Vehicle Parking App

Project Report

Author

Name: Vedant Sahu
Roll Number: 23F2003040
Email: 23f2003040@ds.study.iitm.ac.in

About Me:
I am a diploma level student of IITM BS Degree in Data Science and Applications.

Description

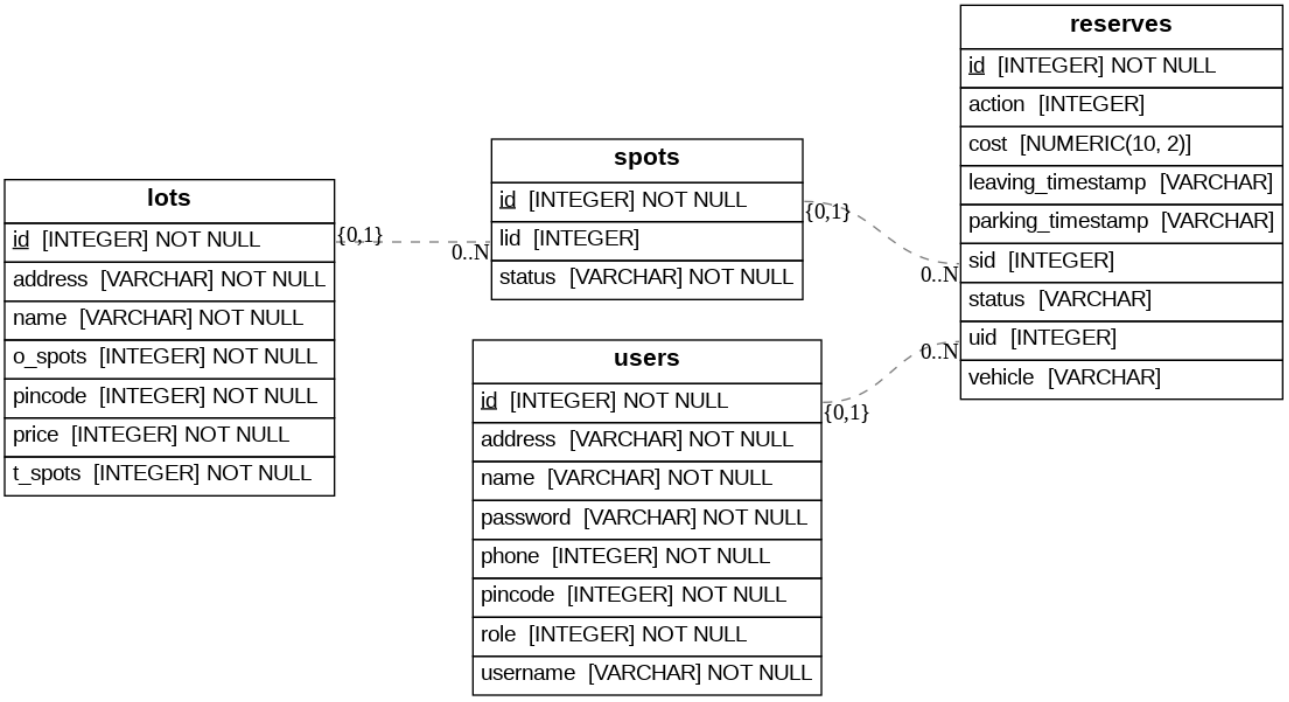
The aim of this project is to create a Vehicle Parking Application which helps the user to find, book, and manage parking spots. It is a multi-user app. There can be multiple Users but only one Admin. Admin and Users have their dashboard. User can book many parking spots for his vehicles. Admin can create Parking Lots and Parking Spots. Every Lots can have several spots. System will Automatically create parking spots based on the maximum capacity of the lot. Search functionality is being included. There is a Summary page which depicts the revenue generated from each parking lots.

Technologies Used

Frontend: HTML, CSS , Jinja, Bootstrap and JQuery have been used for styling of app.
Backend: The application uses Flask, SQL Alchemy to implement core functionalities, in addition, with Matplotlib for generating graphs.
Database: The application uses SQLite for data storage.

DB Schema Design

Design Rationale:
Clear role separation, referential integrity via foreign keys, and cascading deletes ensure smooth data flow and minimal redundancy.



API Design

The API is designed for server-side rendered web pages, not as a RESTful JSON API.

Endpoints:

- **User Authentication & Registration**
  - /login (POST): Authenticates users and admin.
  - /user_register (POST): Registers new users.
- **Dashboard Views**
  - /admin_dashboard (GET): Loads admin dashboard.
  - /user_dashboard (GET): Loads user dashboard.
- **Parking Lot Management**
  - /lotadd (POST): Adds a new parking lot and its spots.
  - /lotedit (POST): Edits parking lot details and spot count.
  - /lotdelete (POST): Deletes a parking lot (if all spots are empty).
- **Parking Spot Actions**

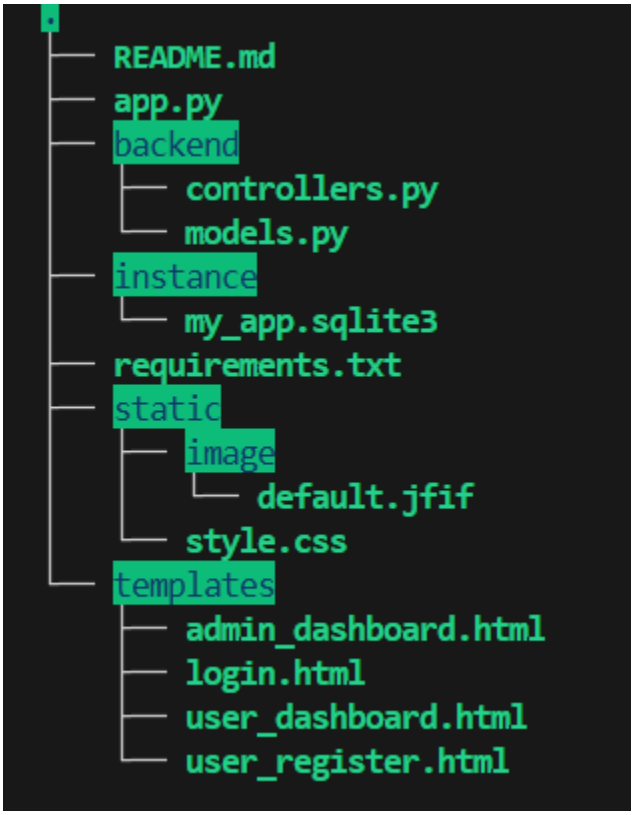
    - /book/<int:userid> (POST): Books a parking spot for a user.
    - /release/<int:userid> (POST): Releases a booked spot.
    - /payment/<int:userid> (POST): Marks payment as completed for a reservation.
- **Search**
    - /search (POST): Searches users or parking lots based on criteria.

Note on Use of AI

AI tools (LLMs) were used during development for debugging and error resolution.

Architecture & Features

Structure:
parking_app_23f2003040/



Features Implemented:

- **User Authentication:** Login and registration for users and admin, with role-based access.

- **Parking Lot Management:** Admin can add, edit, and delete parking lots and manage parking spots.
- **Spot Booking and Release:** Users can book available spots, release them, and pay for their parking.
- **Search Functionality:** Both admin and users can search for lots and users using various criteria.
- **Statistics and Visualization:** The app generates bar, pie, and line charts using Matplotlib to visualize spot status, revenue, and user parking history.

Video

Video Link: https://drive.google.com/drive/folders/1O4DBT_NJkB_4HZXGLAWOiDm3J2RHUorY?usp=sharing