Modern Application Development - 2
Movie Ticket Booking System - Project Report

Introduction

The Movie Ticket Booking System is a web application designed to facilitate the booking of show tickets in theaters. It is built using Flask for the backend, Vue JS for the user interface, SQLite for the database, and utilizes Redis and Celery for caching and batch job processing. The system caters to both users and administrators, offering features for booking tickets, creating theaters, and managing shows.

System Overview

The Movie Ticket Booking System consists of several modules, each serving a specific purpose:

User Module
Fields: id, username, email, phone, password
The system utilizes JWT (JSON Web Tokens) for user authentication and stores user details securely in local storage.

Theatre Module
Fields: id, name, location, capacity
Admin users can create theaters, specifying their names, locations, and seating capacities.

Show Module
Fields: id, name, language, tag, rating
Administrators have the ability to create shows, including information such as show names, languages, tags, and ratings.

Inventory Module
Fields: id, theatre_id, show_id, date, start_time, end_time, retail_price, selling_price, is_active, available seats
Inventories are created for each show, detailing showtimes, pricing, and seat availability.

Booking Module
Fields: id, inventory_id, user_id, ticket_count, date
Users can book tickets for shows, with each booking linked to specific inventory and user.

System Functionality

The Movie Ticket Booking System offers the following key functionalities:

User Registration and Login: Users can sign up and log in to access the booking system.

Admin Privileges: Admin users can create theaters and shows, add inventory and edit and delete them, providing flexibility in managing the system's offerings.

Ticket Booking: Users can book multiple tickets for various movies, with each booking linked to a specific inventory.

System Design

The system follows a client-server architecture with a Vue JS-based front-end interacting with a Flask-based backend. Key design considerations include:

Database: SQLite is used as the database system, offering a lightweight and file-based storage solution. Tables for users, theaters, shows, inventories, and bookings are maintained.
Authentication: User authentication is implemented using JWT (JSON Web Tokens), ensuring secure access to the system.
Caching: Redis is utilized for caching to enhance system performance, reducing the load on the database and speeding up frequently accessed data.
Batch Jobs: Celery is integrated for batch job processing, enabling features like daily reminders and background tasks.
User Experience: Vue JS provides a dynamic and responsive user interface, allowing users to easily navigate, search for shows, and book tickets.

Following is the link to Postman Documentation containing all API contracts:
https://documenter.getpostman.com/view/27430914/2s9YC7UCeG

Link to Presentation Video:
https://drive.google.com/file/d/1xZ6Hn980QacvwaIM9mQeHmvAmDwoiojR/view?usp=sharing

Conclusion

The Movie Ticket Booking System provides a robust platform for users to easily book tickets for their favorite shows while offering administrators the tools they need to manage theaters and shows efficiently. It leverages modern technologies like Flask, Vue JS, and Redis to enhance user experience and system performance. I have completed this project under the course Modern Application Development-2 and it was a great learning experience for me! I would like to thank the entire IITMOD team for this opportunity.

-Munishka Vijayvergiya
21f1005789