Vehicle Parking Application

MAD - 1

Author

Name: Rahul Sharma
Roll Number: 23f1001879
Email-id: 23f1001879@ds.study.iitm.ac.in
Currently, I am a full time student pursuing this degree.

Project Description

This project is a vehicle parking management application that allows users to register, book parking slots, and track their reservations. Users can view available parking lots, reserve a spot, and calculate costs based on time used. Admins have the ability to manage parking lots and view overall system activity. The system ensures smooth parking operations and helps users to manage their parking more efficiently.

Technologies Used

- Flask - for application used
- Jinja2 templates, Bootstrap and HTML/CSS for styling
- SQLite and SQLAchemy for Database work

Architectures and Features

The project follows the **MVC architecture** (Model-View-Controller), where:

- **Models** handle the database interactions.
- **Views** present the frontend templates.
- **Controllers** manage the application logic and routing.

User Features:

- New users can register and log in.
- After logging in, users can:
    - View parking lots and available spots.
    - Make reservations by selecting a slot and time.
    - Check their reservation history.
    - View total cost and status of current/previous bookings.
    - Edit their profile information.

Reservation Logic:

- Users can book a spot if available anytime.
- On exiting, the system calculates the time duration and cost.
- Time is automatically adjusted to local time (e.g., out_time + 5 hours 30 minutes, if needed for display for IST).

Admin Features:

- Admins are added to the database automatically on creation.

- Admins can:
    - Create and manage parking lots.
    - Monitor parking usage and user activity.
    - View and manage parking spots.
    - Delete or update lots and manage users.

DataBase Schema Design

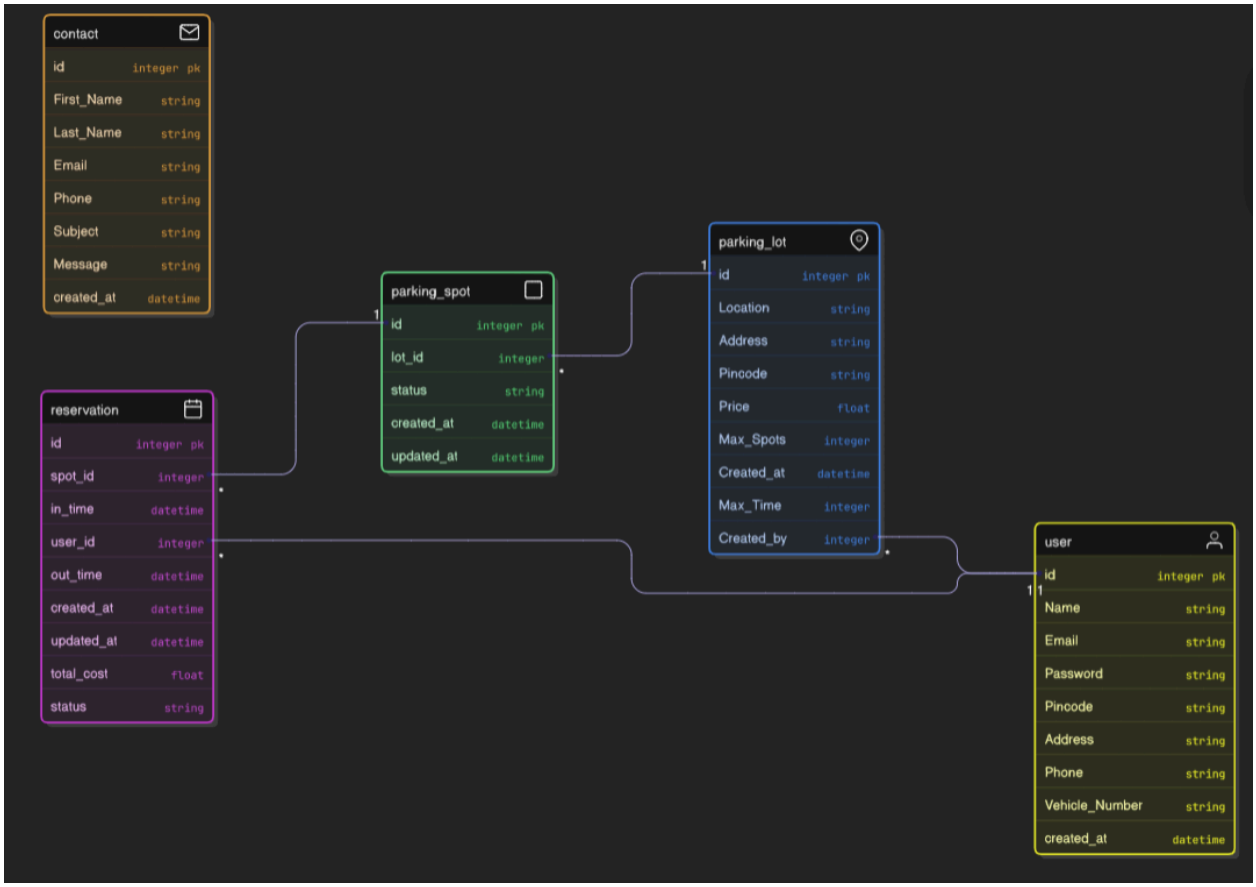

Fig1. ER Diagram of my Vehicle Parking application

The database has been designed to store information about users, parking lots, parking spots, reservations, and contact messages.

Tables

- **User:** id, Name, Email, Password, Pincode, Address, Phone, Vehicle_Number, created_at
- **Parking_Lot:** id, Location, Address, Pincode, Price, Max_Spots, Created_at, Max_Time, Created_by
- **Parking_Spot:** id, lot_id, status, created_at, updated_at
- **Reservation:** id, spot_id, in_time, user_id, out_time, created_at, updated_at, total_cost, status
- **Contact:** id, First_Name, Last_Name, Email, Phone, Subject, Message, created_at

Future Improvements

This project can be improved by adding functionalities for premium park space creation, vehicle types and a payment gateway system. Additionally, the interface can be furnished much more than this if required.

Conclusion

In conclusion, the project was a success in creating a platform for user to book parking spots by just one click and check their info easily. Use of Flask, jinja2 and SQLAlchemy made the development much more effective.

Project Demonstration Links:

- ***Video Drive Folder link:*** ***Vehicle Parking App MAD1 23f1001879***