

MOVIE TICKET BOOKING SYSTEM

A MINI PROJECT REPORT

Submitted By

NITHEESH S

Reg.No.: 3122 22 3003 031

In the partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

BIOMEDICAL ENGINEERING



**Sri Sivasubramaniya Nadar College of Engineering
(An Autonomous Institution, Affiliated to Anna University)
Rajiv Gandhi Salai (OMR), Kalavakkam – 603 110**

JANUARY 2024

BONAFIDE CERTIFICATE

Certified that this project report “**MOVIE TICKET BOOKING SYSTEM**” is the bonafide work of "**NITHEESH S(3122223003031)**" who carried out the mini project work under my supervision.

Mr. SANDANA KARUPPAN A
B.E., M.E., (Ph.D.)

Submitted to the Viva voce Examination held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

MOVIE TICKET BOOKING SYSTEM

INTRODUCTION:

The Movie Ticket Booking System is a Java-based application designed to streamline the process of booking movie tickets and providing essential details about available movies and upcoming releases. This system aims to offer users a convenient platform to reserve tickets for their preferred movies while accessing comprehensive information regarding showtimes, available seats, cast details, and more.

BACKGROUND:

In today's fast-paced world, the entertainment industry plays a pivotal role in people's lives, and cinema remains a popular choice for leisure and relaxation. However, the traditional process of purchasing movie tickets often involves long queues, uncertain seat availability, and limited access to comprehensive movie information. Therefore, there's a need for a user-friendly digital platform that simplifies the ticket booking process and offers real-time updates on movie screenings and related details.

OBJECTIVES:

The primary objective of this system is to bridge the gap between movie enthusiasts and the ticketing process by offering a user-friendly platform that simplifies movie ticket reservations and keeps users informed about the latest movie offerings.

1.Efficient Ticket Booking: Enable users to effortlessly book movie tickets online, providing a seamless and hassle-free experience.

2.Comprehensive Movie Information: Furnish users with essential details about available movies, including titles, genres, available seats, ticket prices, ratings, and cast information.

3.Upcoming Movies Preview: Showcase upcoming movie releases, allowing users to stay informed about upcoming films and plan their movie experiences in advance.

4.Enhanced User Experience: Design an intuitive interface to ensure ease of use, making the ticket booking process simple and enjoyable for users.

FUNCTIONAL REQUIREMENT:

1. Movie Details:

- View Available Movies: Users can view a list of available movies showing at the theater.
- Display Movie Details: Users can access detailed information about a selected movie, including title, genre, available seats, ticket price, cast, and rating.
- Display Upcoming Movies: Users can see a list of upcoming movies with their respective details.

2. Booking Tickets:

- Book Movie Tickets: Users can select a movie and book a specific number of seats for that movie.
- Seat Availability Check: The system should verify seat availability before confirming the booking.
- Calculate Total Bill: The system calculates the total bill amount considering ticket prices and selected snacks.

3. Snacks Ordering:

- Display Snacks Menu: Users can view a list of available snacks with their respective prices.
- Order Snacks: Users can order snacks by selecting from the menu.

4. User Interaction:

- Input Validation: Ensure proper validation for user inputs to prevent errors and maintain system integrity.
- Exit System: Provide an option for users to exit the system.

PROGRAM CLASS:

The main class `MovieTicketBookingSystem` orchestrates the system's operations. It includes functionalities such as displaying available movies, allowing users to select a movie, book tickets, order snacks, and calculate the total bill.

This class handles user interactions through the console (standard input and output) and utilizes instances of the `Theater`, `Movie`, and `Snacks` classes to manage movie-related information, theater details, and snack options. It provides a basic console-based interface for users to interact with the system, choose movies, book tickets, order snacks, and receive a total bill amount.

The provided class `MovieTicketBookingSystem` is responsible for implementing the functionalities of the movie ticket booking system and handling user interactions within the console environment.

SOURCE CODE:

```
import java.util.Scanner;
```

```
class Movie {
```

```
    private String title;
```

```
    private String genre;
```

```
    private int availableSeats;
```

```
    private double ticketPrice;
```

```
    private String[] cast;
```

```
    private double rating;
```

```
    public Movie(String title, String genre, int availableSeats, double ticketPrice,  
String[] cast, double rating) {
```

```
    this.title = title;

    this.genre = genre;

    this.availableSeats = availableSeats;

    this.ticketPrice = ticketPrice;

    this.cast = cast;

    this.rating = rating;
}
```

```
public String getTitle() {
    return title;
}
```

```
public String getGenre() {
    return genre;
}
```

```
public int getAvailableSeats() {
    return availableSeats;
}
```

```
public double getTicketPrice() {
    return ticketPrice;
}
```

```
public String[] getCast() {  
    return cast;  
}
```

```
public double getRating() {  
    return rating;  
}
```

```
public void bookTickets(int numTickets) {  
    if (numTickets <= availableSeats) {  
        availableSeats -= numTickets;  
        System.out.println(numTickets + " ticket(s) booked for " + title);  
    } else {  
        System.out.println("Sorry, not enough seats available for " + title);  
    }  
}
```

```
public void displayMovieDetails() {  
    System.out.println("Title: " + title);  
    System.out.println("Genre: " + genre);  
    System.out.println("Available Seats: " + availableSeats);  
    System.out.println("Ticket Price: Rs." + ticketPrice);  
}
```

```
System.out.println("Rating: " + rating);

System.out.println("Cast:");

for (String actor : cast) {

    System.out.println("- " + actor);

}

}
```

```
public static void displayUpcomingMovies(Movie[] upcomingMovies) {

    System.out.println("\nUpcoming Movies:");

    for (int i = 0; i < upcomingMovies.length; i++) {

        System.out.println((i + 1) + ". Title: " + upcomingMovies[i].getTitle() + " -
Genre: " +

            upcomingMovies[i].getGenre() + ", Available Seats: " +
upcomingMovies[i].getAvailableSeats() +

            ", Ticket Price: Rs." + upcomingMovies[i].getTicketPrice() + ", Rating: "
+ upcomingMovies[i].getRating());

        System.out.println("  Cast: ");

        for (String actor : upcomingMovies[i].getCast()) {

            System.out.println("    - " + actor);

        }

    }

}
```



```
}
```

```
class Snacks {  
    private String name;  
    private double price;  
  
    public Snacks(String name, double price) {  
        this.name = name;  
        this.price = price;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public double getPrice() {  
        return price;  
    }  
}
```

```
class Theater {  
    public String theaterName;
```

```

public String location;

public Movie[] movies;

public Snacks[] snacks;

public Movie[] upcomingMovies;

public Theater(String theaterName, String location) {

    this.theaterName = theaterName;

    this.location = location;

    movies = new Movie[5];

    movies[0] = new Movie("AYALAAN", "Sci-Fantasy", 200, 150, new
String[]{"Sivakarhikeyan", "Rakul Preeth Singh", "Yogi babu"}, 9.2);

    movies[1] = new Movie("CAPTAIN MILLER", "Periodic-Action", 180, 150,
new String[]{"Dhanush", "Priyanka Mohan", "Sundeep Kishan"}, 8.5);

    movies[2] = new Movie("LAL SALAAM", "Sports-Drama", 150, 150, new
String[]{"Rajnikanth", "Vishnu Vishaal", "Vikraanth"}, 8.0);

    movies[3] = new Movie("MERRY CHRISTMAS", "Action-Thriller", 120, 150,
new String[]{"Vijay Sethupathi", "Katrina Kaif", "Radhika Apte"}, 7.7);

    movies[4] = new Movie("THANGALAAN", "Historic", 120,150, new
String[]{"Vikram", "Parvathy ", "Malavika Mohanan"},8.9);


    snacks = new Snacks[8];

    snacks[0] = new Snacks(" Butter Popcorn", 70);

    snacks[1] = new Snacks("Coke", 80);

    snacks[2] = new Snacks("Pepsi", 80);

    snacks[3] = new Snacks("Fries", 60);

    snacks[4] = new Snacks("Naachos", 90);

    snacks[5] = new Snacks("Ice Cream", 40);

```

```

        snacks[6] = new Snacks("Veg Puffs", 50);

        snacks[7] = new Snacks("Chicken Puffs", 80);


        upcomingMovies = new Movie[6];

        upcomingMovies[0] = new Movie("Vidaamuyarchi", "Political-Action", 0, 0, new
String[]{"Ajith Kumar", "Trisha"}, 0);

        upcomingMovies[1] = new Movie("Demonte Colony-2", "Horror", 0, 0, new
String[]{"Arulnithi", "Priya Bhavani Shankar"}, 0);

        upcomingMovies[2] = new Movie("Brother", "Family Drama", 0, 0, new
String[]{"Jayam Ravi", "Priyanka Mohan"}, 0);

        upcomingMovies[3] = new Movie("Kanguva", "Periodic Action", 0, 0,
new String[]{"Surya", "Disha Patni"}, 0);

        upcomingMovies[4] = new Movie("Dhurva Natchatiram", "High Action
Thriller", 0, 0, new String[]{"Vikram", "Ritu Varma"}, 0);

        upcomingMovies[5] = new Movie("Thug Life", "Periodic Action", 0, 0,
new String[]{"Kamal Hassan", "Trisha"}, 0);

    }

```

```

public void displayMoviesAndTheaterDetails() {

    System.out.println("Theater: " + theaterName + " - Location: " + location);

    System.out.println("\nAvailable Movies:");

    for (int i = 0; i < movies.length; i++) {

        System.out.println((i + 1) + ". Title: " + movies[i].getTitle() + " - Genre: " +

            movies[i].getGenre() + ", Available Seats: " +
movies[i].getAvailableSeats() +

```

```
        ", Ticket Price: Rs." + movies[i].getTicketPrice() + ", Rating: " +  
movies[i].getRating());
```

```
        System.out.println("  Cast: ");
```

```
        for (String actor : movies[i].getCast()) {
```

```
            System.out.println("    - " + actor);
```

```
        }
```

```
    }
```

```
}
```

```
public double calculateSnacksBill(int[] snackChoices) {
```

```
    double totalSnacksBill = 0.0;
```

```
    for (int choice : snackChoices) {
```

```
        if (choice > 0 && choice <= snacks.length) {
```

```
            totalSnacksBill += snacks[choice - 1].getPrice();
```

```
        }
```

```
    }
```

```
    return totalSnacksBill;
```

```
}
```

```
    public void bookMovieTickets(int movieIndex, int numTickets) {
```

```
        if (movieIndex >= 0 && movieIndex < movies.length) {
```

```
            movies[movieIndex].bookTickets(numTickets);
```

```
        } else {
```

```
            System.out.println("Invalid movie selection!");
```

```
        }
```

```
}  
}
```

```
public class MovieTicketBookingSystem{
```

```
    public static void main(String[] args) {
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        Theater theater = new Theater("Rohini Silver Screens", "Chennai");
```

```
        boolean bookingActive = true;
```

```
        while (bookingActive) {
```

```
            System.out.println("\nWelcome to Movie Ticket Booking System");
```

```
            System.out.println("Theater: " + theater.theaterName + " - Location: " +  
theater.location);
```

```
            for (int i = 0; i < theater.movies.length; i++) {
```

```
                // Display available movies
```

```
                System.out.println((i + 1) + ". " + theater.movies[i].getTitle());
```

```
            }
```

```
            System.out.print("\nEnter the movie number to view details (1,2,3,4,5) or 0 to  
exit: ");
```

```
int movieChoice = scanner.nextInt();

if (movieChoice == 0) {

    bookingActive = false;

    System.out.println("Thank you for using the Movie Ticket Booking
System!");

    break;

}

int movieIndex = movieChoice - 1;

if (movieIndex >= 0 && movieIndex < theater.movies.length) {

    Movie selectedMovie = theater.movies[movieIndex];

    System.out.println("\nMovie Details for " + selectedMovie.getTitle() + ":");

    System.out.println("Title: " + selectedMovie.getTitle());

    System.out.println("Genre: " + selectedMovie.getGenre());

    System.out.println("Available Seats: " + selectedMovie.getAvailableSeats());

    System.out.println("Ticket Price: Rs." + selectedMovie.getTicketPrice());

    System.out.println("Rating: " + selectedMovie.getRating());

    System.out.println("Cast:");

    for (String actor : selectedMovie.getCast()) {

        System.out.println("- " + actor);

    }

    System.out.print("\nEnter the number of seats to book: ");
```

```
int numSeatsToBook = scanner.nextInt();

if (numSeatsToBook <= selectedMovie.getAvailableSeats()) {
    selectedMovie.bookTickets(numSeatsToBook);

    System.out.println("\nSnacks Menu:");

    for (int i = 0; i < theater.snacks.length; i++) {
        System.out.println((i + 1) + ". " + theater.snacks[i].getName() +
            " - Rs." + theater.snacks[i].getPrice());
    }

    System.out.println("Enter the snack numbers to order (separated by
spaces) or 0 to skip: ");

    scanner.nextLine(); // Consume the newline character

    String snackChoicesInput = scanner.nextLine();

    String[] snackChoicesStr = snackChoicesInput.split(" ");

    int[] snackChoices = new int[snackChoicesStr.length];

    for (int i = 0; i < snackChoicesStr.length; i++) {
        snackChoices[i] = Integer.parseInt(snackChoicesStr[i]);
    }

    double snacksBillAmount = theater.calculateSnacksBill(snackChoices);

    double ticketsBillAmount = selectedMovie.getTicketPrice() *
numSeatsToBook;

    double totalBill = snacksBillAmount + ticketsBillAmount;
```

```
        System.out.println("Total Snacks Bill: Rs." + snacksBillAmount);

        System.out.println("Total Ticket Price: Rs." + ticketsBillAmount);

        System.out.println("Your total bill amount is: Rs." + totalBill);

        Movie.displayUpcomingMovies(theater.upcomingMovies);

        System.out.println("*****Enjoy your Movie
with Snacks.*****");

        System.out.println("*****Thank you for Booking your Movie tickets
and Snacks.*****");

    } else {

        System.out.println("Sorry, not enough seats available for " +
selectedMovie.getTitle());

    }

    } else {

        System.out.println("Invalid movie selection!")

    }

}

}

scanner.close();

}

}
```


ALGORITHM:

1. Initialize the Theater, Movies, and Snacks:

- Create a Theater instance with theater name and location.
- Initialize movie details and upcoming movies with their attributes.
- Define various snack options available at the theater.

2. Display Welcome Message:

- Print a welcome message to the user.

3. Display Available Movies:

- Show the list of available movies with their titles.

4. Prompt User to Select a Movie:

- Ask the user to input the number associated with their desired movie or 0 to exit.

5. If User Chooses a Movie:

a. Display Movie Details:

- Show detailed information about the selected movie: title, genre, available seats, ticket price, rating, and cast.

b. Prompt User to Enter Number of Seats to Book:

- Take input from the user for the number of seats they want to book.

6. Book Tickets:

a. Check Seat Availability:

- Verify if the requested number of seats is available.

b. If Seats Available:

- Book the requested number of seats for the selected movie.

- Display the Snacks Menu.
- Prompt User to Order Snacks:
 - Display snack options and take user input for snack choices.
- Calculate Total Bill:
 - Calculate the total bill amount, including ticket prices and snacks ordered.
- Display Total Bill Amount.

7. Display Upcoming Movies:

- Show a list of upcoming movies with their details.

8. Exit or Continue:

- Provide an option for the user to exit the system or continue booking tickets.

9. Loop Back or Exit:

- If the user chooses to continue, loop back to Step 3.
- If the user decides to exit, display a thank you message and end the program.

10. End Algorithm.

OUTPUT:

1. Giving sample input as 1 , it displays the details of that film. Number of tickets to be booked is done here

```
PS D:\jprg> java MovieTicketBookingSystem

Welcome to Movie Ticket Booking System
Theater: Rohini Silver Screens - Location: Chennai
1. AYALAAN
2. CAPTAIN MILLER
3. LAL SALAAM
4. MERRY CHRISTMAS
5. THANGALAAN

Enter the movie number to view details (1,2,3,4,5) or 0 to exit: 1

Movie Details for AYALAAN:
Title: AYALAAN
Genre: Sci-Fantasy
Available Seats: 200
Ticket Price: Rs.150.0
Rating: 9.2
Cast:
- Sivakarthikeyan
- Rakul Preeth Singh
- Yogi babu

Enter the number of seats to book: 15
15 ticket(s) booked for AYALAAN
```

- 2.To choose snacks from the menu given.

```
Snacks Menu:
1. Butter Popcorn - Rs.70.0
2. Coke - Rs.80.0
3. Pepsi - Rs.80.0
4. Fries - Rs.60.0
5. Naachos - Rs.90.0
6. Ice Cream - Rs.40.0
7. Veg Puffs - Rs.50.0
8. Chicken Puffs - Rs.80.0
Enter the snack numbers to order (separated by spaces) or 0 to skip:
1 2 4 4 7 2 3
```

- 3.It displays the ticket amount, snacks amount and total amount.

```
Total Snacks Bill: Rs.480.0
Total Ticket Price: Rs.2250.0
Your total bill amount is: Rs.2730.0
```

4.It displays the preview of the upcoming movies.

```
Upcoming Movies:
1. Title: Vidaamuyarchi - Genre: Political-Action, Available Seats: 0, Ticket Price: Rs.0.0, Rating: 0.0
   Cast:
   - Ajith Kumar
   - Trisha
2. Title: Demonte Colony-2 - Genre: Horror, Available Seats: 0, Ticket Price: Rs.0.0, Rating: 0.0
   Cast:
   - Arulnithi
   - Priya Bhavani Shankar
3. Title: Brother - Genre: Family Drama, Available Seats: 0, Ticket Price: Rs.0.0, Rating: 0.0
   Cast:
   - Jayam Ravi
   - Priyanka Mohan
4. Title: Manguva - Genre: Periodic Action, Available Seats: 0, Ticket Price: Rs.0.0, Rating: 0.0
   Cast:
   - Surya
   - Disha Patni
5. Title: Dhurva Natchatiram - Genre: High Action Thriller, Available Seats: 0, Ticket Price: Rs.0.0, Rating: 0.0
   Cast:
   - Vikram
   - Ritu Varma
6. Title: Thug Life - Genre: Periodic Action, Available Seats: 0, Ticket Price: Rs.0.0, Rating: 0.0
   Cast:
   - Kamal Hassan
   - Trisha
*****Enjoy your Movie with Snacks.*****
*****Thank you for Booking your Movie tickets and Snacks.*****
```

5.if we didn't want to book tickets , it shows Thank You.

```
PS D:\jprg> javac MovieTicketBookingSystem.java
PS D:\jprg> java MovieTicketBookingSystem

Welcome to Movie Ticket Booking System
Theater: Rohini Silver Screens - Location: Chennai
1. AYALAAAN
2. CAPTAIN MILLER
3. LAL SALAAM
4. MERRY CHRISTMAS
5. THANGALAAAN

Enter the movie number to view details (1,2,3,4,5) or 0 to exit: 0
Thank you for using the Movie Ticket Booking System!
```

CONCLUSION:

The system allows users to view available movies, access detailed information about each movie, book tickets based on seat availability, order snacks, and view upcoming movie releases. It engages users through console-based inputs and outputs, providing an intuitive interface to navigate through available options and book tickets accordingly. The system performs fundamental operations such as seat availability checks, calculating total bills incorporating ticket prices and snack orders, and displaying relevant movie and snack information.

