BOOK MY CINEMA TICKET

A
Mini Project Report Submitted in Partial fulfillment of the Requirement for the Award of the Degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

S.VISHNAVI (229Y1A05G3)

S.ARCHANA (229Y1A05G9)

V.PAVAN (229Y1A05I5)

S.SAJITH (229Y1A05F7)

Under the Guidance of

Dr. V. VENKATA RAMANA M.Tech., Ph.D.

Associate Professor, Dept. of CSE

To

Department of Computer Science and Engineering

K.S.R.M COLLEGE OF ENGINEERING
(UGC - AUTONOMOUS)
Approved by AICTE, New Delhi & Affiliated to JNTUA, Ananthapuramu
(Accredited by NAAC with A+ Grade & B.Tech.(Civil, EEE,Mech,ECE and CSE) Programs by NBA)
KADAPA – 516 005 (A.P.)

2024- 2025

K.S.R.M COLLEGE OF ENGINEERING

(UGC - AUTONOMOUS)

Approved by AICTE, New Delhi & Affiliated to JNTUA, Ananthapuramu

(Accredited by NAAC with A+ Grade & B.Tech.(Civil, EEE,Mech,ECE and CSE) Programs by NBA)

KADAPA – 516 005 (A.P.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION & MISSION

VISION:

To evolve as a recognized center of excellence in the area of Computer Science and Engineering and other related inter-disciplinary fields.

MISSION:

M1:: To produce competent and industry ready professionals through well balanced curriculum and innovative pedagogy.

M2::To provide conducive environment for research by establishing centre of excellence and industry collaborations.

M3:: To instill leadership qualities, ethical values among students through various co-curricular and extracurricular activities.

B. TECH. (COMPUTER SCIENCE AND ENGINEERING)

PROGRAM EDUCATIONAL OBJECTIVES

B.Tech - Computer Science and Engineering Program Objectives.

A graduate of the K.S.R.M.C.E, C.S.E should have a successful career in CSE or a related field, and within three to five years, should

PEO1: To excel in their career as competent software engineer in IT and allied organizations.

PEO2: To pursue higher education and to demonstrate research temper for providing solutions to engineering problems.

PEO3: To contribute for the societal development by exhibiting leadership, through professional, social and ethical values.

PROGRAM OUTCOMES

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/Development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environment.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

PSOs are statements that describe what the graduates of a specific engineering program should be able to do:

PSO1: Professional Skills: The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.

PSO2: Problem-Solving Skills: The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.

PSO3: Successful Career and Entrepreneurship: The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies.

COURSE OUTCOMES

CO 1 : Design static web pages using HTML and CSS

CO 2 : Design and Develop Webpages Using Javascript

CO 3 : Design web pages to authenticate users using Cookies.

CO 4 : Design web pages using HTML, CSS and Angular JS

CO 5 : Develop server side programs using PHP and accessing database through PHP.

CO-PO MAPPING

Course Outcome	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	3	3		3	3		3	3	3		3	3	3	
CO2	3	3	3		3	3		3	3	3		3	3	3	
CO3															
CO4	3	3	3		3	3		3	3	3		3	3	3	
CO5	3	3	3		3	3		3	3	3		3	3	3	

K.S.R.M COLLEGE OF ENGINEERING

(UGC - AUTONOMOUS)

Approved by AICTE, New Delhi & Affiliated to JNTUA, Ananthapuramu

(Accredited by NAAC with A+ Grade & B.Tech.(Civil, EEE,Mech,ECE and CSE) Programs by NBA)

KADAPA – 516 005 (A.P.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certified that the project report entitled " BOOK MY CINEMA TICKET " is being submitted by **S.VISHNAVI (229Y1A05G3) , S.ARCHANA (229Y1A05G9), V.PAVAN (229Y1A05I5) , S.SAJITH (229Y1AO5F7)** to K.S.R.M. College of Engineering (UGC - AUTONOMOUS), Kadapa in partial fulfillment of the requirements for the award of the degree of "BACHELOR OF TECHNOLOGY" in "COMPUTER SCIENCE AND ENGINEERING" is a bonafide record of the project work carried out by them under our supervision during the period 2023-2024.

Project Guide

Dr. V VENKATA RAMANA MTech., Ph.D.

Project Co-ordinator

Dr.V Venkata Ramana M.Tech., Ph.D.

Associate Professor Dept. of CSE

Head of the Department

Sri.A. Ram Prakash Reddy, M.Tech,(Ph.D).

Assistant Professor&HOD OF CSE

Date: **Internal Examiner** **External Examiner**

DECLARATION BY THE CANDIDATE

We VISHNAVI , ARCHANA, PAVAN, SAJITH bearing Roll No: 229Y1A05G3, 229Y1A05G9, 229Y1A05I5, 229Y1A05F7 hereby declare that the Project Report entitled **BOOK MY CINEMA TICKET** under the guidance of **Dr. V Venkata Ramana, M.Tech., Ph.D., Associate Professor**, **Department of CSE** is submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering.**

This is a Record of Bonafide work carried out by me and the results embodied in this Project Report have not been reproduced or copied from any source. The results embodied in this Project Report have not submitted to any other University or Institute for the Award of any other Degree or Diploma.

Signature of the Student

S.VISHNAVI

S.ARCHANA

V.PAVAN

S.SAJITH

ACKNOWLEDGEMENT

An endeavour over a long period can be successful only with the advice and supports of many well-wishers. We take this opportunity to express our gratitude and appreciation to all of them.

We are extremely thankful to our beloved Managing Director **Dr. K Chandra Obul Reddy garu** who took keen interest and encouraged us in every effort throughout this course.

We wish to express our deepest sense of gratitude and pay our sincere thanks to our guide **Dr. V VENKATA RAMANA M.Tech., Ph.D.** Department of Computer Science and Engineering, K.S.R.M. College of Engineering(A), Kadapa for his valuable guidance and suggestions in analysing and testing throughout the period, till the end of project work completion and his timely suggestions and help.

We are very much thankful to **A. Ram Prakash Reddy, M. Tech., Ph.D.,** Professor, Head of the Department for Computer Science and Engineering, K.S.R.M. College of Engineering(A), Kadapa for providing good facilities & congenial atmosphere in our college.

We take this opportunity to express our deep gratitude and appreciation to all those who encouraged us for successfully completion of this Project work. We wish to express our sincere to gratitude to **Dr. T NAGESWARA PRASAD**, **M.Tech., Ph.D**. Vice Principal of K.S.R.M. College of Engineering(A), Kadapa and **Dr. V.S.S. MURTHY, M.Tech., Ph.D**. Principal of K.S.R.M. College of Engineering(A), Kadapa and for their consistent help and encouragement to complete the project work.

We also thankful to all teaching and non-teaching staff of the **Department of Computer Science and Engineering** for their support throughout our B.Tech. course.

We express our heartfelt thanks to **My Parents** for their valuable support and encouragement in completion of my course. Also I express my heartfelt regards to my Friends for being supportive in completion of my project.

Project Associates

S.VISHNAVI (229Y1A05G3)

S.ARCHANA (229Y1A05G9)

V.PAVAN (229Y1A05I5)

S.SAJITH (229Y1A05F7)

TABLES AND CONTENTS

CHAPTER	PAGE NO
Abstract	1
List of Figures	2
CHAPTER 1	
INTRODUCTION	3-5
1.1Domain Description about the area of project	
1.2 Problem Definition	
1.3 Proposed Solution	
1.4 Objectives	
CHAPTER 2	
LITERATURE SURVEY	6-7
2.1. Introduction	
2.2 Related Work	
2.3. Overview	
CHAPTER 3	
SYSTEM ANALYSIS	8-10
3.1. Introduction	
3.2 Existing System & Disadvantages	
3.3 Proposed System &Advantages	
3.4 Software Requirements	
3.5 System Architecture	
CHAPTER 4	
MODULES	11-12
4.1 User Registration and Authentication	
4.2 Movie Catalog and Search	
4.3 Seat Selection	
4.4 Booking and Payment	
4.5 User Reviews and Ratings	
4.6 Admin Dashboard	

4.7 Ticket Cancellation and Refund

4.8 Analytics and Reporting

CHAPTER 5

SOURCECODE **13-30**

5.1 register.html

5.2 login.html

5.3 index.html

5.4seconpage.html

5.5thirdpage.html

5.6receipt.html

CHAPTER 6

RESULTS **31-34**

6.1 Introduction

6.2 Output screens

CHAPTER 7

Conculsion And Future Enhancements **35-36**

7.1 conclusion

7.2 Future Enhancements

References **37-38**

ABSTRACT

The Book My Cinema Ticket is a web-based application designed to simplify the process of booking movie tickets online. The system provides users with a seamless experience, allowing them to choose movies, theaters, showtimes, and seats, and complete the payment process—all from a single platform. The system is built using HTML, CSS, and JavaScript for the frontend, and PHP with MySQL for the backend.The process begins with a login page, where users can authenticate themselves before proceeding with the booking. After logging in, the user is directed to the booking page, where they can select their city from a list. Upon choosing the city, they can pick a theater, followed by selecting a movie and showtime. After choosing the showtime, users are presented with a seat selection page, where they can view and select available seats in the theater. Once the seats are selected, the user is directed to the payment page to enter payment details. The payment gateway securely processes the transaction, and once the payment is confirmed, a booking confirmation with all the details—such as movie name, theater, showtime, seat numbers, and transaction ID—is displayed and sent to the user via email.The system is powered by a MySQL database, which stores user data, movie schedules, theater information, and booking history. PHP is used to handle server-side logic, ensuring that data is securely processed and displayed. The frontend interface is designed to be user-friendly and responsive, ensuring compatibility across different devices. Overall, the Movie Ticket Booking System automates the ticket booking process, improving efficiency for both users and administrators while providing a secure and convenient way to book movie tickets online.

LIST OF FIGURES

S.NO	Figure No	Diagram	Page No
1	1.1	About Book My Cinema Ticket	3
2	3,5	System Architecture	10
3	6.1	Registeration Page	32
4	6.2	Login Page	32
5	6.3	Welcome To Movie Ticket Booking Page	33
6	6.4	Select Your Movie Details Page	33
7	6.5	Seat Booking page	34
8	6.6	Booking Receipt Page	34

CHAPTER 1

INTRODUCTION

The **Book My Cinema Ticket** is a web-based platform designed to help users easily browse and book movie tickets. Initially created using **frontend** technologies like **HTML**, **CSS**, and **JavaScript**, the system was later enhanced with a **backend** built with **PHP**, and **MYSQL** to handle dynamic data and improve performance.

The frontend provides a user-friendly interface to select movies, showtimes, and seats, while the backend ensures real-time management of movie data, user authentication, and seat booking. The integration of the backend allows for features like:

- **Dynamic movie and seat availability**: Real-time updates of movie listings, showtimes, and seat statuses.
- **User authentication**: Secure login and registration.
- **Persistent data storage**: All bookings and user details are stored securely in a database.

This system offers an efficient and scalable solution for booking movie tickets, enhancing both the user experience and backend performance.

Fig 1.1 About Book My Cinema Ticket

1.1 Domain Description about the Area of the Project

The domain of this project is the entertainment industry, specifically focused on movie ticket booking. The rise of online booking systems has made traditional ticket purchasing obsolete. With the increasing popularity of online platforms, the demand for easy, accessible, and user-friendly systems to book movie tickets has surged. The project leverages web technologies (HTML, CSS, JavaScript) to create an intuitive interface for users to explore movies, select showtimes, pick seats, and manage bookings all from the comfort of their homes.

1.2 Problem Definition

Booking movie tickets has traditionally involved standing in long queues, checking for availability, and possibly missing out on preferred seats or showtimes. With the rise of digital platforms, many people still face issues such as a lack of real-time seat availability updates, the inability to select specific seats in some cases, and inadequate user interfaces. Additionally, some booking platforms are not optimized for mobile use, making it harder for users to book tickets on the go.

1.3 Proposed Solution

The Movie Ticket Booking System addresses these problems by offering an online platform where users can:

1. **Create an Account**: Users can register and log in to their accounts.
2. **Choose a Movie**: Users can browse a list of available movies and select one they wish to watch.
3. **Select a City**: The system offers a range of cities where movies are showing, making it convenient for users to choose their local theaters.
4. **Pick a Hall and Show Time**: After selecting a movie, users can choose their preferred theater, hall, and showtime.
5. **Seat Selection**: The system displays available and booked seats in real-time, allowing users to select their preferred seats.

6. **Booking Confirmation**: After selecting seats, users can confirm their booking, with a receipt generated to show the details of the booking.

The platform also provides an intuitive, mobile-responsive user interface and ensures that users are kept up-to-date with seat availability in real-time. This eliminates the need for physical queuing and the frustration of overbooked showtimes.

1.4 Objectives

The objectives of the Movie Ticket Booking System are:

1.User Account Management: Provide functionality for users to create accounts, log in, and securely store their credentials.

2.Real-Time Seat Availability: Display available and booked seats in real-time, preventing users from booking unavailable seats.

3.Multiple City and Movie Options: Offer a variety of cities and movies, giving users ample choices.

4.Simplified Booking Process: Allow users to easily select a movie, time, hall, and seats, streamlining the booking process.

5.Mobile-Responsive Design: Ensure the platform is fully responsive and optimized for mobile users.

6.Secure Payment System (Future Scope): Integrate a secure payment system (which could be added as part of the future development) to handle online payments for bookings.

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

The **Book My Cinema Ticket** is a part of a broader trend of digital transformation in the entertainment industry. With the growing demand for online services, traditional methods of booking movie tickets, such as waiting in long queues or calling theaters, are becoming less practical. Modern movie ticket booking systems leverage **web-based platforms**, mobile applications, and **backend technologies** to provide an efficient, scalable, and convenient solution. This literature survey focuses on understanding the various approaches to designing such systems, reviewing existing solutions, and exploring their limitations and improvements.

2.2 Related Work

Several online movie ticket booking systems and platforms have emerged, each offering unique features and functionality. Below are some notable works in the field:

1. **BookMyShow**: One of the most popular movie ticket booking platforms in India, BookMyShow offers users the ability to choose movies, book tickets, select seats, and pay online. It utilizes a backend system that supports real-time seat availability and user-specific recommendations based on preferences.
2. **Fandango**: Fandango is a well-known online platform for movie ticket booking in the U.S. It provides users with showtimes, ticket purchasing options, and the ability to select seats. Fandango also integrates features like movie trailers, reviews, and ratings.
3. **Cineworld**: Cineworld's booking system allows users to book tickets online through its website or mobile application. It supports seat selection, showtime browsing, and payment integration. The system ensures high availability of tickets, especially during peak hours.
4. **Ticketmaster**: Although primarily known for event ticketing, Ticketmaster’s platform is also used for movie ticket sales. It uses an integrated backend to handle high-traffic volumes and ensure seamless booking.

5. **SimpleTicket**: This is a lightweight movie ticket booking system aimed at small theaters. It focuses on simplicity and ease of use for the user, featuring a minimalistic design and core functionalities like movie selection, ticket purchase, and basic user authentication.

Technological Advances:

1. **Real-Time Booking and Availability**: Many modern systems, like BookMyShow and Fandango, use advanced backend systems to ensure real-time updates on seat availability, showtimes, and ticket bookings. This is often achieved through **AJAX**, **WebSockets**, and **RESTful APIs**.
2. **Personalization**: Several systems have incorporated recommendation algorithms that suggest movies based on user preferences and past bookings. This is commonly implemented using **machine learning** techniques.'
3. **Payment Integration**: Platforms have integrated various payment gateways, such as **PayPal**, **Stripe**, or local payment providers, allowing users to complete their transactions securely online.
4. **Cross-Platform Booking**: With the rise of mobile-first users, most of these systems now provide cross-platform booking capabilities, allowing users to book tickets via websites, mobile applications, or even voice assistants.

2.3 Overview

The goal of this section was to explore the existing landscape of movie ticket booking systems and understand the different techniques and technologies employed in their development. The systems reviewed showcase various features, such as real-time updates, user authentication, and seamless payment integrations, which are essential for a smooth user experience. The next steps in developing an advanced system should focus on improving **scalability**, **security**, and **performance** by integrating **cloud technologies**, **machine learning** for recommendations, and enhanced **user interfaces**.

While existing systems provide significant value, many face challenges such as handling high traffic loads during peak times, ensuring data consistency, and delivering personalized user experiences. Thus, there remains room for innovation, especially in **user-centric features** like **seat selection** and **dynamic pricing**.

CHAPTER 3

SYSTEM ANALYSIS

3.1 **Introduction**

This chapter provides an analysis of the **Book My Cinema Ticket**, highlighting the limitations of existing systems and proposing a new system with enhanced features. The analysis focuses on scalability, user experience, and technological advancements to ensure a seamless and efficient booking process.

3.2 **Existing System & Disadvantages**

Current movie ticket booking systems such as **BookMyShow** and **Fandango** offer basic functionalities but have several limitations:

- **Limited Scalability**: High traffic during peak hours often leads to slow performance or unavailability of tickets.
- **Complex User Interface**: Many systems have cluttered interfaces, making navigation difficult, especially for new users.
- **Lack of Personalization**: Current systems provide limited personalized recommendations.
- **Inefficient Seat Selection**: The process can be slow or prone to errors due to outdated interfaces.
- **Payment Integration Issues**: Payment failures or a lack of localized payment options are common problems.
- **Security Concerns**: Instances of data breaches and fraud have raised concerns about data protection.
- **Multi-Platform Issues**: Seamless integration between different platforms (mobile and web) is often missing.

3.3 **Proposed System & Advantages**

The proposed system addresses these issues with several key improvements:

1. **Scalable Architecture**: Built on a cloud platform to handle high traffic with minimal delays.

2. **Simplified User Interface**: An intuitive, user-friendly interface for easier navigation.
3. **Advanced Personalization**: Machine learning algorithms to offer movie recommendations based on user preferences.
4. **Real-Time Seat Selection**: Instant seat availability updates to prevent errors.
5. **Enhanced Payment Integration**: Multiple payment options and seamless transaction processes.
6. **Security Improvements**: Advanced encryption and compliance with security standards to protect user data.
7. **Multi-Platform Support**: Consistent user experience across web and mobile platforms, with features like push notifications.

Advantages:

- **Improved User Experience**: Intuitive interface and personalized recommendations.
- **Better Performance**: Scalability ensures smooth operation even during peak times.
- **Enhanced Security**: Strong data protection and secure payment systems.
- **Real-Time Updates**: Instant seat selection and booking confirmations.
- **Seamless Access**: Consistent experience across all devices.

3.4 Software Requirements

The software requirements for the "Movie Ticket Booking System" are as follows:

1. Operating System:

• Windows 10/11, macOS, or any Linux distribution that supports PHP, MySQL, and a web server.

2. Web Server:

• Apache (part of XAMPP, WAMP, or LAMP stacks).

3. Database:

• MySQL (compatible with PHP MyAdmin for database management).

4. Backend Programming:

• PHP (Version 7.x or higher).

5. Frontend Technologies:

• HTML5, CSS3, Bootstrap (for responsive design).

• JavaScript (for interactive features).

6. Additional Tools:

• PHP MyAdmin (for database management).

• Text Editor or IDE (e.g., Visual Studio Code, Sublime Text).

7. Browser Compatibility:

• Supports modern web browsers like Google Chrome, Firefox, Microsoft Edge, or Safari.

3.5 System Architecture

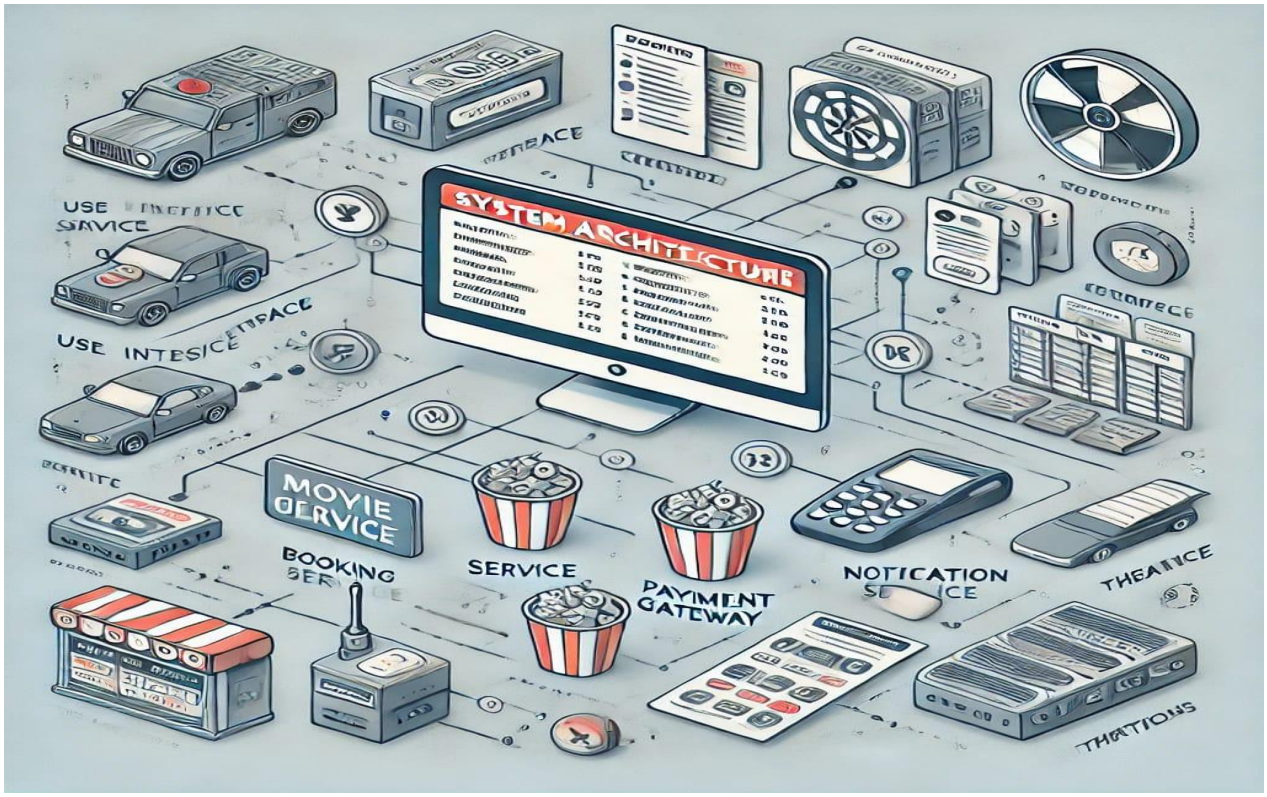

Fig 3.5 System Architecture

CHAPTER 4

MODULES

The Book My Cinema Ticket is divided into several modules, each addressing a specific functionality to ensure a smooth user experience and efficient system management. Below is a detailed description of the key modules:

4.1 User Registration and Authentication

This module handles the registration of new users and the authentication process for existing users. It ensures secure login with features like email verification, password hashing, and options for password recovery. Users can create their profiles by providing essential details, which are then stored securely in the database. Additionally, it may include advanced security measures such as CAPTCHA and Two-Factor Authentication (2FA).

4.2 Movie Catalog and Search

This module is responsible for displaying the list of available movies, complete with details like movie title, genre, rating, language, cast, director, and showtimes. Users can filter movies based on categories such as genre, release date, or ratings. The system will allow users to search for movies by title or actor, making the selection process user-friendly.

4.3 Seat Selection

This module provides an interactive seating arrangement that shows the available and booked seats in real-time. Users can select their preferred seats for a particular showtime. It will update the seat availability dynamically, preventing double bookings and ensuring an accurate view of seating options. This is especially useful for popular shows.

4.4 Booking and Payment

Once the user has selected a movie and seats, this module handles the booking process. It calculates the total price of the tickets, allowing users to proceed with payment. Integrated payment gateways (such as credit/debit cards, wallets, and online banking) will securely process

payments. After a successful transaction, the system generates an electronic ticket and sends it to the user’s email or mobile.

4.5 User Reviews and Ratings

This module allows users to rate movies and share their experiences by writing reviews. The average ratings and reviews for each movie are displayed for other users to view. This not only helps other users decide which movies to watch but also gives feedback to movie producers and theater operators.

4.6 Admin Dashboard

The Admin Dashboard provides theater operators and admins with a centralized interface to manage the entire system. Admins can update movie details, manage user accounts, and handle movie bookings. The dashboard also allows for the generation of reports related to sales, user activity, and overall system performance.

4.7 Ticket Cancellation and Refund

This feature allows users to cancel their bookings and request refunds in accordance with the cinema's cancellation policy. The module ensures that any valid cancellation requests are processed promptly, updating the user’s account and returning funds to their original payment method.

4.8 Analytics and Reporting

The Analytics module provides detailed insights into system performance, ticket sales, popular movies, user activity, and financial reports. Admins can analyze trends, track sales over time, and use this data to optimize the movie offerings, pricing strategies, and marketing campaigns.

Each module plays a crucial role in maintaining the functionality and efficiency of the Movie Ticket Booking System, ensuring users have a seamless, secure, and enjoyable experience from registration to booking and beyond.

CHAPTER 5

SOURCE CODE

5.1 REGISTER.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Register - Movie Ticket Booking</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="container">
    <h1>Create Your Account</h1>
    <p>Join us and book your favorite movie tickets easily!</p>
    <form id="register-form" onsubmit="return registerUser(event)">
      <label for="username">Username</label>
      <input type="text" id="username" name="username" required placeholder="Enter your username">
      <label for="email">Email</label>
      <input type="email" id="email" name="email" required placeholder="Enter your email">
      <label for="password">Password</label>
      <div class="password-container">
        <input type="password" id="password" name="password" required placeholder="Enter your password">
        <span class="eye-icon" onclick="togglePassword()">  </span>
      </div>
      <label for="confirm-password">Confirm Password</label>
      <input type="password" id="confirm-password" name="confirm-password" required placeholder="Confirm your password">
```

```
    <button type="submit">Register</button>
   </form>
   <p>Already have an account? <a href="login.html">Login here</a></p>
  </div>
  <script src="script.js"></script>
</body>
</html>
```

5.2 LOGIN.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login - Movie Ticket Booking</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="container">
    <h1>Login to Your Account</h1>
    <p>Welcome back! Please login to continue.</p>
    <form id="login-form" onsubmit="return loginUser(event)">
      <label for="username">Username</label>
      <input type="text" id="username" name="username" required placeholder="Enter your username">
      <label for="password">Password</label>
      <div class="password-container">
        <input type="password" id="password" name="password" required placeholder="Enter your password">
        <span class="eye-icon" onclick="togglePassword()">  </span>
      </div>
```

```
    <button type="submit">Login</button>
  </form>
  <p>Don't have an account? <a href="register.html">Register here</a></p>
 </div>
 <script src="script.js"></script>
</body>
</html>
```

5.3 INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Movie Ticket Booking System</title>
 <link rel="stylesheet" href="styles.css">
 <style>
  body{
   background-image: url(image.1.png);
   height: 100vh;
   background-size: cover;
   background-position: center;
  }
 </style>
</head>
<body>
 <!-- Main Container -->
 <div class="container">
  <h1>Welcome to the Movie Ticket Booking</h1>
  <p>Click below to start booking your tickets!</p>
```

```
    <button onclick="goToNextPage()">Book Tickets</button>
  </div>
  <script src="script.js"></script>
</body>
</html>
```

5.4 SECONDPAGE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Select Movie Details</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="container">
    <h1>Select Your Movie Details</h1>
    <!-- City Selection -->
    <h2>Select City</h2>
    <select id="city-selection">
      <option value="chennai">Chennai</option>
      <option value="bangalore">Bangalore</option>
      <option value="mumbai">Mumbai</option>
      <option value="delhi">Delhi</option>
      <option value="kolkata">Kolkata</option>
      <option value="hyderabad">Hyderabad</option>
      <option value="pune">Pune</option>
      <option value="chandigarh">Chandigarh</option>
    </select>
```

```
<!-- Movie Selection -->
<h2>Select Movie</h2>
<select id="movie-selection">
  <option value="Pushpa">Pushpa</option>
  <option value="RRR">RRR</option>
  <option value="Vikram">Vikram</option>
  <option value="KGF 2">KGF 2</option>
  <option value="Avatar 2">Avatar 2</option>
  <option value="Mission Impossible">Mission Impossible</option>
  <option value="Doctor Strange">Doctor Strange</option>
  <option value="Spider-Man: No Way Home">Spider-Man: No Way Home</option>
</select>
<!-- Hall Selection -->
<h2>Select Hall</h2>
<select id="hall-selection">
  <option value="PVR">PVR</option>
  <option value="INOX">INOX</option>
  <option value="Cinepolis">Cinepolis</option>
  <option value="GV">GV</option>
  <option value="Sathyam">Sathyam</option>
  <option value="Forum">Forum</option>
</select>
<!-- Show Timing -->
<h2>Select Show Timing</h2>
<select id="timing-selection">
  <option value="11:00 AM">11:00 AM</option>
  <option value="2:30 PM">2:30 PM</option>
  <option value="6:30 PM">6:30 PM</option>
  <option value="9:30 PM">9:30 PM</option>
</select>
```

```
    <!-- Next Button -->
    <button onclick="goToSeatSelectionPage()">Proceed to Select Seats</button>
  </div>
  <script src="script.js"></script>
</body>
</html>
```

5.5 THIRDPAGE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Select Seats</title>
  <link rel="stylesheet" href="styles.css">
  <style>
    body{
      background-image: url(imge.2.jpg);
      height: 100vh;
      background-size: cover;
      background-position: center;
    }
  </style>
</head>
<body>

  <div class="container">
    <h1>Selected Movie Details</h1>
    <div id="movie-details"></div>
```

```
    <h2>Select Your Seats</h2>
    <div id="seat-map"></div>
    <button onclick="bookSeats()">Book Seats</button>
    <p id="message"></p>
    <a href="secondPage.html">Go back to select details</a>
  </div>
  <script src="script.js"></script>
</body>
</html>
```

5.6 RECEIPT.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Booking Receipt</title>
  <link rel="stylesheet" href="styles.css">
  <style>
    body{
      background-image: url(image5.webp);
      height: 100vh;
      background-size: cover;
      background-position: center;
    }
  </style>
</head>
<body>
  <div class="container">
    <h1>Booking Receipt</h1>
    <div id="booking-summary"></div>
```

```
    <!-- Confirm Booking Button -->
    <button onclick="confirmBooking()">Confirm Booking</button>
  </div>
  <script src="script.js"></script>
</body>
</html>
```

5.7 CSS

```
body {
  font-family: Arial, sans-serif;
  background-color: #f7f7f7;
  margin: 0;
  padding: 0;
}
.container {
  width: 100%;
  max-width: 400px;
  margin: 50px auto;
  padding: 30px;
  background-color: rgba(107, 223, 213, 0.7);
  color: #fff;
  border-radius: 8px;
  text-align: center;
}
h1 {
  font-size: 32px;
  color: #ff4081;
  margin-bottom: 15px;
}
h2 {
```

```
  font-size: 24px;
  color: #ff80ab;
  margin-bottom: 20px;
}
form {
  display: flex;
  flex-direction: column;
  align-items: center;
}
input[type="text"],
input[type="email"],
input[type="password"] {
  width: 100%;
  padding: 12px;
  margin: 10px 0;
  font-size: 16px;
  border-radius: 5px;
  border: 1px solid #ddd;
  background-color: #fff;
  color: #333;
}
input[type="password"] {
  margin-left: -15px;
}
input[type="text"]:focus,
input[type="email"]:focus,
input[type="password"]:focus {
  border-color: #ff4081;
  outline: none;
}
```

```
.password-container {
  position: relative;
  width: 100%;
}
.eye-icon {
  position: absolute;
  top: 12px;
  right: 10px;
  cursor: pointer;
  font-size: 20px;
  color: #ff4081;
}
button {
  padding: 12px;
  margin-top: 20px;
  font-size: 16px;
  background-color: #ff4081;
  color: white;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  transition: background-color 0.3s ease;
}
button:hover {
  background-color: #ff80ab;
}
 {
  display: grid;
  grid-template-columns: repeat(10, 30px);
  grid-gap: 5px;
```

```
  justify-content: center;
}
.seat {
  width: 30px;
  height: 30px;
  background-color: #4CAF50;
  border-radius: 5px;
  cursor: pointer;
}
.seat.selected {
  background-color: #2196F3;
}
.seat.booked {
  background-color: #f44336;
  cursor: not-allowed;
}
a {
  color: #ff4081;
  text-decoration: none;
  font-size: 16px;
}
a:hover {
  text-decoration: underline;
}
 (max-width: 500px) {
  .container {
    padding: 20px;
    width: 90%;
  }
  h1 {
```

```
    font-size: 28px;
  }
  h2 {
    font-size: 20px;
  }
  input[type="text"],
  input[type="email"],
  input[type="password"] {
    font-size: 14px;
  }
  button {
    font-size: 14px;
  }
}
```

5.8 JAVASCRIPT

```
let selectedSeats = [];

const moviePrices = {
  'Pushpa': 1000,
  'RRR': 400,
  'Vikram': 180,
  'KGF 2': 220,
  'Avatar 2': 250,
  'Mission Impossible': 300,
  'Doctor Strange': 500,
  'Spider-Man: No Way Home': 410
};
function goToNextPage() {
  window.location.href = "secondPage.html"; // Ensure the path is correct
}
```

```
function goToSeatSelectionPage() {
  const selectedCity = document.getElementById('city-selection').value;
  const selectedMovie = document.getElementById('movie-selection').value;
  const selectedHall = document.getElementById('hall-selection').value;
  const selectedTiming = document.getElementById('timing-selection').value;
    localStorage.setItem('selectedCity', selectedCity);
  localStorage.setItem('selectedMovie', selectedMovie);
  localStorage.setItem('selectedHall', selectedHall);
  localStorage.setItem('selectedTiming', selectedTiming);
  window.location.href = "thirdPage.html";
}
window.onload = function () {
  if (window.location.href.indexOf("thirdPage.html") > -1) {
    const movieDetails = `
      <p>City: ${localStorage.getItem('selectedCity')}</p>
      <p>Movie: ${localStorage.getItem('selectedMovie')}</p>
      <p>Hall: ${localStorage.getItem('selectedHall')}</p>
      <p>Show Timing: ${localStorage.getItem('selectedTiming')}</p>
    `;
    document.getElementById('movie-details').innerHTML = movieDetails;
    generateSeats(); // Generate seat grid
  }
  if (window.location.href.indexOf("receiptPage.html") > -1) {
    const selectedSeatsText = localStorage.getItem('selectedSeats').split(', ').join('<br>');
    const totalPrice = localStorage.getItem('totalPrice');
    const bookingSummary = `
      <p>Movie: ${localStorage.getItem('selectedMovie')}</p>
      <p>Hall: ${localStorage.getItem('selectedHall')}</p>
      <p>Seats: ${selectedSeatsText}</p>
      <p>Total Price: ₹${totalPrice}</p>
```

```
    `;
    document.getElementById('booking-summary').innerHTML = bookingSummary;
  }
};
function generateSeats() {
  const seatMap = document.getElementById('seat-map');
  const selectedMovie = localStorage.getItem('selectedMovie');
  const selectedHall = localStorage.getItem('selectedHall');
    ConstbookedSeats                                                                    =
JSON.parse(localStorage.getItem(`${selectedMovie}_${selectedHall}_bookedSeats`)) || [];
  seatMap.innerHTML = ''; // Clear previous seats
  for (let i = 0; i < 10; i++) {
    for (let j = 0; j < 10; j++) {
      const seat = document.createElement('div');
      seat.classList.add('seat');
      seat.dataset.row = i + 1;
      seat.dataset.col = j + 1;
      const isBooked = bookedSeats.some(s => s.row === i + 1 && s.col === j + 1);
      if (isBooked) {
        seat.classList.add('booked'); // Mark it as booked
      } else {
        seat.onclick = () => toggleSeatSelection(i + 1, j + 1); // Allow selection if not booked
      }
      seatMap.appendChild(seat);
    }
  }
}
function toggleSeatSelection(row, col) {
  const seat = document.querySelector(`[data-row="${row}"][data-col="${col}"]`);
  if (seat.classList.contains('booked')) return; // Prevent selecting booked seats
  if (seat.classList.contains('selected')) {
```

```
    seat.classList.remove('selected');
    selectedSeats = selectedSeats.filter(s => !(s.row === row && s.col === col));
  } else {
    seat.classList.add('selected');
    selectedSeats.push({ row, col });
  }
}
function bookSeats() {
  selectedSeats.forEach(seat => {
    constseatElement=document.querySelector(`[data-row="${seat.row}"][data-
col="${seat.col}"]`);
    seatElement.classList.add('booked');
    seatElement.classList.remove('selected');
  });
    const selectedMovie = localStorage.getItem('selectedMovie');
  const selectedHall = localStorage.getItem('selectedHall');
  LetbookedSeats                                                                    =
JSON.parse(localStorage.getItem(`${selectedMovie}_${selectedHall}_bookedSeats`)) || [];
    selectedSeats.forEach(seat => {
    bookedSeats.push({ row: seat.row, col: seat.col });
  });

    localStorage.setItem(`${selectedMovie}_${selectedHall}_bookedSeats`,
JSON.stringify(bookedSeats));
  const moviePrice = moviePrices[selectedMovie];
    localStorage.setItem('selectedSeats',selectedSeats.map(seat => `${seat.row}-${seat.col}`).join(',
'));
  localStorage.setItem('totalPrice', selectedSeats.length * moviePrice); // Calculate total price
   window.location.href = "receiptPage.html";
}
function confirmBooking() {
  alert('Your booking is confirmed! Thank you for booking with us.');
```

```
    window.location.href = "index.html"; // Make sure index.html is your main booking page
}
function togglePassword() {
  const passwordField = document.querySelectorAll('#password, #confirm-password')[0];
  const eyeIcon = document.querySelector('.eye-icon');
  if (passwordField.type === "password") {
    passwordField.type = "text";
    eyeIcon.innerHTML = "    ";  // Hide icon
  } else {
    passwordField.type = "password";
    eyeIcon.innerHTML = "    ";  // Show icon
  }
}
function registerUser(event) {
  event.preventDefault();
  const username = document.getElementById('username').value;
  const email = document.getElementById('email').value;
  const password = document.getElementById('password').value;
  const confirmPassword = document.getElementById('confirm-password').value;

  if (password !== confirmPassword) {
    alert('Passwords do not match');
    return;
  }
  const users = JSON.parse(localStorage.getItem('users')) || [];
  const userExists = users.some(user => user.username === username || user.email === email);
  if (userExists) {
    alert('Username or email already exists');
    return;
  }
```

```
  users.push({ username, email, password });
  localStorage.setItem('users', JSON.stringify(users))
  alert('Registration successful! You can now log in.');
  window.location.href = 'login.html';  // Redirect to login page
}
function loginUser(event) {
  event.preventDefault();
  const username = document.getElementById('username').value;
  const password = document.getElementById('password').value;
  const users = JSON.parse(localStorage.getItem('users')) || [];
  const user = users.find(user => user.username === username && user.password === password);
  if (user) {
    localStorage.setItem('isLoggedIn', true);
    window.location.href = 'mainPage.html';  // Redirect to the main page
  } else {
    alert('Invalid username or password');
  }
}
function loginUser(event) {
  event.preventDefault();
  const username = document.getElementById('username').value;
  const password = document.getElementById('password').value;
    window.location.href = "index.html"; // Redirect to index.html page
}
function togglePassword() {
  const passwordField = document.getElementById('password');
  const eyeIcon = document.querySelector('.eye-icon');
  if (passwordField.type === 'password') {
    passwordField.type = 'text';
    eyeIcon.textContent = '      ';    } else { passwordField.type = 'password';eyeIcon.textContent =
      '   '; }}
```

CHAPTER 6

RESULTS

6.1 INTRODUCTION

Welcome to our **Book My Cinema Ticket**! This platform allows users to seamlessly browse and book movie tickets in their favorite cities and theaters. Whether you're a new user wanting to create an account or an existing user ready to book your next movie experience, we've got you covered.

- **User Registration & Login:**
    - **Create an Account:** New users can sign up with a username, email, and password.
    - **Login:** Existing users can log in using their credentials for a quick access to booking tickets.
- **Movie Selection:**
    - Choose from a wide variety of popular movies.
    - Select the city where you want to watch the movie.
    - Pick your preferred theater from options like **PVR**, **INOX**, **Cinepolis**, and others.
    - Choose a showtime that suits your schedule.
- **Seat Selection:**
    - Once you've selected a movie, theater, and showtime, you can choose your seats from a dynamic seat map.
    - Seats are color-coded: available seats are green, selected seats are blue, and booked seats are red.
- **Booking Summary & Confirmation:**
    - After selecting your seats, you'll get a detailed summary of your booking including the movie details, seat selection, and total price.
    - Upon confirming your booking, you receive a receipt with all the booking information.
- **User-Friendly Interface:**
    - The website is designed to be intuitive and easy to navigate, whether you're booking from your laptop or mobile device.
    - We've also implemented features like password visibility toggle, form validation, and responsive design for a smooth experience.
- **Secure Login & Registration:**
    - User credentials are stored locally in the browser, ensuring that your login information remains secure. Passwords are hidden and can be toggled for visibility.

6.2 0UTPUT SCREENS

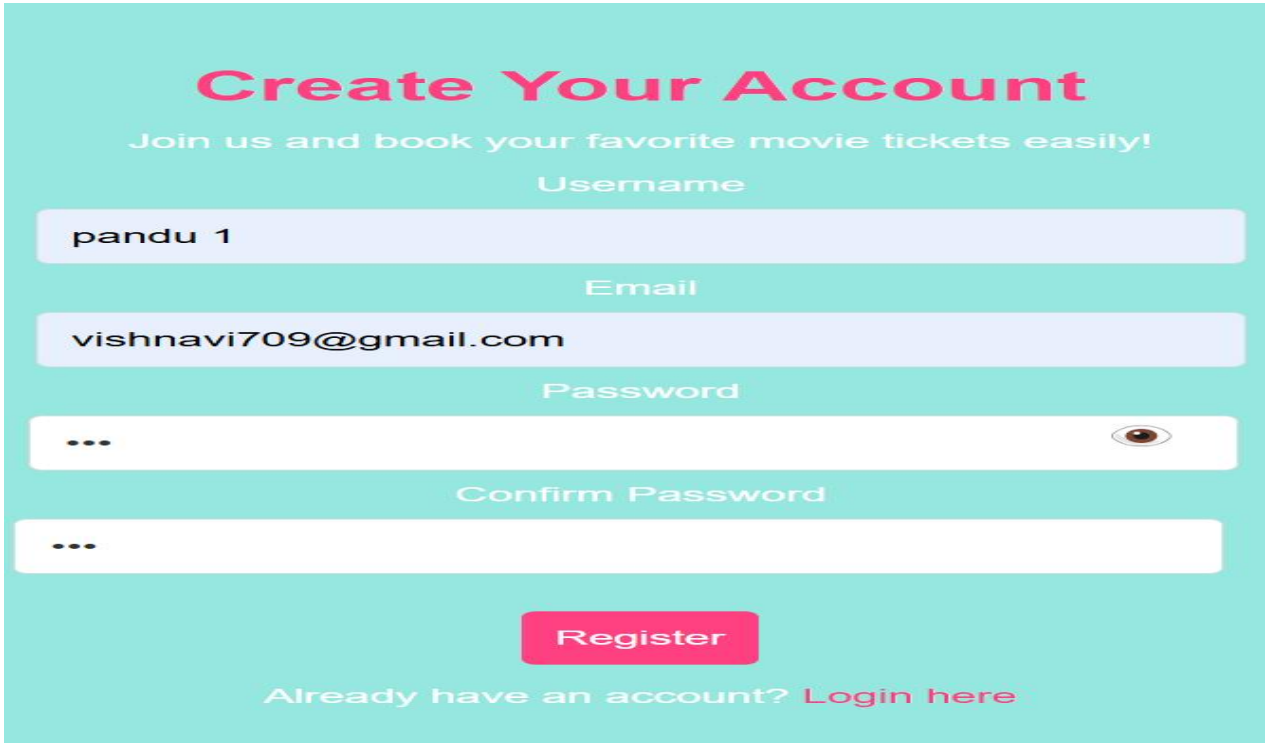

Fig 6.1 Registration Page

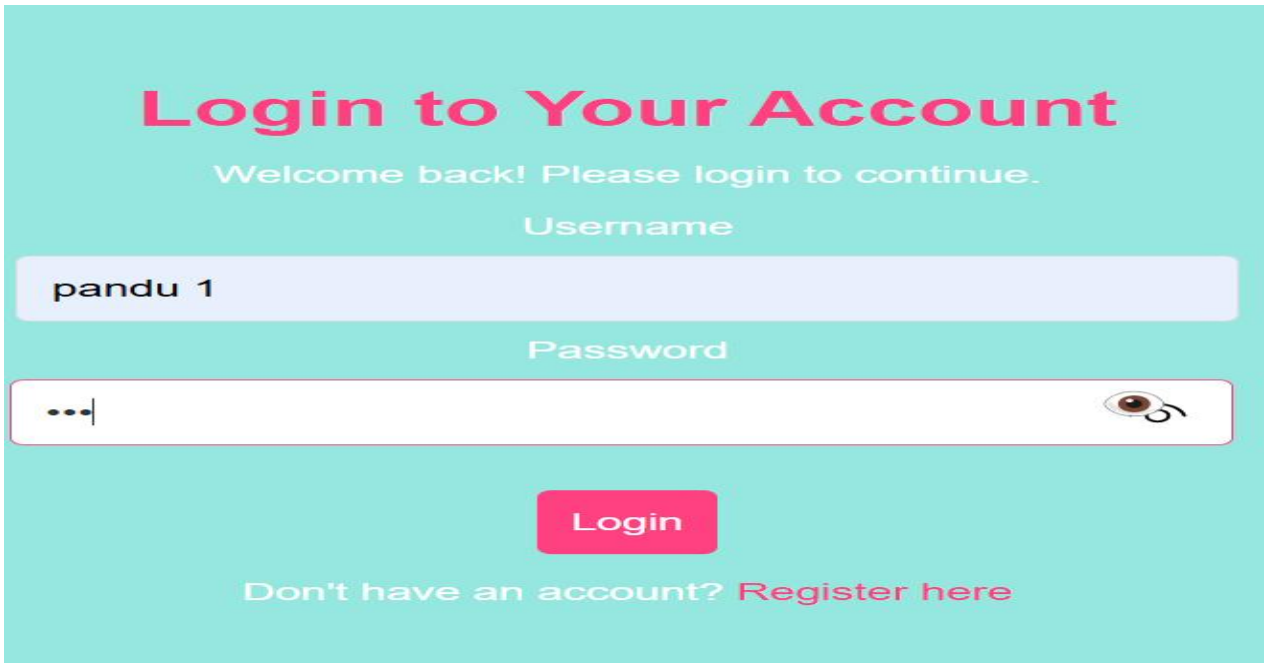

Fig 6.2 Login Page

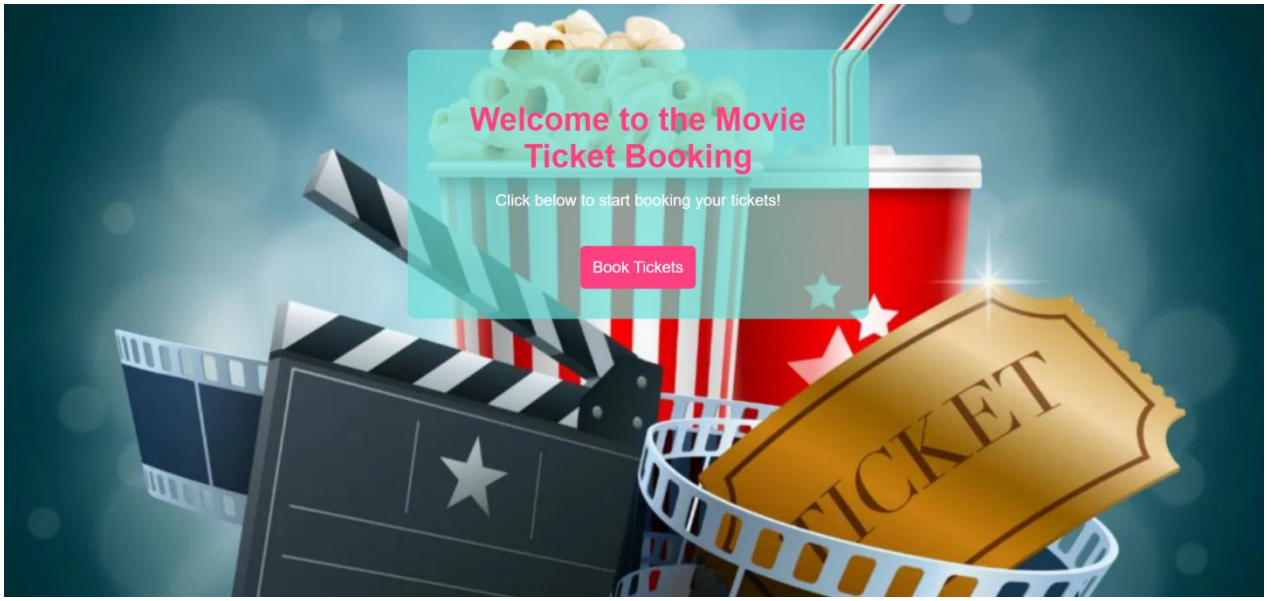

Fig 6.3 Welcome To Move Ticket Booking Page

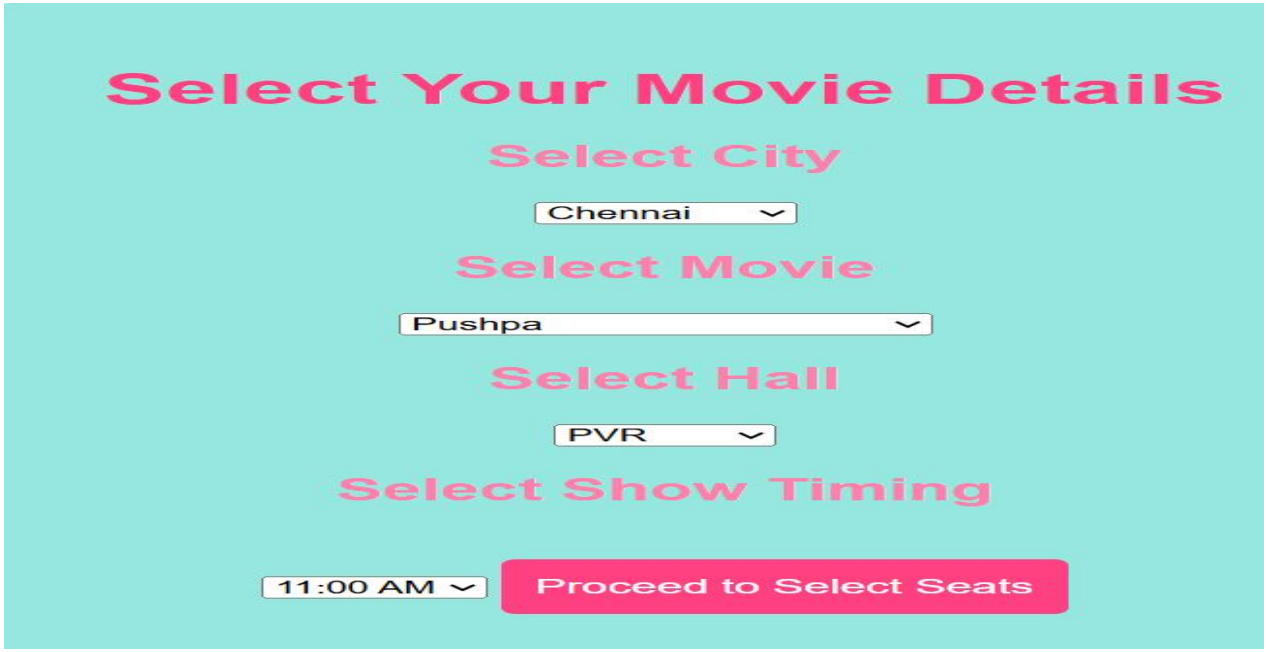

Fig 6.4 Select Your Movie Details Page

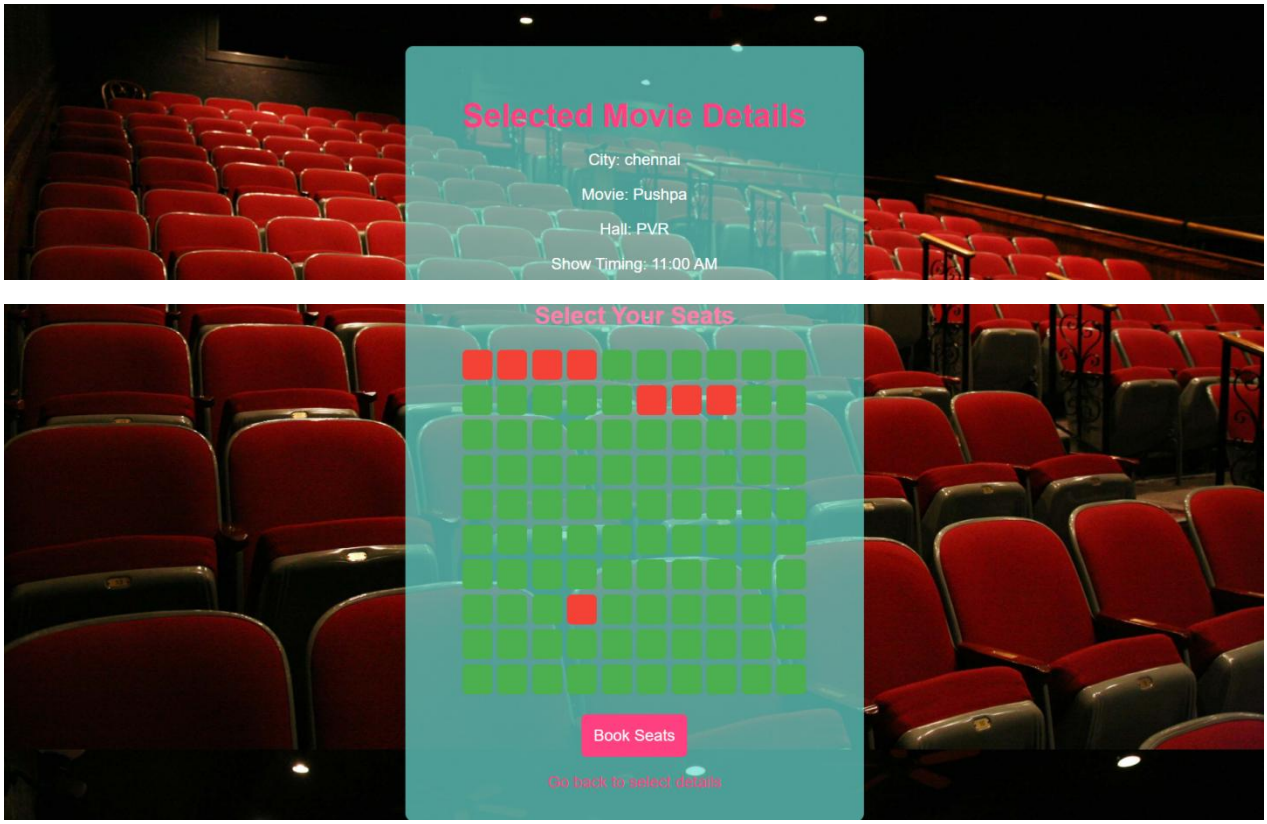

Fig 6.5 Seat Booking Page

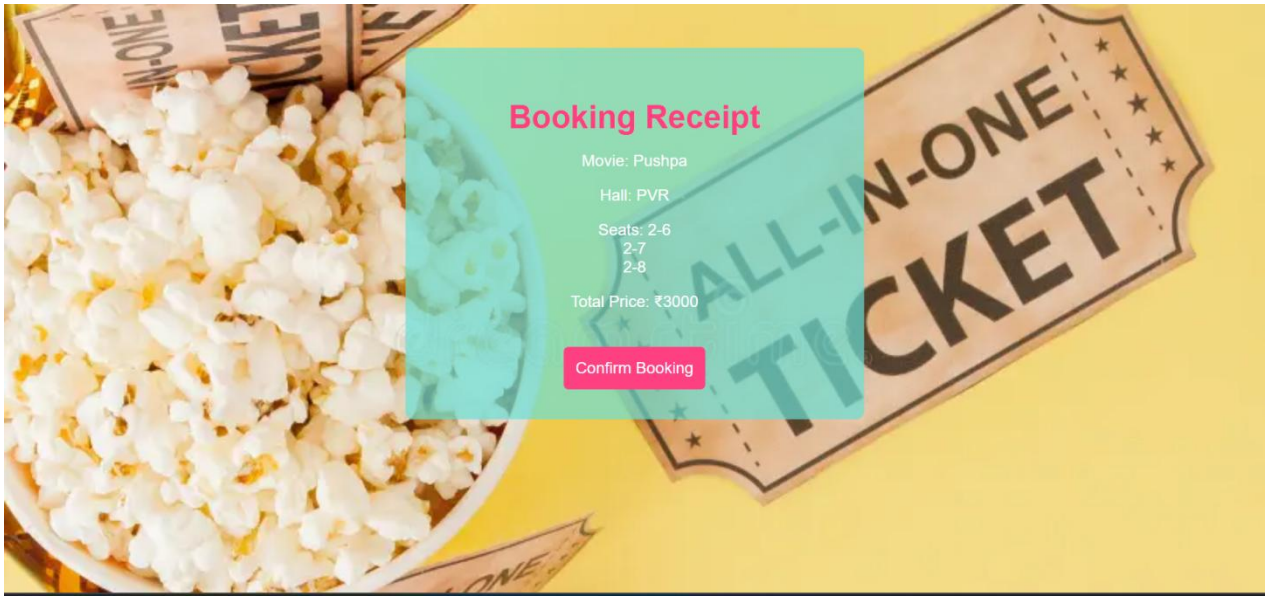

Fig 6.6 Booking Receipt Page

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 CONCLUSION

The Book My Cinema Ticket, built with front-end technologies like HTML, CSS, and JavaScript, offers a basic yet functional platform for users to register, log in, select movies, choose seats, and complete their bookings. However, the system's current reliance on local storage limits its ability to handle large-scale data and ensure secure user authentication.

Integrating a backend using PHP and MySQL would significantly enhance the system's functionality. With PHP, the system could manage user accounts, authenticate logins, and process bookings dynamically. MySQL would provide a database for storing user information, movie details, seat availability, and booking history. This setup would enable real-time updates, secure password storage, and improved scalability.

By using PHP for server-side logic and MySQL for data management, the Movie Ticket Booking System could evolve into a fully functional, secure, and scalable platform that delivers a smooth and reliable experience for users.

7.2 FUTURE ENHANCEMENTS

Back end Integration with PHP and MySQL:

1.User Management: Implement PHP and MySQL for secure user registration, login, and profile management. User credentials and preferences can be stored in a MySQL database.

2.Booking Management: Manage seat availability in real-time and store booking data in MySQL. When a user selects seats, the backend can update seat status and prevent double bookings.

3.Payment Gateway Integration: Integrate popular payment gateways (e.g., PayPal, Stripe) for online payments, enabling users to pay for their tickets securely.

Real-Time Seat Availability:

Implement dynamic seat availability updates using AJAX and PHP. When a user selects a seat, the system can instantly reflect this change in the database, preventing multiple users from booking the same seat at the same time.

Advanced Movie Search and Filters:

Allow users to search for movies by genre, release date, or rating. Enhance the movie selection process by including filters for better user experience.

Admin Panel for Movie and Booking Management:

Create an admin panel where the admin can add, update, and delete movies, manage showtimes, and view booking data. This would enable better control over movie listings and booking management.

Email and SMS Notifications:

Send confirmation emails or SMS messages to users after successful booking. These notifications could include booking details, movie timings, and a QR code for easy access to the cinema.

Mobile Application:

Develop a mobile application (iOS/Android) to allow users to book tickets on the go. The mobile app could also include features like push notifications for movie promotions and upcoming releases.

Admin Authentication and Authorization:

Implement role-based authentication for admins to ensure only authorized users can manage the system's movie listings, bookings, and user data.

Improved User Interface and UX:

Enhance the visual design and user experience (UX) of the booking process. Incorporate more intuitive layouts, animations, and transitions to make the system more engaging.

Multi-language and Currency Support:

Enable the system to support multiple languages and currencies, allowing users from different regions to access the platform in their preferred language and currency.

REFERENCES

1. **Book:**

Software Engineering: A Practitioner's Approach by Roger S. Pressman.

1. This book offers a comprehensive understanding of software engineering principles, methodologies, and best practices for system design, development, and maintenance. It is an essential resource for any software development project.

2. **Article:**

Design and Implementation of Online Movie Ticket Booking System by Dr. S. Srinivasan, et al., *International Journal of Computer Applications (2018)*.

1. Link: https://www.ijcaonline.org/
2. This paper discusses various design strategies and implementation techniques for movie ticket booking systems, focusing on user experience and backend efficiency.

3.Website:

Booking.com: Movie Ticket Booking System – A leading platform in the booking industry, demonstrating user interface design and functionalities that can inspire similar projects.

Link: https://www.booking.com

4.Technical Documentation:

API Documentation for Movie Data and Ticketing Systems by OMDB API.

1. Link: https://www.omdbapi.com/

2.Provides an API to fetch movie details, ratings, and showtimes, a key component for any movie ticketing system.

5.Online Resource:

Stack Overflow Discussion on Building a Movie Ticketing System with PHP.

Link: https://stackoverflow.com/questions/

A valuable discussion thread for developers looking for guidance on building a movie ticketing platform using PHP, MySQL, and other technologies.

Online Forum:

Movie Ticket Booking System: Common Issues and Solutions on GitHub Discussions.

1.Link: https://github.com/discussions/

2. A forum for developers collaborating on building and improving movie ticket booking systems.

Video Tutorial:

Creating a Movie Booking System Using Angular and Node.js by Tech with Tim.

1.Link: https://www.youtube.com/watch?v=

2. A YouTube video series teaching the development of a full-stack movie ticket booking application with modern JavaScript frameworks.

Research Article:

Evaluating the Impact of Automated Systems in Ticket Booking Industry by Sarah Lee, *International Journal of Advanced Technology*.

1.Link: https://www.ijat.com

2.This article evaluates the positive impacts of automation and online platforms in the ticket booking industry.

K.S.R.M COLLEGE OF ENGINEERING

(UGC - AUTONOMOUS)

Approved by AICTE, New Delhi & Affiliated to JNTUA, Ananthapuramu

(Accredited by NAAC with A+ Grade & B.Tech.(Civil, EEE,Mech,ECE and CSE) Programs by NBA)

KADAPA – 516 005 (A.P.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Bio-Data

Roll No	Name of the Student	Student/ Father WhatsApp Nos	Address	Fb id/Insta. Id	Photo
229Y1AO5G3	SIDDAVATAM VISHNAVI	9346434375	MARATHI STREET , KADAPA	Vishnavi12@gmail.com	
229Y1A05G9	SUDA ARCHANA	6303497494	ITI CIRCLE KADAPA	Sudhaarachana@gmail.com	
229Y1AO5I5	VALLEPU PAVAN	9182372120	NAGARIGUTTA, PULIVENDULA	Pavan_vallepu@gmail.com	
229Y1A05F7	SHAIK SAJITH	9392884650	NANDI MANDALAM KADAPA	Sajith456@gmail.com