

Author

Nitesh Mahendra Arya

21f200218

21f2000218@ds.study.iitm.ac.in

I'm a Diploma student and an Engineer from HITS working as a Software Developer at Boeing India Pvt Ltd. I am also an AIML Enthusiast with a good hands on experience related to ML and GenAI.

Description

This application is a Full Stack project involving management and booking of parking spots. Here users can book the parking spot based on their desired location. Admins will have access to creating parking lots which will have multiple spots under it. This application needs to be secured based on the user's username and password, which will be handled by jwt-based authentication. Alongside the same jwt token will be used to provide controlled authorization to different parts of the application (Admin & User).

Redis will be used to provide caching options to optimize the performance, with the further integrations with sqlite for database, flask-mail to send and receive automated mails. ChartJs is used for the interactive dashboard with Bootstrap for different styles, icons and designs to make the application appealing and responsive across devices.

AI/LLM was used for certain static details generation, like creating a seed script which was used to populate database with dummy values which helps in testing the application. Some design inspirations were also taken for simplicity.

Technologies used

Backend

1. Flask - Used as a backend framework to create APIs serving the main business logics
2. Flask_jwt_extended - This was used for operations related to jwt tokens (Creation, identity etc)
3. Flask_SQLAlchemy - This was used for managing database, including creation of tables, manipulating rows and storing data.
4. Flask_Marshmallow - It is used to build a layer around apis which serializes and deserializes the response and requests. This also simplifies the validation logic by handling them at request level encapsulated at class level.
5. Celery - schedule jobs like sending mails, notifications and reports are achieved through celery
6. Redis - This was used for caching the apis as this is used for optimizations.
7. Flask-Mail - sending mails

Frontend

VueJs - Framework to create single page applications

Vuex - state management which stores data related to user at application state level.

Routers - Helps in managing routes/navigations of applications

Axios - making api calls to the backend

Bootstrap - CSS framework for components and designs, also bootstrap icons were used for appealing icons

ChartJs - Used to create charts

jsPdf - library to create PDFs

DB Schema Design

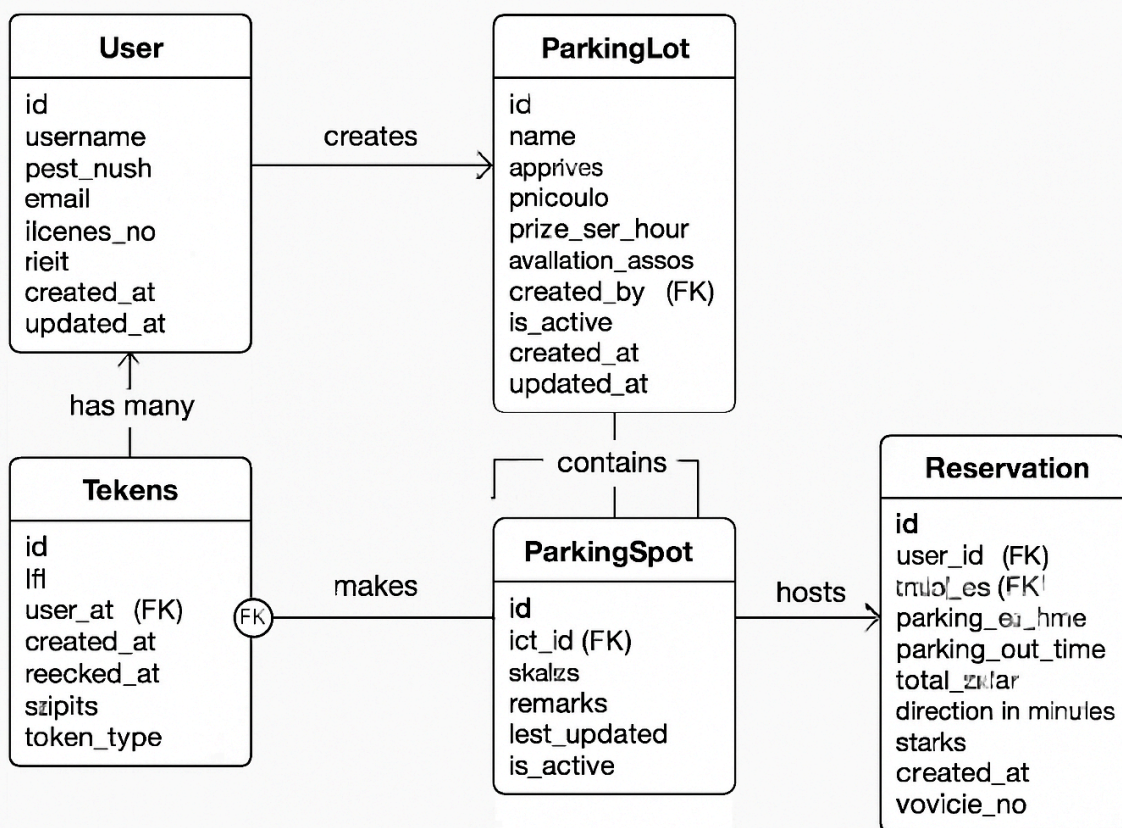
DB design consists of 5 tables:

ParkingSpot, ParkingLot, Users, Revoked_Tokens and Reservation. Also Enums were used to handle the flags related to statuses.

Relationships:

Table had mainly one-to-one relationships which were:

- User -> token
- user-> Reservation
- User -> ParkingLot
- ParkingLot -> ParkingSpot
- ParkingSpot -> Reservation



API Design

Modular design principle was used to design the APIs where logic for the apis were separated in service layer, db operations were made through models. And api request , responses were handled through schema objects. This gives an edge as endpoints need not to be changed everytime, changing service layer logic will cascade to apis.

Link - https://drive.google.com/file/d/1_rSTvtvwUjrFQKZWob-tWpZxftH_h-nj/view?usp=sharing

Architecture and Features

The project architecture follows an MVC pattern where different tasks are encapsulated as separated entities. The backend was separated from the frontend. Frontend was developed based on the principle of Single Page Application (SPAs).

Backend

It is organised in multiple subdirectories, each kind of representing a separate layer. Mainly, Factory architecture was adopted here, where every object was declared in an extension and application startup was handled by separate functions.

Frontend

SPA was the principle used where, routes was considered as the core and interceptors were used to handle management of api calls and vuex store was used was state management

Video

Link - https://drive.google.com/file/d/1Mv5kHRHHsw_zjNARWVoq8DgjU58hqAA5/view?usp=sharing