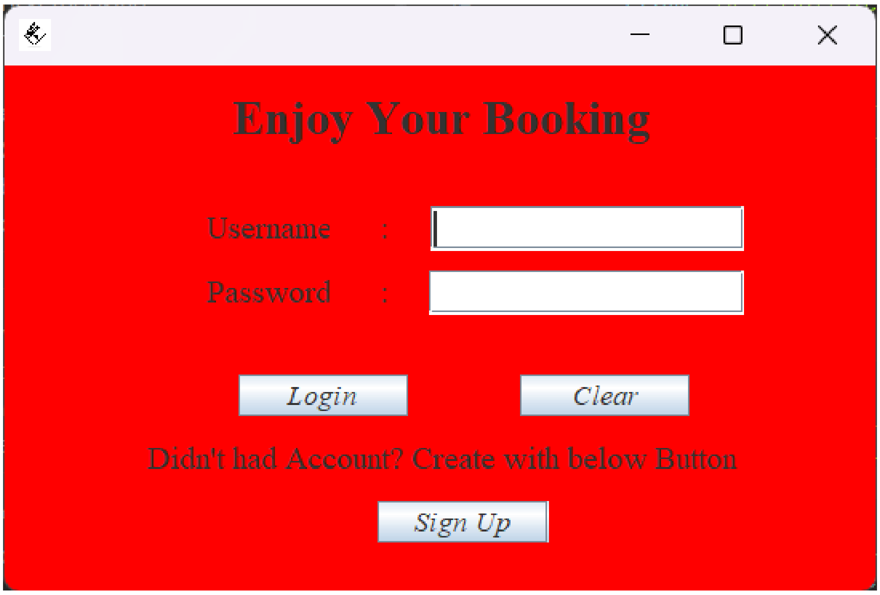
```
package Application;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.SwingConstants;
import java.awt.Font;
import java.awt.Toolkit;
import java.awt.Color;
import javax.swing.ImageIcon;

public class EndingPage {
	private JFrame frame;

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

	/**
	 * Create the application.
	 */
	public EndingPage() {
		initialize();
		frame.setVisible(true);
	}

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

	frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\nithi\\OneDrive\\Deskto
p\\movie-icon.jpg"));
```

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

        JLabel lblNewLabel = new JLabel("THANK YOU");
        lblNewLabel.setForeground(new Color(255, 255, 255));
        lblNewLabel.setVerticalAlignment(SwingConstants.BOTTOM);
        lblNewLabel.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 24));
        lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
        lblNewLabel.setBounds(10, 25, 416, 103);
        frame.getContentPane().add(lblNewLabel);

        JLabel lblTheEnd = new JLabel("THE END");
        lblTheEnd.setForeground(new Color(255, 255, 255));
        lblTheEnd.setVerticalAlignment(SwingConstants.TOP);
        lblTheEnd.setHorizontalAlignment(SwingConstants.CENTER);
        lblTheEnd.setFont(new Font("Times New Roman", Font.BOLD | Font.ITALIC, 24));
        lblTheEnd.setBounds(10, 132, 416, 103);
        frame.getContentPane().add(lblTheEnd);

        JLabel lblNewLabel_1 = new JLabel("");
        lblNewLabel_1.setIcon(new ImageIcon("C:\\Users\\nithi\\Downloads\\Create a visual 1
(6) (1).png"));
        lblNewLabel_1.setBounds(0, 0, 436, 258);
        frame.getContentPane().add(lblNewLabel_1);
    }

}
```

CHAPTER 5

RESULT SCREENS

SCREEN 1:



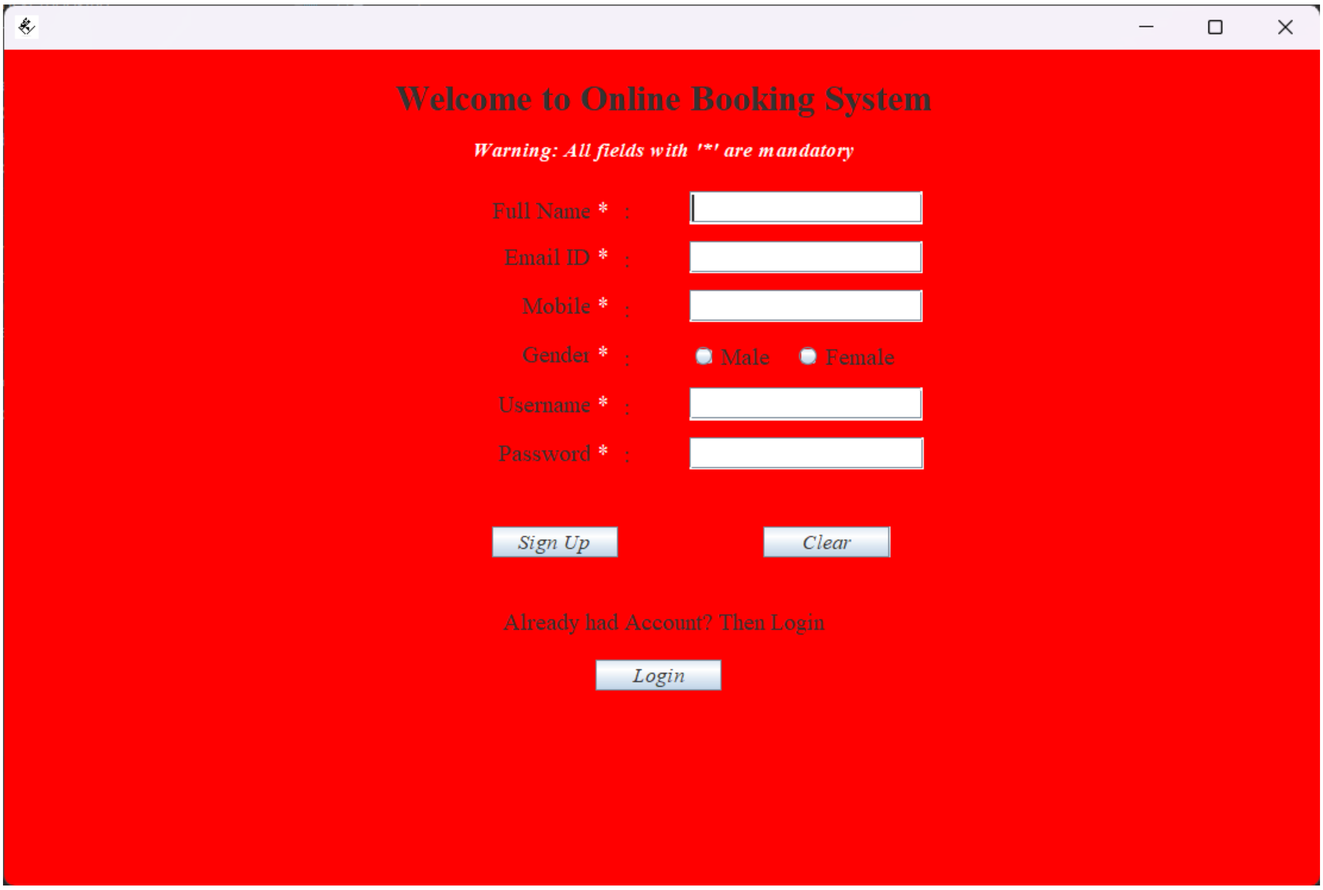
Enjoy Your Booking

Username :

Password :

Didn't had Account? Create with below Button

SCREEN 2:



Welcome to Online Booking System

Warning: All fields with '' are mandatory*

Full Name * :

Email ID * :

Mobile * :

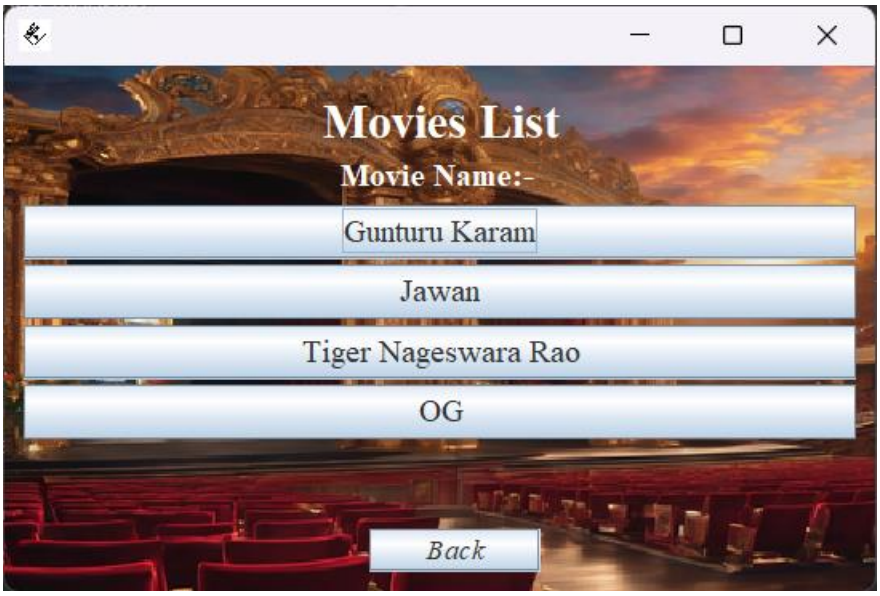
Gender * : ○ Male ○ Female

Username * :

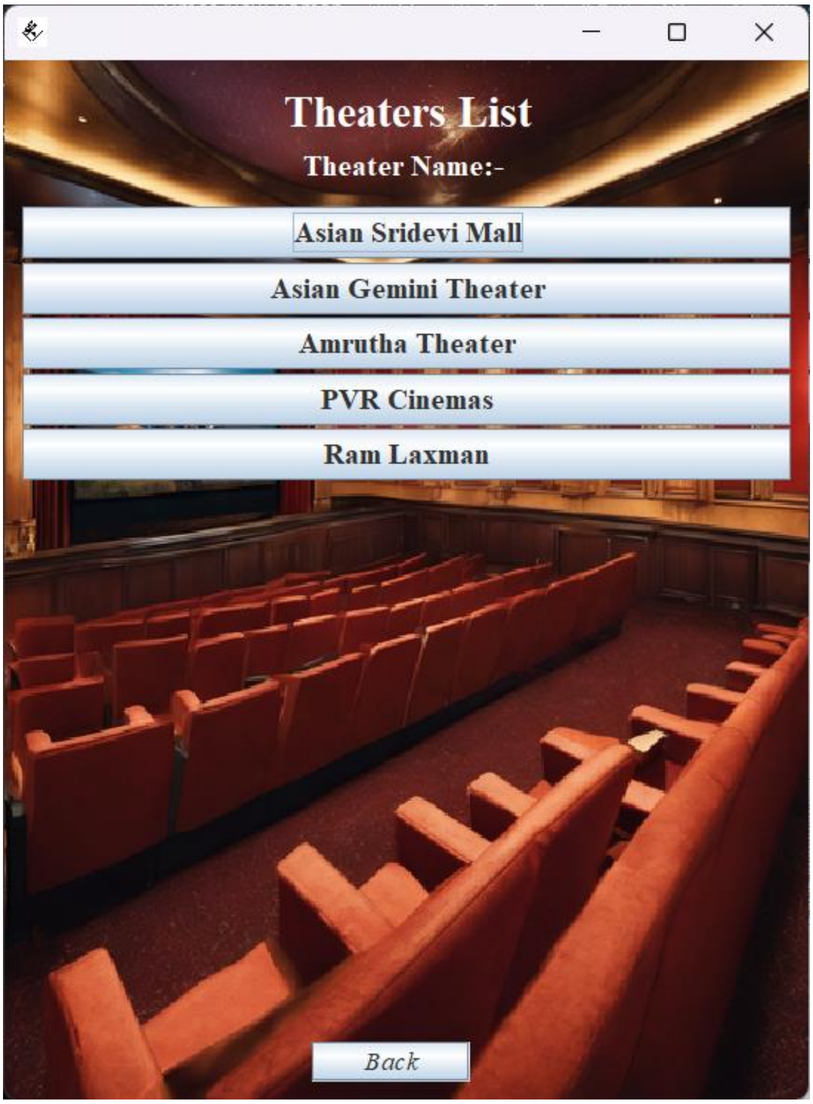
Password * :

Already had Account? Then Login

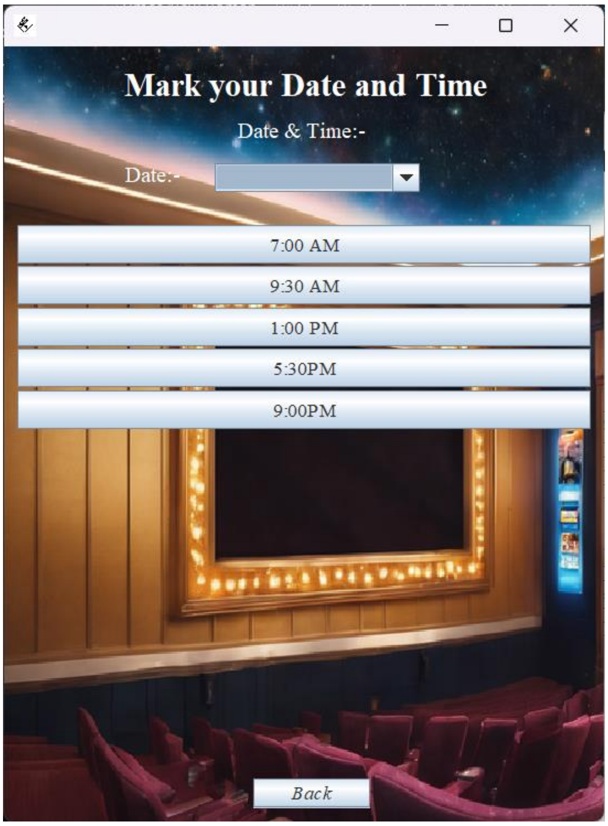
SCREEN 3:



SCREEN 4:



SCREEN 5:

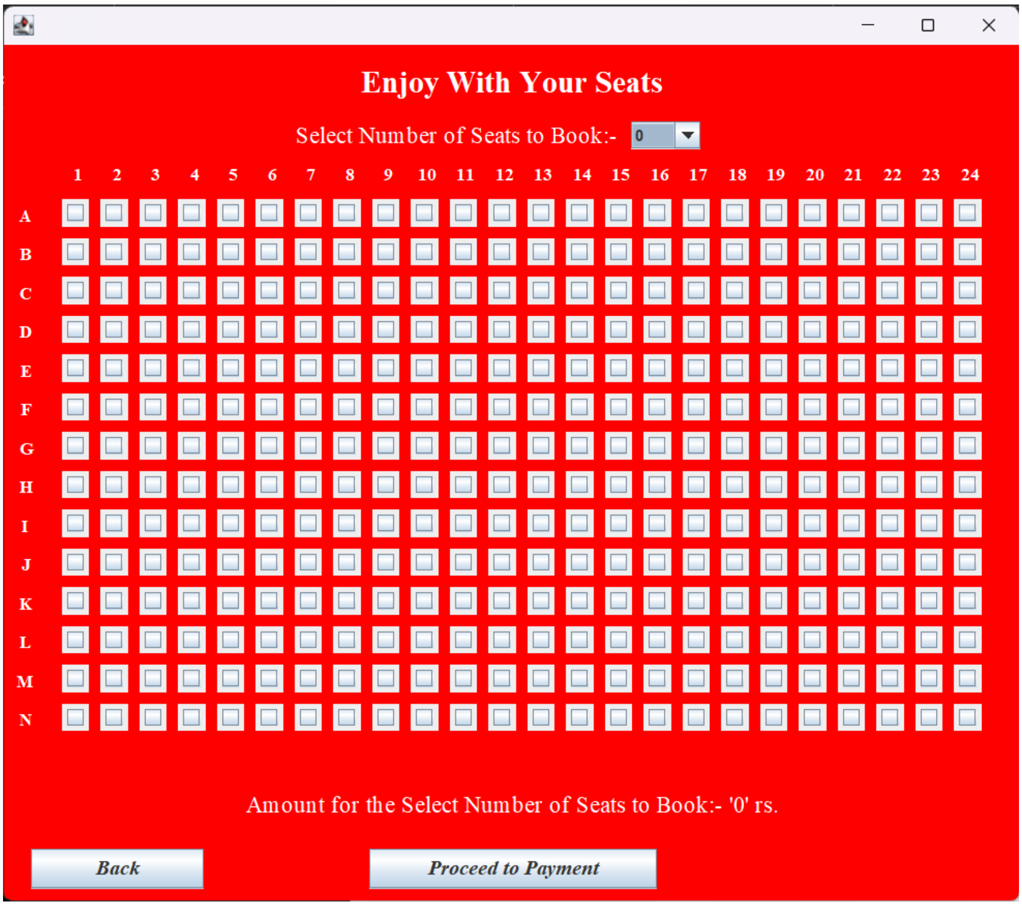


Mark your Date and Time

Date & Time:-

Date:-

SCREEN 6:



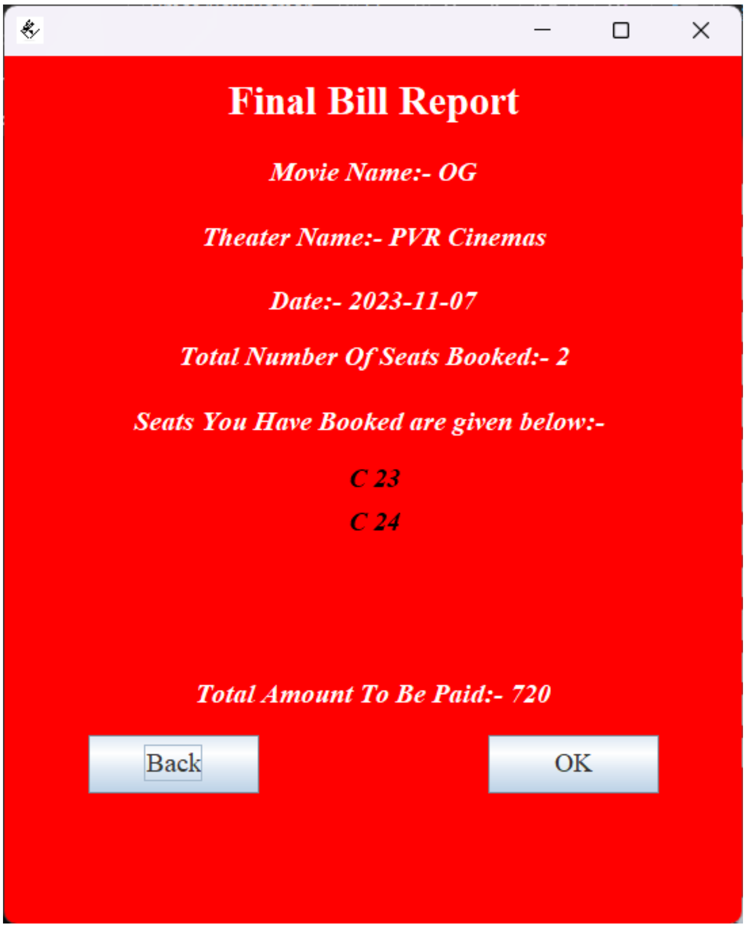
Enjoy With Your Seats

Select Number of Seats to Book:- 0

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
B	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
C	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
D	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
E	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
F	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
G	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
H	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
I	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
J	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
K	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
L	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
M	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
N	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□

Amount for the Select Number of Seats to Book:- '0' rs.

SCREEN 7:



Final Bill Report

Movie Name:- OG

Theater Name:- PVR Cinemas

Date:- 2023-11-07

Total Number Of Seats Booked:- 2

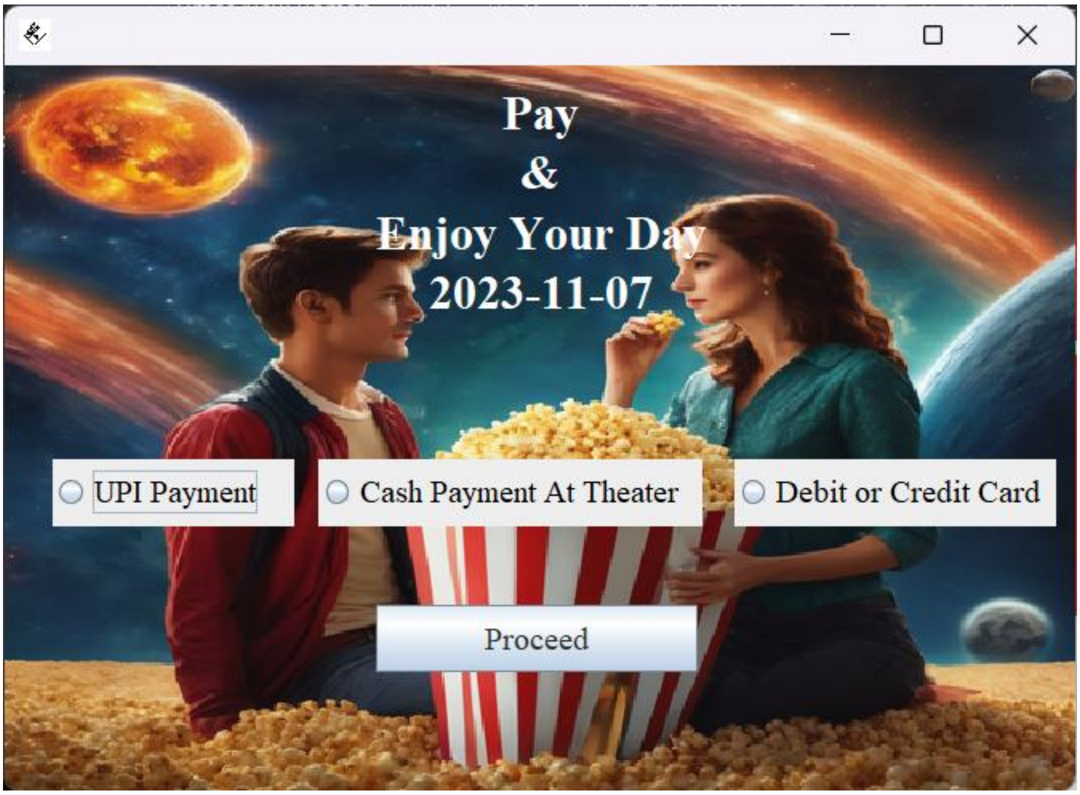
Seats You Have Booked are given below:-

C 23

C 24

Total Amount To Be Paid:- 720

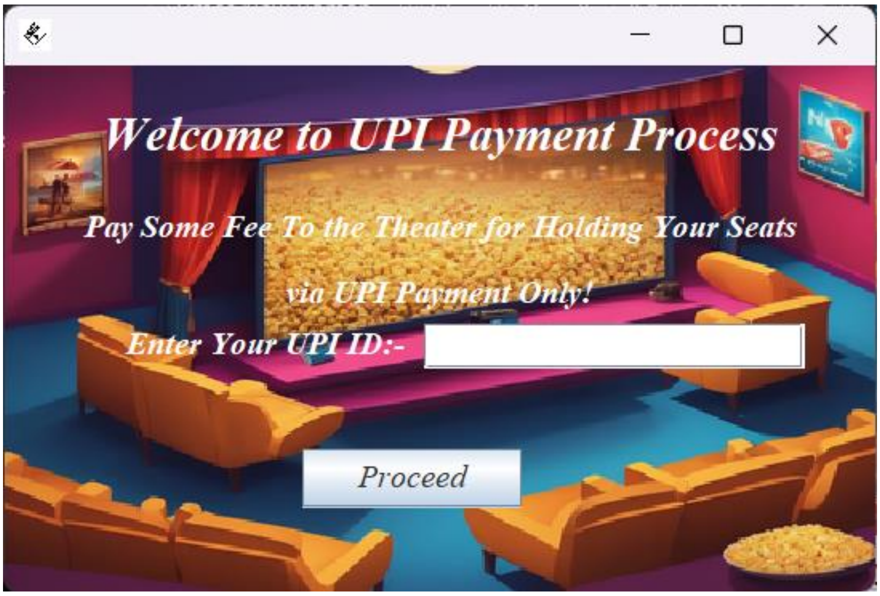
SCREEN 8:



Pay
&
Enjoy Your Day
2023-11-07

○ UPI Payment ○ Cash Payment At Theater ○ Debit or Credit Card

SCREEN 9:



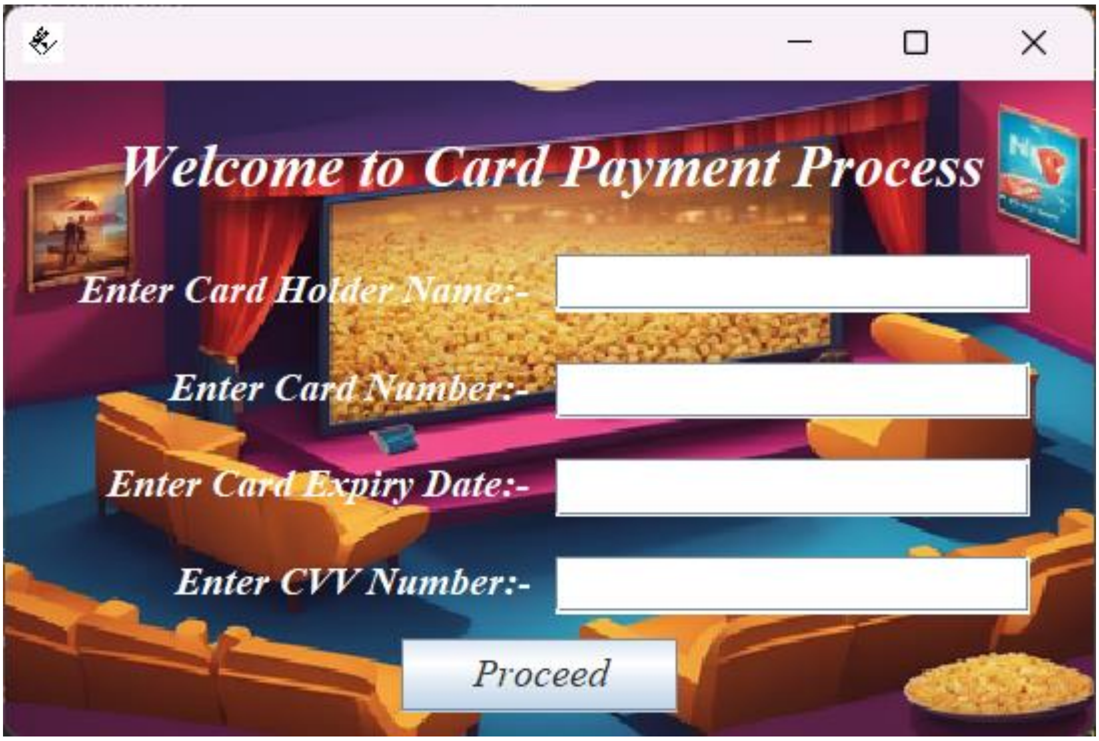
Welcome to UPI Payment Process

Pay Some Fee To the Theater for Holding Your Seats

via UPI Payment Only!

Enter Your UPI ID:-

SCREEN 10:



Welcome to Card Payment Process

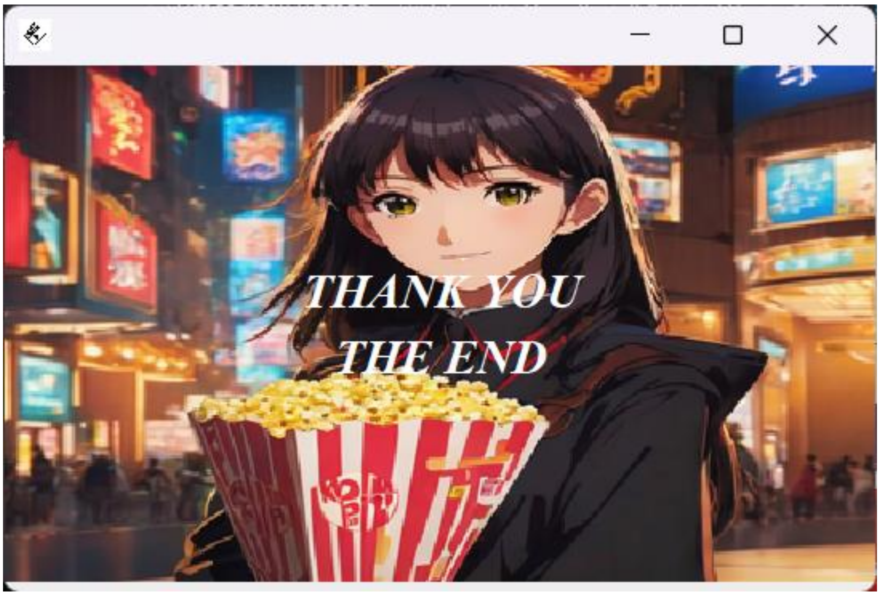
Enter Card Holder Name:-

Enter Card Number:-

Enter Card Expiry Date:-

Enter CVV Number:-

SCREEN 11:



CHAPTER 6

CONCLUSION

Finally, the creation of the “**ONLINE MOVIE TICKET BOOKING SYSTEM”**, as well as the Java Swing framework and Eclipse software, has resulted in a complex and user-friendly platform that significantly improves the moviegoing experience. This project exemplifies the key ideas of OOPC, such as encapsulation, inheritance, and polymorphism, which have been critical in developing a well-organized and maintainable project.

The usage of Java Swing for user interface design produced a visually beautiful and intuitive platform. Customers can easily browse movie offerings, pick showtimes, and purchase tickets, making it a simple and quick experience for movie lovers. Swing's components and layout managers were used to create a responsive and interactive user interface that allows users to easily traverse and make their decisions.

The ultimate result is a system that demonstrates the power of OOPC in Java, Java Swing, and Eclipse in building a safe, user-friendly, and highly extensible online movie ticket purchasing platform. It simplifies the ticket ordering process, lays the groundwork for future maintenance and enhancements, and ultimately adds to a better moviegoing experience for users.