Project Documentation

Project Name: I-Cinema

Project Description: I-Cinema is a web-based e-commerce platform that allows users to book tickets online. The platform features a user-friendly interface that makes it easy for customers to browse Movies, check available seats, and shows available in different theatres. Then, they can book the tickets and do the payment either through a debit or credit card. It is built using modern web technologies and is designed to be highly scalable, secure, and reliable.

Major Modules:

- Movie Catalog Management
- Theatre management
- Seating management
- Booking Management
- Payment Gateway Integration

Architecture:

This application uses Microservice Architecture to make it a robust, scalable, and easy to maintain application. Each microservice is responsible for individual task and different microservices communicate with each other using REST APIs. The following microservices to be implemented:

- **Movies Microservice:** This microservice will manage Movie library and Movie catalogues. It also keeps track of new releases, popular among the audience. It integrates with other microservices, such as the Booking Microservice, to enable users to book tickets for their preferred movies.

- **Theatre Microservice:** This microservice will manage Theatre library and shows catalogues. It also keeps track of show timings and seats availability for each show in all theatres.
- **Seating Microservice:** This microservice will manage seat availability. It also keeps seats status (Available, booked or selected).
- **Booking Microservice**: This microservice will manage the booking of shows.
- **Payment Microservice**: This microservice will handle the payment process. It will integrate with a payment gateway to allow users to make payments for their bookings.
- **Gateway Service:**
    - Acts as the entry point for all requests to the microservices.
    - Uses Spring Cloud Gateway to handle routing, load balancing, and traffic management.
    - Uses Spring Cloud Consul for service discovery and registration.

Technology Stack:

- **Backend:** Spring Boot, Spring Data, Spring REST, Spring Cloud Consul
- **Frontend:** Angular/React, TypeScript/JavaScript, HTML5, CSS3, Bootstrap5
- **Database:** MySQL/NOSQL
- **Security:** OAuth/Spring Security

Non-Functional Requirements:

- Performance: The application should be fast and responsive, with quick load times and minimal lag or delay.
- Scalability: The application should be able to handle large amounts of traffic and scale as needed to accommodate growth.
- Security: The application should be secure and protect user data from unauthorized access, with measures such as encryption and secure authentication.
- Reliability: The application should be reliable and available, with minimal downtime or outages.
- Usability: The application should be easy to use and intuitive, with a clear and consistent user interface.
- Maintainability: The application should be easy to maintain and update, with clear and well-organized code that is easy to understand and modify.
- Compatibility: The application should be compatible with a wide range of devices and browsers, with support for different screen sizes and resolutions.

Final Deliverables:

- Application archive (.jar) with source code
- Database DDL Script
- Complete Source code
- Sample screenshots of important screens

User Stories:

US 01: View Movies

As a User, I should be able to view the list of movies that are screening currently, so that I can book the movie I want.

Acceptance criteria:

- The user must be able to access the Movies from the home page.
- The user must be able to view all the available Movies with their names, Genre, language, and images in tabular format.
- Upon clicking any movie, it should redirect to that movie details page.

US 02: Search Movies

As a user, I should be able to search for a movie by its name so that I can quickly find the movie I am looking for and to book tickets for that movie.

Acceptance criteria:

- The search bar should be visible and accessible in-home page of the application.
- The user must be able to search for a movie by its name.
- The search results must be displayed in a visually appealing way.
- The search results must include the movies with their names, Genre, language, and images.
- On clicking any movie from the search result, it should redirect to the respective movie's details page.
- The user must be able to filter the search results by rating, genre, or language.
- The most relevant search results must be displayed first.
- Display proper error messages in case the movie is not available.

US 03: Filter Movies

As a user, I want to be able to filter movies by rating, genre, or language so that I can find the movies that are most relevant to my interests and needs.

Acceptance criteria:

- The user must be able to filter movies by rating, genre, and language.
- The filter categories should be available in an expandable way with all types of filters available.

- On expanding a filter category, it should show the corresponding filter options of that filter category.
- Examples for filter categories and options:
    - Rating: 4+, 3+, 2+,1+.
    - Genre: Comedy, Action, Horror, Thriller, etc.
    - Language: Hindi, English, Telugu, Tamil, etc.
- The user must be able to combine multiple filters to narrow down their search results.
- The user must be able to see the number of movies that match each filter.
- The filtered results must be displayed in a way that is visually appealing, user-friendly, and consistent with the overall design of the application.
- Display proper error messages in case the movie is not available.

US 04: View Movie details
As a user, I want to be able to view the details of a movie so that I can know more about it and decide whether I want to watch it.

Acceptance criteria:

- The user should be able to view the movie details by clicking on a movie from the home page or from the search results page.
- The movie details page must include the following information:
    - Name
    - High-quality image
    - Release date
    - Genre
    - Rating
    - Censor Rating
    - Description
    - Add your rating (input box)
- The movie details page must be displayed in a full-screen view that is user-friendly and readable.
- The user must be able to easily navigate back to the list of movies.
- It should have a Book My Show" button to book that movie
- Display proper messages in case of errors or exceptions.

US 05: Movie Booking

As a user, I should be able to book tickets for the movies of my preference from the movie details page.

Acceptance criteria:

- The user should be able to book the movie by clicking on an " Book My Show" button from the movie details page and it will redirect to the screen matrix page.
- The Screen matrix page should display list of theaters screening the movie and the shows available in each.
- Display proper messages in case of errors or exceptions.

US 06: Browse a movie showtime:

As a user, I want to be able to browse and select a theater, shows, and seats so that I can book my tickets and enjoy my movie experience.

Acceptance criteria:

- The user should be able to view all the theatres that are screening that movie and the show times available.
- On selecting the theatre and show time, they should be redirected to a page that displays the price per seat along with an input box where they can enter the number of seats they require. This will allow the user to see the total cost of their tickets before they proceed to payment.
- If the number of seats required is more than the number of seats available for the show, then “Required number of seats are not available” error message should pop up along with a button “book some other show”.
- On clicking “book some other show”, it should redirect the theatres and the show times available page for that movie.
- Display proper messages in case of any show, in which all seats are booked already.

User Story 07: Display and Select Seats

As a user, I want to be able to see a seating chart of the theatre that shows which seats are available, booked, and selected in different colors so that I can easily choose the seats I want.

Acceptance Criteria:

- On the booking page, I should be able to see a seating chart of the theatre.
- The seating chart should display available seats in green, booked seats in red, and selected seats in yellow.
- When I hover over a seat, I should be able to see information about the seat, such as the seat type and price.
- I should be able to select seats by clicking on them.
- After selecting seats, the seat color should be updated to indicate that they are selected.
- I should be able to see the total cost of the tickets before proceeding to checkout.
- Proper error or exception messages should be displayed when necessary.

US 08: View Booking details:

As a user, I should know the booking details like total price for seats and Convenience fee so that I can book the tickets online.

Acceptance criteria:

- The user should be able to view the pricing details for their booking like price for seats, convenience fee, GST, and the total payment amount.
- The total payment amount can be calculated by adding price for seats (price per seat * number of seats selected by the user), convenience fee and GST.
- The total price of the tickets should be displayed to the user along with a “pay” button.
- On clicking “Pay button” the user will be procced for the payment.
- Display proper messages in case of errors or exceptions.

US 09: Make Payment:

As a user, I should be able to provide my card detail so that I can proceed with payment.

Acceptance criteria:

- The user should be able to use a secure payment method to complete the purchase.
- The user can make payment by selecting the card type (credit card or debit card) with the help of radio button.
- If payment is through a Credit card, a 10% discount should be applied else 5% discount should be applied.
- The user must be able to enter their card details, including their card number, expiration date, and CVV code, on a secure payment page.
    - Card number:
        - The card number must be a valid credit or debit card number.
        - The card number must be 16 digits long.
    - Expiration date:
        - The expiration date must be in the future.
        - The expiration date must be in a valid format (MM/YY).
    - CVV code:
        - The CVV code must be a 3-digit number.
- The user must be able to review their card details before completing the payment.
- The user must be able to receive a confirmation message once the payment is processed successfully.
- Display proper messages in case of errors or exceptions.

US 10: Rate the Movie

As a user, I want to be able to rate movies after watching them so that I can share my thoughts and experiences with other viewers and help them to make informed decisions about which movies to watch.

Acceptance criteria:

- The user must be able to rate a movie on a scale of 1 to 5 stars.
- The average rating of the movie must be displayed on the movie's page.
- Users must be able to filter movies by their average rating.

- Display proper messages after submitting the rating successfully.
- Display proper messages in case of errors or exceptions.