Movie-Ticket Booking v2 Report

Author:

- **Name**: Prabuddh Mathur
- **Roll No**: 21f2001016
- **Email**: 21f2001016@ds.study.iitm.ac.in
- **About Me**: Currently pursuing Data Science and Programming and have a working hand in Python. Always ready to learn new things and try out new technologies, all along with being a connected friend.

Description:

A Movie Ticket Booking site is to be made using Flask, SQLAlchemy, and Vue3.js. There is an Admin role who is able to create/edit/delete venues and add shows in them. User is able to create/edit/delete and rate their show bookings

Technologies Used:

- **Python:** responsible for developing the controllers and serving as the host programming language for the app
- **Vue.js:** used to develop the front-end of the app
- **HTML:** responsible for developing the required Vue components and templates
- **Bootstrap:** used to make the front-end appealing and easy to navigate
- **SQLite:** serves as the database for the app
- **Flask:** serves as the web-framework for the app
- **Flask-SQLAlchemy:** used to access and modify the app's SQLite database
- **Flask-Celery**: used for asynchronous background jobs at the backend
- **Flask-Caching**: used for caching API outputs and increasing performance
- **Redis**: used as an in-memory database for the API cache and as a message broker for celery
- **Matplotlib:** used to create the various required charts
- **Git:** responsible for version control

Database Schema:

The database has four tables and the schema is as follows:

User Table	**Shows Table**
● **Id** (Integer): Primary Key, Auto Increment ● **Username** (String): Not Null ● **Email** (String): Unique, Not Null ● **Password** (String): Not Null ● **isAdmin** (Boolean): Not Null, Default 0	● **show_id** (Integer): Primary Key, Auto Increment ● **show_name**(String): Not Null ● **show_timing** (String): Not Null ● **show_rating** (Integer): Not Null ● **show_tags** (String): Not Null ● **show_ticketprice** (Integer): Not Null, Default 0

Venues Table	**Bookings Table**
● **venue_id** (Integer): Primary Key, Auto Increment ● **venue_name** (Integer): Foreign Key (Post.roll), Not Null ● **venue_place**(String): Foreign Key (User.username), Not Null ● **venue_location**(String): Not Null ● **venue_capacity**(Integer) ● **venue_show**(Relationship)	● **booking_id** (Integer): Primary Key, Auto Increment ● **booking_tickets** (Integer): Foreign Key (User.username), Not Null ● **user_id** (String): Foreign Key (user.username), Not Null ● **show_id**(String): Foreign Key(show.show_id) ● **user_rating**(Integer)

Architecture and Features:

The architecture of the Movie Ticket Booking follows a client-server model, where Vue.js serves as the front-end framework and Python-Flask as the back-end framework. Vue.js handles the presentation layer and manages user interactions through its MVVM architecture, while Python-Flask handles the server-side logic, such as HTTP requests and responses, asynchronous tasks, and database interactions.

The features of the application are as follows:

- **User authentication**: Signup and Login
- **User-specific API tokens**: Generate tokens to use user-specific requests
- **Venue analytics**: Graphs showing Show details and Venue details
- **Data export**: Download venues' shows and analytics as a CSV file
- **Venue management**: Create, view, edit, and delete posts
- **Show management**: Create, view, edit, and delete user accounts
- **RESTful API**: API available for posts, users, comments, and follows
- **Export blogs**: Export blog content as PDFs
- **Reminders**: Receive daily reminders to visit the website
- **Monthly engagement report**: Admin receives a report as an email summarizing engagement for the month

Video:

For the video, click here!