

Project Name: BoxOfficeBuddy

Branch DAC March-2024

Documentation On

**“BoxOfficeBuddy”**

DAC2024

**Guided By: Snehal Somvanshi**

Submitted By:

**Group No: 02**

* **Suraj Mundada 240343020101**
* **Prathamesh Sawant 240343020073**
* **Sumit Karjule 240343020099**
* **Sumit Jagadale 240343020100**

**Table of Contents**

1. [Introduction 5](#_TOC_250016)
   1. [Document Purpose **5**](#_TOC_250015)
   2. [Project Background **5**](#_TOC_250014)
   3. [Aim & Objectives **5**](#_TOC_250013)
2. [Business Requirements Overview 6](#_TOC_250012)
3. Functional Requirements 6
   1. [Customer Module **6**](#_TOC_250011)
   2. [Theare Owner Module **7**](#_TOC_250010)
   3. [Admin Module **7**](#_TOC_250009)
4. [Non-Functional Requirement 7](#_TOC_250008)
5. [Use Case Diagram 8](#_TOC_250007)
   1. [Customer **8**](#_TOC_250006)
   2. [Theatre Owner Module **9**](#_TOC_250005)
   3. Admin 10
6. [Project Architecture Diagram 11](#_TOC_250004)
7. [Database Design 12](#_TOC_250003)
   1. User table **12**
   2. Theatre table **13**
   3. Movie table **14**
   4. Theatre\_Screens table 15
   5. Theatre\_screen\_show table **15**
   6. Booking\_details table **16**
   7. Screen\_Seats table **16**
   8. Roles table **17**
   9. Booking table **17**
   10. State table **17**
   11. City table 17
8. [E-R Diagram 18](#_TOC_250002)
9. [Snapshots 19](#_TOC_250001)
10. [Conclusion and Future scope](#_TOC_250000) 28

**List of Figures**

## Use Case Diagrams 17

Fig 1 Project Architecture………………………………………………………………. 11

Fig2 Customer 8

Fig3Theatre owner 9

Fig4Admin 10

Fig5ER Diagram 14

# Introduction:

## Document Purpose:

This document communicates the business requirements and scope for developing BoxOfficeBuddy - Movie tickets booking system. The scope of this document is to define the functional and non-functional requirements, business rules and other constraints requirements.

## Project Background:

The idea of developing an online movie booking system arises from the need to provide a convenient, quick, and efficient way for users to book movie tickets. This system will address various points associated with traditional booking methods, such as long waiting times, limited availability of tickets, and the inconvenience of traveling to the movie theater just to buy tickets. Additionally, it will help theater owners manage bookings more efficiently, reduce the workload on their staff, and improve overall customer satisfaction.

## Aim & Objectives:

The main objective of this project is building a website which is a user-friendly interface that allows users to browse movies, check showtimes, and book tickets effortlessly from any device with internet access. Reduce the need for physical tickets, thereby contributing to environmental sustainability by decreasing paper usage. Implement robust security measures to protect user data and payment information. Provide detailed information about movies, including synopses, trailers, cast details, and user reviews. Offer an easy-to-use backend system for theater owners to manage movie schedules, ticket pricing, and seat availability

# Business Requirements Overview:

* BoxOfficeBuddy is the public web application.
* The BoxOfficeBuddy System will initially target global users, with a primary focus on India

in phase 1.

* There are mainly two types of user. One is the user and other is theater owners
* Users can search for movie showtimes, ticket availability, and prices.
* Theater Owners can list their movies, showtimes, and additional information.
* The BoxOfficeBuddy System provides functions that efficiently connect users with movies,

facilitating effortless ticket booking.

* BoxOfficeBuddy System is maintained by Administrator.

# Functional Requirements Overview:

BoxOfficeBuddy consists of three modules described below.

1. Customer Module
2. Theatre Owner Module
3. Admin Module

# Customer Module

* + - Customer can register and create their own accounts on the platform.
    - Customer can search for movies by title.
    - Customer can choose show times for movies they want to watch
    - Customer can select seat for the specific chosen movie.
    - Customer can book tickets for selected movies and show times

# Theatre Owner Module

* + - Theatre Owner can request for registration.
    - Theatre Owner can login to the system.
    - Theatre Owner can add the theatre.
    - Theatre Owner can add the screens to the respective theatre.
    - Theatre Owner can select the movie for adding to shows.
    - Theatre Owner can arrange seating arrangement for shows.
    - Theatre Owner setup movies which are going to be screened.

# Admin Module

* + - Admin will approve the registration request of theatre owner.
    - Admin will add the movie.
    - Admin will do crud operations for movie.

# Non-Functional Requirement:

* The web site should use professional design, look and feel and color scheme.
* Users will have no limitations for accessing the application through the Internet. Theportalbeinganinternetapplication,itisdifficulttospecifyanexactnumberof visitorsorusers.Hence,wewilltargetthesystemtosupportsufficientusersonthe launch of phase 1.
* Beingapublicwebsite,thesitemustfollowgeneralusabilityguidelinesformenus, navigation, colors, links and other actions provided on the screens.
* The system should be designed in such a manner that the user will be able to complete tasks in a minimum number of steps.

# Use-Case Diagram

## Customer:

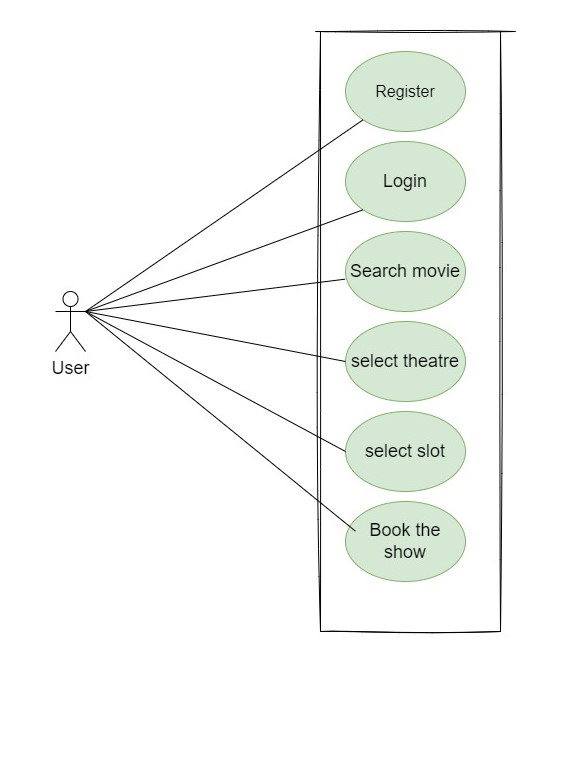
****

Fig. Use-CaseDiagramfor Customer

## Theatre Owner:

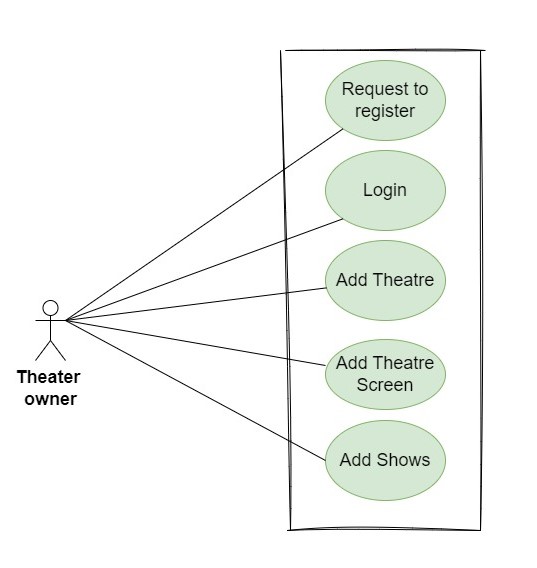
****

Fig. Use-CaseDiagramforTheatre Owner

* 1. **:Admin:**

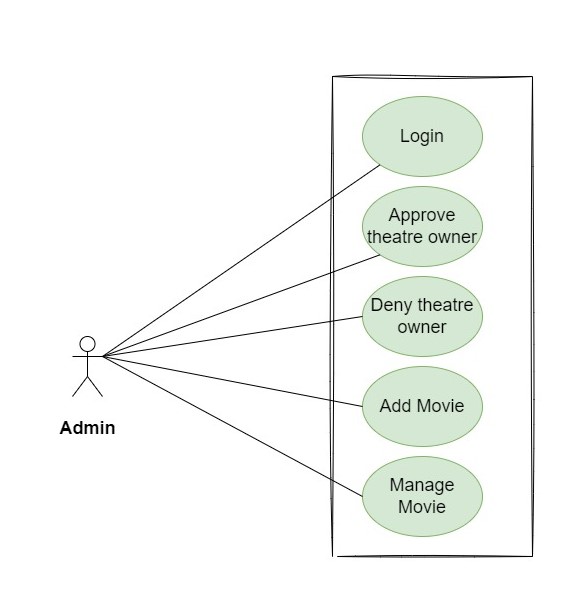
****

Fig. Use-CaseDiagramfor Admin

# Project Architecture Diagram:

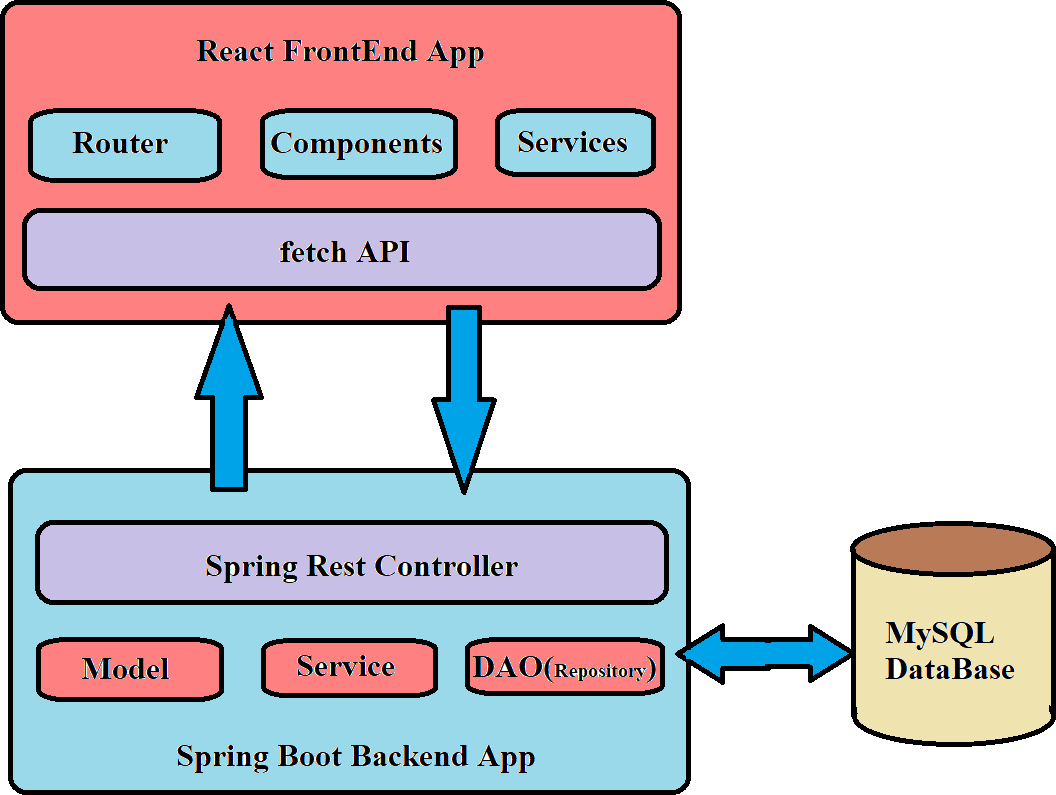


Fig.Project Architecture Diagram

# Database Design:

1. **​User table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| User\_id | Integer |  | PrimaryKey | NotNull | UniqueUserID |
| Fullname | Varchar(45) |  |  | NotNull | FullnameofUser |
| Password | Varchar(45) |  |  | NotNull | PasswordofUser |
| Pincode | Varchar(45) |  |  | NotNull | PincodeofCity |
| Adress\_line1 | Varchar(45) |  |  | NotNull | Address Line1ofUser |
| Adress\_line2 | Varchar(45) |  |  | NotNull | Address Line2ofUser |
| Mobile\_no | Varchar(45) |  |  | NotNull | Mobilenoofuser |
| Email\_id | Varchar(45) |  |  | NotNull | EmailofUser |
| City\_id | Integer |  | Foreignkey | NotNull |  |
| Role\_id | Integer |  | Foreignkey | NotNull | LoginIDofcustomer |
| Status | Varchar(45) |  |  | NotNull | LoginIDofcustomer |

1. **​Theatre table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| Theatre\_id | Integer |  | Primarykey | NotNull | UniqueTheatre |
| No\_of\_Screen | Integer |  |  | NotNull | NumberforScreens |
| Theatre\_name | Varchar(45) |  |  | NotNull | NameofTheatre |
| GST\_No | Varchar(45) |  |  | NotNull | GSTnoofTheatre |
| address\_line1 | Varchar(45) |  |  | NotNull | AddressLine1ofTheatre |
| address\_line2 | Varchar(45) |  |  | NotNull | AddressLine2ofTheatre |
| Pincode | Varchar(45) |  |  | NotNull | PincodeofCity |
| user\_id | Integer |  | Foreignkey | NotNull | UniqueIDofUser |
| Phone | Integer |  |  | NotNull | PhonenoforTheatre |

1. **Movie table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| movie\_id | Integer |  | Primarykey | Null | Unique Movie Identifier |
| Movie\_title | Varchar(45) |  |  | Null | |  | | --- | | Title of the Movie |  |  | | --- | |  | |
| Movie\_relase\_date | Varchar(45) |  |  | Null | |  | | --- | | Release Date of the Movie |  |  | | --- | |  | |
| Movie\_duration | Varchar(45) |  |  | Null | |  | | --- | | Duration of the Movie |  |  | | --- | |  | |
| Movie\_language | Integer |  |  | Null | Registrationyearofthe vehicle |
| Movie\_category\_id | Integer |  | Foreign key | Null | |  | | --- | | Foreign Key to Movie Category |  |  | | --- | |  | |
| Movie\_banner\_path | Integer |  |  | Null | |  | | --- | | Banner Path for the Movie |  |  | | --- | |  | |

1. **Theater\_Screens table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| screen\_id | Integer |  | Primary key | Null | |  | | --- | | Unique Screen Identifier |  |  | | --- | |  | |
| screen\_name | |  | | --- | | Varchar(45) |  |  | | --- | |  | |  |  | Null | |  | | --- | | Name of the Screen |  |  | | --- | |  | |
| total\_seats | |  | | --- | | Varchar(45) |  |  | | --- | |  | |  |  | Null | |  | | --- | | Total Seats |  |  | | --- | |  | |
| |  | | --- | | theatre\_id |  |  | | --- | |  | | Integer |  | Foreign key | Null | Foreign Key to Theater |

1. **Theatre screen show table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| show\_id | Integer |  | Primary key | Null | |  | | --- | | Unique Show Identifier |  |  | | --- | |  | |
| show\_startdate | Tinyint |  |  | Null | |  | | --- | | Start Date and Time of the Show |  |  | | --- | |  | |
| show\_enddate | Datetime |  |  | Null | |  | | --- | | End Date and Time of the Show |  |  | | --- | |  | |
| 9\_am\_slot | Varchar(45) |  |  | Null | Slot for booking |
| 12\_am\_slot | Varchar(45) |  |  | Null | Slot for booking |
| 3\_pm\_slot | Varchar(45) |  |  | Null | Slot for booking |
| 6\_pm\_slot | Varchar(45) |  |  |  | Slot for booking |
| 9\_pm\_slot | Varchar(45) |  |  |  | Slot for booking |
| movie\_id | Integer |  | Foreign key | Null | |  | | --- | | Foreign Key to Movie |  |  | | --- | |  | |
| screen\_id | Integer |  | Foreign key | Null | |  | | --- | | Foreign Key to Screen |  |  | | --- | |  | |

1. **Booking\_Details table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| booking\_detail\_id | Integer |  | Primary key | Null | |  | | --- | | Unique Booking Detail Identifier |  |  | | --- | |  | |
| price | Datetime |  |  | Null | |  | | --- | | Price of the Booking |  |  | | --- | |  | |
| seat\_id | Datetime |  | Foreign Key | Null | |  | | --- | | Foreign Key to Seat |  |  | | --- | |  | |
| booking\_id | Datetime |  | Foreign Key | Null | Foreign Key to Booking |

1. **Screen\_Seats table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| seat\_id | Integer |  | Primary key | Null | |  | | --- | | Unique Seat Identifier |  |  | | --- | |  | |
| seat\_no | Varchar(45) |  |  | Null | |  | | --- | | Seat Number |  |  | | --- | |  | |
| seat\_category\_id | Integer |  | Foreign | Null | |  | | --- | | Foreign Key to Seat Category |  |  | | --- | |  | |
| screen\_id | Integer |  | Foreign | Null | Foreign Key to Screen |

1. **Roles table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| |  | | --- | | role\_id |  |  | | --- | |  | | Integer |  | Primary key | Null | Uniquerole ID |
| role\_name | Varchar(45) |  |  | Null | Nameoftheroleofperson who is loggingin |

1. **Booking table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| booking\_id | Integer |  | **-** | Null | |  | | --- | | Unique Booking Identifier |  |  | | --- | |  | |
| seats\_booked | Integer |  |  | Null | |  | | --- | | Total Number of Seats Booked |  |  | | --- | |  | |
| Booking\_date\_time |  |  |  |  | |  | | --- | | Date and Time of Booking |  |  | | --- | |  | |
| Total\_amount |  |  |  |  | Total Amount for Booking |

1. **State table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| state\_id | Integer |  |  | Null | |  | | --- | | Unique State Identifier |  |  | | --- | |  | |
| state\_name | Varchar(45) |  |  | Null | NameoftheState |

1. **City table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| city\_id | Integer |  | Primary |  | |  | | --- | | Unique City Identifier |  |  | | --- | |  | |
| |  | | --- | | city\_name |  |  | | --- | |  | | Varchar(45) |  |  |  | Name of the city |
| state\_id | Integer |  | foreign |  | Foreign Key to State |

# ER-Diagram:

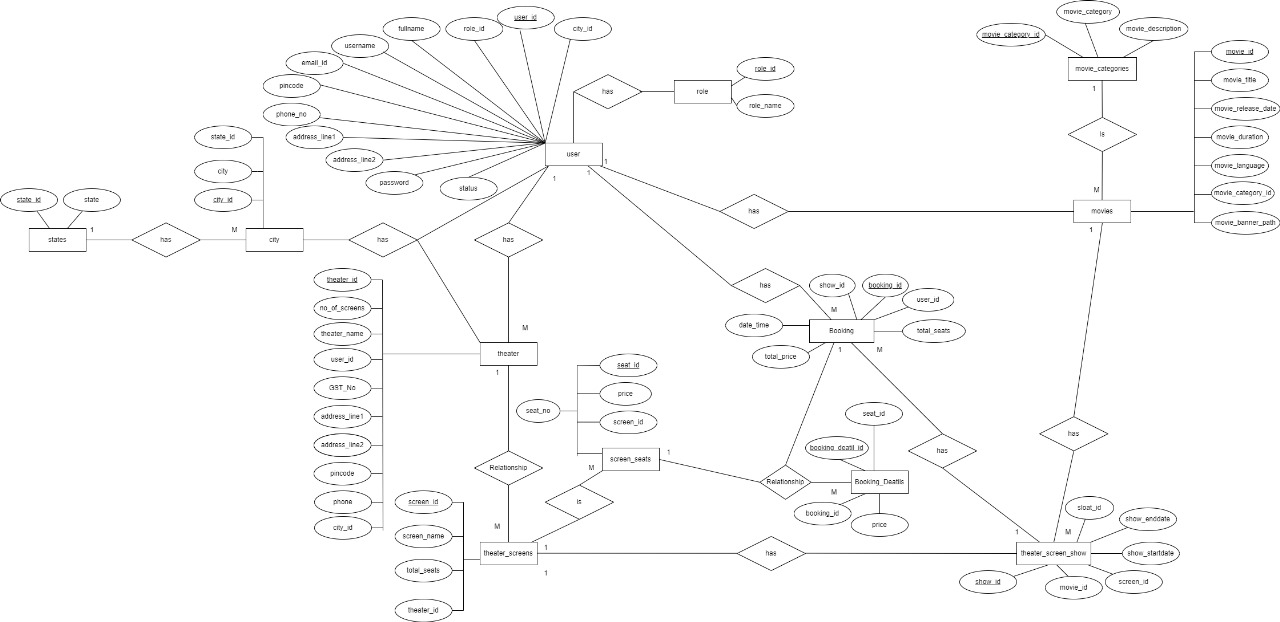
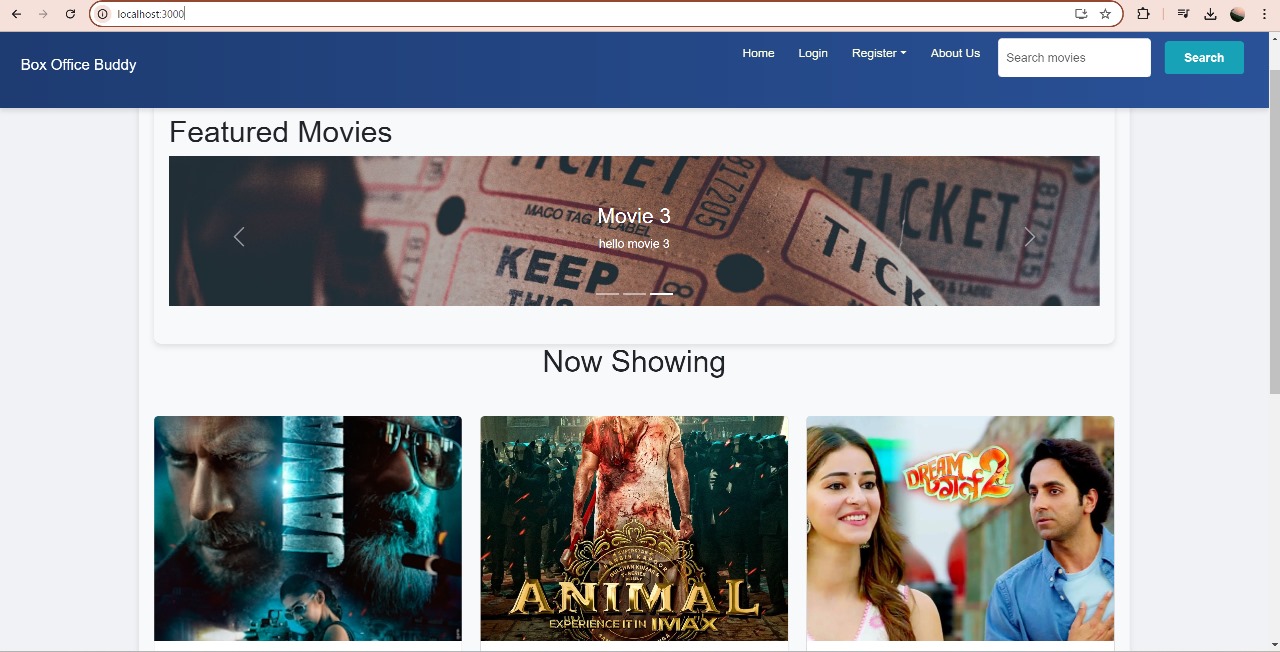
****

Fig. E-Rdiagram of BoxOfficeBuddy System

# Snapshots:

* 1. **Home Page:**

Following snapshot shows the Home page for BoxOfficeBuddy.

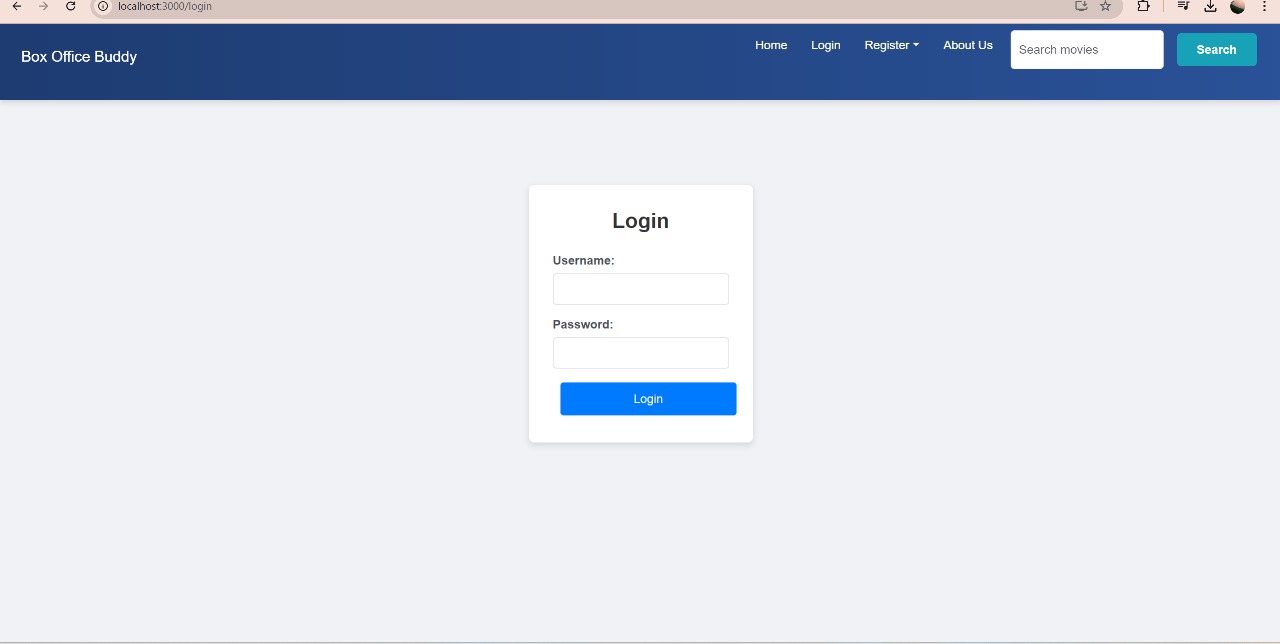


This page contains following controls

* Home
* About Us
* Search Movies
* Login
* Register
  + Customer Registration
  + Theatre Registration

# Login Page:

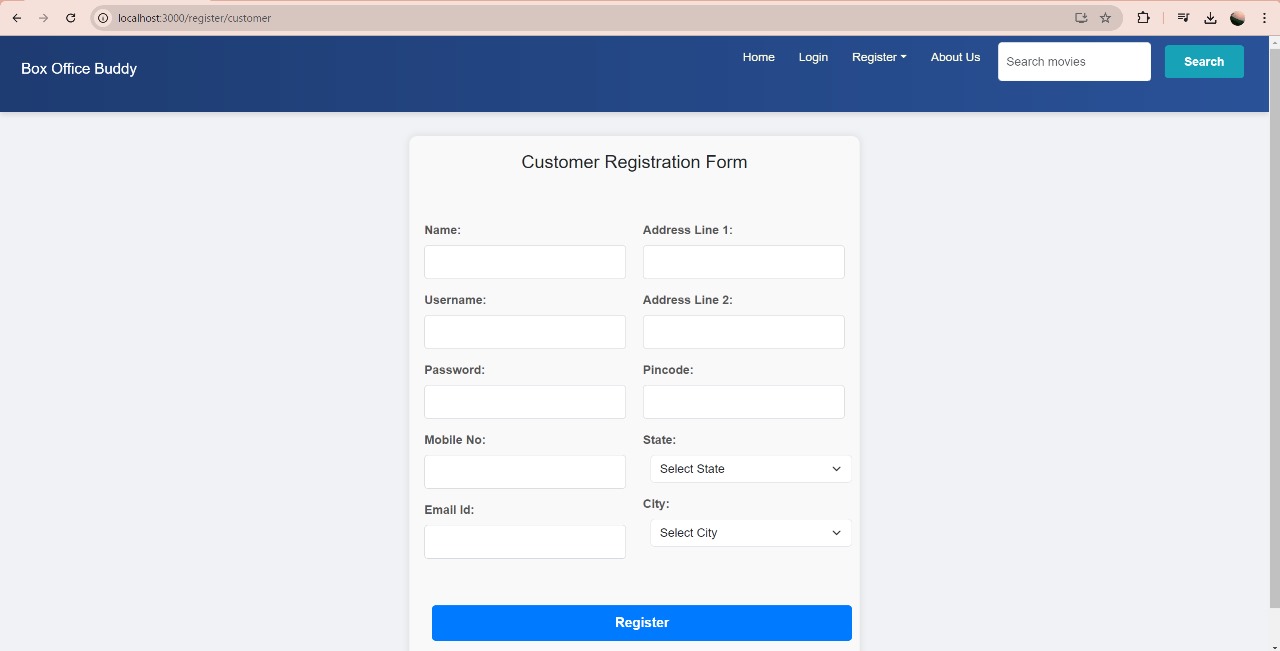
Following snapshot shows the Login page for BoxOfficeBuddy



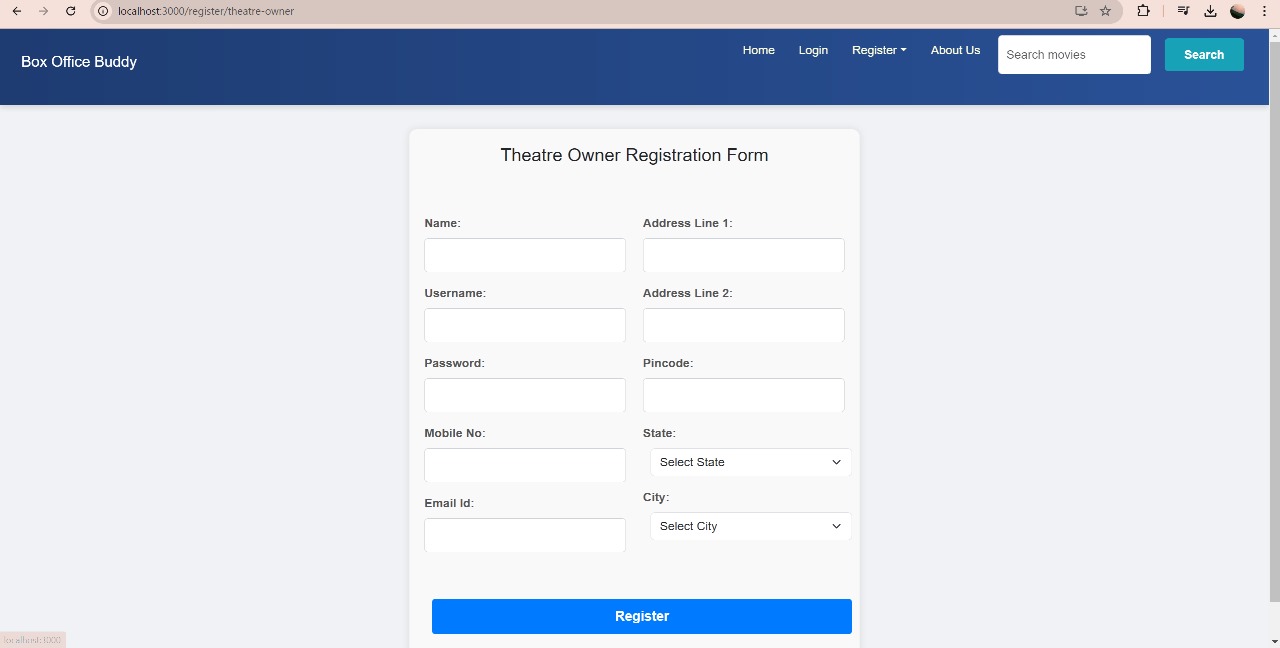
# Registration Page:

Following snapshot shows the registration page for BoxOfficeBuddy

* Customer Registration Page:

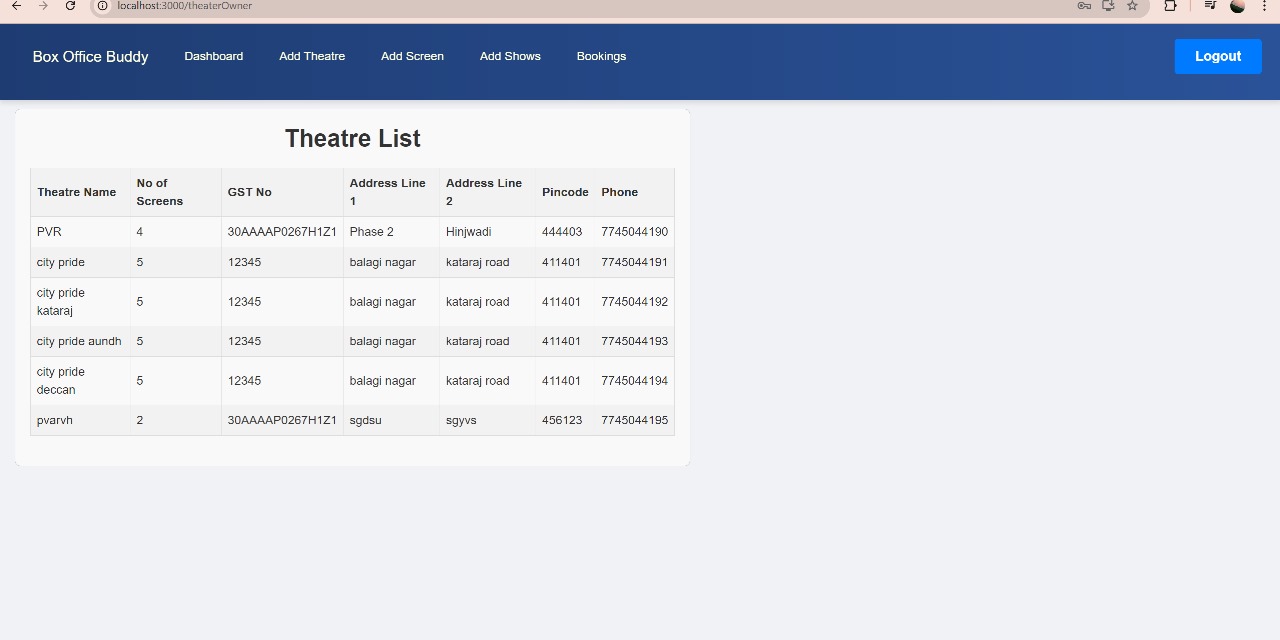


* Theater Owner Registration page:

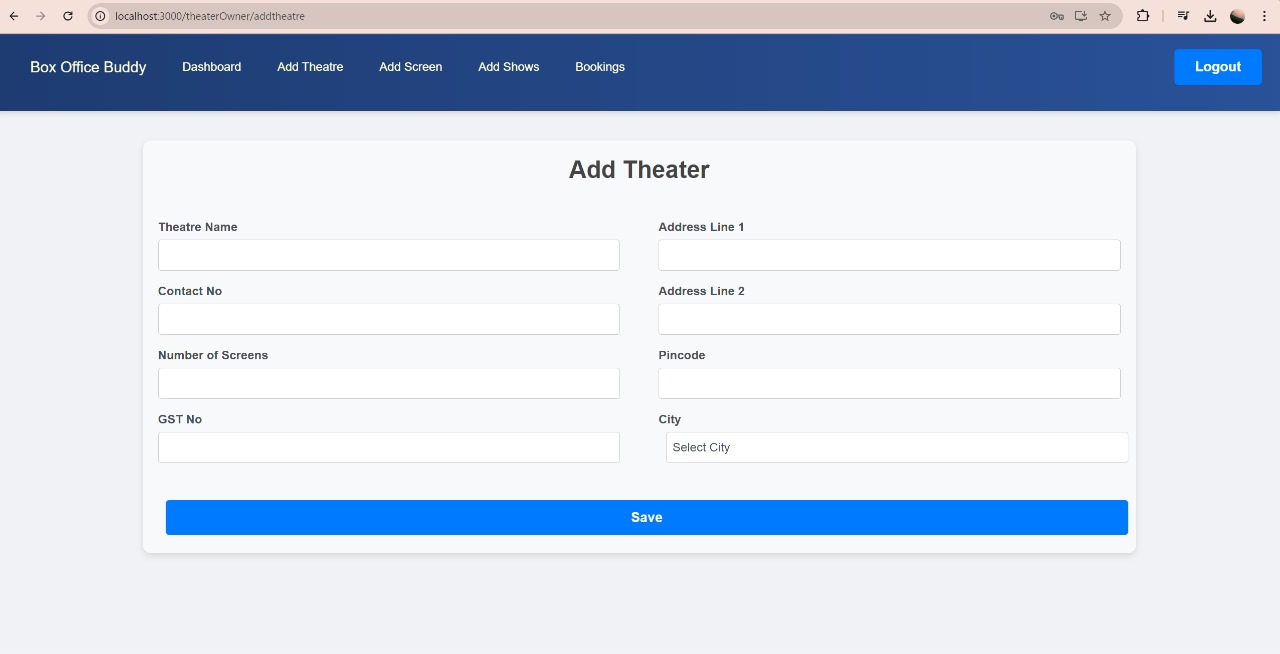


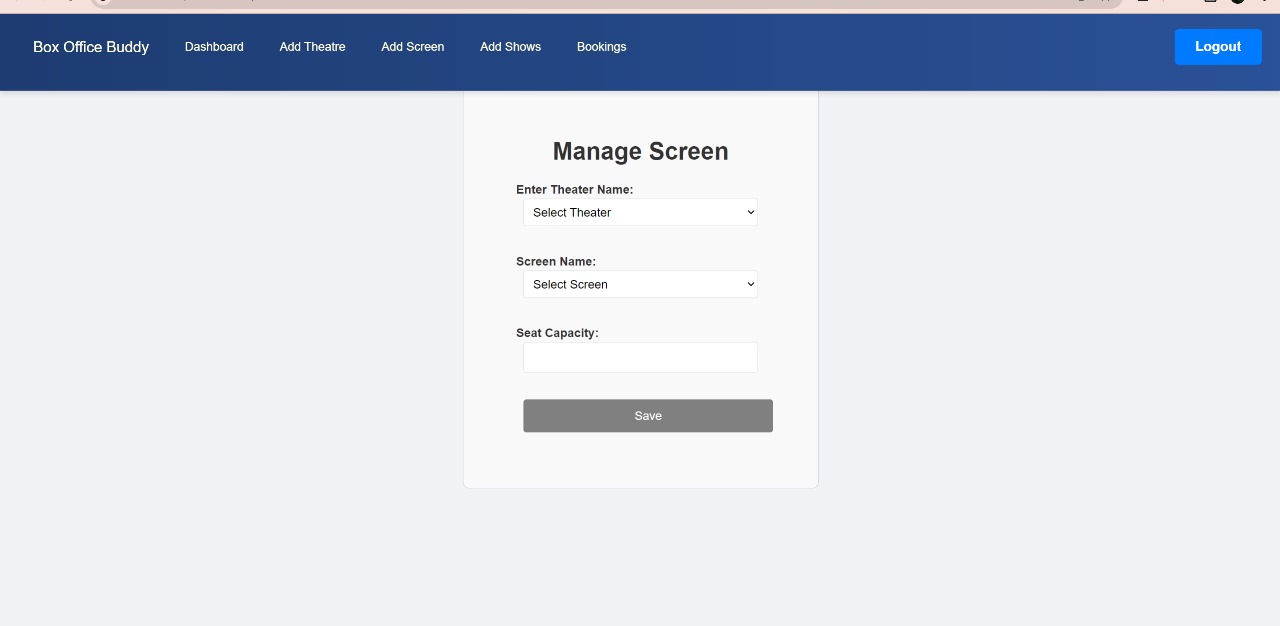
# Theatre owner flow:

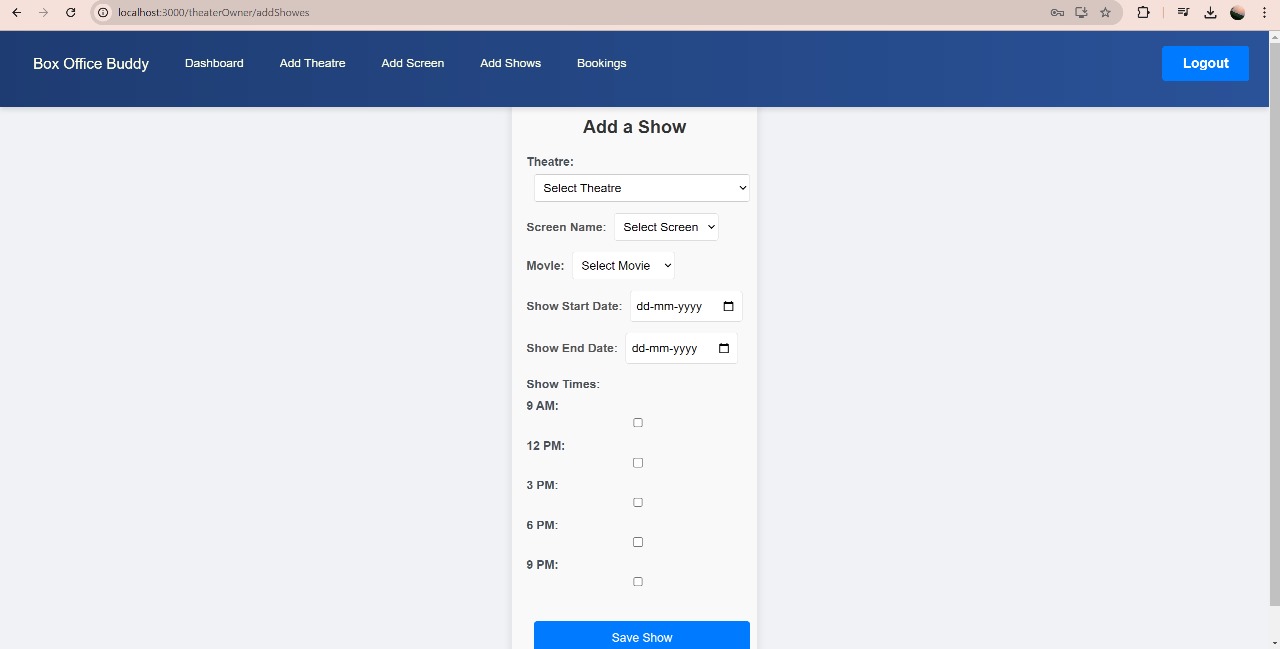
* View theatre list



* Add Theatre

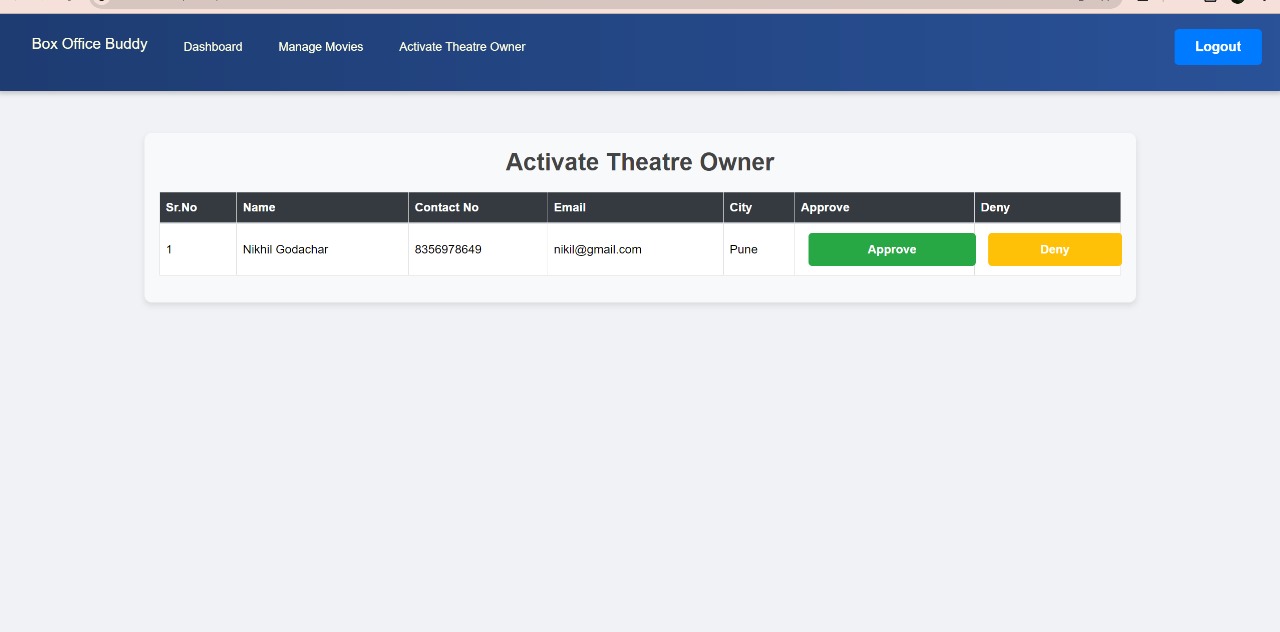


* Add screens
* Add shows

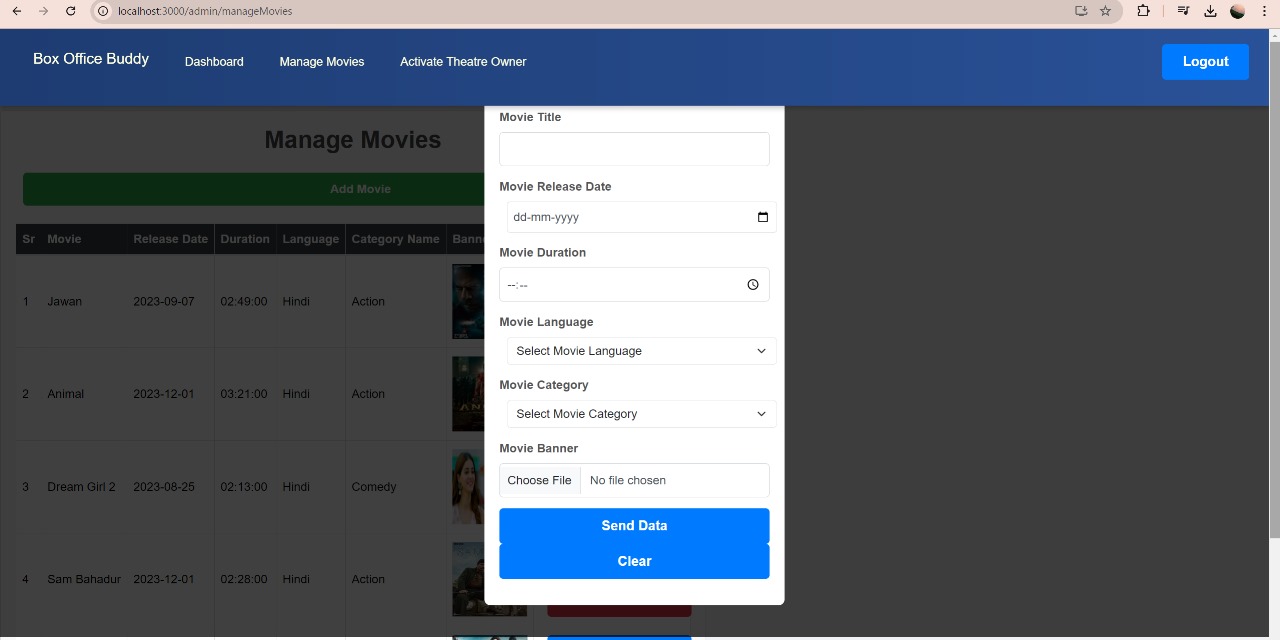


# Admin flow of snapshot:

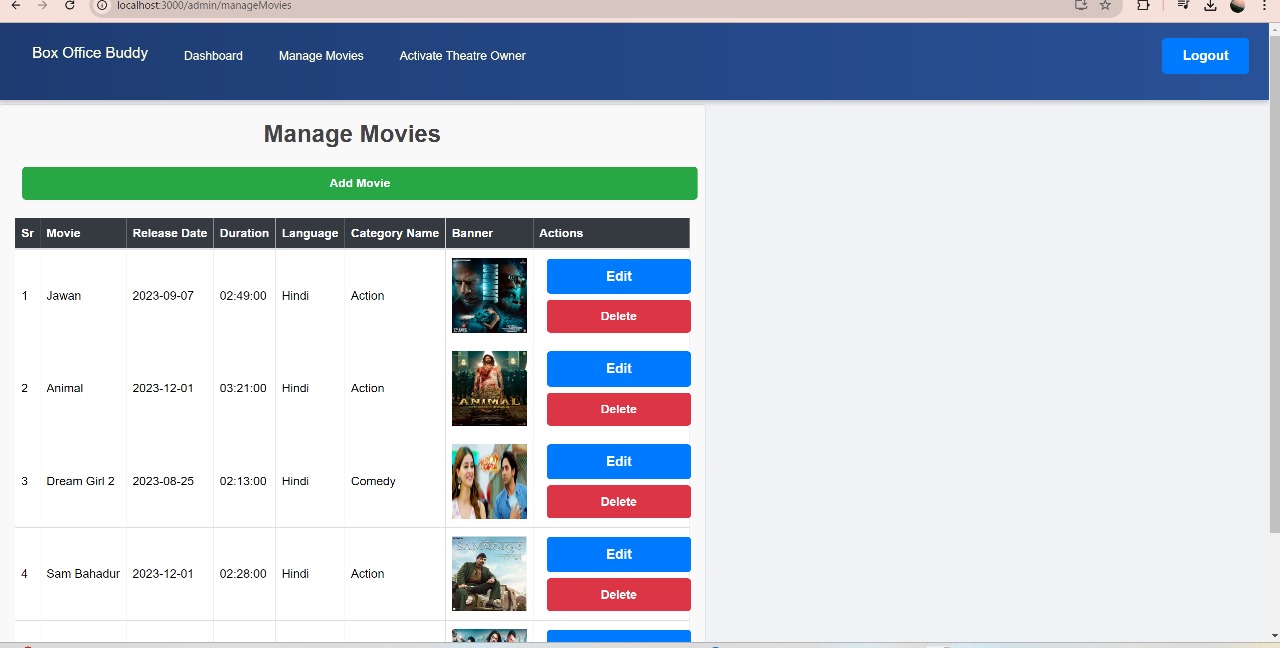
* Approve Theatre Owner



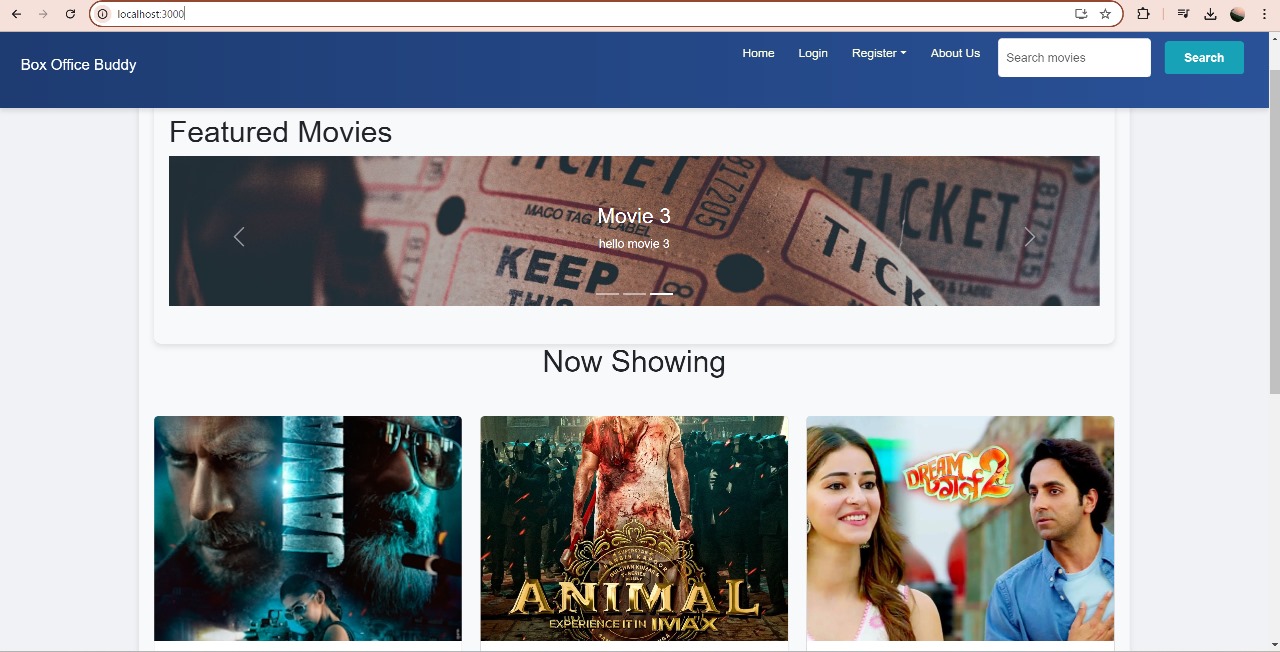
* Add Movie



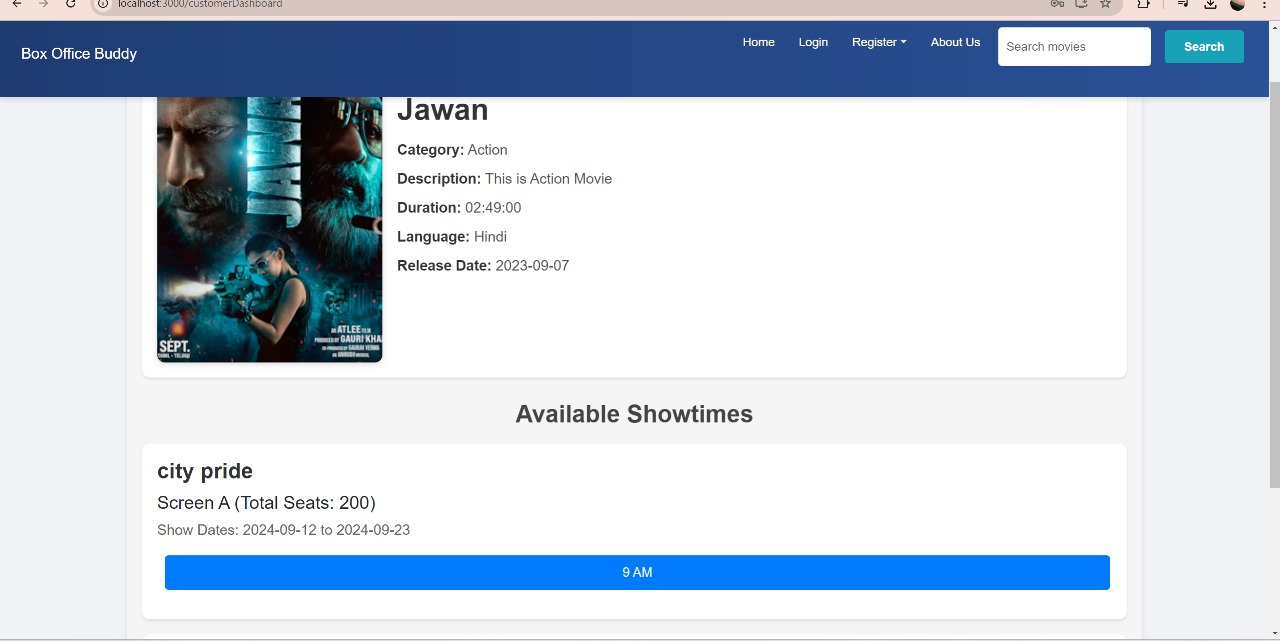
* Manage Movie



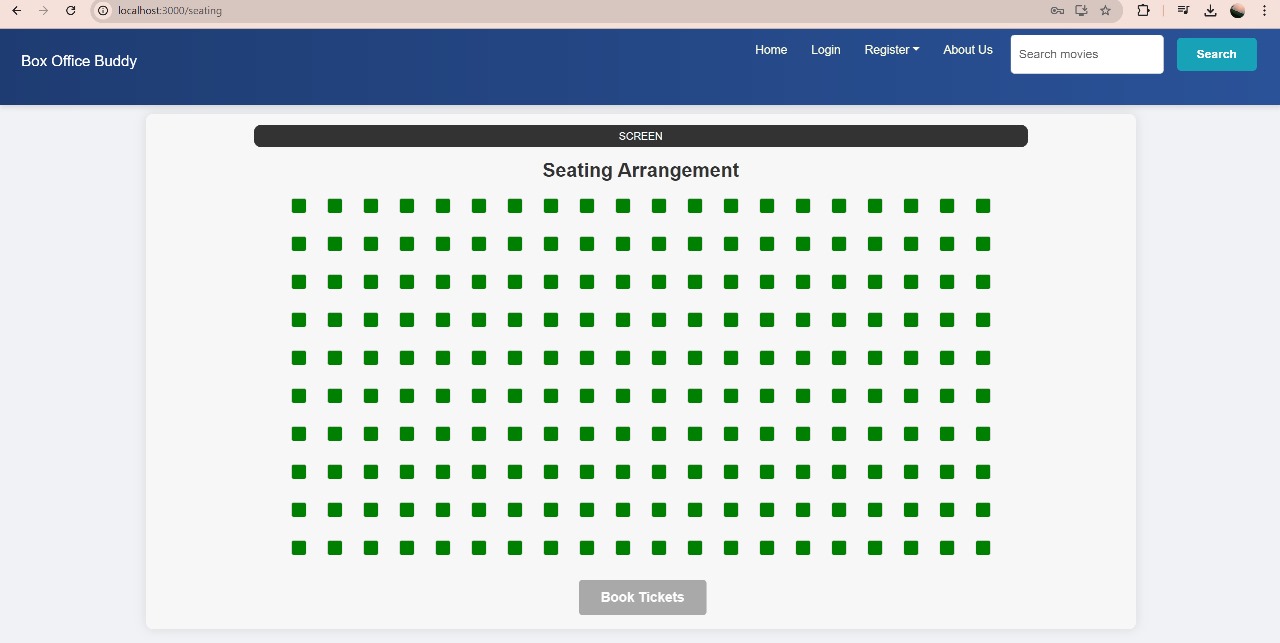
* 1. **Customer flow of snapshot**
* Choose movie



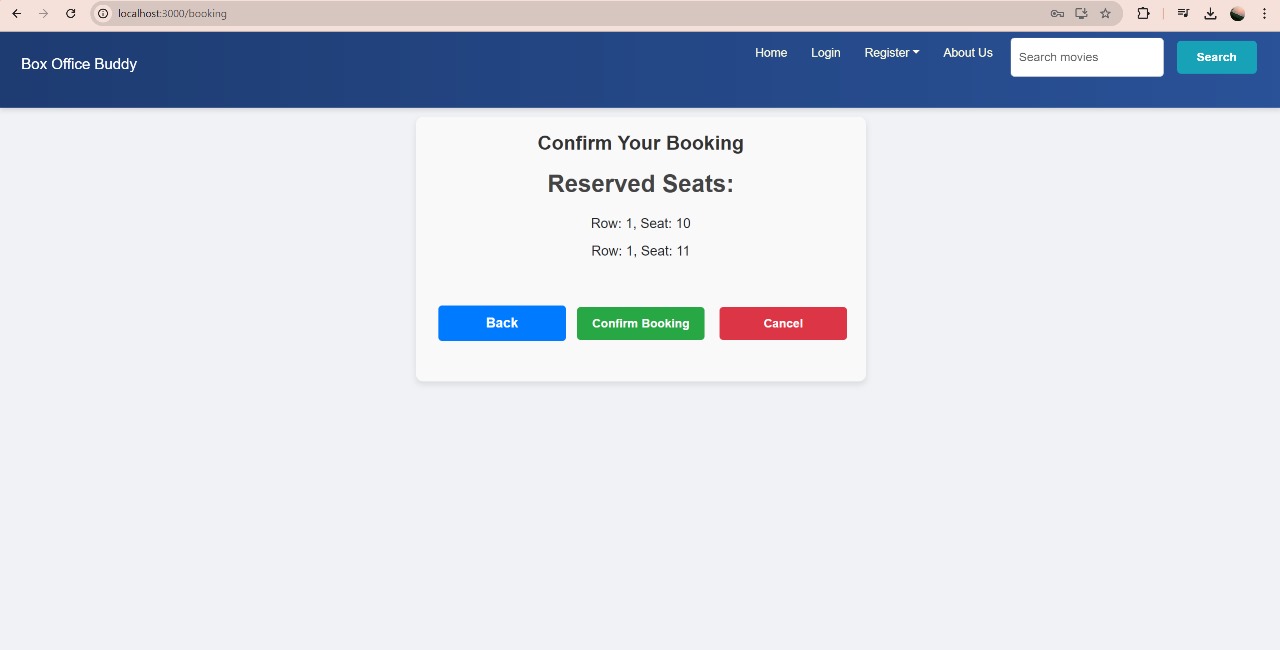
* Book Slots



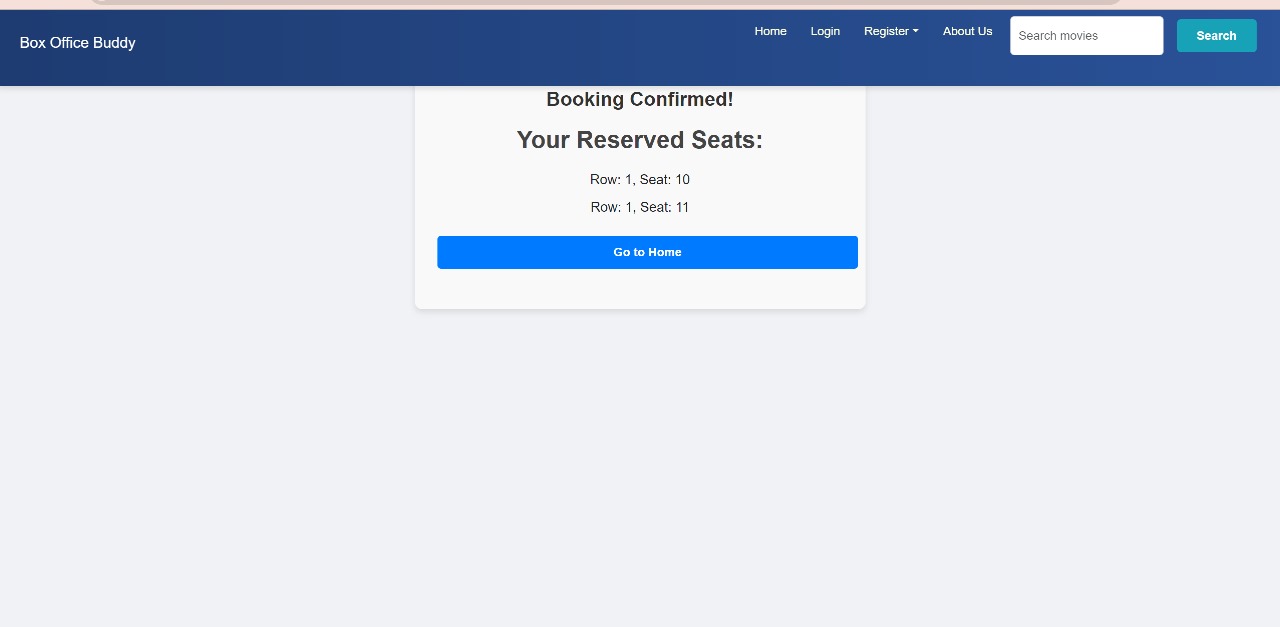
* Choose seats



* Confirm booking



* Booking Done



# CONCLUSION AND FUTURE SCOPE

The **BoxOfficeBuddy** system could evolve in several ways to enhance the user experience, improve business operations, and leverage emerging technologies. Here are some future scopes for such a system.

Provide details about seat comfort, such as legroom or recline features, using user reviews

or sensor data.

Our system provides a very user-friendly platform where customer can easily register theatre, login, search movies , book ticket, also service center will provide current status of the service progress.

Few things which we have kept aside for right no was our plat for market but in future we desire to add are to establish this service provision across pan India, to create flexible packages as per customer’s requirement and to notify customer about vehicle’s service progress, service completion via SMS or email.