

Author

Laxman bafna

Roll Number: 21f2000485

Email: 21f2000485@ds.study.iitm.ac.in

I am a computer science student with a passion for web development.

Description

My project aims to create a Ticket Show platform for booking show tickets in theaters. It allows users to browse available shows, book tickets, and receive reminders. The project uses Flask for the backend API, Vue.js for the frontend, and Redis for caching and background tasks.

Technologies Used

Flask for API development

Vue.js for the frontend UI

Redis for caching and as a message broker

SQLite for the database

Flask-Caching for caching management

Flask-RESTful for building API endpoints

Celery for backend tasks

DB Schema Design

The database schema consists of tables for users, theaters, shows, bookings, and more. The users table stores user information, including authentication details. The theaters and shows tables hold details about venues and available shows. The bookings table records user reservations. Foreign keys and indexes ensure data integrity and efficient queries.

API Design

I have created API endpoints using Flask-RESTful for various functionalities, including user authentication, theater management, show listings, and bookings. The API was designed with a clear structure and supports CRUD operations for relevant resources. The API specification is available in a separate YAML file.

Architecture and Features

The Ticket Show project follows a well-organized structure that separates the backend and frontend components. In the backend, Flask is used to build the API, and the project's controllers are located in designated packages. API endpoints for user authentication, theater management, show bookings, and more are defined in these controllers. The Flask-RESTful extension facilitates the creation of clean and efficient API routes. The frontend is built using Vue.js, with Vue components organized in a structured manner. Templates and static assets are located in their respective directories. The project's configuration settings, such as database connections and cache settings, are stored centrally for easy management.

Video

https://drive.google.com/file/d/1zEfg3ankoJyehbxgcaNT7lc7AKrEk5-c/view?usp=drive_link