MAD2 Ticket Booking App Report

Author:

Shubhankar Jaiswal
Roll No. 21f1006828
Email: 21f1006828@ds.study.iitm.ac.in

I'm a data science student from Maharajganj, Uttar Pradesh, with a diploma in data science from IITM.

Project Summary:

Developed a Flask-based API for a movie booking system. Users can register, log in, and book tickets for shows. Admin users can manage theaters, shows, and user roles. The API supports operations such as registration, login, user promotion, theatre creation, show addition, ticket booking, and data export.

Technologies Used:

Flask: Backend foundation for routing, request handling, and database integration.
Flask-Migrate: Database migration and schema management.
Flask-Security: User authentication, authorization, and role management.
Flask-JWT-Extended: JWT integration for secure user sessions.
Flask-Marshmallow: Object serialization for converting data types.
Flask-Caching: Caching to optimize performance.
Flask-CORS: Handling CORS for frontend requests.
SQLite: Lightweight database engine for development.
SQLAlchemy: ORM library simplifying database interactions.
Vue.js: Dynamic frontend components.
Axios: Making HTTP requests from frontend to backend.
Vue Router: Handling navigation in SPAs.
JSON Web Tokens (JWT): Secure information transmission.
Python: Backend logic and functionality.
JavaScript: Frontend interactivity.

Database Schema:

User Table: User information including authentication data and roles.
Role Table: Defines user roles for permissions.
Theatre Table: Information about theaters.
Show Table: Details of individual shows.
Booking Table: Tracks user reservations.
CsvEntry Table: Stores CSV files associated with theaters.

API Design:

User Registration and Login: /register, /login endpoints for registration and authentication.
Admin Role Management: /promote endpoint to promote users to admin.
Theatre and Show Management: /admin/theatres and related endpoints for admin functions.
User Dashboard and Booking: /user/dashboard, /bookings/{showId}, /user/booked-tickets endpoints for user interactions.
CSV Export: Endpoints for exporting theater details in CSV format.

Architecture and Features:

Frontend in src/components directory with Vue.js components.
Backend with app.py as Flask core and model.py defining database models.
Features include user authentication, role-based access control, ticket booking, and background tasks using Celery for exporting CSV files and scheduling jobs.

Video:
Project Demonstration Video Link
https://drive.google.com/file/d/1t1ojM2-ChLIgWr9zrYqJMv3YQrAD69V3/view?usp=sharing