# Vehicle Parking Management System — Project Report

Student Name: Patel Megh Shirishkumar
Student Roll/Registration Number: 24f2002973
Email: 24f2002973@ds.study.iitm.ac.in
Course: Modern Application Development I - Project

## 1. Project Details:

### a) Problem Statement

To build a web-based Vehicle Parking System using Python Flask and SQLite, supporting multiple users and an admin role. The application should allow the admin to add/edit/delete parking lots, users to register/login, reserve/release parking spots, and display an overview/dashboard. Only the admin has the power to add or remove lots, and lots may only be deleted if all spots are free. The database must be created programmatically from the code, and all standard security and usability practices are to be followed.

### b) Approach

I broke the problem down as follows:

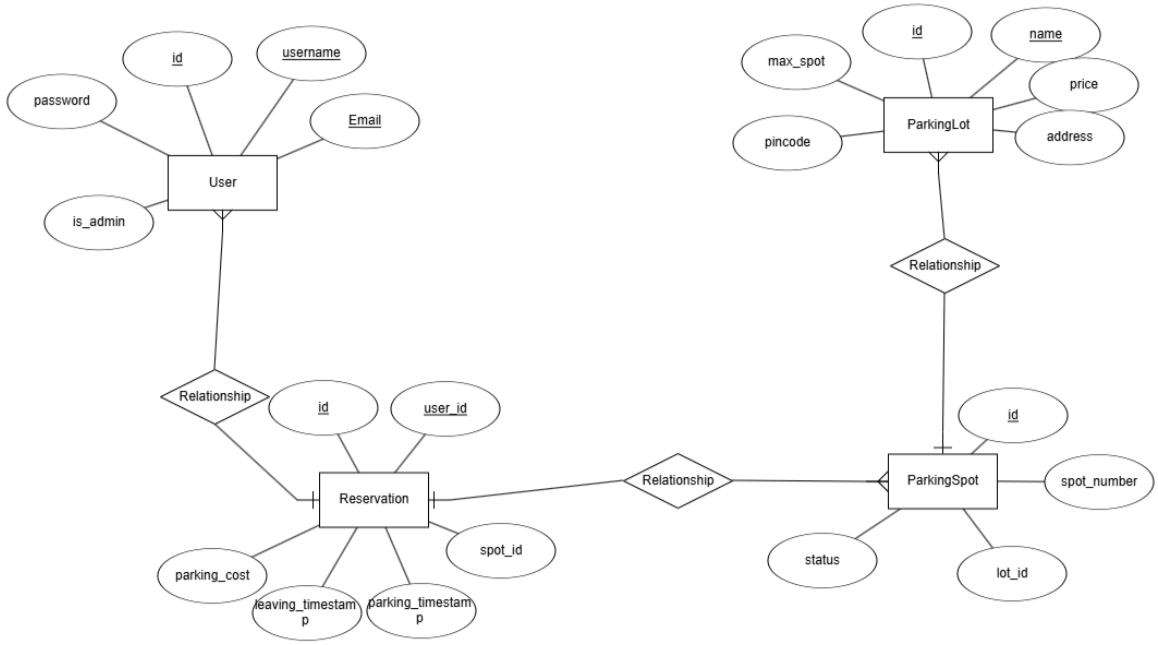
- Identified the core entities: User, Admin, ParkingLot, ParkingSpot, Reservation
- Designed the ER diagram and corresponding SQLAlchemy models
- Created a Flask application with routes for both user and admin functionalities
- Maintained session-based authentication and role distinction, with secure password handling
- Developed and linked Jinja2 HTML templates for the various pages
- Used Bootstrap for a clean, responsive UI
- Implemented database creation via a CLI/init script
- Ensured that deletion of lots is database-safe (cascade deletes), and only possible if all spots are free

## 2. Frameworks and Libraries Used:

- Python 3
- Flask: Backend web framework
- Flask-SQLAlchemy: Database ORM for managing all DB operations from Python code
- Werkzeug: For secure password hashing
- Bootstrap 5: (via CDN) for styling and layout of the web pages

- SQLite: Lightweight embedded database

## 3. Entity-Relationship (ER) Diagram:




Description of relationships:

- One user can have many reservations.
- One spot can be reserved many times (different reservations at different times).
- One lot can have many spots.

## 4. API Resource Endpoints:

The following endpoints were created as HTML form/pages:

- /login — Login for user/admin
- /register — User registration
- /logout — Logout
- /admin — Admin dashboard
- /admin/lots — Add/view parking lots
- /admin/lots/delete/<lot_id> — Delete a lot
- /admin/spots/<lot_id> — View all spots in a lot
- /admin/users — View all users
- /user — User dashboard
- /user/lots — View lots (user)

- /user/reserve/<lot_id> — Reserve a spot
- /user/release — Release a spot

**5. Video Presentation Link:**

https://drive.google.com/file/d/1nAh7QkgYwMt8vcrAMjUCIBkhPU05UKQH/view?usp=sharing

**6. Use of AI/LLM Assistance:**

Yes, I used generative AI tools for:

- Debugging errors and getting code explanations
- Suggestions for Frontend design

Final logic and features were developed and tested by myself. AI suggestions were reviewed and adjusted to fit the assignment.

End of report.