Indian Institute of Technology Mandi

CS-303(Software Engineering)

Group No: 02

Assignment: Smart Parking System

Group Details:

Luv Sharma (B22115)

Sameer Gupta(B22127)

Krish Mittal (B22214)

Bilal Muhammad Khan(B22293)

Ch Sunil Patra (B22294)

Contents:

Topics **Page No**

I. **Project Overview** -- **01**

II. **Technical Implementations:** --- **02-04**

a. Technology Stack

b. System Architecture

c. Database Design

III. **Features & Functionalities** -- **04-05**

IV. **ER Diagram** -- **05-06**

V. **User Flow** -- **06-07**

VI. **Competitive Benchmarking** -- **07-08**

VII. **Future Scope** -- **08**

VIII. **Conclusions** -- **09-10**

Project Overview

- **Smart and Convenient Parking:** It is a web-based parking management system that helps users find and book parking slots easily.

- **Real-Time Slot Booking & Monitoring:** Users can see available parking spots in real-time and reserve them instantly.

- **Multi-Level Parking Management:** The system supports parking across multiple basement levels, making it efficient for large facilities.

- **User Authentication & Booking History:** Every user needs to sign up and log in to book a slot. They can also view their past and current bookings for better trac

Technical Implementations:

I. Technology Stacks:

- **Frontend (User Interface & Experience):**

  - **React.js with Vite –** A fast and lightweight framework to build a responsive and interactive web interface**.**
  - **Material-UI Components –** Pre-designed UI components that enhance the look and feel of the application.
  - **JWT (JSON Web Token) for Authentication –** Ensures secure user login and session management.
  - **Axios for API Calls –** Handles communication between the frontend and backend efficiently.

- **Backend (Server & Database Management):**

  - **Flask (Python) –** A lightweight and powerful web framework that handles server-side logic.
  - **SQLite Database –** A simple yet effective database to store user and booking data.
  - **Flask-JWT-Extended –** Manages secure authentication and user sessions**.**

- ♦ **Flask-CORS –** Allows seamless interaction between the frontend and backend, even when hosted separately.

- **System Requirements: Python 3.8+, Node.js 14+, npm /yarn, Visual Studio Code, Git .**

II. System Architecture:

Our Smart Parking System follows a structured architecture where different components communicate efficiently to provide a seamless experience.

A [Client/Frontend] --> B [Flask Server/Backend]

B --> C [SQLite Database]

A --> D [Authentication Service]

D --> B

Details:

1. **Client/Frontend (A → B)**
   - ○ The user interacts with the React.js frontend, which sends requests (like booking a slot or logging in) to the Flask backend.
2. **Flask Server/Backend (B → C)**

  - The Flask server processes user requests and communicates with the SQLite database to store and retrieve data, such as user details and parking slot information.

3. **Authentication Service (A → D → B)**

  - When a user logs in, the JWT authentication service verifies their credentials before allowing access to the system.
  - Once verified, the backend grants access to features like booking slots and viewing past reservations.

III. Database Design

Users
Username
Email
User_id
Password
Contact

Parking
Slot_id
Level
User_id
Slot_No
Status

Booking
Book_id
User_id
Slot_id

Feature and Functionalities

1. User Management Features:

User registration with unique credential, Secure login/authentication system, Profile management with email and phone verification and Session management using JWT tokens.

2. Parking Management Features:

Multi-level parking structure (3 levels), 100 parking slots per level, Real-time slot availability tracking, Interactive parking map visualization and Color-coded slot status indication (White: Available, Red: Occupied).

3. Booking System Features:

Real-time slot booking, Slot selection from interactive map, Booking confirmation system, Multiple active bookings tracking, Booking history maintenance and Auto-calculation of parking charges.

4. Security Features:

Password encryption, JWT token-based authentication, Session timeout management, Input validation and sanitization and Secure API endpoints.

5. Database Management:

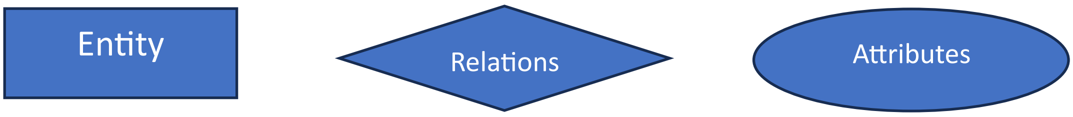
User data persistence, Booking records maintenance and Parking slot status tracking.

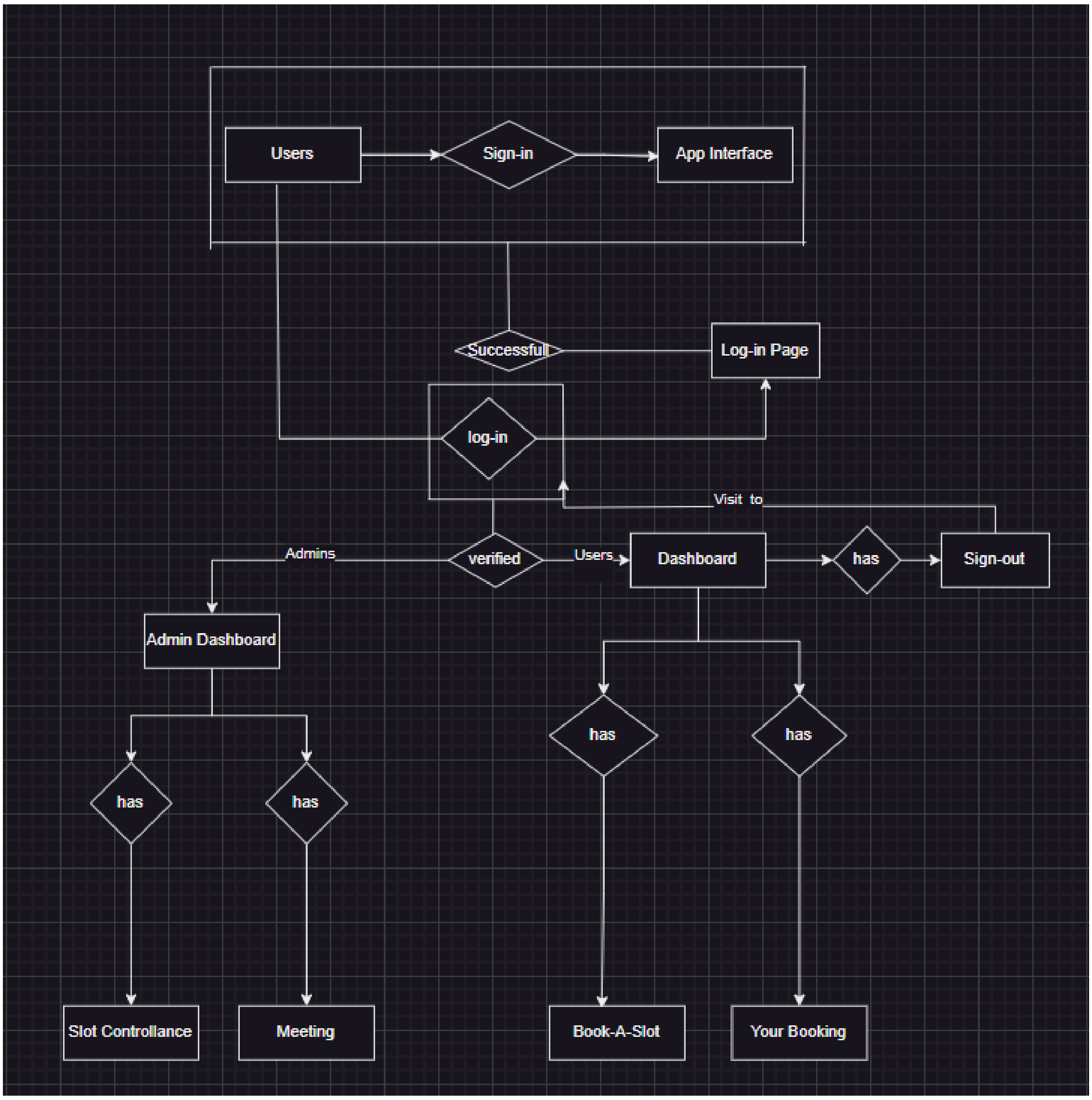
6. Technical Features:

Real-time data synchronization, Cross-platform compatibility, Error handling and logging and Database backup and recovery.

ER Diagram:

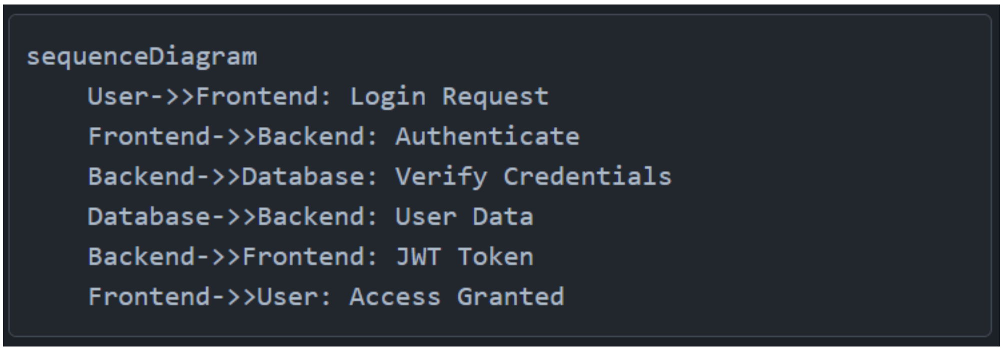
Representation of different elements:





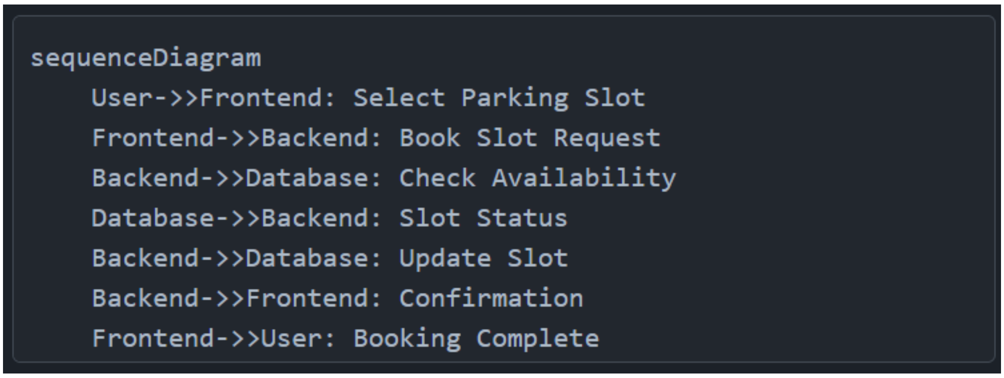
User Flow:

A. User Authentication Flow:



```
sequenceDiagram
    User->>Frontend: Login Request
    Frontend->>Backend: Authenticate
    Backend->>Database: Verify Credentials
    Database->>Backend: User Data
    Backend->>Frontend: JWT Token
    Frontend->>User: Access Granted
```

B. Booking Flow:



```
sequenceDiagram
    User->>Frontend: Select Parking Slot
    Frontend->>Backend: Book Slot Request
    Backend->>Database: Check Availability
    Database->>Backend: Slot Status
    Backend->>Database: Update Slot
    Backend->>Frontend: Confirmation
    Frontend->>User: Booking Complete
```

Competitive Benchmarking:

1. Compared to traditional parking systems, our **Smart Parking System** offers:
    - **Real-time slot availability** to minimize manual errors.
    - **User-friendly dashboard** for easy slot booking and management.
    - **Secure authentication** to ensure user privacy.
    - **Digital slot allocation** to optimize space utilization.
2. Competes with existing solutions by offering a **cost-effective**, **efficient**, and **scalable** parking management system.

Future Scope:

Our Smart Parking System has the potential to grow and improve significantly, making parking management even more convenient, automated, and user-friendly. By following ways:

1. Smart IoT Integration:

Sensor-based occupancy detection, Real-time monitoring systems, Automated barrier control and Environmental monitoring.

2. AI-Powered Slot Recommendations:

Predictive parking availability, Smart vehicle identification, Automated license plate recognition and Optimal slot recommendation.

3. Dedicated Mobile App:

Native Android/iOS apps, Real-time notifications, QR code-based access and Voice-guided navigation.

4. Digital Payment Integration:

Online payment gateway, Digital wallet integration, Subscription-based parking, Dynamic pricing model and Multi-Location Parking Support.

6. Enhanced Security Features:

Biometric authentication, Two-factor authentication, OAuth integration, SSO capabilities, Eco-Friendly Initiatives, CCTV integration, Security incident tracking and Automated alert system.

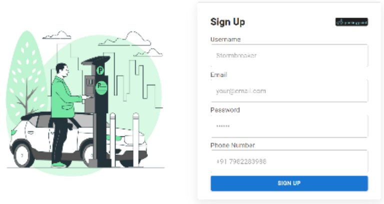
8. Data-Driven Insights & Reports:

Navigation apps integration, Weather service integration, Traffic management systems and Emergency services alerts.

Conclusion:

It is a simple and efficient way to manage parking spaces. It helps users find, book, and track parking slots easily through a digital platform. With features like real-time slot tracking, secure login, and a user-friendly interface, it makes parking hassle-free. This system saves time, reduces confusion, and ensures better space utilization. As it grows, it can be improved with more features like automated payments, smart sensors, and AI-based predictions, making parking even more convenient and efficient.

INTERFACE OF OUR WEBSITE:



Sign Up

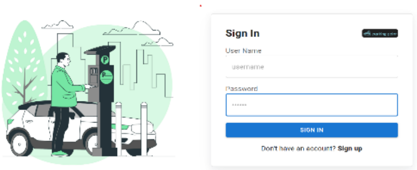
Username: Stormbreaker

Email: your@email.com

Password: ••••••

Phone Number: +91 7982283988

Signup

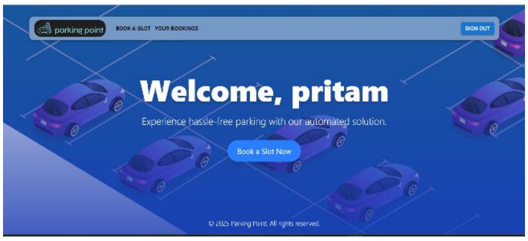


Sign In

User Name: username

Password: ••••••

Sign in

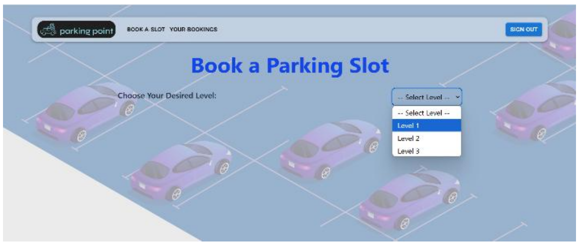


Welcome, pritam

Experience hassle-free parking with our automated solution.

© 2025 Parking Point. All rights reserved.

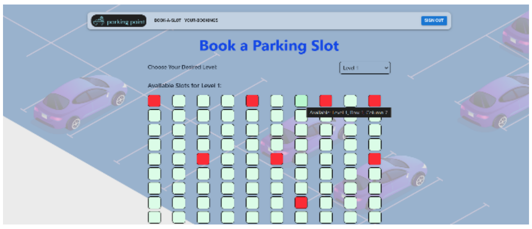
Dashboard



Book a Parking Slot

Choose Your Desired Level: -- Select Level --

Book Page



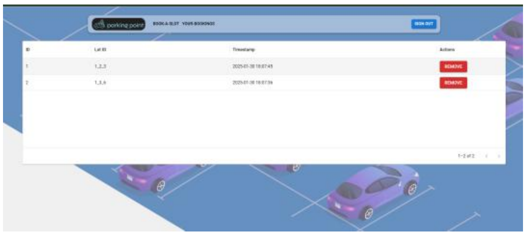
Book a Parking Slot

Choose Your Desired Level: Level 1

Available Slots for Level 1:

Available: Level 1, Row 1, Column 7

Slot View



ID	Lot ID	Timestamp	Actions
1	1_2_3	2025-01-30 18:07:45	REMOVE
2	1_3_6	2025-01-30 18:07:36	REMOVE

1–2 of 2

Your Bookings