SMART PARK

Software Requirements Specification (SRS)

Team Members
R.Prabhakara Arjun (2022503003)
V. Guru Aravindan (2022503001)

Introduction

Urban cities are growing at a phenomenal pace resulting in high population density and vehicle density which constantly requires the deployment of proper parking to relieve traffic load, better use of available space. This is because in most cases one has to wait to get himself parked. Many times cars create unnecessary waiting and blockage and the driver and the parking lot owner both get wars of frustration. Therefore we have developed **Smart Park**, which is a new *Parking System Management* which address the problems of inefficient occupation of parking areas.

Product Description

The **Smart Park** is a complete solution for the management of Parking Systems which includes vehicle entry and exit system, maximal utility of parking spaces, and tracking the occupancy status of all the parking spaces. The system enhances user experience as well as administration by presenting functions such as reservations, payments, and reservations daily reports.

Intended Audience

Smart Park is designed for drivers seeking efficient parking solutions and for parking lot operators aiming to streamline operations through technology.

Functional requirements:

1.Vehicle Entry and Exit Management*:*

The system will keep records of vehicle entry and exit and times and will produce and dispense tickets for vehicles based on the time it was parked.

2.Parking Space Allocation

Allocation of Parking Space The allocation of the parking lots will be automatic according to the vehicle space in order to make more effective use of the available areas. .

3.Reservation System

Reservation System Users are able to book spaces through a mobile and web interface in advance and the system does not permit the use of the reserved spaces and they are blocked till used or canceled.

4.Real-Time Availability Tracking

The system will update the users on the availability of parking space in a specification of how long the updates will be displayed on signage and how soon other aspects of the web and mobile application can be accessed.

5.Multi-Level Facility Support

Smart Park will guide vehicles to appropriate levels and sections, distributing the parking load evenly across all levels.*(if possible)*

6.Payment and Billing

The system will be flexible in that it will accept cash, cards as well as mobile payments while also producing and issuing invoices for cost incurred on an hourly or daily basis.

7.Report Generation

Admins can generate reports on usage, revenue, and space utilization, with the option to export reports in PDF or Excel formats.

Non Functional requirements:

1.Scalability:- *(Volume Handling):*** The system should be able to accommodate a large number of vehicles and additional facilities.

2.Reliability:- *(Uptime):* to ensure minimal downtime to handle failures.

3.Performance:- *(Response Time):* Quick responses for entry/exit actions and real-time updates, even during peak times.

4.Security:**-** *(Access Control):* Prevent unauthorized access to the management interface.

5.Usability:-(*User Interface)***:** Provide an intuitive and user-friendly interface for mobile/web and on-site digital boards, useable by a general public and layman

6 .Maintainability*:(Maintenance):* The system should be easy to maintain with clear documentation and configurable settings.

Tools and languages:

Front End:for front end will be developing using languages like **Java with JavaFX**,.*As a backup option*, **HTML, CSS, and JavaScript** will be considered if additional web-based functionality is required.

Back End: The back end will be supported by ***Java Spring Boot***.

Database:*SQL* will be the primary choice for database management, ***MongoDB*** will be considered as an alternative if a NoSQL solution is needed for handling more complex data structures

Assumptions :

It is assumed that Smart Park will be highly beneficial for parking lot operators and drivers, gaining widespread adoption over time.

Use Case Diagram: