# Vehicle Parking Management System - Final Project Report

## Author

Name: Balasubramaniam

Roll Number: 21f3000842

Email: 21f3000842@ds.study.iitm.ac.in

I am Balasubramaniam, a student in the BS Data Science program.

## Description

This project implements a multi-user vehicle parking system using Flask. Users can reserve parking spots, and administrators can manage lots and users. ChatGPT was used occasionally to resolve syntax errors and UI issues. AI/LLM usage: ~10% - Limited to bug fixing and HTML syntax reference.

## Technologies Used

- Python, Flask
- SQLite with SQLAlchemy ORM
- Jinja2 for HTML templating
- HTML/CSS with Bootstrap
- Flask extensions: flask_sqlalchemy, flask

These were chosen for rapid web development and ease of integration.

## DB Schema Design

The database contains the following tables:

- User: Stores user credentials and links to roles.
- Role: Defines user roles (admin/user).
- UserRole: Associates users with roles.
- ParkingLot: Contains metadata about parking lots.
- ParkingSpot: Individual parking spots linked to lots.
- ReserveParkingSpot: Reservation history with spot_id, lot_id, lot_name, timestamps.

Check constraints and foreign keys enforce data integrity.

## API Design

No REST APIs were implemented. All interactions are server-rendered with Flask.

## Architecture and Features

The project is structured into modular files:
- `routes.py` contains all Flask route logic.
- `models.py` defines the database schema using SQLAlchemy.
- HTML templates are in the `templates/` folder using Jinja2.
- Static files (CSS/JS) are in the `static/` folder.

Features:
- Admin login and lot management (create/edit/delete lots).
- User login and reservation.
- Automatic billing based on time parked.
- Admin and user summary pages for reservation history.

## Video

Link to the demo video:
https://drive.google.com/file/d/1xQ9KIDryXKmA_aacuzPIQ6lQlfpBMzWC/view?usp=sharing