# Project Report

**Author**
**Name:** Krishna Vallabha Goswami
**Roll Number:** 23F3002697
**Email:** 23f3002697@ds.study.iitm.ac.in

**About Me:**

I am currently a Diploma level student BS Degree in Data Science and Applications at IIT Madras with a strong interest in Artificial Intelligence, Machine Learning, and technology-driven problem solving. He is passionate about building impactful, real-world applications by combining data-driven insights with intuitive design

**About Project**

The objective of this project is to design and develop a smart vehicle parking system that allows users to view available slots, book them in real-time, and manage parking efficiently. The system should simulate or reflect real-life parking scenarios to improve urban mobility and reduce congestion.

**Technologies Used**

**Backend (Python, Flask)**

1. Flask
   The main web framework used to build the backend of the application. It handles routing, rendering templates, and managing server-side logic.
2. Flask-SQLAlchemy
   Used as the Object Relational Mapper (ORM) to interact with the SQLite database in a Pythonic way.
3. Flask-Bcrypt
   Enables secure password hashing for storing user passwords safely in the database.
4. Flask-Login
   Handles user authentication, session management, and login/logout functionalities.
5. Flask-WTF
   Provides integration with WTForms for rendering secure HTML forms with CSRF protection.
6. Jinja2
   A templating engine used by Flask to render dynamic content in HTML files.
7. Werkzeug
   Provides WSGI utilities and tools that power Flask’s request and response handling mechanisms.

**Frontend (HTML5, CSS3)**

1. HTML5 & CSS3
   Used to structure and style the web pages, making them responsive and visually appealing.
2. Chart.js
   A JavaScript library used to display dynamic and interactive charts for analytical purposes.

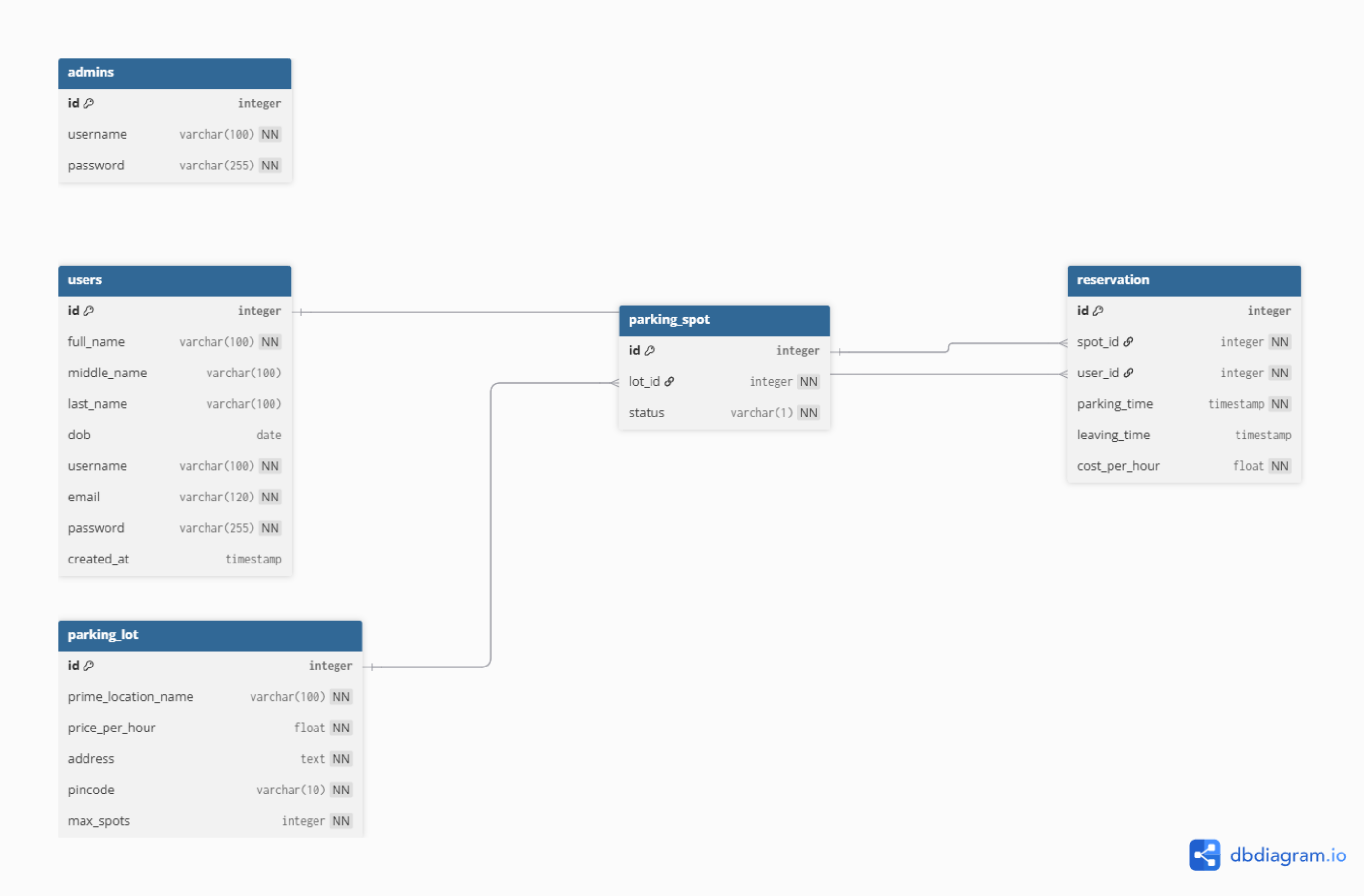
**Database**

1. SQLite
   A lightweight and self-contained database engine used for storing user data, login credentials, and other records.

**Development Tools**

1. Virtual Environment (venv)
   Used to manage dependencies and isolate the project environment.
2. pip
   Python's package manager used to install and manage libraries listed in requirements.txt.
3. Git
   Used for version control, code tracking, and collaboration.

### DB Schema Design



| admins | | |
|---|---|---|
| id | integer | |
| username | varchar(100) | NN |
| password | varchar(255) | NN |

| users | | |
|---|---|---|
| id | integer | |
| full_name | varchar(100) | NN |
| middle_name | varchar(100) | |
| last_name | varchar(100) | |
| dob | date | |
| username | varchar(100) | NN |
| email | varchar(120) | NN |
| password | varchar(255) | NN |
| created_at | timestamp | |

| parking_spot | | |
|---|---|---|
| id | integer | |
| lot_id | integer | NN |
| status | varchar(1) | NN |

| reservation | | |
|---|---|---|
| id | integer | |
| spot_id | integer | NN |
| user_id | integer | NN |
| parking_time | timestamp | NN |
| leaving_time | timestamp | |
| cost_per_hour | float | NN |

| parking_lot | | |
|---|---|---|
| id | integer | |
| prime_location_name | varchar(100) | NN |
| price_per_hour | float | NN |
| address | text | NN |
| pincode | varchar(10) | NN |
| max_spots | integer | NN |

### Architecture and Features

**Architecture:**

- Follows the MVC (Model-View-Controller) architecture.
- Controllers manage the logic and routing using Flask.
- Templates (HTML + CSS) are used for front-end layout and rendering views.
- Models handle database interactions for users, reservations, and parking lot data.

User Features:

- Dashboard showing available parking lots.
- Search parking lots by pin code, address, or name.
- View rate per hour for each lot.
- Access personal analytics, including:
- Total reservations made so far.
- Total amount spent.
- Monthly expenditure.
- Average parking duration.
- Monthly spend trend (graph).
- Usage by day of the week.
- Favorite parking lot.
- Peak usage hours.
- View parking history.
- Edit profile option.
- Logout option.
- Book / Release a parking spot.
- Extend Parking lot reserved time.

**Admin Features:**

1. Dashboard Layout:
    - Home, Dashboard, Users, Sales, Summary.
    - Parking Lot & Spot Management:
    - View and search parking lots.
    - Add / Update lots.
    - Manage spot status (Available / Reserved).

2. Edit:
   - Maximum spots.
   - Rate per hour.
   - Area, lot name, and address.
3. User Section:
   - View registered users with:
   - Username, name, email.
   - Total reservations made.
   - Active reservations.
4. Sales Section:
   - Total sales (All-time).
   - Sales this month.
   - Monthly sales trend.
   - Monthly reservation count.
   - Recent reservations
5. Summary Section:
   - See total number of parking lots and overview analytics.

**Demonstration Video**

https://drive.google.com/file/d/1zYm-MVre_wSEUoF5rxHQIClvgVAQuMwl/view?usp=drive_link