Author

Lalit K

22f2001140

22f2001140@ds.study.iitm.ac.in

I am a final Year CSE student from Tamil Nadu, who is interested in Data science and Web development

Description

A Flask based web application for the users to book tickets for the latest shows in their desired venues. It has a role-based authentication. The admin can create, update, and remove shows and venues. Schedule alerts to users, send monthly reports, export venue report as csv.

Technologies used

Flask – Framework used to create API for web application

Flask-SQL Alchemy – To define the database models and to manipulate data

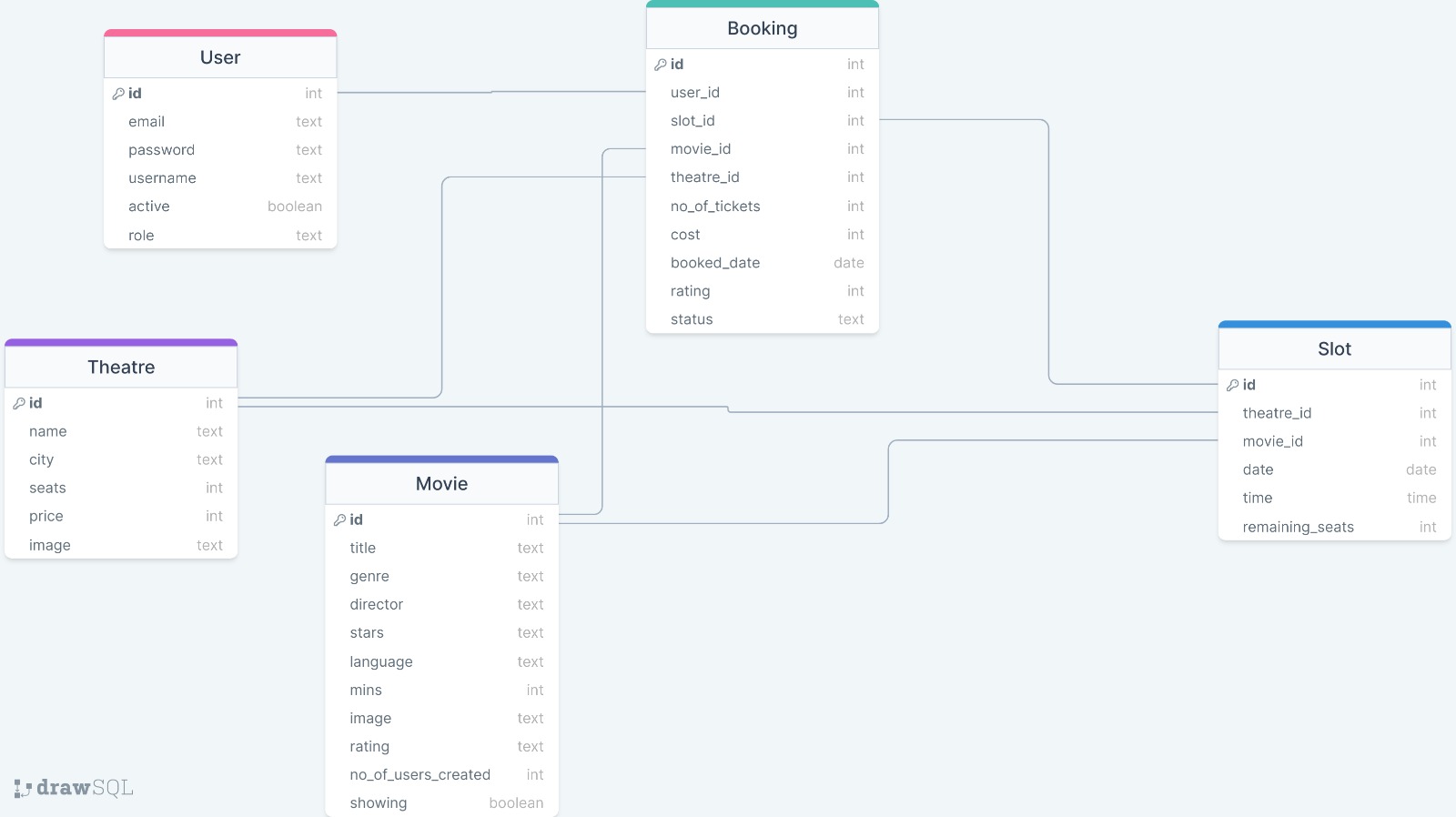
VueJS - For Client-Side Development (Front-End)

Axios - For API

Redis – For temporary data storage and faster access

Celery – To Schedule and export jobs/tasks

DB Schema Design



The table User and Booking has a one-many relationship, because a single user can book for multiple movies in multiple theatres.

The table Movie and Theatre has a one-many relationship, because a single theatre can have multiple movies running.

The table Slots and Movie has a one-many relationship, because a single movie can have multiple slots running.

Architecture and Features

Project : TicketShow

Project Tree:

|  |  |  |
| --- | --- | --- |
| **└─── backend**  **└───\_\_pycache\_\_**  **└─── main.py**  **└─── celeryconfig.py**  **└───**  **└─── instance**  **│** database.db  **└───website**  **├───\_\_pycache\_\_**  **│. env**  **│** models.py  │ \_\_init\_\_.py  | api.py  | config.py  | controllers.py  | dataaccess.py  **│** tasks.py  │ worker.py  **├───static**  **├───movies**  |…  **├ theatres**  |…  **├───exports**  |… | **└─── frontend**  **└─── public**  │ favicon.ico  │ index.html  **└─── src**  **│ App.vue**  **│ main.js**  **└─── components**  **└─── auth**  │ Login.vue  │ Signup.vue  **└─── admin**  │ CreateMovie.vue  **│** CreateTheatre.vue  │ CreateAdmin.vue  **│** DeleteMovie.vue  │ DeleteTheatre.vue  **│** DeleteSlot.vue  │ UpdateMovie.vue  │ UpdateTheatre.vue  | Addslot.vue  **└─── users**  **|** Movies.vue | | Theatres.vue  | selectedMovie.vue  | SelectedTheatre.vue  **|** selectedSlot.vue  | Booking.vue  | Profile.vue  **└─── utils**  │ Navbar.vue  **└─── router**  │ router.js  **└─── assets**  │ movies  | …  │ theatres  |…  │ index.jpg  │ logo.png  │ bg1.jpg  **└───** .gitignore  └─── README.md  └─── babel.config.js  └─── jsonify.json  └─── package-lock.json  └─── package.json  └─── vue.config.json |

The views are defined in the file api.py and the models are defined in models.py. \_\_init\_\_.py creates the app and database. The templates are stored as vue files and routing is performed in index.js . And some images used are stored in the static folder and assets. The celery tasks are scheduled in tasks.py and configured in worker.py.

Features Implemented:

CRUD for shows, CRUD for Venues, Redis Cache, Celery scheduled Jobs, Axios for API,

Video :