Parkbuddy
MAD 2 Project Report

Author:

- Name: Jaskirat Singh Riyat
- Roll No: - 23f1000718
- Email: 23f1000718@ds.study.iitm.ac.in

Description:

ParkBuddy is a robust smart parking management system designed for both administrators and regular users. The platform features a secure multi-user system with separate interfaces for admins and users, enabling seamless parking lot management, real-time spot allocation, and automatic billing based on parking duration. Admins benefit from a comprehensive dashboard with analytics, full CRUD operations for lots, user management, real-time spot monitoring, and revenue tracking. Users can easily register, book parking spots, manage active sessions, review their parking history, and download CSV reports. The system leverages JWT-based authentication, Redis caching for high performance, and Celery-powered background jobs for tasks like daily reminders, monthly reports, and CSV exports. Optimized queries and intelligent caching ensure a smooth, real-time experience for all users.

Technology Stack:

Backend

- **Flask**: Web framework for API development
- **SQLAlchemy**: ORM for database operations
- **SQLite**: Database (as per requirements)
- **Redis**: Caching and message broker
- **Celery**: Background task processing
- **JWT**: Token-based authentication
- **Flask-Mail**: Email functionality (implemented via mailhog)

Frontend

- **Vue.js 3**: Progressive JavaScript framework
- **Bootstrap 5**: CSS framework for responsive design
- **Chart.js**: Data visualization
- **Bootstrap Icons**: Icon library

API Endpoints

Authentication:

- POST /api/auth/register - User registration
- POST /api/auth/login - User login
- POST /api/auth/admin/login - Admin login

Admin Endpoints:

- GET /api/admin/dashboard-stats - Dashboard statistics
- GET /api/admin/users - List all users
- GET /api/admin/lot-details/<id> - Parking lot details
- POST /api/lots - Create parking lot
- PUT /api/admin/edit-lot/<id> - Edit parking lot
- DELETE /api/lots/<id> - Delete parking lot

User Endpoints:

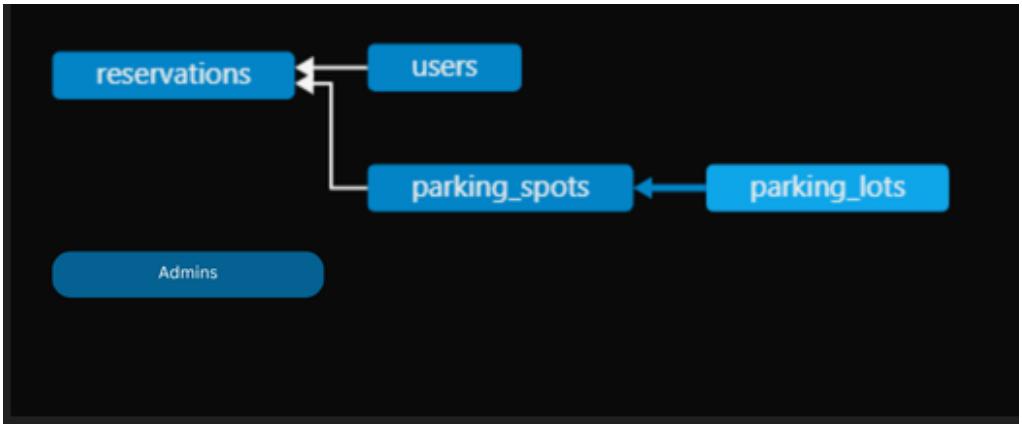
- GET /api/user/stats - User statistics
- GET /api/user/reservations - Parking history
- GET /api/user/active-reservations - Active reservations
- POST /api/user/reserve/<lot_id> - Reserve parking spot
- POST /api/user/release/<reservation_id> - Release parking spot
- POST /api/user/export-csv - Trigger CSV export

Common Endpoints:

- GET /api/lots - List all parking lots

Database Schema

- **users – handles user accounts**
- **admins – manages admin profiles**
- **parking_lots – stores lot info**
- **parking_spots – tracks individual spots**
- **reservations – logs bookings & billing**



Video demo:

https://drive.google.com/file/d/17fBeA3Sbcg4lWByT0QaPaiUt8fuYhdej/view?usp=sharing

(The AI/LLM percentage for the app is around 40-50%)