Author

Devathi Sai Mani Kumar
21f3003246
21f3003246@student.onlinedegree.iitm.ac.in
I am a technology enthusiast and an autodidact person.

Description

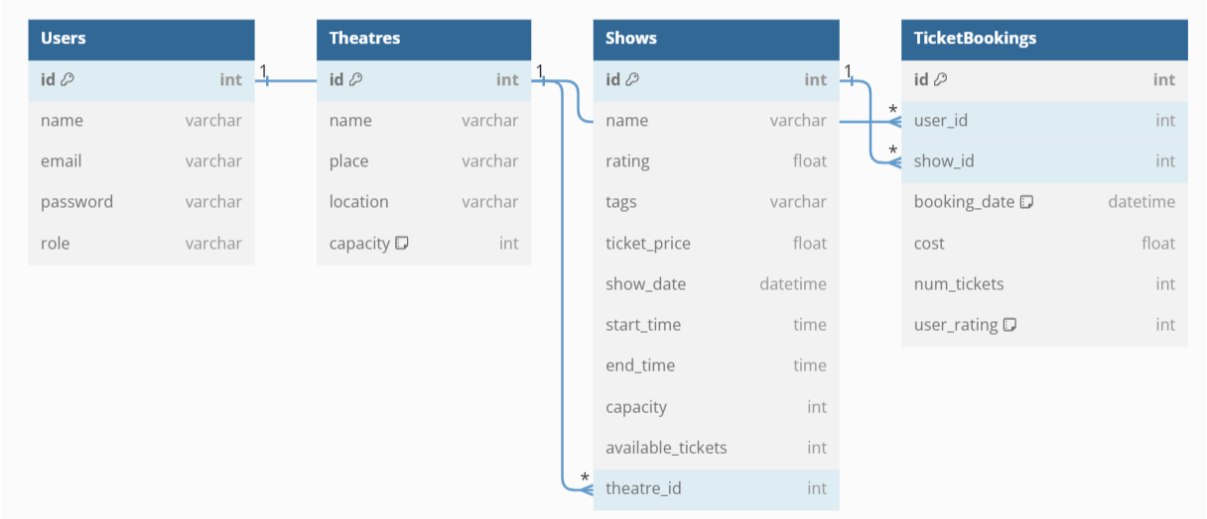
The Ticket Booking System is designed to streamline the process of booking tickets for shows at different theatres. It offers users the ability to register, log in, and browse available shows. Users can select their preferred showtimes and the number of tickets they wish to book. The system ensures ticket availability, handles user authentication, and provides booking confirmations. Additionally, the system equips admins with functionalities to manage theatres, shows, and view revenue reports for better overall control.

Technologies used

1. Vue.js: For building the frontend user interface, providing a smooth and interactive experience to users.
2. Flask: As the backend web framework, handling API requests, data processing, and business logic.
3. Flask-RESTful: For creating RESTful APIs to manage users, theatres, shows, and bookings.
4. SQLAlchemy: To interact with the database and manage data models effectively.
5. Flask-JWT-Extended: For implementing user authentication and securing API endpoints.
6. Flask-Caching: To cache certain endpoints and improve system performance.
7. Flask-Bcrypt: For securely hashing and managing user passwords.
8. Redis: Redis is used for caching and as a message broker, improving system performance and enabling task queue management.
9. Celery: Integrated with Redis, Celery handles background tasks, ensuring efficient processing of time-consuming operations.

DB Schema Design

The database schema includes the following tables:
1. Users: To store user details like name, email, password, and role.
2. Theatres: To store theatre information such as name, place, location, and capacity.
3. Shows: To store show details like name, rating, tags, ticket price, date, time, capacity, and available tickets.
4. Bookings: To store user bookings, including the show ID, user ID, booking date, number of tickets, and cost.



API Design

The API includes endpoints for various functionalities:

1. User Endpoints: For user registration and login.
2. Show Endpoints: To browse available shows and book tickets.
3. Admin Endpoints: For theatre and show management, as well as revenue reports.
4. JWT Authentication: The API implements JWT-based authentication to secure user endpoints.

Architecture and Features

The Ticket Show App follows a structured MVC (Model-View-Controller) architecture:
- Controllers: Handled API endpoints, interacted with models, and performed data operations.
- Models: Represented database entities and managed data interactions through SQLAlchemy.
- Views (Vue.js Components): Renders data fetched from API responses for users, ensuring a smooth and engaging frontend experience.
Key Features:
1. User Registration & Login: Allow users to create accounts and log in securely.
2. Show Browsing & Booking: Enable users to browse available shows and book tickets seamlessly.
3. Theatre & Show Management: Admins can manage theatres and shows efficiently.
4. Revenue Reports: Admins can access revenue reports for analysis and decision-making.
5. Email Notifications: Send email notifications for booking confirmations and updates.
6. Show Rating: Allow users to rate shows, enhancing the user experience and providing valuable feedback.

Video

https://drive.google.com/file/d/1e7CaDdTUinCcRtiShLa5ZGOoXIaWP4EX/view?usp=sharing