**MOVIE TICKET BOOKING SYSTEM**

MINOR PROJECT REPORT

By

**Shalok Gupta (RA2211003010727)**

**Samarth Agarwal (RA2211003010727)**

Under the guidance of   
**Dr. S. Ashwini***In partial fulfilment for the Course*

of

**21CSC206P – ADVANCED OBJECT-ORIENTED AND PROGRAMMING**

in in C-tech



**FACULTY OF ENGINEERING AND TECHNOLOGY**

**SCHOOL OF COMPUTING**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**KATTANKULATHUR**

**NOVEMBER 2023**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**(Under Section 3 of UGC Act, 1956)**

**BONAFIDE CERTIFICATE**

Certified that this minor project report for the course **21CSC206P** **ADVANCED OBJECT ORIENTED AND PROGRAMMING** entitled in " **MOVIE TICKET BOOKING SYSTEM** " is the bonafide work of **Shalok Gupta (RA2211003010727) and Samarth Agarwal (RA2211003010727)** who carried out the work under my supervision.

# SIGNATURE

**Dr. S. Ashwini**

**Assistant Professor**

**C-tech**

SRM Institute of Science and Technology

Kattankulathur

# ABSTRACT

Web-based Movie Ticket Booking/ Reservation System for cinema halls and multiplex movie theaters which can be accessed over the internet. Presently most movie lovers book their tickets via Paytm or Bookmyshow mobile apps and websites. This web project will automate the reservation of movie tickets and inquiries about the availability of seats. This online java based application includes email confirmation for the movie tickets with seat numbers and movie time. This project was developed using Java & MySQL & Java Swing.

# ACKNOWLEDGEMENT

We express our heartfelt thanks to our honorable **Vice Chancellor Dr. C. Muthamizhchelvan**, for being the beacon in all our endeavors.

We would like to express my warmth of gratitude to our **Registrar Dr. S. Ponnusamy,** for his encouragement.

We express our profound gratitude to our **Dean (College of Engineering and Technology) Dr. T. V. Gopal,** for bringing out novelty in all executions.

We would like to express my heartfelt thanks to Chairperson, School of Computing **Dr. Revathi Venkataraman,** for imparting confidence to complete my course project

We wish to express my sincere thanks to **Course Audit Professors Dr. Vadivu. G, Professor, Department of Data Science and Business Systems and Dr. Sasikala. E Professor, Department of Data Science and Business Systems** and **Course Coordinators** for their constant encouragement and support.

We are highly thankful to my Course project Faculty **Dr. S. Ashwini**

**Assistant Professor, C-tech**for his/herassistance, timely suggestion and guidance throughout this course project.

We extend my gratitude to our **HoD Dr. M Lakshmi, Professor, Department of Data Science and Business Systems** and my Departmental colleagues for their Support.

Finally, we thank our parents and friends near and dear ones who directly and indirectly contributed to the successful completion of our project. Above all, I thank the almighty for showering his blessings on me to complete my Course project.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **CONTENTS** | **PAGE NO** |
| **1** | **INTRODUCTION** |  |
|  | 1.1 Motivation |  |
|  | 1.2 Objective |  |
|  | 1.3 Problem Statement |  |
|  | 1.4 Challenges |  |
| **2** | **LITERATURE SURVEY** |  |
| **3** | **REQUIREMENT ANALYSIS** |  |
| **4** | **ARCHITECTURE & DESIGN** |  |
| **5** | **IMPLEMENTATION** |  |
| **6** | **EXPERIMENT RESULTS & ANALYSIS** |  |
| **7** | **CONCLUSION** |  |
| **8** | **REFERENCES** |  |

1. **INTRODUCTION**

A movie ticket booking system is a software application or platform that allows customers to reserve and purchase tickets for movies showing in theaters. It simplifies the process of planning a trip to the cinema by providing an online or offline interface for browsing available movies, showtimes, theater locations, and seats, and then making reservations or purchases.

Here is an introduction to the key components and features of a movie ticket booking system:

* **Movie Listings**: The system displays a list of movies currently playing in theaters. It includes details like movie title, poster, genre, rating, and a brief synopsis.
* **Showtimes**: Customers can choose from a range of showtimes for a particular movie. These showtimes may vary based on factors like time of day, day of the week, and theater location.
* **Theater Locations**: Customers can select a specific theater location from a list. Each location will offer a different set of showtimes for the selected movie.
* **Seat Selection**: Once the customer has chosen a movie, showtime, and theater, they can select the seats they want. The system typically displays a seating layout for each theater, showing which seats are available and which are already booked.
* **Pricing**: The system calculates the total ticket price based on the number of seats selected and the ticket price for each seat. Pricing may vary for adults, children, seniors, or other special categories.
* **Payment Processing**: Customers provide payment information to complete the booking. The system should support various payment methods such as credit cards, digital wallets, and cash.
* **Booking Confirmation**: After a successful payment, customers receive a booking confirmation that includes details like the movie title, showtime, theater location, seat numbers, and a unique booking reference number.
* **User Accounts**: Many movie ticket booking systems allow customers to create user accounts. This enables them to store personal information, view booking history, and receive notifications about upcoming movies and promotions.
* **Accessibility**: These systems often provide accessibility features, such as selecting wheelchair-accessible seats or offering subtitles for hearing-impaired customers.
* **Feedback and Reviews**: Some systems allow customers to provide feedback and reviews for movies, theaters, and their overall experience.

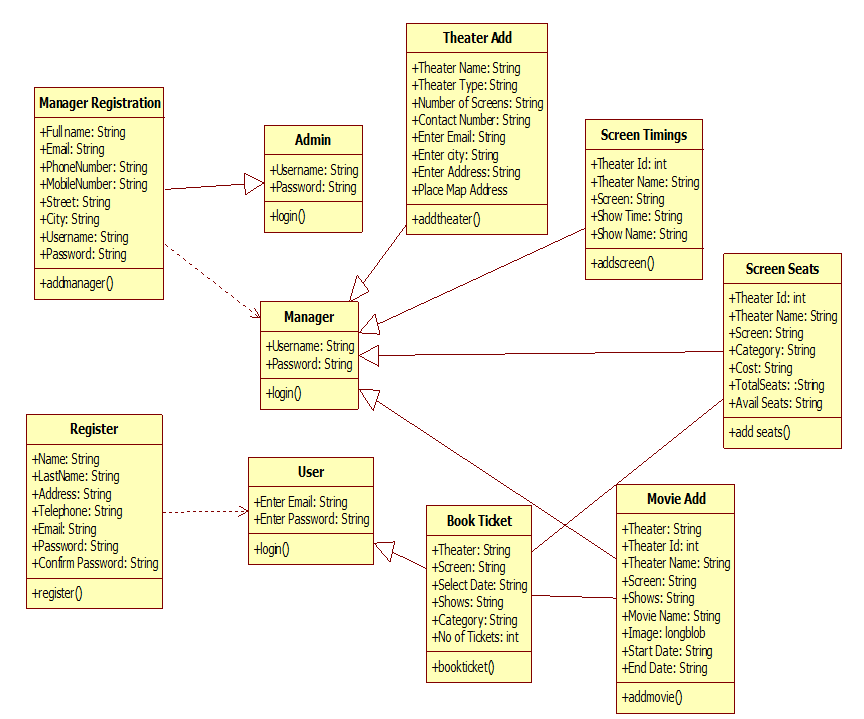
1. **LITERATURE SURVEY**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **AUTHORS** | **YEAR OF PUBLICATION** | **NAME** |
| 1. | Nielsen, N. V. | 2014 | E-commerce: Evolution or revolution in the fast-moving consumer goods world. |
| 2. | Rishab Jain, C., & Kaluri, R | 2015 | Design of automation scripts |
| 3. | Gangeshwer, D. K | 2013 | E-commerce or internet marketing: |
| 4. | Tjhin, V. U., Egeten, A. E. J., & Ciptasari, N. | 2017 | Effects of e-marketing and consumer lifestyle towards the style of decision making in online purchase of movie ticket. |
| 5. | Duan, W., Gu, B., & Whinston, A. B. | 2012 | Do online reviews matter?—An empirical investigation of panel data. |

1. **REQUIREMENTS** 
   1. **Requirement Analysis**

* **Java Development Kit (JDK):** You will need a JDK to write and compile Java code. You can download the latest version of the JDK from the Oracle website or use an open-source alternative like OpenJDK.
* **Integrated Development Environment (IDE):** An IDE makes development easier. Popular choices for Java development include Eclipse, IntelliJ IDEA, and NetBeans.
* **Database Management System (DBMS):** You will need a DBMS to store data related to movies, theaters, bookings, and customer information. Common choices include MySQL, PostgreSQL, Oracle, or SQLite. Java provides database connectivity through JDBC (Java Database Connectivity).

1. **ARCHITECTURE AND DESIGN**

****

1. **IMPLEMENTATION**

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.text.DecimalFormat;

import javax.swing.JButton;

import javax.swing.JComboBox;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JTextArea;

import javax.swing.SwingUtilities;

public class MovieTicketBookingSystem {

    private JFrame frame;

    private JComboBox<String> movieComboBox;

    private JComboBox<String> timeComboBox;

    private JComboBox<String> locationComboBox;

    private JPanel seatPanel;

    private JButton calculateButton;

    private JTextArea billTextArea;

    private String[] movies = { "Avengers", "Leo", "Jawan", "Tiger", "Annabelle" };

    private String[] times = { "10:00 AM", "1:00 PM", "5:00 PM", "8:00 PM" };

    private String[] locations = { "Anna Nagar", "Guindy", "Tambaram", "Vandalur", "OMR" };

    private double ticketPrice = 120.0;

    private int selectedSeats = 0;

    public MovieTicketBookingSystem() {

        frame = new JFrame("Movie Ticket Booking System");

        frame.setLayout(new GridLayout(2, 1));

        JPanel inputPanel = new JPanel();

        inputPanel.setLayout(new FlowLayout());

        movieComboBox = new JComboBox<>(movies);

        timeComboBox = new JComboBox<>(times);

        locationComboBox = new JComboBox<>(locations);

        inputPanel.add(new JLabel("Movie:"));

        inputPanel.add(movieComboBox);

        inputPanel.add(new JLabel("Time:"));

        inputPanel.add(timeComboBox);

        inputPanel.add(new JLabel("Location:"));

        inputPanel.add(locationComboBox);

        frame.add(inputPanel);

        seatPanel = new JPanel();

        seatPanel.setLayout(new GridLayout(5, 5));

        // Create seat buttons and add action listeners

        for (int i = 1; i <= 25; i++) {

            JButton seatButton = new JButton("Seat " + i);

            seatButton.addActionListener(new ActionListener() {

                @Override

                public void actionPerformed(ActionEvent e) {

                    JButton button = (JButton) e.getSource();

                    if (button.getBackground() == Color.GREEN) {

                        button.setBackground(null);

                        selectedSeats--;

                    } else {

                        button.setBackground(Color.GREEN);

                        selectedSeats++;

                    }

                }

            });

            seatPanel.add(seatButton);

        }

        frame.add(seatPanel);

        calculateButton = new JButton("Calculate");

        calculateButton.addActionListener(new ActionListener() {

            @Override

            public void actionPerformed(ActionEvent e) {

                calculateTotalPrice();

            }

        });

        billTextArea = new JTextArea(10, 40);

        billTextArea.setEditable(false);

        frame.add(calculateButton);

        frame.add(billTextArea);

        frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        frame.pack();

        frame.setVisible(true);

    }

    private void calculateTotalPrice() {

        double totalPrice = selectedSeats \* ticketPrice;

        String selectedMovie = movieComboBox.getSelectedItem().toString();

        String selectedTime = timeComboBox.getSelectedItem().toString();

        String selectedLocation = locationComboBox.getSelectedItem().toString();

        DecimalFormat df = new DecimalFormat("#.00");

        String bill = "Movie: " + selectedMovie + "\n" +

                "Time: " + selectedTime + "\n" +

                "Location: " + selectedLocation + "\n" +

                "Selected Seats: " + selectedSeats + "\n" +

                "Total Price: Rs." + df.format(totalPrice);

        billTextArea.setText(bill);

    }

    public static void main(String[] args) {

        SwingUtilities.invokeLater(new Runnable() {

            @Override

            public void run() {

                new MovieTicketBookingSystem();

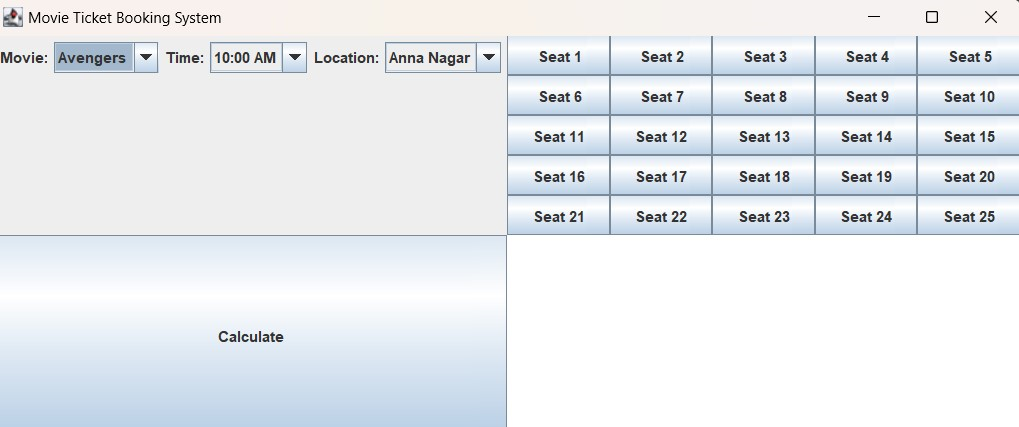
            }

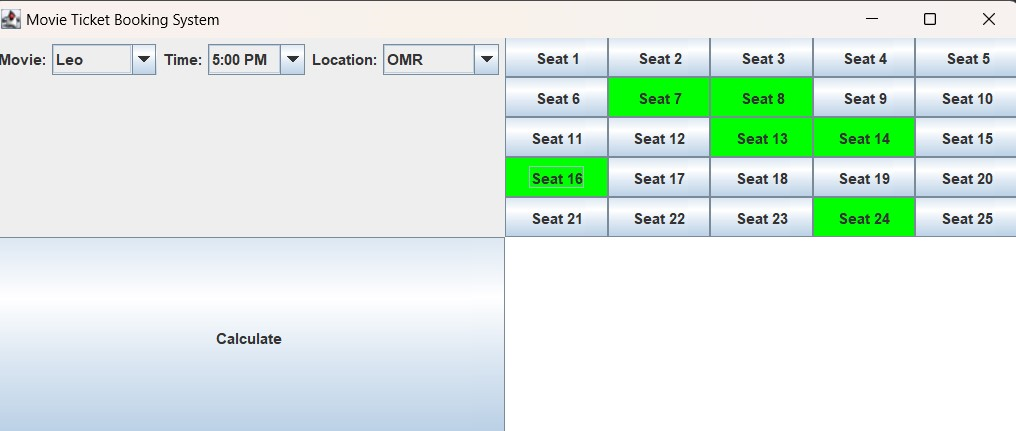
        });

    }

}

1. **RESULTS AND DISCUSSION**







1. **CONCLUSION**

In conclusion, a movie ticket booking system is a software application that facilitates the process of purchasing movie tickets. It typically includes features like movie selection, showtime options, seat selection, ticket price calculation, and billing. Such systems are widely used in the entertainment industry to streamline the ticket booking process for customers and theaters.

In the era of online ticket booking, a well-designed movie ticket booking system can greatly improve the user experience and streamline the ticket purchasing process for both customers and theaters.

**REFERENCES**

* [www.github.com](http://www.github.com)
* [www.geeksforgeeks.com](http://www.geeksforgeeks.com)
* [www.wikipedia.com](http://www.wikipedia.com)
* [www.researchgate.net](http://www.researchgate.net)