

bookmyshow

Design Document

Jeevan Reddy T	(2023BCS069)
David Sure	(2023BCS066)
Sameer Y	(2023BCS073)

TABLE OF CONTENTS

[1. Introduction](#)

[2. Design considerations](#)

[2.1 Assumptions](#)

[2.2 Design constraints](#)

[2.3 Design methodology](#)

[2.4 System environment](#)

[3. Architecture](#)

[3.1 System Design](#)

[3.2 Functional Decomposition tree](#)

[3.3 Context Diagram](#)

[3.4 Data flow Diagram](#)

[3.5 Data Dictionary](#)

[4. Component Design](#)

[4.1 Activity Diagram \(system\)](#)

[4.2 Activity Diagram \(admin\)](#)

[5. User Interface Design](#)

1. Introduction

BookMyShow is a web-based application that operates within a centralized network, designed for booking tickets for movies, concerts, theater performances, and other entertainment events. The system allows users to reserve seats, cancel bookings, and check showtimes, ensuring quick and hassle-free reservations.

BookMyShow is developed to digitize and streamline the traditional ticket booking process, making it more efficient and accessible. It maintains customer details, event listings, ticket availability, and booking history.

To achieve this design, **HTML, JavaScript, and CSS** were used for the front-end interface, providing a user-friendly experience, while the **back-end was developed using MySQL** to handle database management, transactions, and user authentication. The implemented software enhances customer experience and improves event management operations.

It is recommended that additional functionalities such as **email notifications for ticket confirmation and reminders, integration of online payments via credit/debit cards and digital wallets, as well as QR-based ticket validation for seamless event entry** be incorporated into the system for further enhancement.

2. Design Considerations

This section outlines key issues that need to be addressed before finalizing the complete design solution. This document is based on **Design Document v1.0** and aligns with the **SRS document** for reference in case any part is unclear or requires further elaboration.

2.1 Assumptions

The **BookMyShow** design assumes specific software and hardware requirements as outlined in the SRS. Both the user interface and the database must meet the specified environmental operating conditions. The system is designed to function within the following constraints:

- The system will have access to the necessary computing resources, including **sufficient memory, storage space, and CPU processing power** for smooth execution.
- The user machine will have the **MySQL database components installed**, as they are required for system operation.
- The necessary **database setup and configurations** will be pre-installed before system deployment.

2.2 Design Constraints

The **BookMyShow** system shall be a **web-based platform** designed using the following technologies:

- **Front-end:** HTML, CSS, JavaScript
- **Back-end:** PHP, MySQL database

2.3 Design Methodology

The **Waterfall Model** is chosen for the system development process. This model is well-suited for structuring and documenting the system's functionality and ensures a **systematic approach** to design. The development will follow these steps:

1. **Database Design:** Structuring the database to store customer details, event listings, bookings, and payment records.
2. **Creating Relationships:** Establishing logical relationships between different entities (e.g., users, events, bookings, and payments).

3. **User Interface & System Process Design:** Developing the user interface for seamless navigation and ensuring system processes align with business logic.

2.4 System Environment

The system is designed with a **scalable and secure architecture**, ensuring it can accommodate **future expansions and modifications** based on evolving business needs. Key aspects of the system environment include:

- **Scalability:** The system will allow **easy expansion and modification** to support additional features, such as more event categories or third-party integrations.
- **Security:** Proper measures will be implemented to protect **user data, payment transactions, and system access** from unauthorized use.

This design approach ensures that **BookMyShow** remains flexible, robust, and capable of handling a growing user base efficiently.

3. Architecture

3.1. System design

After the system has been implemented the mapping shall take place according to following:

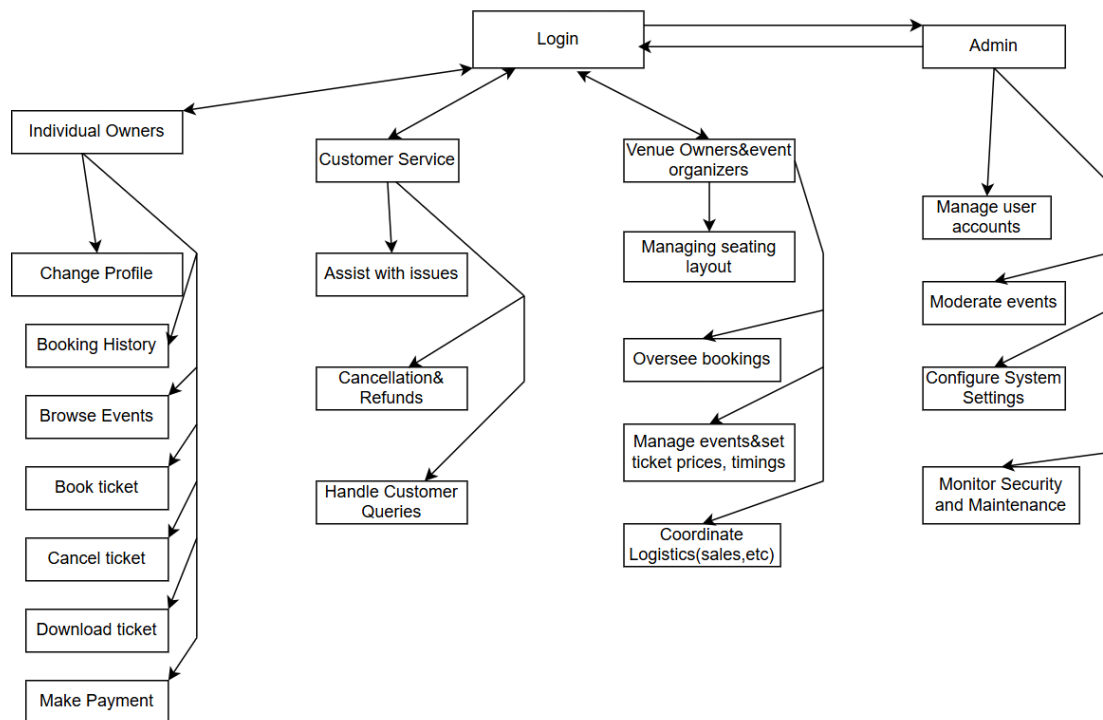


Figure 1 System Design

3.2. Functional Decomposition Tree

The main functions of the system are decomposed into smaller sub functions or sub-modules and further. The System shall take place following structure of organization after implementation. The decomposition is stable and functions should be made highly cohesive.

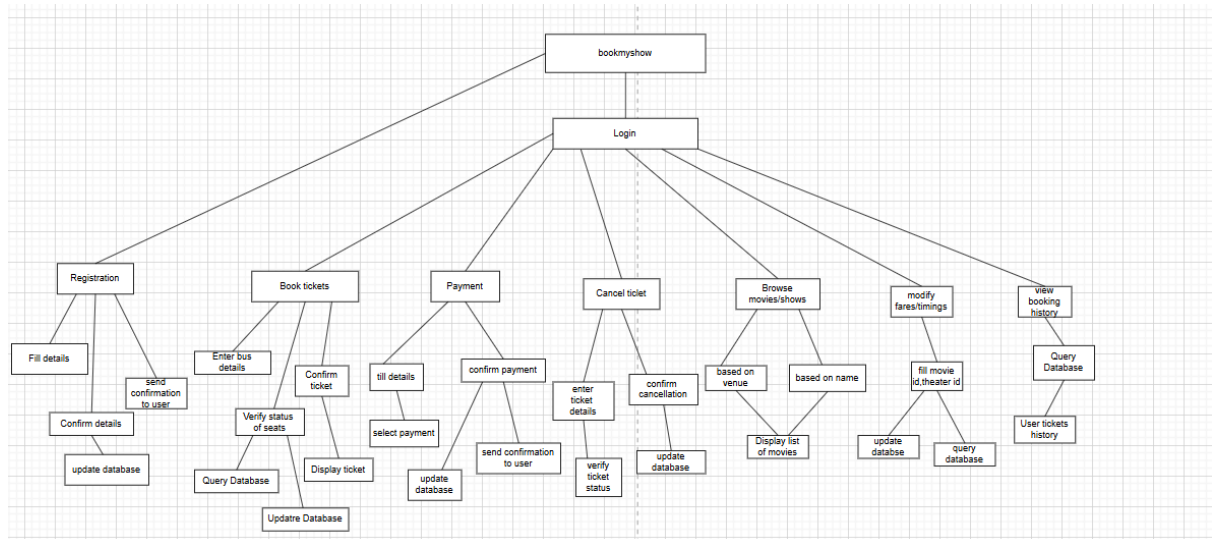


Figure 2 Functional Decomposition Tree

Modules involved in the system:

1. **User Registration:** This module handles new user/vendor registration.
 - **1.1 Fill details:** Input user-entered details via the interface and store them in a data structure.
 - **1.2 Confirm details:** Verify details stored in the data structure.
 - **1.2.1 Update database:** Add new user to the database.
 - **1.3 Send confirmation:** Notify the user about registration credentials and confirmation via email.
2. **Browse Movies & Shows:** Allow users to explore available movies and events.
 - **2.1 Select Location:** Choose a city/region to filter available shows.
 - **2.2 Search Movies/Events:** Search for a movie/event by name or category.

- **2.3 View Showtimes:** Display available time slots for a selected movie/event.
 - **2.3.1 Query database:** Fetch show details based on location and selection.

3. **Book Tickets:** Enable users to book movie/event tickets.

- **3.1 Select Show:** Choose a specific show from available options.
- **3.2 Choose Seats:** Pick seats from the available layout.
 - **3.2.1 Query database:** Check seat availability.
 - **3.2.2 Update database:** Reserve selected seats in the database.
- **3.3 Confirm Booking:** Review booking details before proceeding to payment.

4. **Payment Processing:** Handle payment transactions.

- **4.1 Fill payment details:** Enter payment mode and credentials.
- **4.2 Confirm payment:** Validate payment details and process the transaction.
 - **4.2.1 Update database:** Update booking status upon successful payment.
- **4.3 Send confirmation:** Send booking confirmation and e-ticket via email/SMS.

5. **Cancel Booking:** Allow users to cancel booked tickets within allowed limits.

- **5.1 Enter booking details:** Input booking reference number and credentials.
 - **5.1.1 Verify status:** Check if the ticket is eligible for cancellation.
 - **5.2 Confirm cancellation:** Confirm cancellation and process a refund (if applicable).
 - **5.2.1 Update database:** Mark the ticket as canceled and update seat availability.
6. **Modify Booking:** Enable users to change their booking (if allowed).
- **6.1 Enter booking details:** Provide the booking reference number.
 - **6.2 Modify selection:** Choose new showtime, seats, or other available options.
 - **6.2.1 Query database:** Check availability for modifications.
 - **6.2.2 Update database:** Update the booking details accordingly.
7. **View Booking History:** Let users access their past and upcoming bookings.
- **7.1 Query database:** Fetch booking history for a particular user.
 - **7.2 Display list:** Show the retrieved booking history on the interface.
8. **Add/Remove Movies or Events (Admin/Vendor Module):** Manage available listings.
- **8.1 Fill details:** Input movie/event details to be added/removed.

- **8.1.1 Update database:** Reflect the changes in the system.
- **8.1.2 Query database:** Verify details before applying changes.

9. **Modify Ticket Prices (Admin/Vendor Module):** Adjust pricing for different shows.

- **9.1 Enter Movie/Event ID:** Identify the movie/event to modify pricing.
 - **9.1.1 Query database:** Fetch existing ticket prices.
 - **9.1.2 Update database:** Apply updated pricing.

10. **Search Theaters or Venues:** Allow users to look up nearby venues.

- **10.1 Enter location:** Provide a city or locality.
- **10.2 Query database:** Retrieve a list of available theaters/venues.
- **10.3 Display results:** Show the retrieved list on the interface.

11. **Loyalty Program & Offers:** Provide discounts and loyalty benefits.

- **11.1 Apply promo codes:** Users can enter discount codes while booking.
- **11.2 Earn loyalty points:** Users accumulate points based on transactions.
 - **11.2.1 Update database:** Add loyalty points to the user account.

- **11.3 Redeem benefits:** Allow users to redeem loyalty points for discounts.

12. **Customer Support:** Provide assistance for user queries and issues.

- **12.1 Raise a query:** Users can submit complaints or queries.
- **12.2 Track request:** View the status of a support request.
- **12.3 Query database:** Fetch previous interactions with support.

3.3 Context diagram

Context diagram describes the main actors interacting with the system.

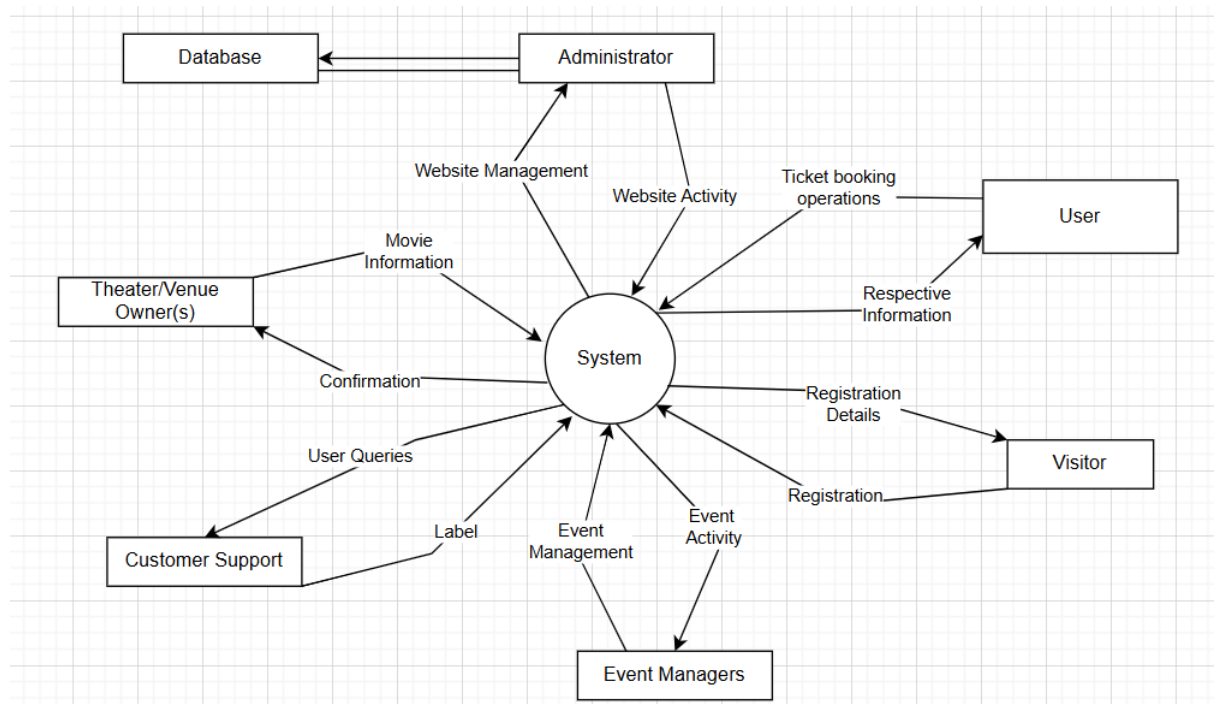


Figure 3 Context diagram

3.4. Data flow diagrams

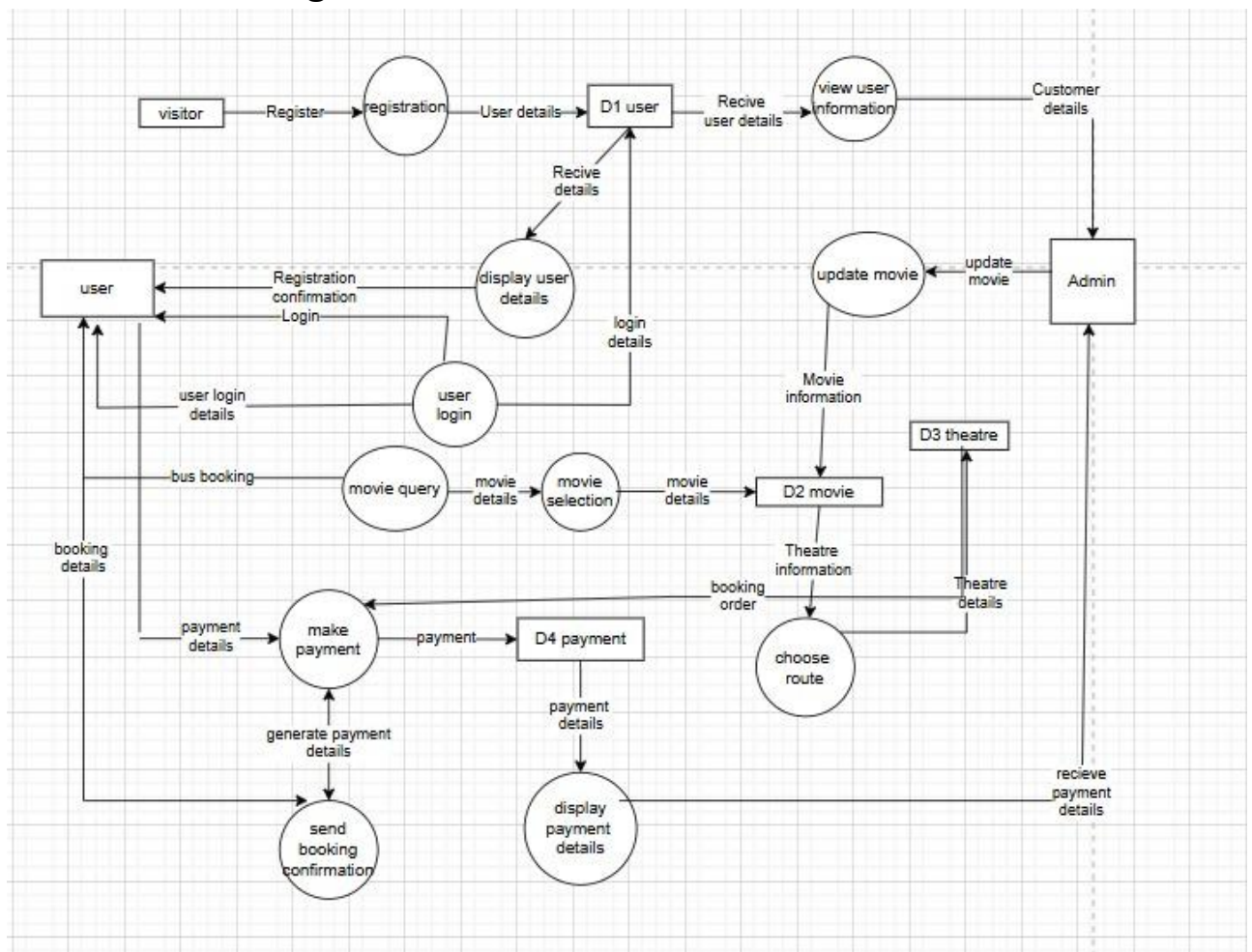


Figure 4 Level 1 DFD for system

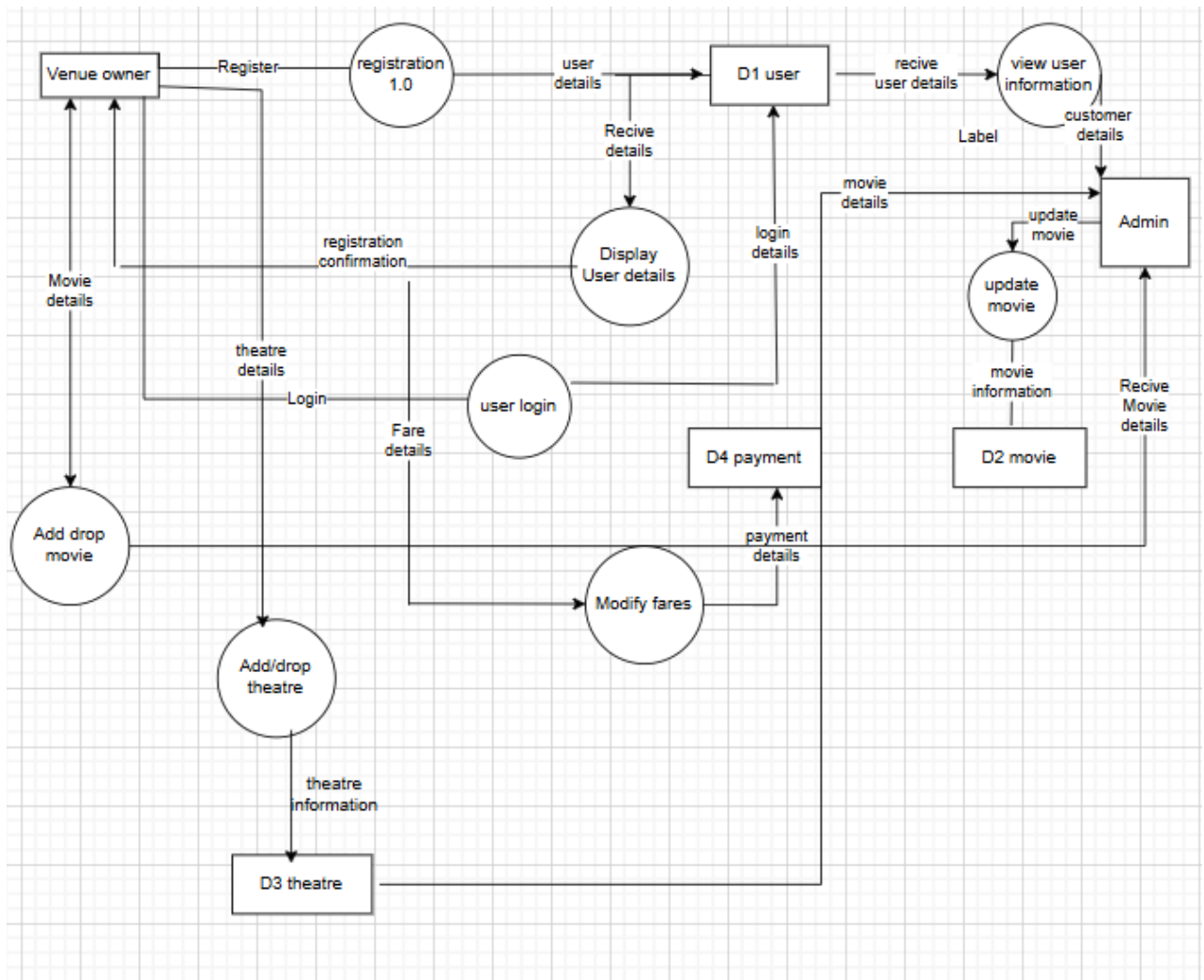


Figure 5 Level 1 DFD for Venue Owner

3.5. Data dictionary

Table 1. Admin

Field	Type	NULL	Default
uname	varchar(30)	NO	None
Password	varchar(30)	NO	None
Name	varchar(30)	NO	None

Table 2. Movie

Field	Type	NULL	Default
Movieid	varchar(30)	NO	None
Movie	varchar(30)	NO	None
Type_ac	char(3)	NO	None
Type_sl	char(3)	NO	None

Table 3. Card

Field	Type	NULL	Default
Num	varchar(16)	NO	None
type	varchar(50)	NO	None
Expdate	date	NO	None
cvv	int(3)	NO	None
Bank	varchar(30)	NO	None

Table 4. Net banking

Field	Type	NULL	Default
uname	varchar(30)	NO	None
password	varchar(30)	NO	None
bank	varchar(30)	NO	None

Table 5. User

Field	Type	NULL	Default
uid	varchar(11)	NO	None
name	varchar(20)	NO	None
email	varchar(50)	NO	None
mobile	varchar(10)	NO	None

Table 6. Ticket

Field	Type	NULL	Default
movieid	int(11)	NO	None
tid	int(11)	YES	NULL
uid	varchar(11)	YES	NULL
Status	varchar(11)	YES	NULL
dot	timestamp	NO	current_timestamp

Table 7. Theater

Field	Type	NULL	Default
tid	int(11)	NO	NULL
bid	varchar(11)	YES	NULL
loc	varchar(10)	YES	NULL
fare	Double	YES	NULL
date	Date	NO	None
start_time	Time	YES	NULL
end_time	Time	YES	NULL
aval_seats	Int(10)	NO	100
max_seats	Int(10)	NO	100

4.Component design

4.1. Activity Diagram for system

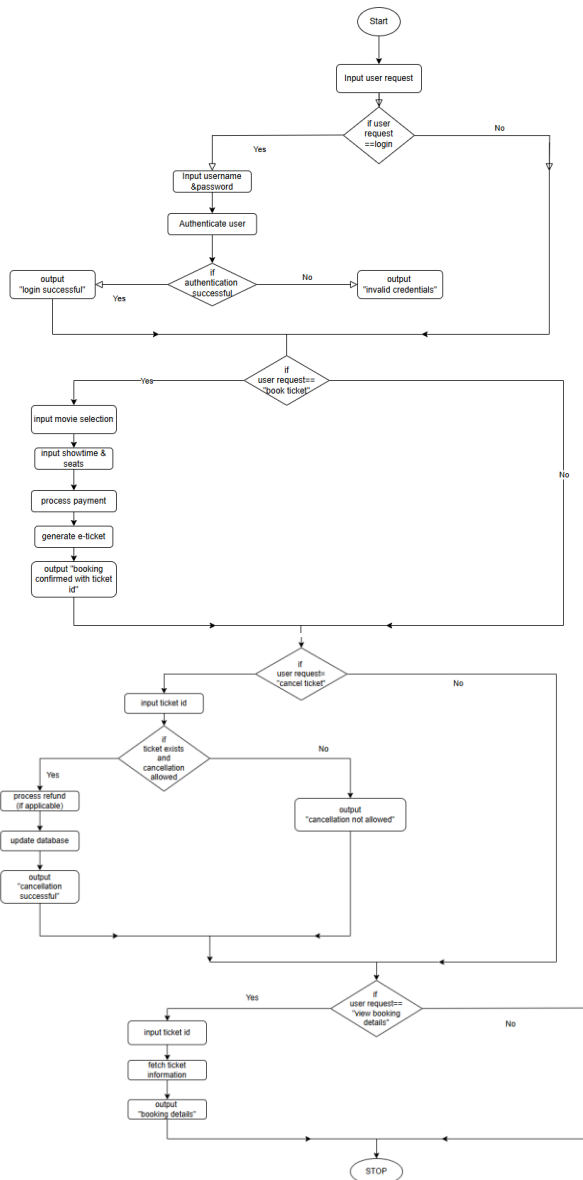


Figure 6 Activity diagram of entire system

4.2. Activity Diagram for admin

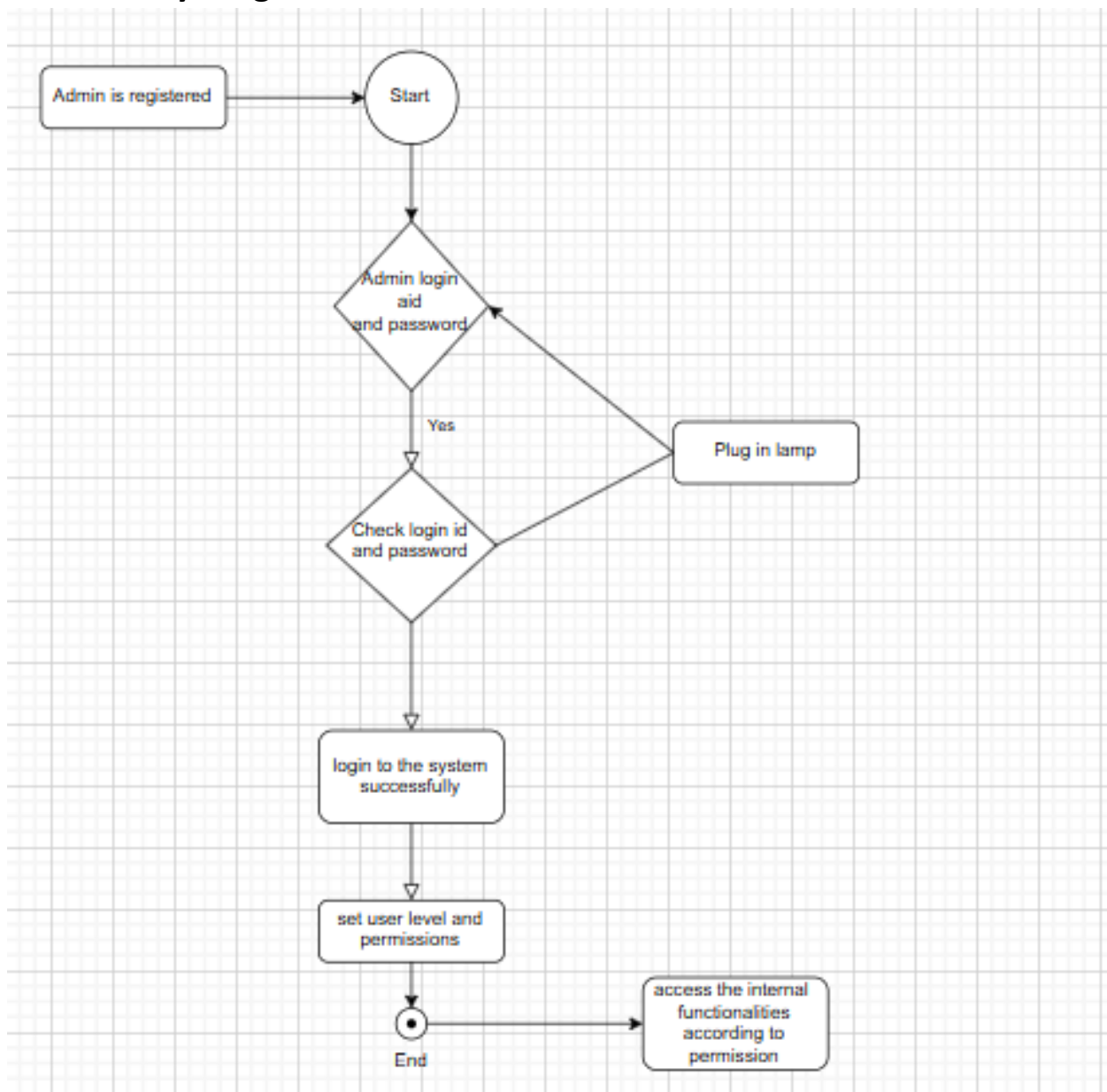


Figure 7 Activity Diagram for admin

5. UI Design Principles

Structure Principle: The interface is well-organized, grouping related features together while keeping unrelated elements separate.

Simplicity Principle: The UI is intuitive and easy to navigate. In case of mistakes, the system provides clear error messages.

Visibility Principle: All important functions are easily accessible without overwhelming the user with too many options.

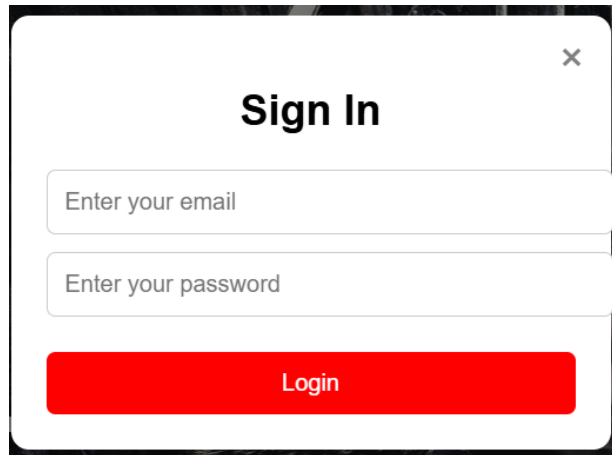
Feedback Principle: System messages notify users about successful actions, errors, or required confirmations.

Reuse Principle: Consistent naming and layouts are used across different pages to reduce confusion.

Main UI Screens

1. **Home Page:** Acts as the main navigation hub for users.
2. **Login Page:** Users enter credentials to access their accounts.
3. **Movie Selection Page:** Users can browse, filter, and select movies.
4. **Seat Selection Page:** Allows users to choose seats for their selected show.
5. **Payment Page:** Handles ticket payments through secure third-party services.
6. **Booking Confirmation Page:** Displays confirmed booking details with an option to download or print tickets.
7. **Booking History Page:** Shows past and upcoming bookings.

8. **Cancellation Page:** Users can cancel eligible bookings after logging in.
9. **Profile & Settings Page:** Users can manage their account details, preferences, and notifications.
10. **Contact Us Page:** Provides customer support options for inquiries and feedback.



A white modal box with a close button (X) in the top right corner. The title "Sign In" is centered at the top. Below the title are two input fields: "Enter your email" and "Enter your password". At the bottom is a red button labeled "Login".

Figure 8 Login page

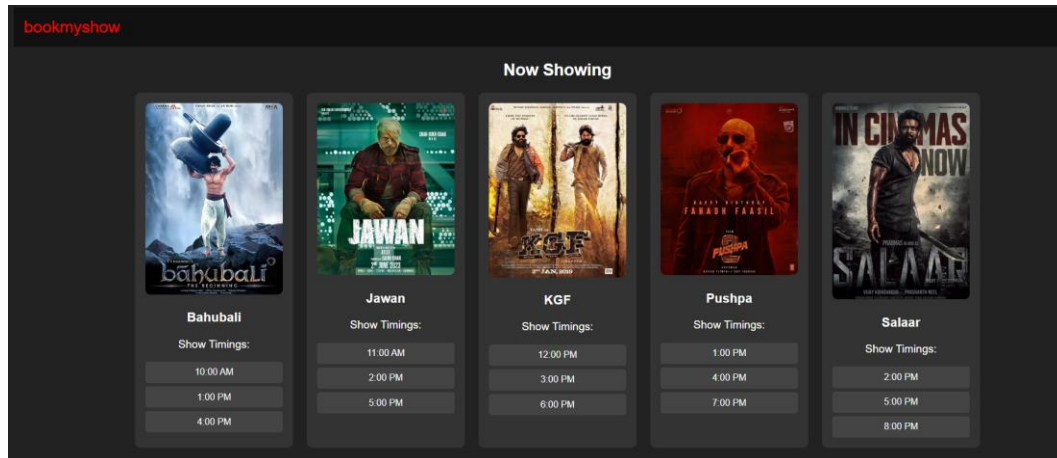


Figure 9 Movie Selection Page

bookmyshow

Seat Booking

Movie: Jawan

Time: 11:00 AM

SCREEN

A	1	2	3	4	5	6	7	8	9	10
B	1	2	3	4	5	6	7	8	9	10
C	1	2	3	4	5	6	7	8	9	10
D	1	2	3	4	5	6	7	8	9	10
E	1	2	3	4	5	6	7	8	9	10
F	1	2	3	4	5	6	7	8	9	10
G	1	2	3	4	5	6	7	8	9	10
H	1	2	3	4	5	6	7	8	9	10
I	1	2	3	4	5	6	7	8	9	10
J	1	2	3	4	5	6	7	8	9	10

Tickets: 2

Selected Seats: B-3, B-7

Total: Rs. 220

Proceed to Payment

Figure 10 Seat Selection Page

Select Payment Method

Movie: Jawan

Time: 11:00 AM

Selected Seats: B-3, B-7

Total Amount: Rs. 220

Credit/Debit Card

UPI (PhonePe)

UPI (GPay)

UPI (Paytm)

Enter Payment Details

Card Number:

XXXX-XXXX-XXXX-XXXX

Name on Card:

Full Name

Expiry Date:

MM/YY

CVV:

XXX

Proceed to Pay

Figure 11 Payment Page

🎉 Booking Confirmed! 🎉

Movie: Pushpa

Time: 1:00 PM

Seats: C-9

Total Amount Paid: Rs. 110

Payment Method: UPI

Go to Home

Figure 12 Booking Confirmation Page

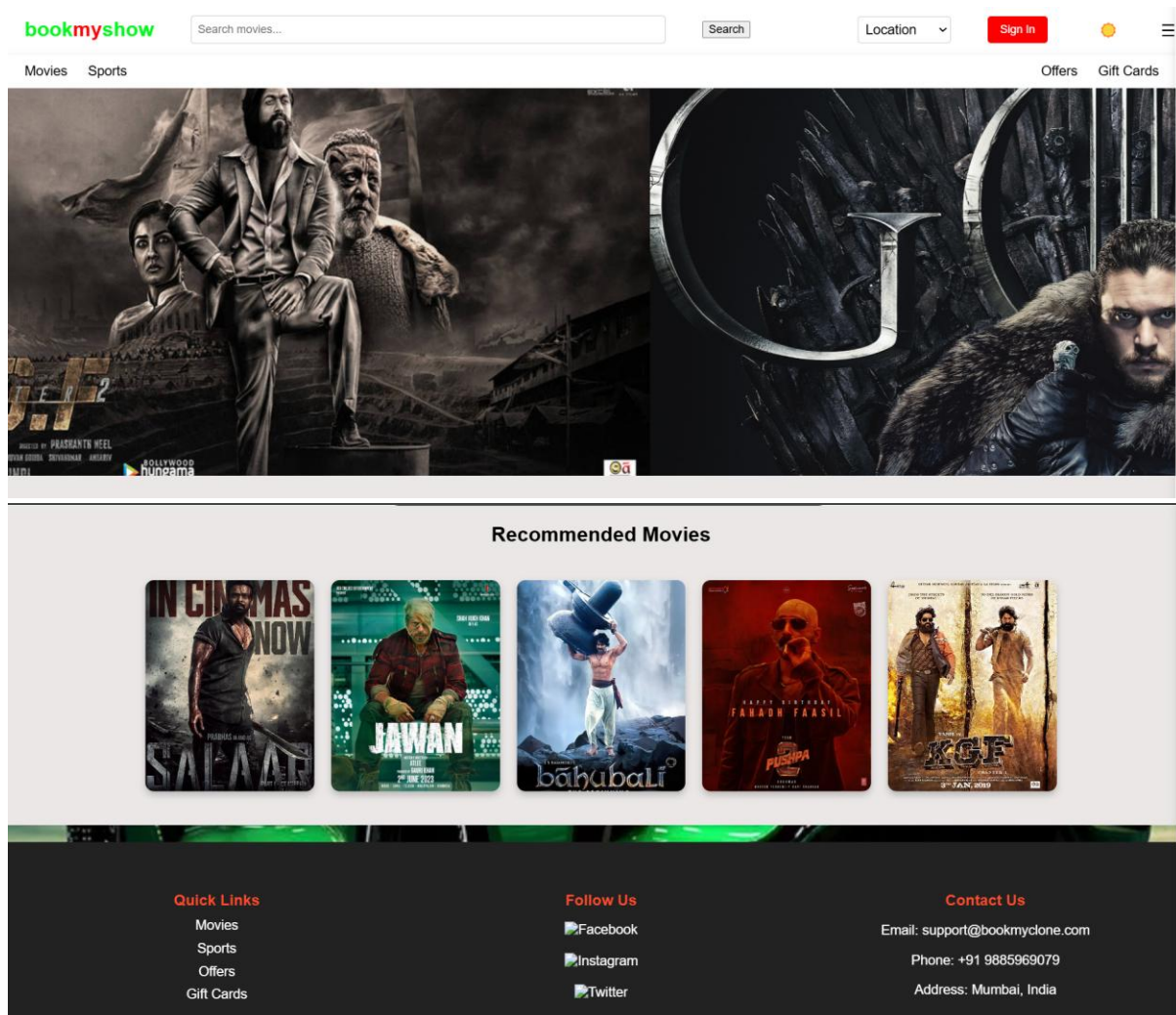


Figure 13 Home Page