Author

Puja Nilesh Rachchh
23f3003284
23f3003284@ds.study.iitm.ac.in
I am a diploma level student of IITM BS Degree in Data Science and Applications.

Description

I developed a multi-user **Vehicle Parking App** for 4-wheeler management using **Flask**, **SQLite**, and **Jinja2 templating**. The app includes two roles: an admin, who can create and manage parking lots and view the status of all spots, and regular users, who can register, reserve an available spot, and vacate it when done. The system dynamically handles spot allocation and prevents deletion of occupied spots. Time-based billing is implemented, and charts summarizing parking statistics are generated using **Matplotlib**. The database is created programmatically, and all core functionalities are implemented as per the project requirements.
Total percentage of AI/LLM used was about 15-20% including both frontend and backend.

Technologies used

Frontend – HTML, CSS and Bootstrap have been used for styling and responsiveness of app.

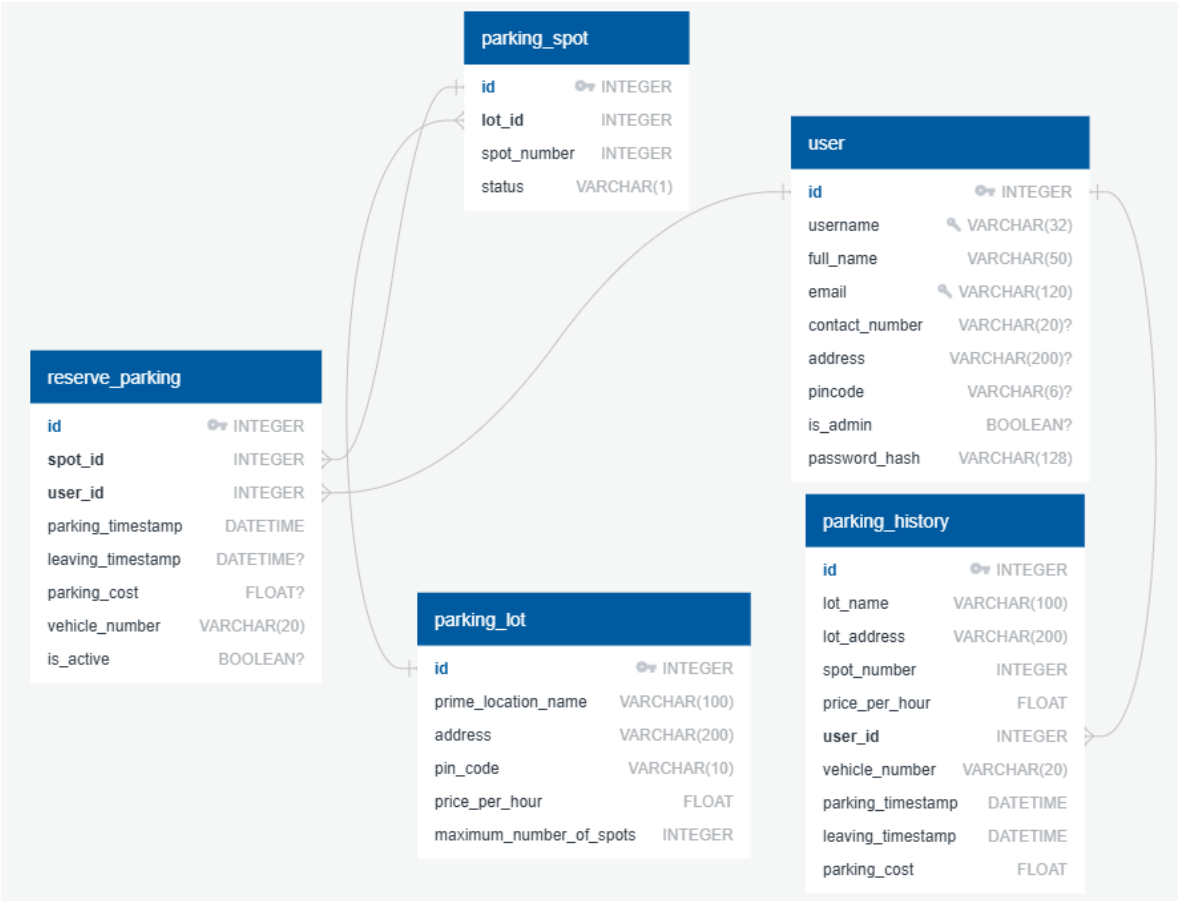
Template Engine - Jinja2 has been used for HTML template generation.

Backend - The application uses Flask, SQL Alchemy to implement core functionalities, in addition, with matplotlib for generating graphs.

Data Storage - The application uses SQLite for data storage.

IDE - The application has been developed in Visual Studio Code using Virtual Environment.

DB Schema Design



Architecture and Features

Architecture:

The root directory includes **app.py** to launch the application, and **config.py** to manage configuration variables. Database models are defined in **models.py**. The **templates/** folder is structured with two subfolders—**Admin/** and **Users/** for Admin and User pages respectively. All styling and images and CSS are placed in the **static/** folder. Core application logic and route handling are implemented in **backend/routes.py**

Features:

The project offers essential features such as **user registration**, **login**, **parking spot booking**, and dashboards tailored for both users and admins. Admins can **add or edit parking lots**, **manage users**, and **monitor spot status**. Users **can book, release,** and **view their parking details**. Advanced features like **real-time cost calculation during spot release** and **summary charts using Matplotlib** enhance usability. These functionalities are implemented using **Flask** for routing, **SQLAlchemy** for database operations, and **Jinja2** templates for rendering dynamic content.

Video

https://drive.google.com/file/d/1VzoqZ_jwRR3xByXjjLxvJqTvcvoJwKuM/view?usp=sharing