LA GRANDEE INTERNATIONAL COLLEGE

Simalchaur Pokhara, Nepal

A Final Report

on

Movie Ticket Booking System

Submitted to

La Grandee International College

Bachelor of Computer Application (BCA) Program

In partial fulfilment of the requirements for the degree of Bachelor of Computer Application (BCA) under Pokhara University

Submitted by

Name	Program	Semester	P.U Registration No.
Priyanka Ghimire	BCA	6th	2021-1-53-0360
Manisha Sharma	BCA	6th	2021-1-53-0354
Diptan Gurung	BCA	6th	2021-1-53-0350

Date: 2081-10-06

Acknowledgement

We have presented this report focusing on the topic “Movie Ticket Booking System”. This report has been prepared for partial fulfilment of the requirement for the degree of BCA and to have practical experience.

We are heartily thankful to the faculty of IT, LA Grandee international college and our supervisor **Mr. Sunil Sapkota** and **BCA Co-ordinator Mr. Kundan Chaudary** for their role to motivate and lead for this report. We obliged towards their constant guidance, supervision and feedbacks which enabled us to prepare a well-executed report.

Further, we express our gratitude to LA Grandee family, classmates, seniors and teachers who have directly and indirectly supported us during our project.

Sincerely,

Priyanka Ghimire

Manisha Sharma

Diptan Gurung

Declaration for

"Movie Ticket Booking System"

Student's Declaration

We **Priyanka Ghimire**, **Manisha Sharma** and **Diptan Gurung** being students of sixth semester at LA GRANDEE International College, Faculty of Science and Technology (Kha), Pokhara University, do hereby declare that the project submitted to the aforementioned institution is an original work completed by us in partial fulfilments of the requirements for the Bachelors of Computer Application (BCA) program, under the supervision of professor **Mr.Sunil Sapkota**. We further state that no resources other than those specifically listed have been utilized in the completion of project.

Name of the student: Priyanka Ghimire

Class Roll No: 16

Exam Roll No: 21535161

Semester: 6th

Signature: _____

Name of the student: Manisha Sharma

Class Roll No: 10

Exam Roll No: 21535155

Semester: 6th

Signature: _____

Name of the student: Diptan Gurung

Class Roll No: 6

Exam Roll No: 21535151

Semester: 6th

Signature: _____

Supervisor's Declaration

I hereby recommended that this project entitled "**Movie Ticket Booking System**" is done under my supervision by **Priyanka Ghimire, Manisha Sharma, Diptan Gurung** during their 6th semester in partial fulfilment of the requirements for the degree of **Bachelors in Computer Application (BCA)** under **Pokhara University** is completed to my satisfaction and be processed for final evaluation.

_____

Mr.Sunil Sapkota

Date: 2081-10-06

Letter of Approval

We certify that we have examined this report entitled "**Movie Ticket Booking System**" and are satisfied with the project defence. It is satisfactory in the scope and qualify as project in partial fulfilment of the requirements for the degree of **Bachelor in Computer Application (BCA)** under **Pokhara University.**

Supervisor	**External Examiner**	**Program Coordinator**
Mr.Sunil Sapkora	**……**	**Mr. Kundan Chaudary**
Ast.Professor		**Ast.Professor**
LAGrandee International College	**Date:** 2081-10-06	LA Grandee International College

Abstract

Movie Ticket Booking System is a web based system developed using HTML, CSS, JavaScript, PHP, and MySQL to streamline the ticket booking process. The system allows users to browse movies, select showtimes, choose seats, and purchase tickets via a responsive and user-friendly interface.

Table of contents

Table of figures

Table of Table

Abbreviations

HTML	Hypertext Markup Language
Css	Cascading style sheet
PHP	Hypertext Preprocessor
API	Application programming interface
Mysql	My Structured Query Language
DFD	Data flow diagram
ER	Entity relationship

1. Introduction

In the modern digital age, online ticket booking has transformed the entertainment industry, offering convenience and efficiency over traditional methods like visiting theaters or booking via phone. This project successfully developed a **Movie Ticket Booking System** using HTML, CSS, JavaScript, PHP, and MySQL, providing users with an intuitive platform to browse movies, select showtimes, choose seats, and book tickets seamlessly. The frontend was designed to be visually appealing and accessible across devices, while JavaScript enabled interactive features like real-time seat selection. The backend, powered by PHP and MySQL, ensured secure user authentication, efficient movie management, and reliable booking operations.

In its final phase, the system was enhanced with the integration of the Khalti API**,** enabling secure and smooth payment processing for users. Additionally, an admin dashboard was designed, allowing theater operators to manage movies efficiently, including adding, deleting, and editing movie details and schedules. These features made the system more functional and scalable, catering to the needs of both users and administrators.

This project achieved its goal of streamlining the ticket booking process while providing real-time updates on schedules, seat availability, and booking status. By combining modern web technologies, the system ensures convenience, reliability, and an enhanced user experience.

2. Problem Statement

- Complex payment integration
- Poor user interface and user experience

3. Objectives

- To improve user experience and accessibility.
- To make a seamless payment integration

4. Background Study

In today's digital era, online systems have transformed businesses, including the entertainment industry. The **Movie Ticket Booking System** has revolutionized ticket purchasing by allowing users to browse movies, select showtimes, choose seats, and pay online from the comfort of their devices.

The old manual system faced numerous challenges, such as tedious record-keeping, unsystematic data storage, difficulty retrieving information, and inefficient error correction. Users often dealt with long queues and sold-out shows, making the process inconvenient and time-consuming.

With advancements in internet technology and the rise of smartphones, online booking systems have addressed these issues, offering convenience, flexibility, and secure digital payment options. Users can now book tickets anytime, anywhere, without visiting theaters in person.

The system enhances efficiency for cinemas by increasing revenue and streamlining operations, while providing users with a hassle-free and efficient ticket booking experience.

5. Requirement document

The purpose of this document is to outline the functional and non-functional requirements for a Movie Ticket Booking System. This document will be used as a guide for the development team during the design and implementation phases of the project.

5.1 Functional Requirements:

User Features

Registration and Login:

- New users can register and log in using secure credentials.
- Sessions are managed to keep users logged in during their visit.

Movie Listings and Search:

- Users can browse movies by genre or search for a specific movie.
- Detailed movie information, including timing and description, is displayed.

Seat Selection and Booking:

- Users can view and select available seats for a show.
- Real-time updates ensure no double booking occurs.

Payment Processing:

- Integrated payment gateway allows users to pay for bookings securely.
- Users receive confirmation after successful payment.

Admin Features

Movie Management:

- Admins can add, update, or delete movie records.
- Includes fields like title, description, genre, and timings.

Booking Management:

- Admins can view and manage bookings made by users.

5.2 Non-Fnctional requirements

Non-Functional requirements are set of specifications that describe the system's operation capabilities and constraints and attempt to improve its functionality.
Some non-functional requirements are:

Security:

- User passwords must be stored securely using encryption.
- The system must ensure secure payment processing by integrating with a reliable payment gateway.
- Implement user authentication to prevent unauthorized access.

Usability:

- The interface should be intuitive and easy to use for customers with basic computer or smartphone skills.
- The system should provide clear error messages and guidance for users when issues occur (e.g., invalid login credentials or unavailable seats).

Maintainability:

- The system should be modular, making it easy to update or add new features (e.g., additional payment methods or seat layouts).
- Bugs and issues should be easy to identify and resolve without impacting ongoing bookings.

These non-functional requirements ensure that the system meets the expectations of performance, reliability, security, and usability, contributing to a smooth experience for both users and theater operators.

6. System Design

Dataflow, Entity-relationship diagram are used for understanding the system's design and its functionalities, and both are important for creating proper documentation.

6.1 Data Flow Diagram

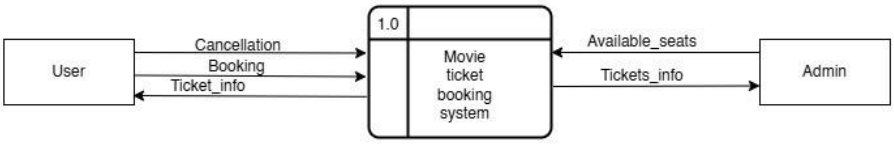


Figure 6.1 Level-0-DFD

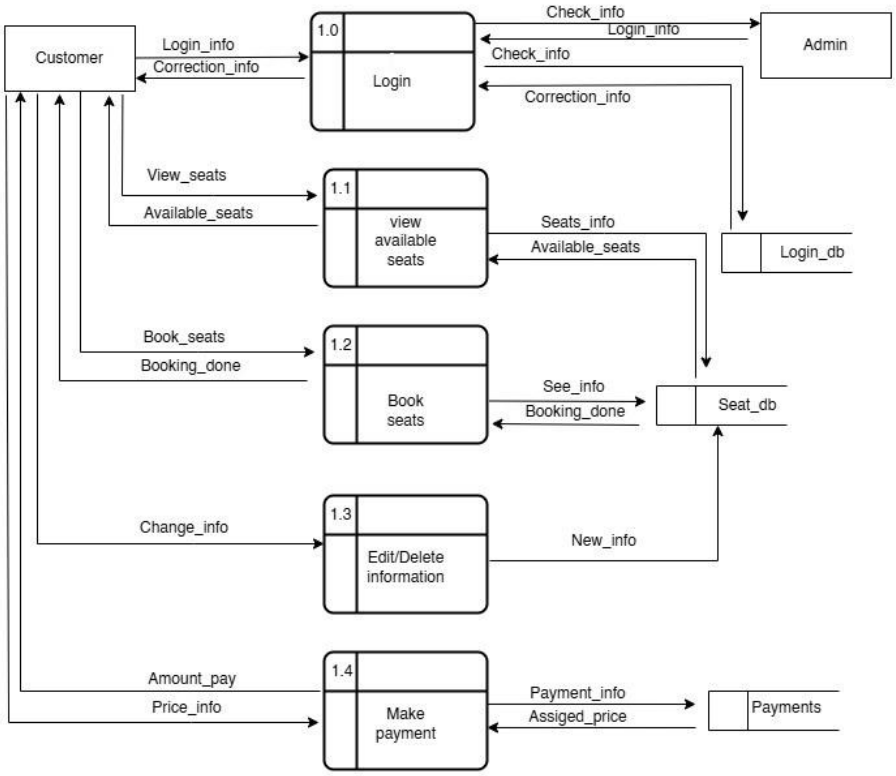


Figure 6.2 Level-1-DFD

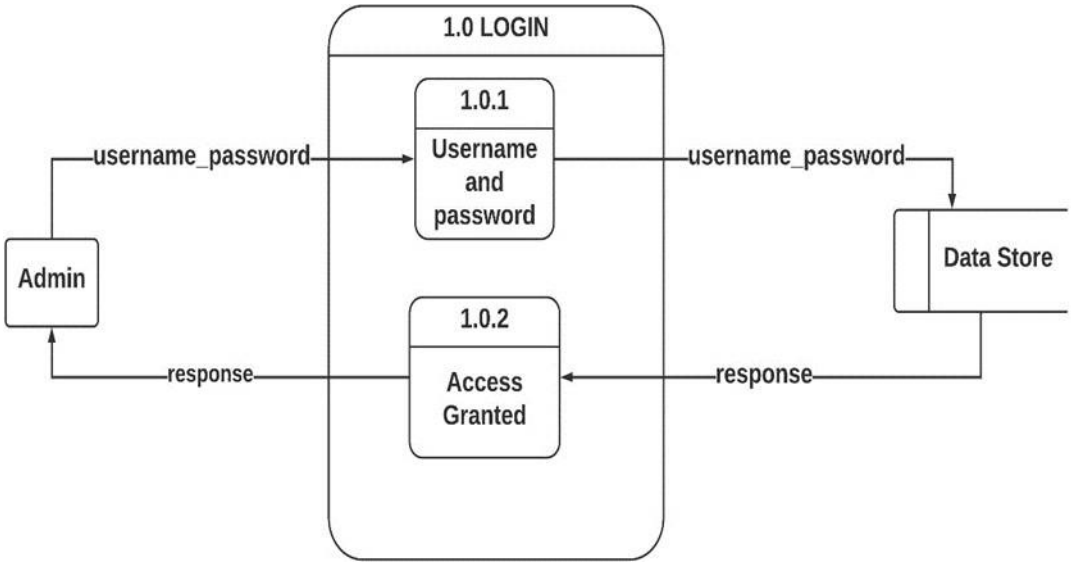


Figure 6.3 Level-2-DFD Admin Login

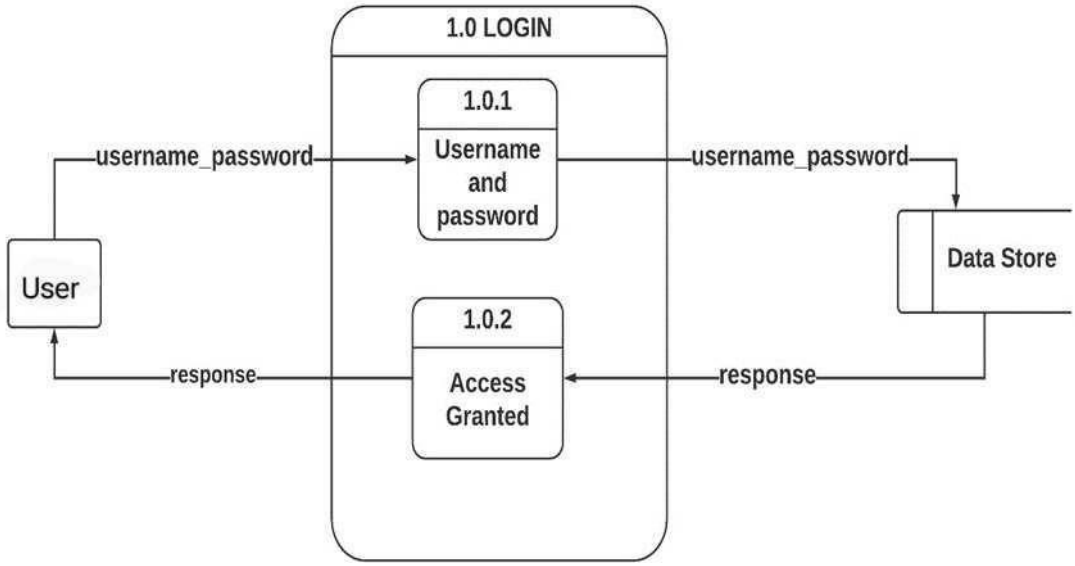


Figure 6.4 Level-2 DFD User Login

6.2 Entity Relationship Diagram(ER-Diagram)

An Entity Relationship Diagram is a diagram that represents relationships among entities in a database. It is commonly known as an ER Diagram.

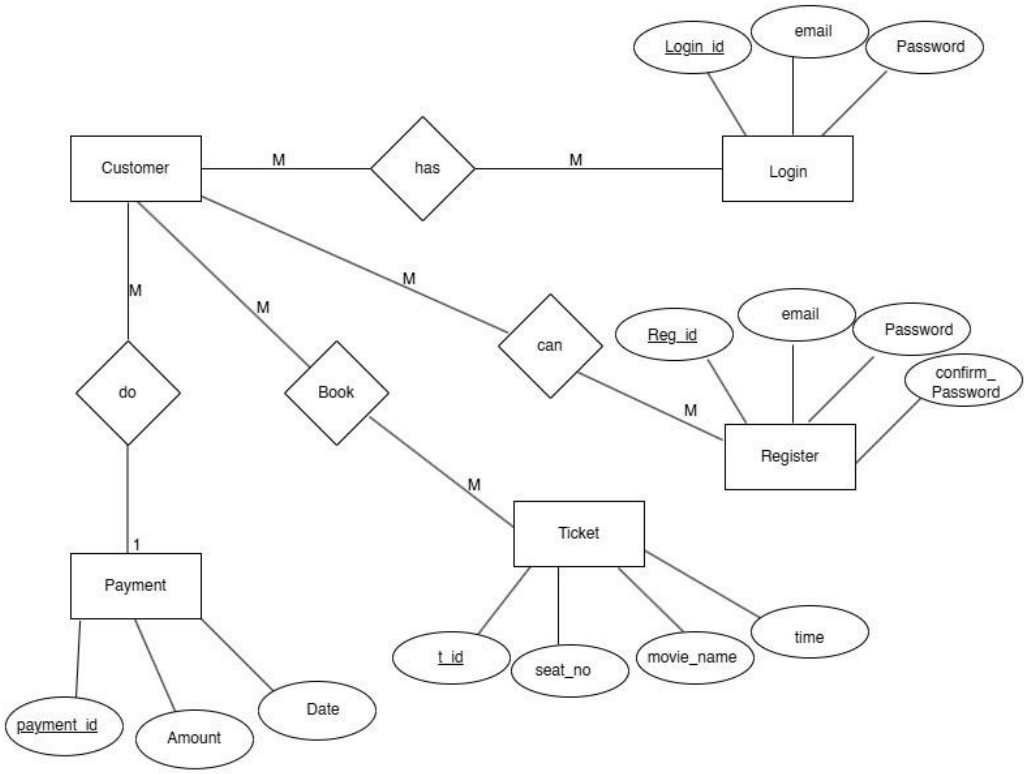


Figure 6.5 ER Diagram

6.3 Use Case Diagram

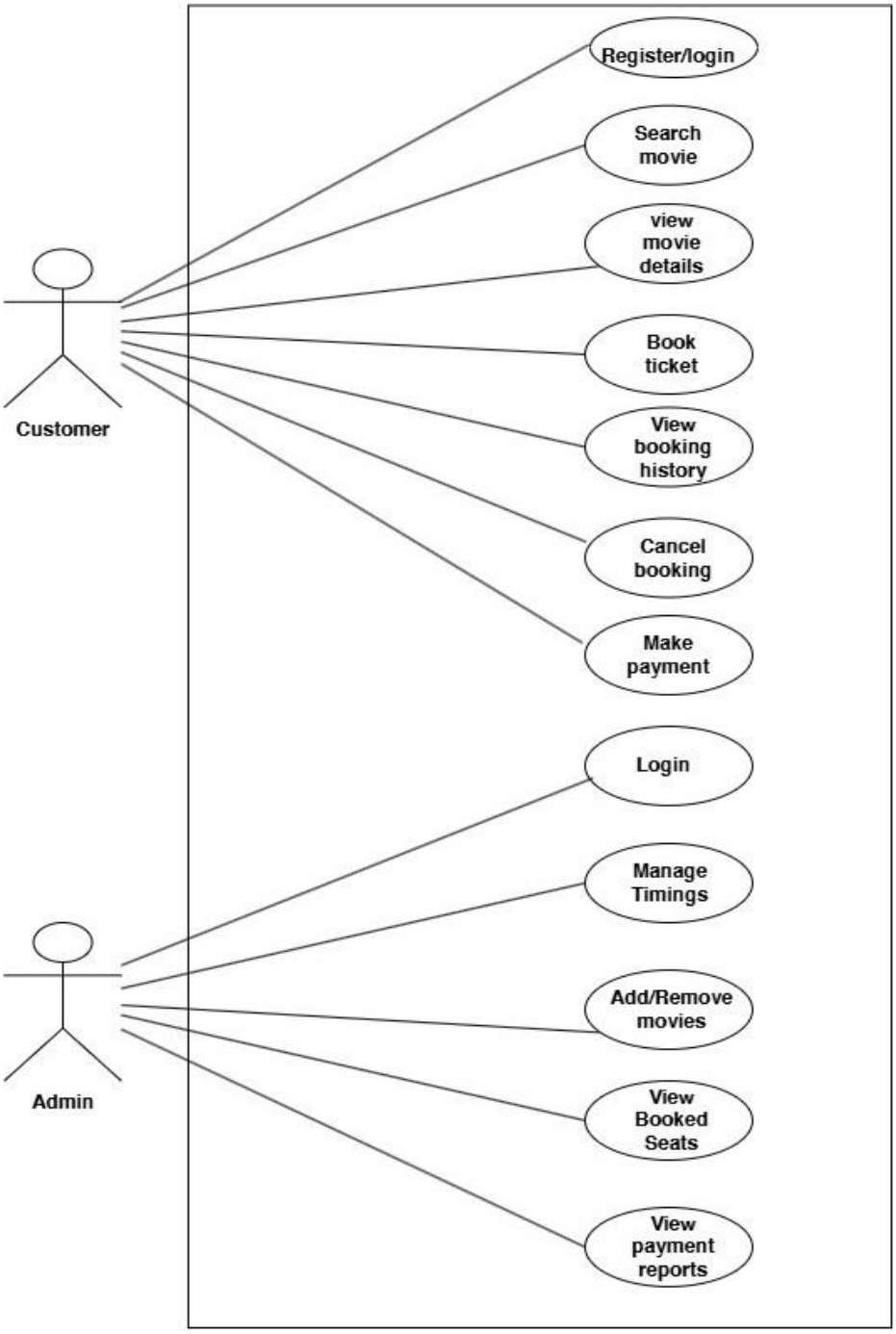


Figure 6.6 Use case diagram

7. Methodology

We are employing Agile Methodology as our chosen approach for project development. This model follows an iterative process that emphasizes flexibility, collaboration and continuous improvement. It follows the iterative as well as incremental approach that emphasizes the importance of delivering of working product very quickly. Agile methodology is a project management framework that breaks projects down into several dynamic phases, commonly known as sprints. The Agile framework is an iterative methodology. After every sprint, teams reflect and look back to see if there was anything that could be improved so they can adjust their strategy for the next sprint.

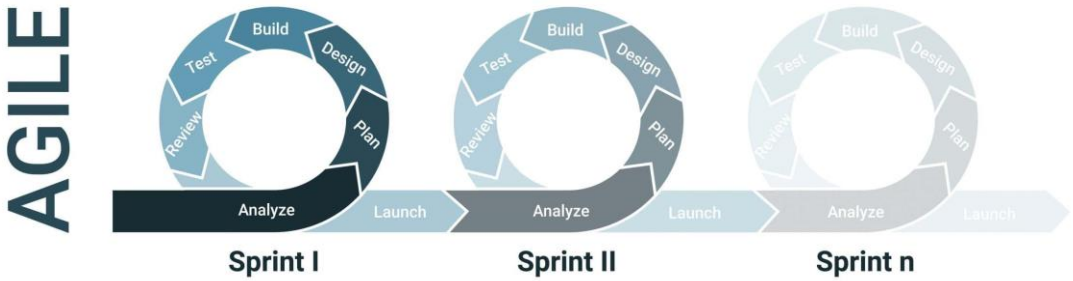


Figure 7.1 Agile Method

(Paliwa, 2024)

The project was divided into sprints, with each sprint focusing on specific features.

Sprint 1: User Registration and Authentication

Frontend: HTML/CSS for forms, JavaScript for validation.

Backend: PHP for handling registration and login logic.

Database: User data stored in the Users table with encrypted passwords.

Testing: Ensured successful login/logout and session handling.

Sprint 2: Movie Listings and Search Functionality

Frontend: Designed a responsive interface to display movie listings dynamically.

Backend: PHP used to fetch and display data from the Movies table.

Search and Filters: Added search functionality (e.g., by title, genre) using SQL queries.

Sprint 3: Seat Selection Interface

Frontend: Developed an interactive seat selection UI using JavaScript. Seats dynamically update their status (available/booked).

Backend: Integrated PHP and MySQL to validate seat availability.

Sprint 4: Payment Gateway Integration

Payment : Created a secure payment.

Booking Confirmation: Updated the database after successful payments.

Sprint 5:Admin Dashboard

Developed a separate admin interface to manage: Adding, editing,viewing and deleting movies.

8. Testing

Software testing is the process of evaluating a software product to ensure it functions as intended. It involves running test cases to identify bugs and verify that the software meets expectations.

Purpose To find issues early in the development process, To reduce the risk of software failure, To improve performance, and to prevent bugs.

We had performed following test cases :

Test Cases

Test Case 1

Title: Module testing of a Login function– Authentication of users

Description: A user should be able to log in with a registered email and password.

Precondition: The user and admin must have an email and password that is registered.

Test Case#	Test Case	Test Steps	Test Data	Expected Results
001.	Verify Login	1. Go to Login 2. Enter Email 3. Enter Password 4. Click Login	email = ab Password= my	Login Unsuccessful
002.	Verify Login	1. Go to Login 2. Enter email 3. Enter Password 4. Click Login	email = aa Password= aa	Login Successful

003.	Verify Login	1. Go to Login 2. Enter email 3. Enter Password 4. Click Login Repeat the process for 3 times	email =aa Password=123	Login Unsuccessful.

Table 1 Test case for login

Test Case 2:

Title: Module Testing of Dashboard function –Admin Add New Movies

Description: An admin should be able to add the new movie information.

Test Steps:

1. After successfully login to Admin.

2. Enter the name of the movie and its description.

Test Case#	Test Case	Test Steps	Test Data	Expected Result
004	Verify Dashboard	1. Go to Admin PAGE 2. Add New Movie	1	Adding New Movie is Successful.

Table 2 Test case for add movie

Test Case 3:

Title: Module Testing of Dashboard function: View Movies Details

Description: A user can view the details of the movies.

Test Steps:

1. After successfully login to Dashboard.

2. Can view the movie name, genre, duration, duration, description and poster.

Test Case#	Test Case	Test Steps	Test Data	Expected Result
006	Verifying View Movie Details	1. Goto Dashboard 2. View Movie Details	4	Viewed details successfully.

Table 3 Test case for View movie

Test Case 4:

Title: Module Testing of Dashboard function: Edit Movies Details

Description: An admin should be able to update the movie details.

Test Steps:

1. After successfully login to the Admin Dashboard.

2. Enter Edit Button.

Test Case#	Test Case	Test Steps	Test Data	Expected Result
007	Verifying Edit the Movie Details	1. Goto Dashboard 2. Enter Edit Button	5	Updated Successfully

Table 4Test case for Editing movies info

Test Case 5:

Title: Module Testing of Dashboard Function: Delete Details

Description: An admin can delete the movie details.

Test Steps:

1. After successfully logging in to the admin dashboard

2. Can delete the Movie details.

Test Case#	Test Case	Test Steps	Test Data	Expected Result
008	Verify Delete Details	1. Goto Dashboard 2. Enter Delete	2	Deleted details successfully.

Table 5 Test case for delete movie

Test Case 6:

Title: Module Testing of Dashboard Function: Exit

Description: A system will be stopped if the exit operation is done.

Test Steps:

1. After successfully login to the Dashboard

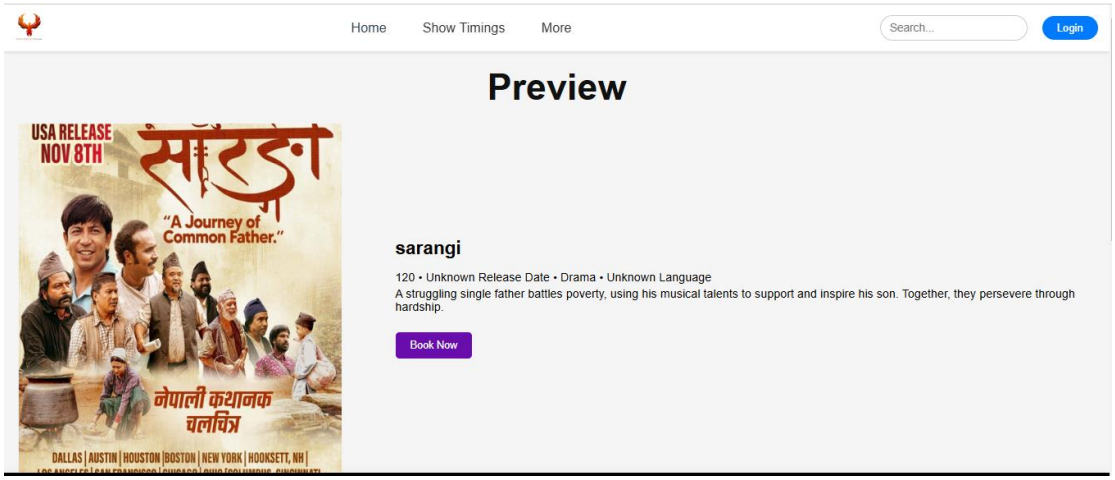
2. Can exit from the admin Dashboard.

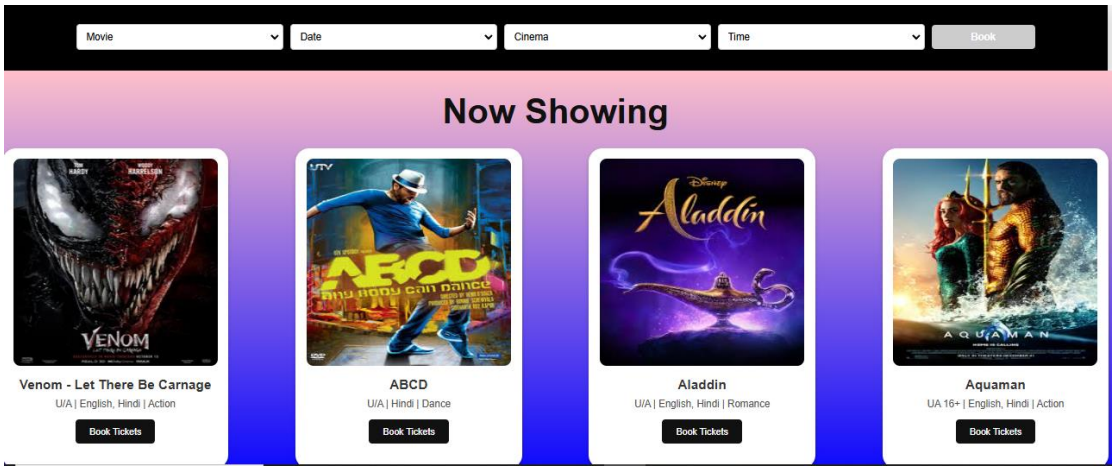
Test Case#	Test Case	Test Steps	Test Data	Expected Result
008	Verify Exit	1. Goto Dashboard 2. Enter Exit	6	Admin will exit from the admin Dashboard.

Table 6 Test case for exit

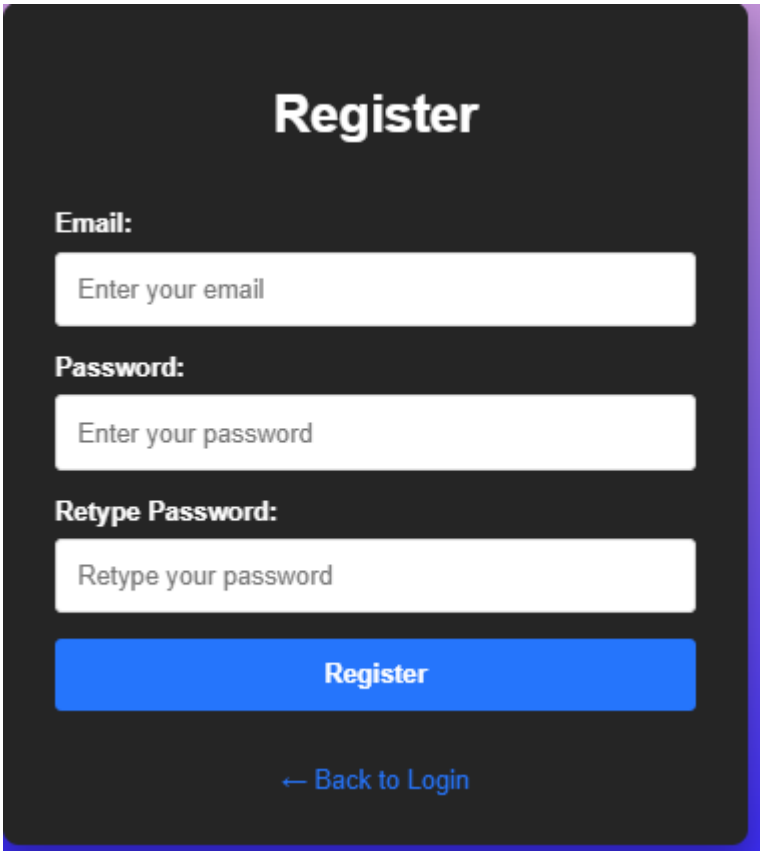
9. Project Result

Home Page

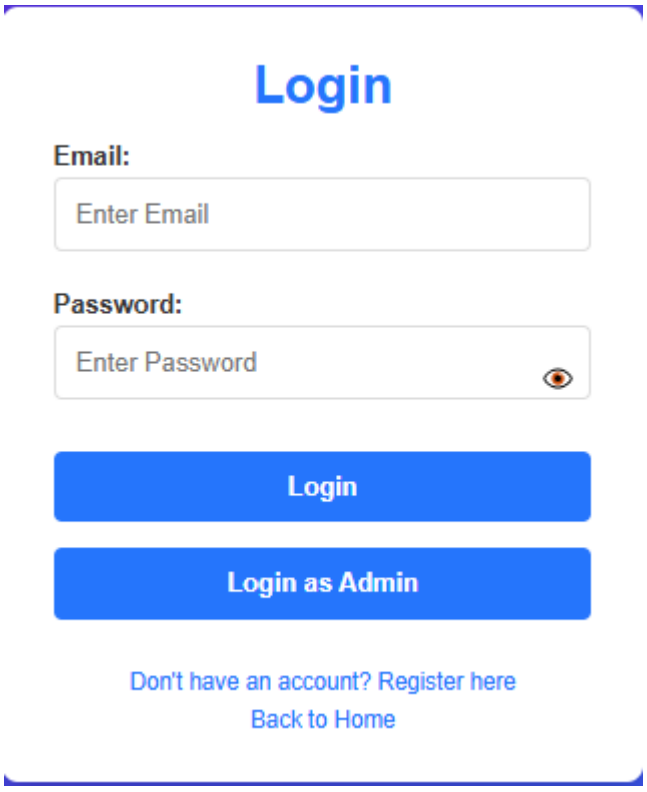




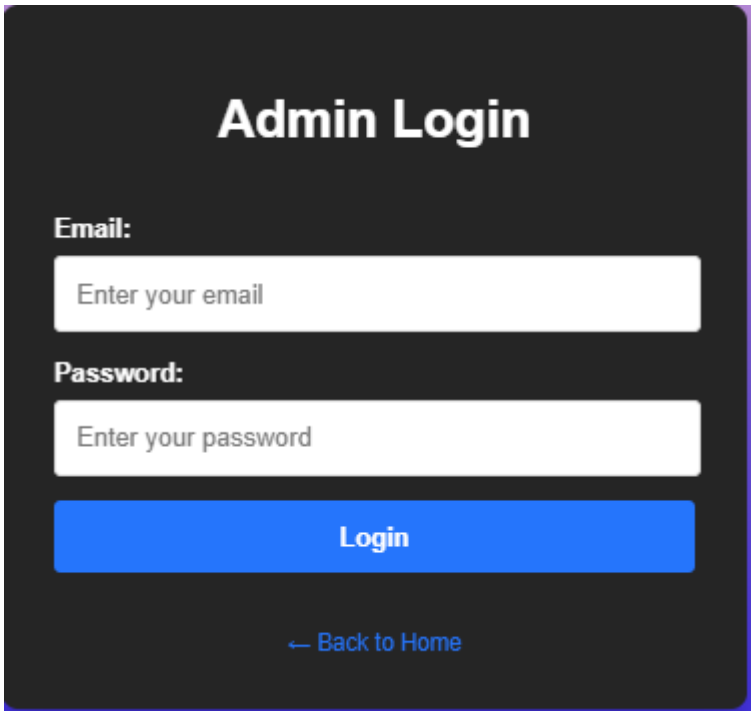
User Registration



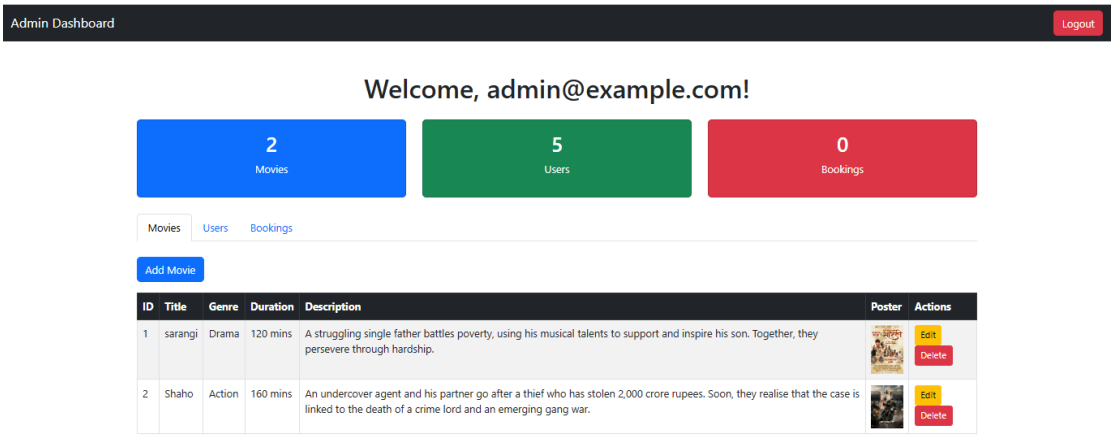
User Login



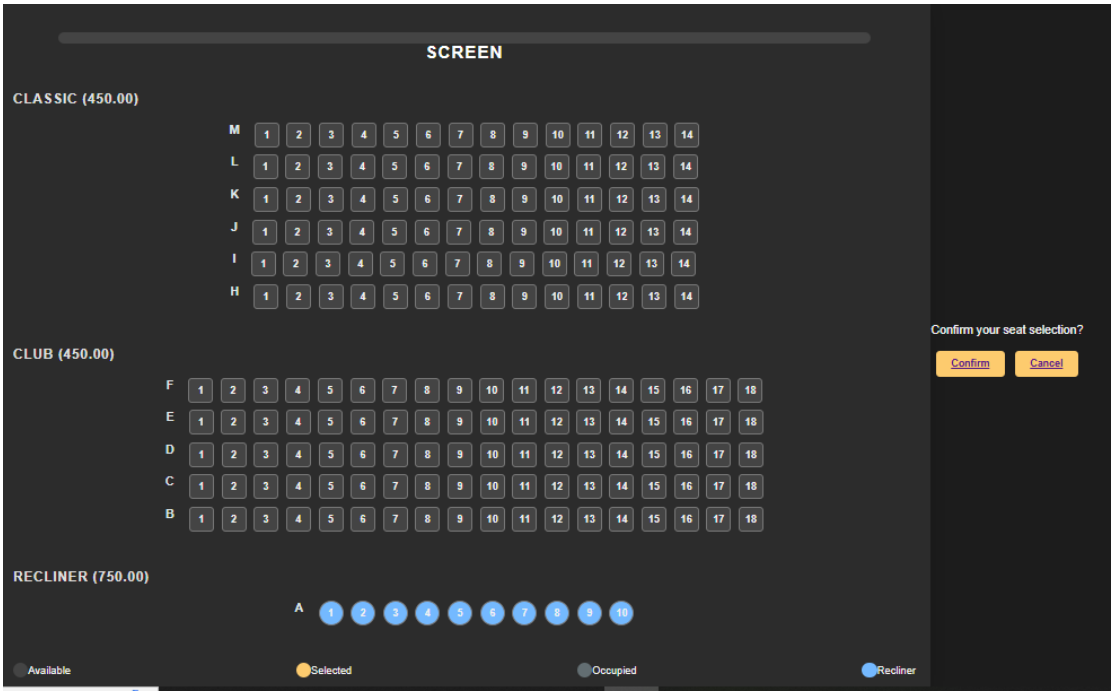
Admin Login



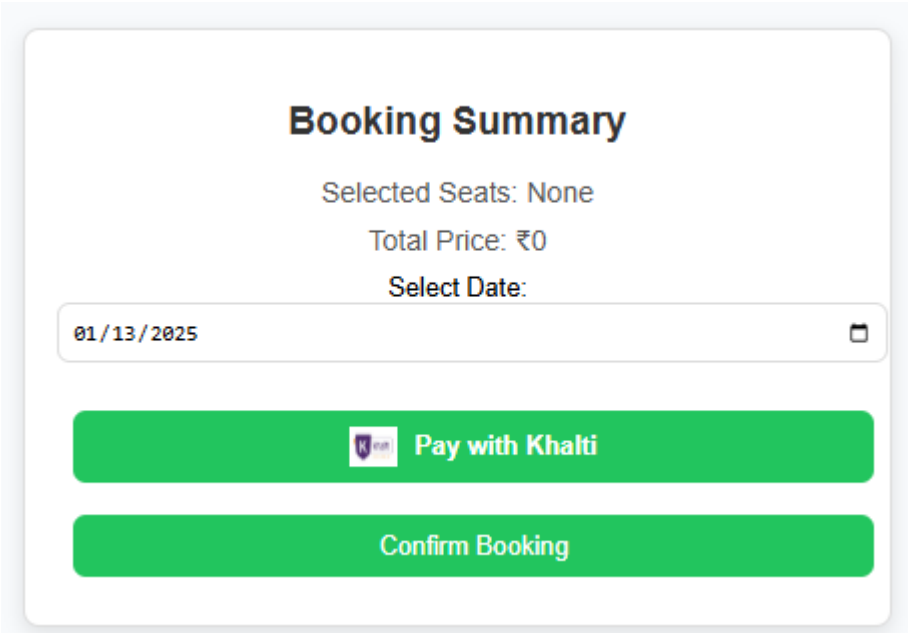
Admin Dashboard

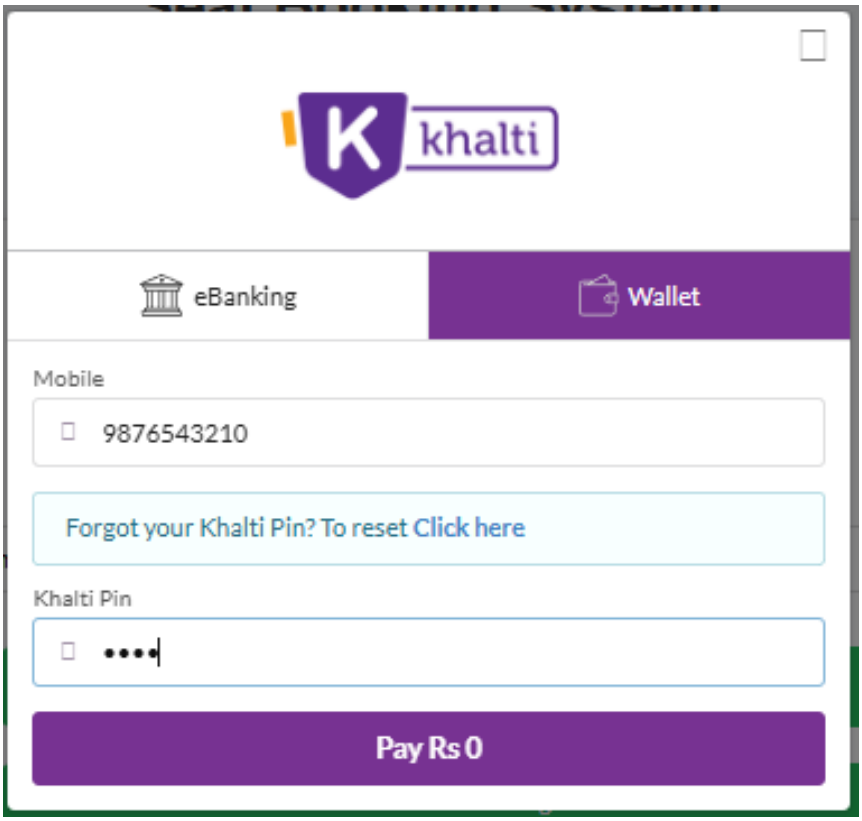


Seat Selection



Payment system





10. Future Enhancements

- Integration with Multiple Payment Gateways
- Push Notifications
- User Feedback System

11. Conclusion

The **Movie Ticket Booking System** was successfully implemented with all planned features.
Key achievements include:

- A secure, user-friendly platform for movie ticket booking.
- An intuitive admin dashboard for effective management.
- Reliable real-time updates for seat selection and payment processing.

This project demonstrates the practical application of web technologies like HTML, CSS, JavaScript, PHP, and MySQL. It also highlights the importance of structured development, testing, and user feedback in creating a functional and robust system.

13. References

Paliwa, P. (2024, september 23). *What is Agile Methodology?* Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org

researchgate.net. (January 2024). *A_CASE_STUDY_ON_ONLINE_TICKET_BOOKING_SYSTEM_PROJECT*. SSRN Electronic Journal..

Scribd.com. (Oct 29,2018). *PHP and MYSQL project report on movie ticket booking system.* FreeProjectz.com.