

Author:

Name: Kirupa Krishan G

Roll No.: 21f1006252

Email: 21f1006352@ds.study.iitm.ac.in

About Me: I am currently in my final year of engineering at Vel Tech University . I am eager and excited to build applications around data science and ML . I hope to utilize these knowledge and apply them towards my core domain of robotics

Description:

The project requires the development of a ticket booking application that utilizes Vue.js and Flask. The main tasks involve designing and building the application's frontend and backend, creating, and managing a database, and integrating the three components.

Technologies Used:

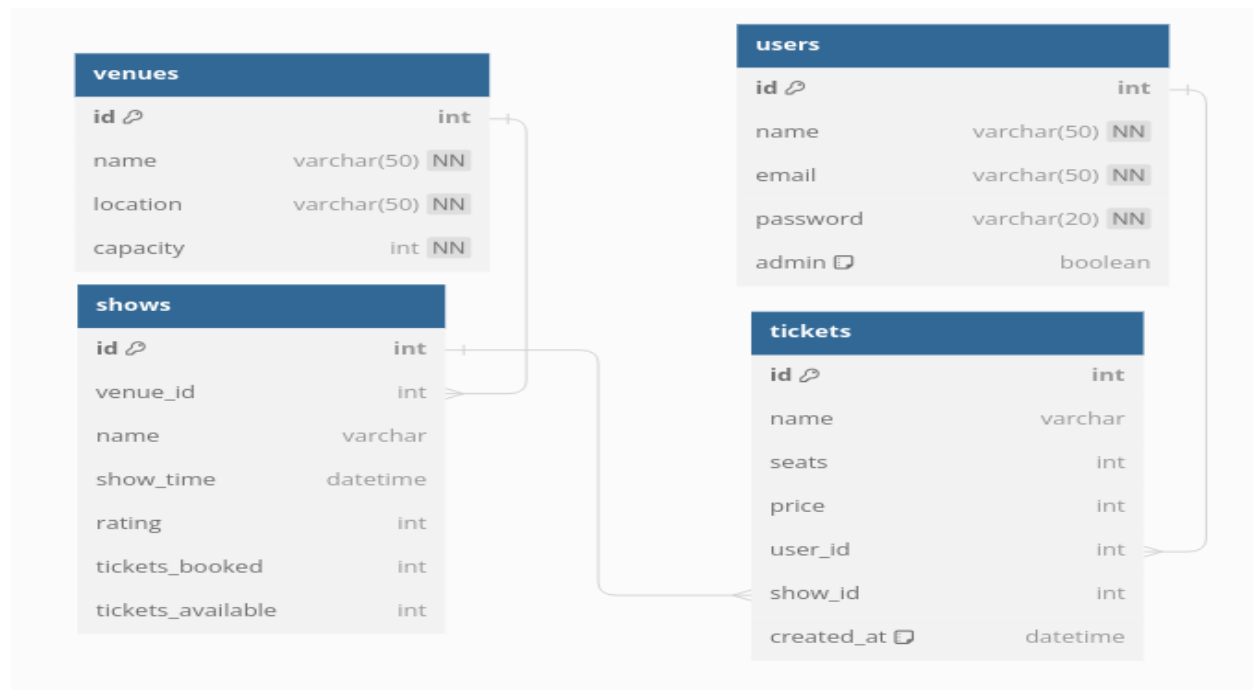
1. Vue Js: It is used for building the front end of the applica on.
2. Flask: It is used to create the backend server and to connect the frontend and database together.
3. Redis: It is used for caching the images used in the web applica on.
4. Flask-SQLAlchemy: It is used to connect Flask with SQLite database.
5. Flask CORS: It is used to enable CORS policy.
6. Sqlite
7. Vuex
8. Bootstrap-vue

API Design:

Routes:

1. Admin Route
2. Auth Route
3. Booking Route
4. Home Route

Ticket Book RouteDB Schema Design:



Architectures and Features:

The web application starts with the login page and the user can choose to create an account if they don't have one. Once logged in successfully, the user is shown all the available movies to book. There is a navbar where the user can choose to:

- navigate to the page where he can look at all his booked tickets.
- Filter movies based on location, name and rating.
- Signout

A movie dashboard where the user can book tickets.

If the logged in user is an admin of the portal he/she is directed to the admin portal that is different from the normal users. The admin has a navbar with special abilities to perform CRUD operation on the venues and shows. Each venue and show is marked with its basic data and along with their id to perform CRUD operations.

Features:

- Mail – Threads runs in the background and sends mail everyday at 5pm and also a monthly report is sent on the first day of the month.
- Caching – Redis is used to cache all the images.
- CRUD operations on Venues and Shows
- User Triggered Async Jobs – Export all the profile data as either pdf or csv file.

Video Links:

<https://drive.google.com/file/d/185fF1AsLtV71KEs9nkl7av2pOyy2tY1l/view?usp=sharing>