# MAD I

# Vehicle Parking App ( Project Report )

**Author**

Name: Sivani Krishna T

Roll No.: 23f1000209

Email: 23f1000209@ds.study.iitm.ac.in

**Description:**

This project involves developing a user-friendly platform for managing parking by implementing user roles (Admin and User) involving real-time spot booking features. Admin can add, edit, or delete parking lots and spots, while users can book and release spots. The system provides clear dashboards, booking history, and live availability updates, making parking management and usage simple, efficient, and organized.

**Technologies used:**

**HTML** -- Defines the structure and content of web pages.

**CSS** -- Adds styles and layout to the frontend UI.

**Jinja2** -- Templating engine for dynamic HTML rendering with Flask.

**Bootstrap --** Used for responsive design and prebuilt UI components.

**Chart.js** -- Displays interactive charts for booking and usage data.

**Python** -- Primary programming language used for application logic.

**Flask** -- Web framework for building and handling backend routes.

**SQLite3** -- Lightweight database to store user, lot, and booking data.

**Architecture and Features:**

The root folder consists of ‘app.py’ that has all the major configurations related to initiating the app and consist of a folder template with all the html files included. There is an application folder which consists of ‘controllers.py’ defining routes , ‘database.py’ for initiating database object and ‘models.py’ storing the table models. Instance is created during runtime by Flask and static consists of images The project is built using HTML, CSS, Jinja2 templates, Bootstrap , SQLite3, and Python with Flask.

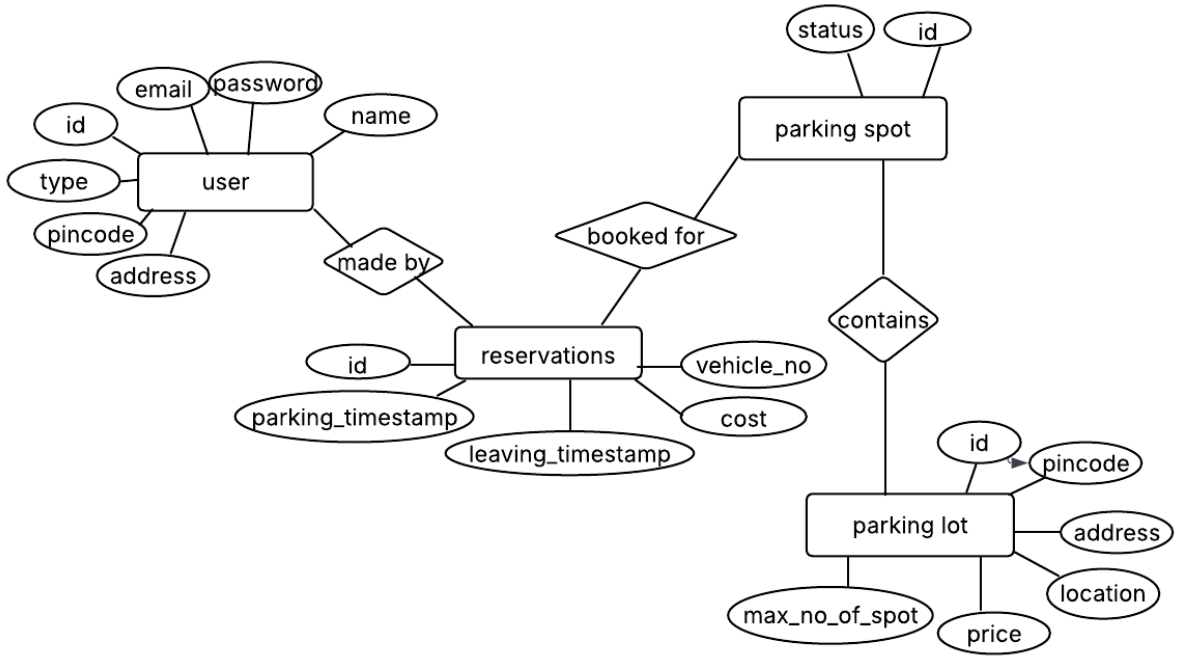
**Default Features:**

- User Authentication: Sign up, login, and session handling using Flask and Flask-Session.
- Parking Lot & Spot Management: Admins manage lots/spots with CRUD operations using Flask-SQLAlchemy.
- Real-Time Booking: Users book or release spots dynamically; status updates are reflected immediately.
- Search Functionality: Users can search lots by location or pin code using filtered queries.
- Responsive Design: Implemented using Bootstrap for smooth access across devices.

**Additional Features:**

- Chart Visualization: Recent booking activity shown via interactive charts using Chart.js.
- Profile Editing: Users can update their profile details.

**DB Schema Design:**




**Video:**

https://drive.google.com/file/d/1KIUXlZSlPGck3OrlwyBpBSLIbvz1xoAO/view?usp=drive_link