Vehicle Parking App

MAD 2 Project Report

Submitted by

Name: Praveen MK

Roll number: 22f3000299@ds.study.iitm.ac.in

IITM Online BS Degree Program,

Indian Institute of Technology, Madras, Chennai

Tamil Nadu, India, 600036

Table of Contents

Introduction

The Car Parking Management System is a web-based application designed to streamline the process of booking, tracking, and reporting parking activities. The system provides a user-friendly interface for both end-users and administrators, enabling efficient management of parking lots, user authentication, booking his- tory, and automated reporting. The primary goal is to enhance the parking expe- rience through automation, transparency, and ease of use.

Technologies Used

- Backend: Python 3.8+, Flask, Flask-SQLAlchemy, Flask-Migrate, Flask-Caching, Celery, Redis
- Frontend: HTML, CSS, Bootstrap, Vue.js, Vue CLI
- Database: SQLite (can be replaced with PostgreSQL/MySQL)
- Libraries: Flask-JWT-Extended for authentication, Flask-Mail for email notifications, Flasgger for API documentation, Chart.js for data visualization
- Others: Docker (optional, for containerization)

Installation

Prerequisites

- Python 3.8 or higher
- Node.js 14+ and npm 6+
- Redis server (for caching and Celery

Backend Setup

1. Navigate to the backend directory:

   ```
   cd backend
   ```

2. Create and activate a virtual environment:

   ```
   python -m venv venv
   source venv/bin/activate
   ```
   (Linux/Mac) or `venv\Scripts\activate` (Win- dows)

3. Install dependencies:

   ```
   pip install -r requirements.txt
   ```

4. Set environment variables (see `.env.example` or `README.md`).
5. Initialize the database:
   ```
   flask db upgrade
   ```
6. Start the backend server:
   ```
   flask run
   ```
7. Start Celery workers (in a new terminal):
   ```
   celery -A celery_app.celery worker --loglevel=info
   celery -A celery_app.celery beat --loglevel=info
   ```

Frontend Setup

8. Navigate to the frontend directory:
   ```
   cd frontend
   ```
9. Install dependencies:
   ```
   npm install
   ```
10. Start the development server:
   ```
   npm run serve
   ```

Design docs

1. ERDiagram: mad 2 ERdiagram.png

2 API docs here, 22f3000299\API.md

Usage

Project presentation: mad2 demo.mp4

5. Conclusion

The Car Parking Management System leverages modern web technologies to provide a robust, scalable, and user-friendly solution for parking management. With features such as user authentication, real-time booking, automated reporting, and performance optimization through caching and background processing, the system is well-suited for deployment in real-world environments. The modular architecture ensures ease of maintenance and future scalability.