"SMART PARKING MANAGEMENT SYSTEM"

A Project Report of Major Project Phase-II Submitted to

Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal

Towards Partial Fulfillment of the Degree of

Bachelor of Technology in

Computer Science and Engineering

Guided by:

Ms. Neha Mehra
Assistant Professor
Dept. of Computer Engg.

Ms. Priyanka Kokate
Assistant Professor
Dept. of Computer Engg.

Submitted by:

Abhishek Patidar (0801CS201007)
Anurag Choudhary (0801CS201017)
Pankaj Nagava (0801CS201065)
Vivek Dawar (0801EC201104)

Department of Computer Engineering

Shri G.S. Institute of Technology and Science,

Indore (M.P.)

2023-2024

SHRI G.S. INSTITUTE OF TECHNOLOGY AND SCIENCE

INDORE (M.P.)

SESSION: 2023-24

RECOMMENDATION

The project report of Major Project Phase-II entitled **"Smart Parking Management System"** submitted by: **0801CS201007 - Abhishek Patidar, 0801CS201017 - Anurag Choudhary, 0801CS201065 - Pankaj Nagava, 0801EC201104 - Vivek Dawar**, students of Bachelor of Technology IV year in the session 2023-24, towards fulfillment of the degree of **B. Tech. in Computer Science and Engineering** of Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal is a satisfactory account of their work towards CO44998: Major Project (Phase II) and is recommended for the award of degree.

Ms. Neha Mehra
Assistant Professor
Project Guide
Dept. of Computer Engg.

Dr. Vandan Tewari
Head of Department
Dept. of Computer Engg.

Ms. Priyanka Kokate
Assistant Professor
Project Co-Guide
Dept. of Computer Engg.

Dean (Academics)
S.G.S.I.T.S. Indore

SHRI G.S. INSTITUTE OF TECHNOLOGY AND SCIENCE

INDORE (M.P.)

(A Govt. Aided Autonomous Institute, Affiliated to RGPV, Bhopal)

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

The Project Report for CO44998: Major Project (Phase II) entitled **"Smart Parking Management System"** submitted by **Abhishek Patidar (0801CS201007), Anurag Choudhary (0801CS201017), Pankaj Nagava (0801CS201065), and Vivek Dawar (0801EC201104)**, students of B.Tech. IV year in the session 2023-24, towards partial fulfillment of the degree of **Bachelor of Technology in Computer Science and Engineering** of Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal is a satisfactory account of their work.

Internal Examiner
Date:

External Examiner
Date:

SHRI G.S. INSTITUTE OF TECHNOLOGY AND SCIENCE

INDORE (M.P.)

DECLARATION

We, **Abhishek Patidar(0801CS201007), Anurag Choudhary(0801CS201017), Pankaj Nagava(0801CS201065), and Vivek Dawar(0801EC201104)**, declare that the Major Project (Phase-II) work on **"Smart Parking Management System"** is our own work conducted under the supervision of **Ms. Neha Mehra**, Assistant Professor, Computer Engg. and **Ms. Priyanka Kokate**, Assistant Professor, Computer Engg. We further declare that to the best of our knowledge, this project work does not contain any part of any work submitted for the award of any degree or any other work in this University or any other University/website without proper citation and is plagiarism-free.

0801CS201007 - Abhishek Patidar

0801CS201017 - Anurag Choudhary

0801CS201065 - Pankaj Nagava

0801EC201104 - Vivek Dawar

ACKNOWLEDGEMENT

With great pleasure and a sense of obligation, we express our heartfelt gratitude to our esteemed guides **Ms. Neha Mehra**, Assistant Professor, Department of Computer Engineering and **Ms. Priyanka Kokate**, Assistant Professor, Department of Computer Engineering, whose constant encouragement enabled us to work enthusiastically. Our project guides, despite their heavy work commitments and busy schedules, have taken the time to bestow upon us their invaluable guidance and support.

We are grateful to **Dr. Vandan Tewari**, Head, Department of Computer Engineering and **Dr. Rakesh Saxena**, Director, S.G.S.I.T.S. Indore, for the facilities provided in the completion of this project.

We sincerely wish to express our gratitude to all the members of the staff of the Computer Engineering Department who have extended their cooperation at all times and contributed to developing the project.

0801CS201007 - Abhishek Patidar
0801CS201017 - Anurag Choudhary
0801CS201065 - Pankaj Nagava
0801EC201104 - Vivek Dawar

Place: Indore
Date: .....

ABSTRACT

Parking in bustling city environments presents significant challenges, contributing to traffic congestion, larger waiting times, and fuel consumption[1]. Studies indicate that 30 percent of cars in traffic spend an average of eight minutes searching for available spaces, resulting in fuel wastage, monetary expenses, and time loss.[2]

Existing solutions to address parking inefficiencies include traditional parking systems, manual reservation processes, and Singapore's parking.sg[3]. However, these methods often lack real-time information on parking availability and fail to streamline the parking process effectively.

To mitigate these challenges, we propose the implementation of a "Smart Parking Management System." Leveraging IoT technologies and OTP-based entry for heightened security, our system allows users to book parking spaces in advance through a user-friendly mobile application. A cloud-based application facilitates efficient real-time management of parking spaces and maintains a secure database of registered drivers, enabling online payment transactions. The primary objective of the Smart Parking Management System is to streamline the often time-consuming process of finding an open parking slot in crowded lots. By offering real-time information on parking availability and enabling users to reserve parking slots through a dedicated application, our system enhances user convenience and contributes to more effective land and traffic management in urban areas.

The proposed Smart Parking Management System will utilize PHP and MYSQL for the development of both the IoT infrastructure and the Web application, ensuring seamless integration and efficient performance.

Table of Contents

Recommendation ii

Certificate iii

Acknowledgement v

Abstract vi

List of Figures x

1 INTRODUCTION 1

1.1 Preamble . . . 2

1.2 Need of the Project . . . 3

1.3 Problem Statement . . . 4

1.4 Project Objective . . . 4

1.5 Proposed Approach . . . 4

1.6 Organization of the Report . . . 4

2 BACKGROUND STUDY 6

2.1 Area of Concern . . . 7

2.2 Existing System . . . 7

2.3 Tools and Technologies . . . 8

3 Literature Review 10

3.1 Inception . . . 11

3.2 Tools Required for Implementation . . . 11

3.3 A Study of Available Approaches . . . 12

3.3.1 Traditional Approach[6] . . . 12

3.3.2 IoT (Internet of Things) Based Approach . . . 12

3.4 Analysis of Drawbacks & Improvements . . . 13

3.4.1 Traditional Parking Management Systems[6]: . . . 13

3.4.2 IoT-based Parking Management Systems without Real-time Updates: 13

3.5 Summary . . . 14

4 ANALYSIS **15**

4.1 Problem Statement: Smart Parking Management System 16

4.1.1 Identified Problems in Current Parking Management Systems . . 16

4.1.2 Proposed Solution: Smart Parking Management System 16

4.2 Functional Requirements 17

4.2.1 User Authentication and Access Control: 17

4.2.2 Real-time Parking Spot Updates: 17

4.2.3 Slot Booking Application: 18

4.2.4 Payment Integration: 18

4.2.5 Centralized Cloud-Based System: 18

4.2.6 Resource Optimization: 18

4.2.7 User Notifications: 18

4.2.8 Security Measures: 18

4.3 Non-Functional Requirements 19

4.4 System Requirements 19

4.5 Use-case descriptions 19

4.5.1 Search Parking 19

4.5.2 Request Slot 20

4.5.3 Contact Attendant 21

4.5.4 View History 21

4.5.5 Approve or Reject Requests 22

4.5.6 Print Receipt 22

4.5.7 Send Notification 23

4.5.8 Vehicle Authentication 23

4.5.9 Add New Parking 24

4.5.10 Add New Attendant 24

4.5.11 Feasibility Study 25

5 DESIGN **26**

5.1 System Architecture 27

5.2 Data Flow Diagram 27

5.2.1 Data Flow Diagram (Level 0) 27

5.3 Activity Diagram 29

5.3.1 Online Booking 29
5.4 Algorithm 30

6 Implementation **32**

6.1 Code Restructuring 33
6.1.1 Enhancement of Existing Features 33
6.1.2 Implementation of New Features 33
6.1.3 Optimization of System Performance 33
6.1.4 Improvement of Data Quality and Accuracy 34
6.1.5 Reduction of System Downtime 34
6.1.6 Minimizing HTTP Requests 34
6.2 Different functionality implementations 34

7 TESTING AND RESULT **42**

7.1 Test Cases 43

8 Conclusion **50**

8.1 Features 51
8.2 Benefits of the project 51
8.3 Limitations of Proposed Approach 52
8.3.1 Future Enhancements 52

9 References **53**

LIST OF FIGURES

Fig. No.	Title	Page No.
5.1	System Architecture	27
5.2	Top level data flow diagram of the entire system (Level 0)	28
5.3	Detailed Data flow diagram of the system with sub-processes (DFD Level 1)	28
5.4	Online Booking Activity Diagram	29
6.1	User Login	34
6.2	User Interface	35
6.3	Search Along Street or Area	35
6.4	View Location	36
6.5	Slot Information	36
6.6	Request Submitted	37
6.7	Already Requested Slot	37
6.8	Booking History	37
6.9	Contact Page	38
6.10	Admin or Attendant Login	38
6.11	Admin Interface	38
6.12	Upload New Parking	39
6.13	Add New Parking Attendant	39
6.14	View, Search, or Download Requests	39
6.15	Rejected Request	40
6.16	Approve or Reject Request by Attendant	40
6.17	Approved Request and Code with Notification Sent	40
6.18	Email Notification	41
6.19	*IoT Model*	41

Chapter 1

INTRODUCTION

This chapter presents a brief introduction to the problem, thereby building the context needed to understand the need for the proposed system. It also states the problem statement, expatiates the objectives, the proposed approach, and the organization of the report.

1.1 Preamble

In today's fast-paced working environment, people (motorists) greatly depend on automobiles to commute to their destinations. Automobiles include motor vehicles, motorbikes, and trucks to mention but a few. The use of these automobiles has increasingly posed a demand for infrastructure to manage the parking.

The Smart Car Parking System is an innovative solution that leverages advanced technologies to address the challenges and inefficiencies of traditional parking systems in urban areas. With the increasing number of vehicles on the roads and limited parking space availability, finding convenient parking spots has become a time-consuming and frustrating experience for drivers. The Smart Car Parking System aims to revolutionize the parking process by integrating IoT (Internet of Things) devices and Android applications to provide real-time parking spot availability, seamless reservation and payment processes, and efficient space management.

The system utilizes IoT devices, such as sensors and actuators, to monitor parking spaces and collect data on occupancy status. This data is then transmitted to a cloud infrastructure for real-time analysis and processing. Through a user-friendly Android application, drivers can access up-to-date information on available parking spots, reserve a space in advance, and complete the payment process conveniently.

The Smart Car Parking System not only improves the parking experience for drivers but also offers numerous benefits to parking operators and urban infrastructure managers. It enables efficient space management, optimizing parking utilization and reducing congestion. By providing real-time data on parking spot availability, it helps drivers save time, reduce fuel consumption, and contribute to a greener environment. Additionally, the integration of secure payment gateways ensures convenient and secure transactions, eliminating the need for manual ticketing or cash transactions.

The Smart Car Parking System represents a significant advancement in parking management, utilizing IoT and Android technologies to create a smarter, more efficient, and user-friendly parking solution. By leveraging real-time data, seamless reservation and

payment processes, and navigation assistance, the system aims to enhance the overall parking experience, optimize space utilization, and contribute to the development of smart and sustainable urban environments.

1.2 Need of the Project

The increasing number of vehicles on the roads has led to a significant challenge in efficiently managing parking spaces. Traditional parking systems often result in wasted time and frustration for drivers who struggle to find available parking spots. There is a need for a smart car parking system that utilizes advanced technologies to streamline the parking process and provide real-time information to drivers.

The Need for the Smart Car Parking System can be summarized as follows:

(1) Inadequate Parking Space Management: Existing parking systems often lack efficient management techniques to handle increasing vehicle populations, resulting in parking congestion and inefficient space utilization.

(2) Time-consuming Parking Spot Search: Drivers waste valuable time searching for available parking spaces, leading to increased traffic congestion, frustration, and fuel consumption.

(3) Lack of Real-time Parking Information: Drivers lack access to real-time information about parking spot availability, making it difficult to plan their parking in advance and leading to unnecessary trips around parking areas.

(4) Manual Reservation and Payment Process: Traditional parking systems often require manual reservation and payment processes, involving cash transactions and limited payment options, which can be inconvenient and less secure for users.

(5) Manual and Inefficient Parking Space Monitoring: The absence of a reliable monitoring system for parking spaces makes it difficult for parking operators to manage and optimize the utilization of available parking spots efficiently.

(6) The Smart Car Parking System aims to address these challenges by leveraging IoT, Android, and other technologies to provide drivers with real-time parking spot availability, seamless reservation and payment processes, and efficient space management. Doing so enhances the overall parking experience, reduces congestion, and maximizes the utilization of parking spaces.

1.3 Problem Statement

Inefficient traditional parking systems lead to congestion, larger waiting times, and a poor user experience, because of which a platform is required to manage parking space efficiently. Smart parking systems using IoT technology with real-time slot booking applications can optimize parking operations, reduce waiting time, and enhance the user experience.

1.4 Project Objective

The objective of the Smart Car Parking System is to provide efficient and convenient parking solutions by leveraging advanced technologies. The Smart Car Parking System aims to improve parking efficiency, reduce congestion, enhance user satisfaction, and promote sustainable urban mobility.

1.5 Proposed Approach

Smart parking systems use IoT technology to collect real-time data on parking occupancy. These systems can automate parking management using sensors and other technology. Smart parking systems can improve the user experience by providing real-time parking information, reducing wait times, and simplifying payment processes.

1.6 Organization of the Report

- **Chapter 1:** A brief introduction about the project giving the need for the project, the problem is going to solve, objectives, and also the approach to be used.
- **Chapter 2**: This chapter describes the background study done to study the concepts of the existing system.
- **Chapter 3**: This chapter describes the literature survey done to analyze the description of studies done in research papers.
- **Chapter 4**: This Chapter describes existing systems and the detailed problem statement of the project work. It deals with a detailed analysis of the project with the functional requirements and non-functional requirements.
- **Chapter 5:** This chapter focuses on design details consisting of the basic architecture of the system and the flow of activities of each sub-problem and activity diagram.

- **Chapter 6:** This chapter revolves around the implementation details providing the full logical view of the modules.
- **Chapter 7:** This chapter contains the testing which shows the tests conducted while optimizing the infrastructure and algorithm and also the results obtained.
- **Chapter 8:** This chapter contains the conclusion of the project, limitations, and scope for future work.
- **Chapter 9:** This chapter contains the references used in the project.

Chapter 2

BACKGROUND STUDY

This chapter provides the background on information systems, existing solutions and enlists the chosen tools and technologies.

2.1 Area of Concern

The common conception about parking is that it is difficult to find an empty parking slot in urban areas, and this often leads to drivers circling around the parking lot, causing traffic congestion and wasting fuel. Traditionally, parking management has relied on manual monitoring and ticketing systems, which are time-consuming, inefficient, and prone to errors. The current system does not provide real-time information on parking availability, which leads to frustration and dissatisfaction among drivers.

For businesses and commercial establishments, providing parking for their customers is a major concern. The cost of building and maintaining a parking lot is high, and the lack of effective parking management systems often leads to underutilization of available spaces. The available parking management systems are expensive and still fail to address the problem of inefficient utilization of parking spaces. Sometimes, there is a need to hire parking attendants to manage the parking lot, which is costly and not always effective.

2.2 Existing System

In traditional manual parking management, personnel such as parking attendants or security guards are assigned to observe and regulate parking spaces. This manual approach relies on human observation and communication rather than technological solutions. While it may initially be cost-effective, it often becomes impractical for larger parking facilities or areas with high traffic due to scalability issues.

A sensor-based approach involves installing IoT sensors in each parking spot to detect occupancy. However, without real-time updates, the system may not provide accurate or up-to-date information on parking space availability. This limitation hinders the system from fully leveraging the potential of IoT technology to provide real-time insights and optimize the efficiency of parking management.

In the case of Singapore's Parking.sg application[3], it introduces a user-friendly method for citizens to enter parking spaces by scanning their license plates. This solution, while innovative, may involve certain costs related to implementation and maintenance. The success of this application also relies on widespread adoption, and its scalability may be affected by factors such as cost considerations and user adoption rates.

2.3 Tools and Technologies

The technology stack used in this project is a combination of programming languages, frameworks, and tools that helped us build this web app and IoT device.

HARDWARE TOOLS

In the implementation of our smart parking system, we have opted for an authentication approach using keypad and servo motor instead of relying on IoT (Internet of Things) sensors. This strategy involves deploying keypads at entry gates and utilizing servo motors as barriers to grant or restrict access based on user authentication.

Keypads at the entry gate serve as the means for users to input a secure code, replacing the traditional role of IoT sensors in detecting the availability of parking spaces. Once the user enters a valid code, the servo motor acts as a barrier, allowing authorized users to proceed and restricting unauthorized access.

While the conventional IoT sensors, such as infrared, ultrasonic, magnetic, and video cameras, are renowned for capturing real-time data on parking space occupancy, our approach emphasizes the use of manual input through keypads for user authentication. The servo motor, in turn, efficiently regulates access based on the authentication status.

This authentication-centric system ensures secure access control and eliminates the need for continuous monitoring of parking spaces. The absence of IoT sensors shifts the focus to the keypad and servo motor combination, providing a reliable and user-friendly means of managing access to parking facilities. The system still maintains its effectiveness in optimizing parking utilization, reducing congestion, and enhancing overall parking management in urban areas.

SOFTWARE TOOLS

- **Fronted Tools:**

  - **Visual Studio** Visual Studio is a code editor used for software development. It helps in creating websites, web apps, and web services. The software contains completion tools, compilers, and other features to facilitate the software development process.

  - **Arduino IDE:** The Arduino IDE serves as the primary development environment for Arduino-based projects. It provides a user-friendly interface for writing, compiling, and uploading code to Arduino microcontrollers. With the Arduino IDE, developers can seamlessly program and deploy code for a wide range of

Arduino-compatible boards, enabling the creation of diverse embedded systems and IoT applications. The IDE supports various libraries and facilitates the integration of sensors, actuators, and other components into Arduino projects.

- **Backend Tools:**

- **phpMyAdmin:** phpMyAdmin is a web-based administration tool for managing MySQL and MariaDB databases. It provides a graphical interface to handle database operations, such as creating, modifying, and deleting databases, tables, and fields. With phpMyAdmin, users can execute SQL queries, import and export data, and manage user privileges. This user-friendly tool simplifies database administration tasks, offering a convenient way to interact with MySQL databases through a web browser. Its features include a visual query builder, server monitoring, and the ability to perform common database tasks efficiently.

- **MySQL Workbench** MySQL Workbench is a visual database design tool. It provides an integrated environment for development, administration, database design, creation and maintenance for MySQL database systems.

Chapter 3

Literature Review

In the Literature Review section, we look at what other experts and researchers have already studied about our topic. It helps us understand what's already known, what questions remain unanswered, and how our project fits into the bigger picture.

3.1 Inception

A review of completed and ongoing research has been conducted to identify current knowledge or methodologies in the field of face detection and recognition. Several efforts were made in the past to implement such systems, which have motivated and guided us throughout the process. After undertaking the project, there were four big questions to answer. This review will answer these questions sequentially.

What tools should be used to implement the project?

What approach should be taken for authentication of cars?

What are the drawbacks of the earlier systems?

What are the improvements?

3.2 Tools Required for Implementation

Keypad and Servo Motor:

- The keypad will be installed at the entry gate for users to input authentication codes.

- The servo motor will act as a barrier, allowing or restricting access based on the entered codes

Microcontrollers:

- Microcontrollers will be employed to manage the authentication process, connecting the keypad and servo motor to the central system.

Central System:

- This cloud-based system will play a central role in processing authentication data, managing parking slots, and ensuring a seamless user experience.

Slot Booking Web Application:

- Developed using PHP and MySQL, the web application will enable users to view available parking slots, book in advance, and receive unique codes via email.

Email Notification Service:

- An email notification service will be integrated to send users their unique codes and provide timely reminders.

3.3 A Study of Available Approaches

3.3.1 Traditional Approach[6]

The traditional approach to parking management typically involves a manual system where parking attendants physically monitor parking lots and collect fees from users. Here are some key features of this approach:

1. **Manual monitoring:** Parking attendants manually monitor the parking lot to ensure users park in designated spots and follow parking rules. They may also collect parking fees.

2. **Cash-based transactions:** Payment is usually made in cash, either at entry or exit. Physical tickets or a manual system track the parking duration and calculate the fee.

3. **Limited information:** Due to manual processes, users have limited information about parking spot availability and the status of booked slots. Physical checks may be necessary.

4. **Limited automation:** Automation is minimal, with no use of IoT sensors or cloud-based systems. This can lead to inefficiencies and errors.

3.3.2 IoT (Internet of Things) Based Approach

An IoT-based parking management system without real-time updates involves installing IoT sensors in each parking spot to detect occupancy. The information is stored in a central database, and drivers can access this database through a display at the parking lot entrance to check parking spot availability.

However, since the system does not provide real-time updates, the availability of parking spots may not be accurate or up-to-date. This may lead to inefficiencies and difficulties for drivers in finding available parking spots. While an improvement over traditional systems, this approach does not fully leverage the potential of IoT technology for real-time information and improved efficiency.

3.4 Analysis of Drawbacks & Improvements

The drawbacks of traditional parking management systems and IoT-based parking management systems without real-time updates are as follows:

3.4.1 Traditional Parking Management Systems[6]:

- Manual monitoring of parking lots can lead to inefficiencies and errors, as attendants may not accurately track occupancy or fees.
- Cash-based transactions can be inconvenient for drivers, who may not have exact change or may have to wait in line to pay.
- Limited information about available parking spots can lead to difficulties for drivers in finding a parking spot, especially during peak hours.

3.4.2 IoT-based Parking Management Systems without Real-time Updates:

- The availability of parking spots may not be accurate or up-to-date, which can lead to difficulties for drivers in finding an available spot.[3]
- The system may not fully leverage the potential of IoT technology to provide real-time information and improve the efficiency of the parking management system.

In both cases, the lack of real-time information can lead to inefficiencies and difficulties for drivers in finding available parking spots, as well as limitations in the ability of parking lot operators to manage their resources and optimize pricing.

An IoT-based parking management system with slot booking and real-time updates can provide several improvements over traditional and IoT-based systems without real-time updates, including:

- Improved accuracy and real-time availability information: With IoT sensors providing real-time occupancy information, drivers can view the availability of parking spots in real-time and book a slot accordingly, which can help reduce the time spent searching for parking spots.
- Increased efficiency and reduced waiting time: By enabling users to book parking spots in advance and reserving them until the user arrives, this system can reduce the wait times for users looking for parking and optimize the utilization of parking spots.

- Enhanced payment options: With the option to pay electronically through the mobile app, the system can reduce cash transactions and provide greater convenience to drivers.
- Improved resource management: By analyzing real-time occupancy data, parking lot operators can optimize pricing and better manage their resources, ensuring that all available parking spots are being utilized efficiently.
- Enhanced user experience: By providing real-time updates and enabling users to book parking spots in advance, the system can provide a more user-friendly and convenient experience for drivers, reducing the stress and frustration associated with finding parking spots.

Hence, an IoT-based parking management system with slot booking and real-time updates can provide numerous benefits for drivers and parking lot operators, improving the efficiency, accuracy, and convenience of parking management.

3.5 Summary

The traditional approach to parking management is inefficient and error-prone, with attendants manually monitoring parking lots and cash transactions causing inconvenience for drivers. IoT-based approaches without real-time updates can also be problematic, as the availability of parking spots may not be accurate or up-to-date. However, an IoT-based system with slot booking and real-time updates can provide several improvements, including increased efficiency and reduced wait times, enhanced payment options, improved resource management, and an enhanced user experience.[5]

Chapter 4

ANALYSIS

This chapter includes the findings and details of the analysis phase. It presents a detailed problem statement and further provides the requirement analysis for the system.

4.1 Problem Statement: Smart Parking Management System

In urban areas, the increasing number of vehicles has led to a critical problem of finding suitable parking spaces efficiently. The traditional methods of parking management, often manual and reliant on attendants, are proving to be inefficient and inconvenient for both drivers and parking lot operators. Additionally, existing IoT-based approaches lack real-time updates, impacting the accuracy of parking spot availability information.

4.1.1 Identified Problems in Current Parking Management Systems

Inefficiency and Errors in Traditional Systems:

- **Manual Monitoring:** The manual monitoring of parking lots by attendants is prone to inefficiencies and errors, leading to challenges in accurately tracking occupancy and managing fees.
- **Cash-based Transactions:** Cash transactions at parking lots can be inconvenient for drivers, causing delays and requiring exact change.

Shortcomings of Existing IoT-Based Systems:

- **Real-time Updates Missing:** Current IoT-based systems lack real-time updates on parking spot availability, making it challenging for drivers to find accurate and up-to-date information.

4.1.2 Proposed Solution: Smart Parking Management System

To address these challenges, our project aims to implement a comprehensive Smart Parking Management System that leverages modern technologies, including IoT devices, a cloud-based central system, and a user-friendly slot booking application. The proposed system seeks to overcome the limitations of traditional and existing IoT-based approaches by introducing the following key features:

- **Authentication and Access Control:** Utilizing a keypad and servo motor, the system ensures secure authentication of users, allowing only authorized vehicles to access the parking area.

- **Real-time Updates and Slot Booking:** Employing IoT sensors, the system provides real-time updates on parking spot availability. A user-friendly slot booking application enables drivers to reserve parking spaces in advance.
- **Efficient Payment Systems:** Integrating electronic payment options through a mobile application enhances user convenience, reduces cash transactions, and streamlines the payment process.
- **Enhanced Resource Management:** The cloud-based central system analyzes real-time occupancy data to optimize pricing and efficiently manage parking resources, ensuring maximum utilization.
- **Improved User Experience:** By offering real-time updates, convenient slot booking, and electronic payment options, the system aims to enhance the overall user experience, reducing the stress associated with finding parking spots.

The implementation of this Smart Parking Management System intends to revolutionize parking facilities in urban areas, providing a seamless, efficient, and user-friendly solution for both drivers and parking lot operators.

4.2 Functional Requirements

4.2.1 User Authentication and Access Control:

- The system should provide a secure user authentication mechanism at the entry gate using a keypad.
- Only authorized users with a valid authentication code should be granted access through the servo motor-controlled gate.

4.2.2 Real-time Parking Spot Updates:

- The system should employ IoT sensors to provide real-time updates on the availability of parking spots.
- Users should be able to view the real-time status of parking spots through the slot booking application.

4.2.3 Slot Booking Application:

- The system should feature a user-friendly slot booking application.
- Users should be able to view available parking spots, book a slot in advance, and receive confirmation.

4.2.4 Payment Integration:

- The slot booking application should integrate electronic payment options.
- Users should be able to pay for their booked slots securely through the mobile application.

4.2.5 Centralized Cloud-Based System:

- A cloud-based central system should manage and process authentication data, parking slot availability, and user bookings.
- The system should ensure efficient communication between IoT devices, the keypad, and the slot booking application.

4.2.6 Resource Optimization:

- The central system should analyze real-time occupancy data to optimize pricing and efficiently manage parking resources.
- Parking lot operators should have tools to adjust pricing based on demand and availability.

4.2.7 User Notifications:

- The system should send notifications to users, including booking confirmations, reminders, and payment receipts.
- Notifications should be delivered through the slot booking application or email.

4.2.8 Security Measures:

- The system should implement encryption and secure communication protocols to protect user data.

- Regular security audits and updates should be conducted to identify and address vulnerabilities.

4.3 Non-Functional Requirements

These requirements are the several answers to the question "How is the system supposed to do". The Non-Functional requirements of the proposed system are as follows:

- Performance: Offline availability of the prefetched data increases the application's performance. The application should not take more than 3 seconds to load the initial screen. Should take less response time to complete desired tasks
- Usability: Making the user interface more interactive by proper placing of buttons, icons, etc increases usability. Using icons related to the information makes it easy for the user to see the information quickly. Users should be able to use the app without any previous knowledge of interacting with such applications.
- Availability: Commonplace from where the user has access to install and update the application. The contact of either the developer or the handling team should be given in the application.

4.4 System Requirements

- Operating Systems - Windows 7 SP1 or later (64-bit), x86-64 based
- Any mobile phone with internet connectivity
- UI development - Visual Studio Code, Arduino IDE

4.5 Use-case descriptions

4.5.1 Search Parking

Main Flows:

- The user navigates to the search page of the Smart Car Parking System.
- The system displays a search interface that allows the user to input criteria for parking search.

- The system retrieves and displays relevant parking information based on the user's search criteria.

Alternate Flows:

- If the system encounters an error during the search process, it displays an error message and prompts the user to refine their search parameters.

Preconditions:

- The Smart Car Parking System must have parking data available.

Postconditions:

- The user is presented with a list of available parking matching the search criteria.

4.5.2 Request Slot

Main Flows:

- The user selects a parking slot and requests to book it through the Smart Car Parking System.
- The system prompts the user to confirm the booking request.
- Upon confirmation, the system processes the request and reserves the selected parking slot for the user.

Alternate Flows:

- If the selected parking slot is not available or encounters an issue, the system notifies the user and suggests alternative slots if applicable.

Preconditions:

- The user must be logged into the Smart Car Parking System.

Postconditions:

- The requested parking slot is reserved for the user.

4.5.3 Contact Attendant

Main Flows:

- The user accesses the contact section of the Smart Car Parking System.
- The system provides contact details for the parking lot attendant.
- The user initiates contact with the attendant through the provided information.

Alternate Flows:

- If contact details are not available or there is an issue, the system notifies the user and suggests alternative means of communication.

Preconditions:

- The user must be logged into the Smart Car Parking System.

Postconditions:

- The user successfully contacts the parking lot attendant.

4.5.4 View History

Main Flows:

- The user navigates to the history section of the Smart Car Parking System.
- The system displays a comprehensive history of the user's parking activities, including past bookings, payments, and interactions.

Alternate Flows:

- If there is no history available, the system informs the user that there is currently no history to display.

Preconditions:

- The user must be logged into the Smart Car Parking System.

Postconditions:

- The user is presented with a detailed history of their parking activities.

4.5.5 Approve or Reject Requests

Main Flows:

- The admin accesses the request management page of the Smart Car Parking System.
- The system displays a list of pending requests for parking slots.
- The admin reviews and approves or rejects each request based on availability and criteria.

Alternate Flows:

- If there is an issue with the approval process, the system notifies the admin and suggests alternative actions.

Preconditions:

- The admin must be logged into the Smart Car Parking System.

Postconditions:

- The selected requests are either approved, and parking slots are assigned, or rejected with appropriate notifications.

4.5.6 Print Receipt

Main Flows:

- The user navigates to the payment history section of the Smart Car Parking System.
- The system displays a list of completed transactions.
- The user selects a specific transaction, and the system generates and displays a printable receipt.

Alternate Flows:

- If there is an issue generating the receipt, the system notifies the user and provides alternatives.

Preconditions:

- The user must be logged into the Smart Car Parking System.

Postconditions:

- The user successfully prints a receipt for the selected transaction.

4.5.7 Send Notification

Main Flows:

- The admin accesses the notification management page of the Smart Car Parking System.
- The system provides options to send notifications to users for announcements, updates, or specific events.

Alternate Flows:

- If there is an issue with sending notifications, the system notifies the admin and suggests alternative actions.

Preconditions:

- The admin must be logged into the Smart Car Parking System.

Postconditions:

- Users receive relevant notifications based on the admin's actions.

4.5.8 Vehicle Authentication

Main Flows:

- The user arrives at the parking gate with a booked slot.
- The system prompts the user to authenticate their booking through a keypad.
- Upon successful authentication, the system controls the servo motor to allow entry to the parking area.

Alternate Flows:

- If there is an issue with authentication, the system notifies the user and provides guidance on resolving the issue.

Preconditions:

- The user must have a valid booking and be at the parking gate.

Postconditions:

- The user successfully enters the parking area after authentication.

4.5.9 Add New Parking

Main Flows:

- The admin navigates to the parking management page of the Smart Car Parking System.
- The system displays a form that allows the admin to enter details for new parking spaces.
- The system adds the new parking details to the Smart Car Parking System.

Alternate Flows:

- If the system encounters an error while adding new parking, it displays an error message and prompts the admin to try again later.

Preconditions:

- The Smart Car Parking System must have admin privileges for adding new parking.

Postconditions:

- The new parking spaces are added to the system.

4.5.10 Add New Attendant

Main Flows:

- The admin navigates to the attendant management page of the Smart Car Parking System.
- The system displays a form that allows the admin to enter details for a new parking lot attendant.
- The system adds the new attendant details to the Smart Car Parking System.

Alternate Flows:

- If the system encounters an error while adding a new attendant, it displays an error message and prompts the admin to try again later.

Preconditions:

- The Smart Car Parking System must have admin privileges for adding new attendants.

Postconditions:

- The new attendant details are added to the system.

4.5.11 Feasibility Study

Feasibility study is an assessment of the practicality of a proposed project or system. A feasibility study aims to objectively and rationally uncover the strengths and weaknesses of an existing or proposed solutions, opportunities and threats present in the natural environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained.

- **Technical Feasibility:** It is an evaluation of the hardware and software and how it meets the need of the proposed system. All the hardware and software tools used in this system are easily available. They are easy to understand and easy to get expertise.

- **Operational Feasibility:** Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition. All the components used in this system are maintainable, reliable and replaceable at nominal cost.

- **Economic Feasibility:** The purpose of an economic feasibility study is to demonstrate the net benefit of a proposed project for accepting or disbursing funds or benefits. A simple economic analysis which gives the actual comparison of costs and benefit are much meaningful in this case.

Chapter 5

DESIGN

5.1 System Architecture

A system architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system.

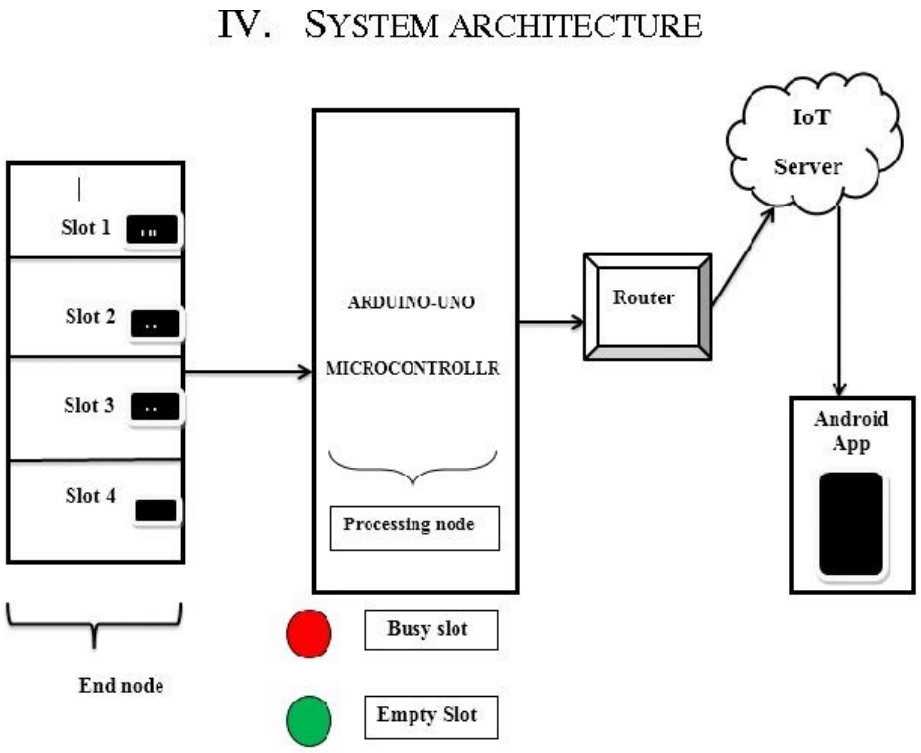

Figure 5.1: System Architecture

5.2 Data Flow Diagram

A data-flow diagram is a way of representing a flow of data of a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow; there are no decision rules and no loops.

5.2.1 Data Flow Diagram (Level 0)

Level 0 is a top level (also known as "context diagram") data flow diagram. It only contains one process node ("Process 0") that generalizes the function of the entire system in relationship to external entities.

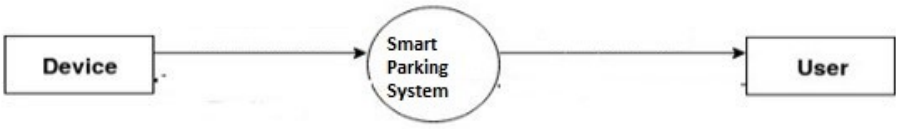

Figure 5.2: Top level data flow diagram of the entire system (Level 0)

Data Flow Diagram (Level 1)

A level 1 data flow diagram (DFD) which is shown in figure 5.3 is more detailed than a level 0 DFD. It breaks down the main processes into sub-processes that can then be analyzed and improved on a more intimate level.

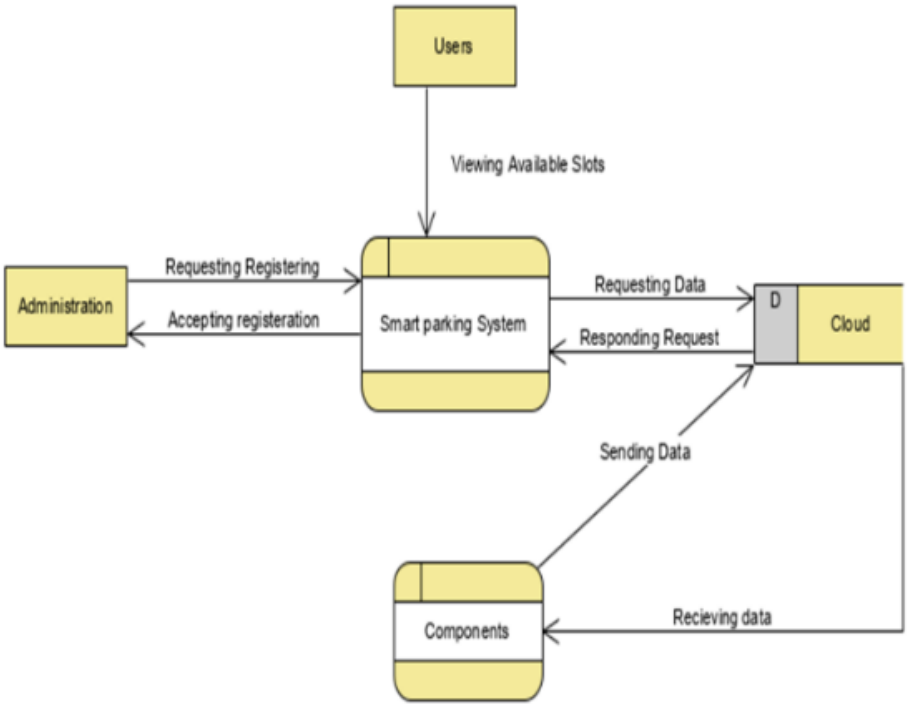

Figure 5.3: Detailed Data flow diagram of the system with sub-processes (DFD Level 1)

5.3 Activity Diagram

Activity diagrams are used to describe the dynamic aspects of the system, representing the flow from one activity to another.

5.3.1 Online Booking

Activity diagram of online booking is shown in figure 5.4.

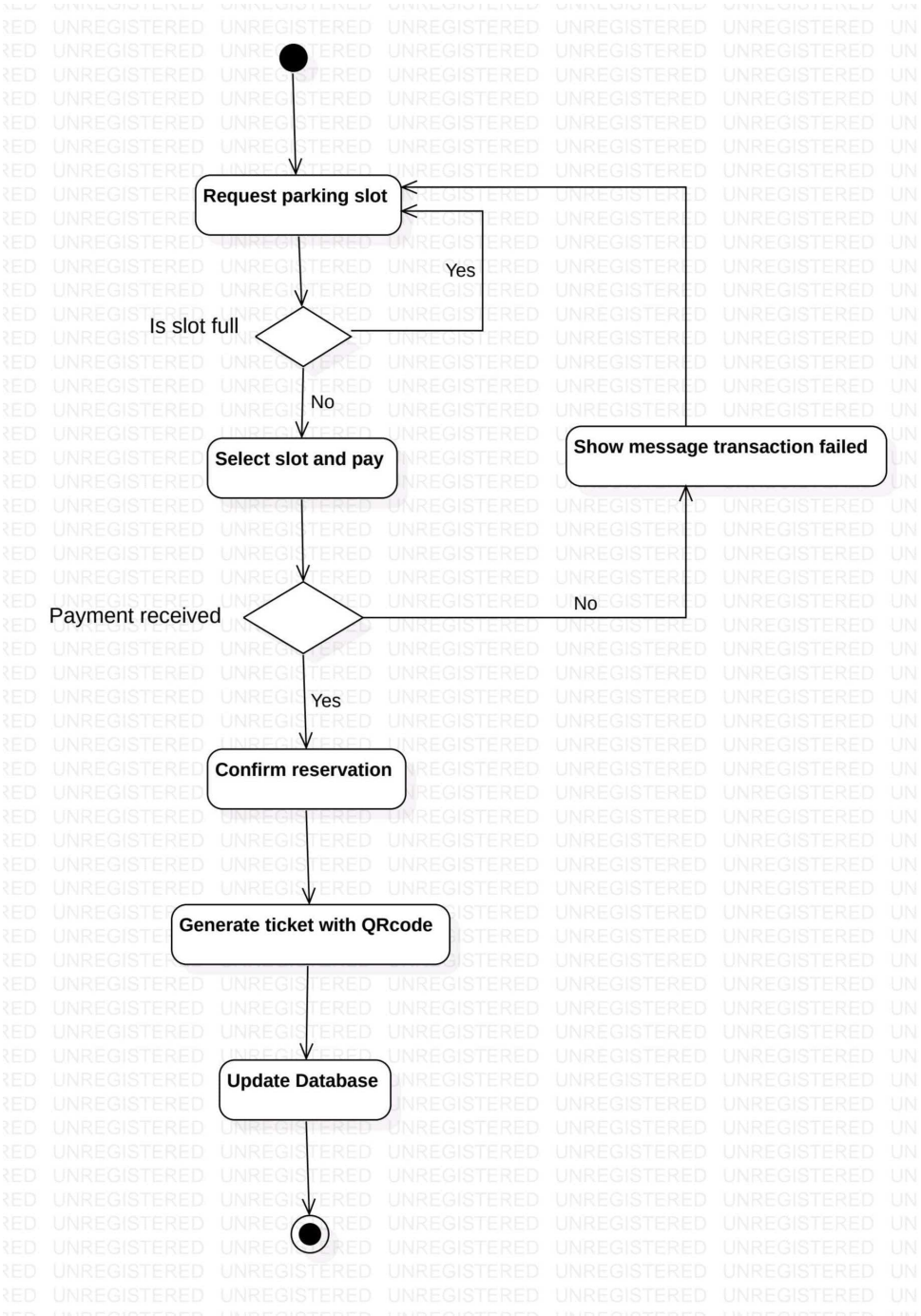

Figure 5.4: Online Booking Activity Diagram

5.4 Algorithm

1. Install Servo Motor and Keypad at Entry Gate for Authentication:

- Install a servo motor and a keypad at the entry gate of the smart parking system.
- The servo motor controls the gate's opening and closing mechanism.
- The keypad serves as a means of authentication for users entering the parking area.
- Only users with a valid authentication code will be allowed to pass through the gate.

2. Connect IoT Devices to a Central System:

- Deploy IoT devices, such as sensors, at each parking spot to detect occupancy.
- Establish a connection between these IoT devices and a central system.
- The central system serves as the brain of the smart parking infrastructure, managing data from IoT devices and controlling various aspects of the system.

3. Develop a Slot Booking Application:

- Create a user-friendly slot booking application for both Android and iOS platforms.
- The application should provide users with real-time information about available parking spots.
- Users can browse the available slots, select a preferred spot, and proceed to book it through the application.

4. User Requests Parking Slot - Sent Request to Attendant:

- When a user requests a parking slot through the application, the request is sent to the parking lot attendant for approval.
- The attendant reviews the request and decides whether to approve or reject it based on availability and other criteria.

5. Attendant Approval - Notification with Payment Link and Authentication Code:

- If the parking lot attendant approves the user's request, a notification is sent to the user.

- The notification contains a payment link for the parking fee and a unique authentication code for accessing the parking area.

6. User Payment and Arrival at Parking:

- The user follows the payment link in the notification to complete the payment for the booked parking slot.
- Upon successful payment, the user proceeds to the smart parking area.

7. Enter Code at Smart Parking Gate for Automatic Access:

- At the smart parking gate, the user enters the authentication code received in the notification.
- The system verifies the code, and if valid, the servo motor automatically opens the gate for the user to enter.

8. Authenticated User Parks Car:

- Authenticated users can now park their cars hassle-free in the designated parking area.
- The smart parking system ensures a seamless and secure experience for users from booking to parking.

Chapter 6

Implementation

This chapter presents the overall implementation progress of the Smart Parking System project. The primary objective is to demonstrate the execution of the proposed solutions and recommendations. Implementation involves the actualization of new features, improvements to existing functionalities, optimization of system performance, enhancement of data quality and accuracy, and the reduction of system downtime. Key points associated with these goals include:

6.1 Code Restructuring

The Code Restructuring aims to establish a more reliable, efficient, and user-friendly Smart Parking System, aligning it with user needs. The focus is on:

- Enhancing existing features
- Implementing new features
- Optimizing system performance
- Improving data quality and accuracy
- Reducing system downtime
- Minimizing HTTP requests

6.1.1 Enhancement of Existing Features

The implementation includes refining and improving the functionalities that were part of the initial system, ensuring a seamless and enhanced user experience.

6.1.2 Implementation of New Features

New features are added to expand the capabilities of the Smart Parking System, providing additional value to users.

6.1.3 Optimization of System Performance

Efforts are directed towards optimizing the overall performance of the system, ensuring swift and efficient operations.

6.1.4 Improvement of Data Quality and Accuracy

Steps are taken to enhance the accuracy and quality of data within the Smart Parking System, contributing to better decision-making.

6.1.5 Reduction of System Downtime

Strategies are implemented to minimize system downtime, ensuring continuous availability and reliability.

6.1.6 Minimizing HTTP Requests

To improve the system's responsiveness, the number of HTTP requests is minimized, contributing to faster load times.

6.2 Different functionality implementations

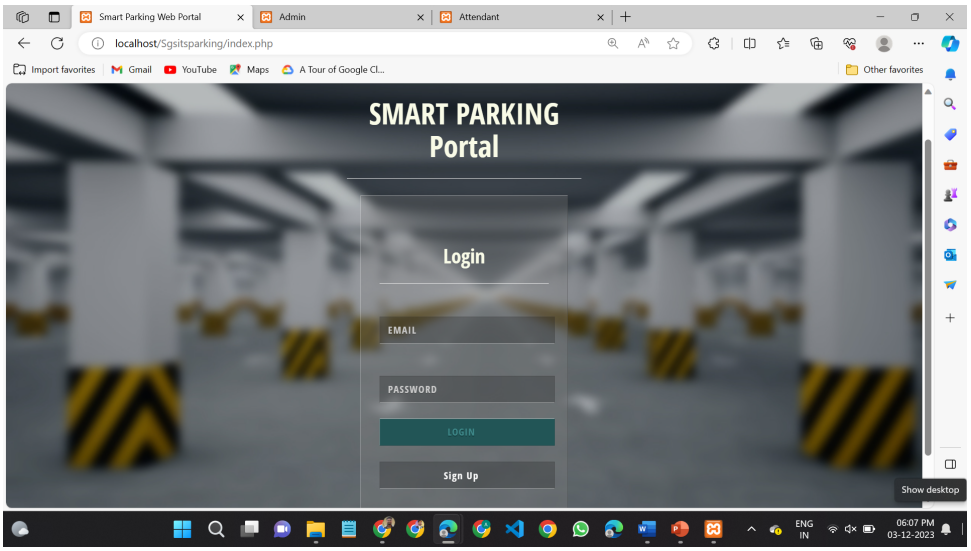

Figure 6.1: User Login

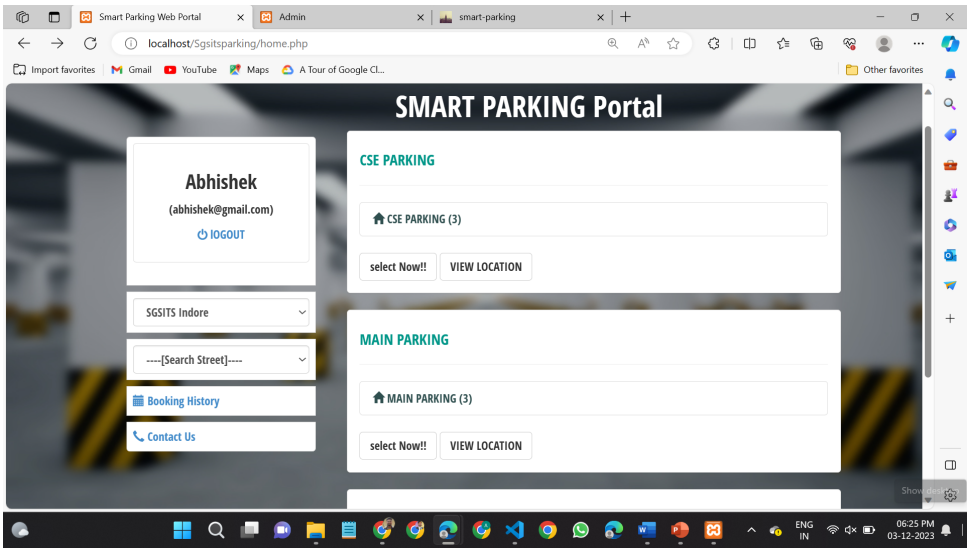

Figure 6.2: User Interface

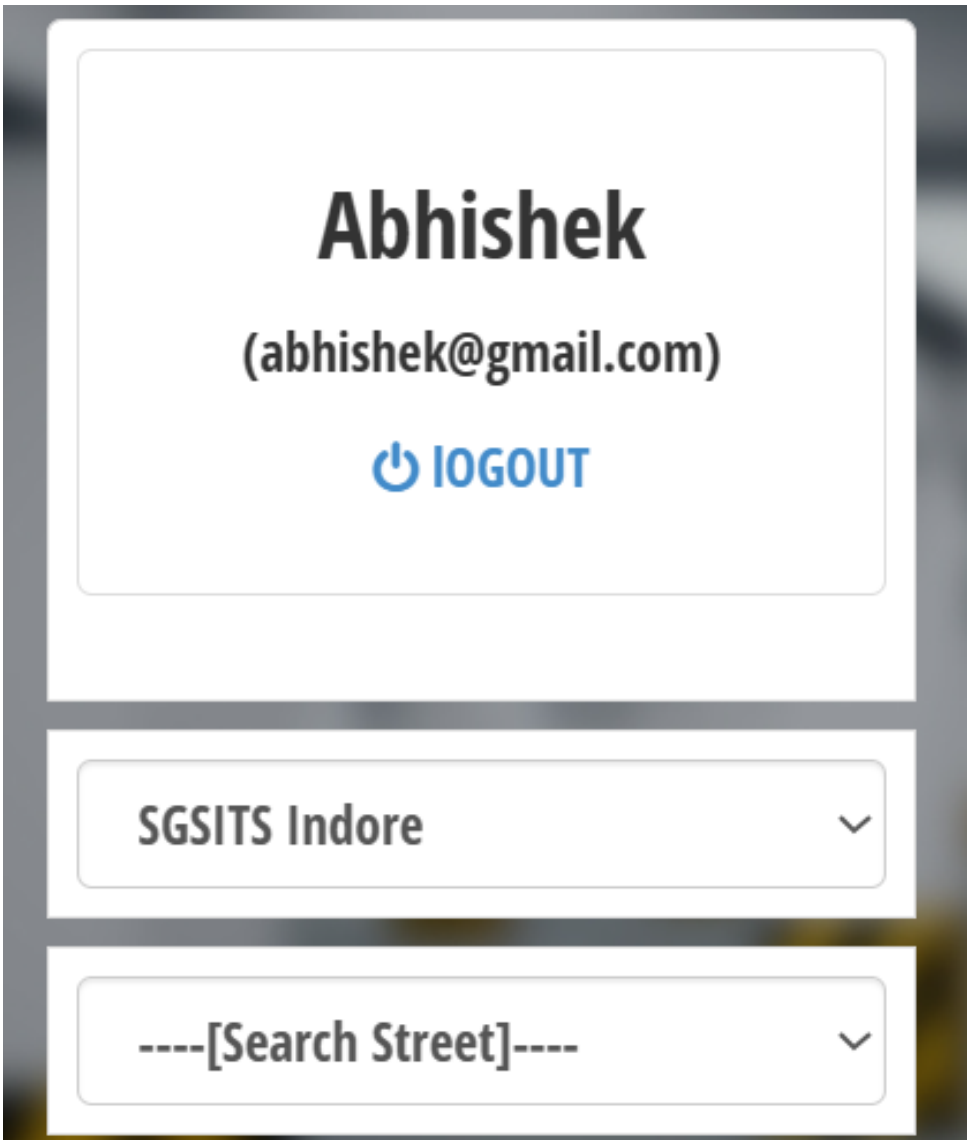

Figure 6.3: Search Along Street or Area

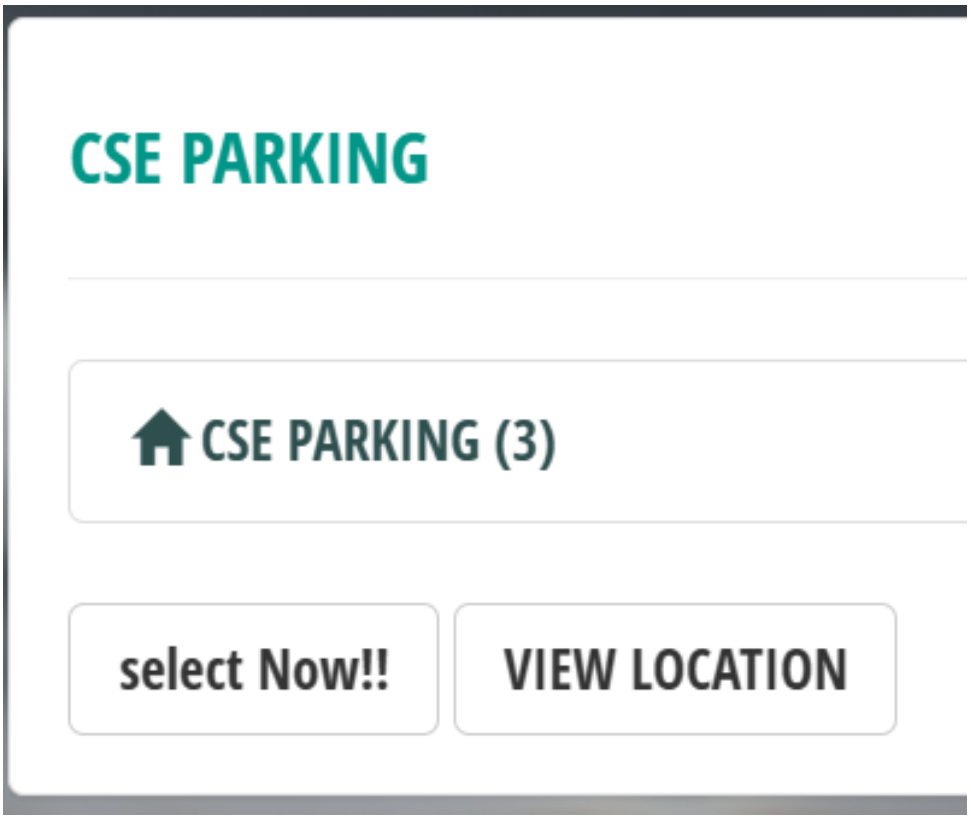

Figure 6.4: View Location

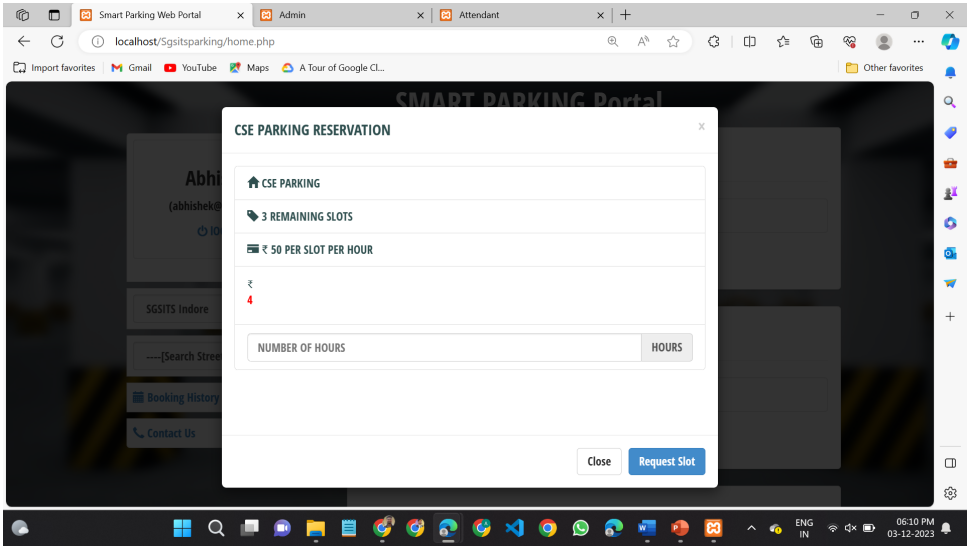

Figure 6.5: Slot Information

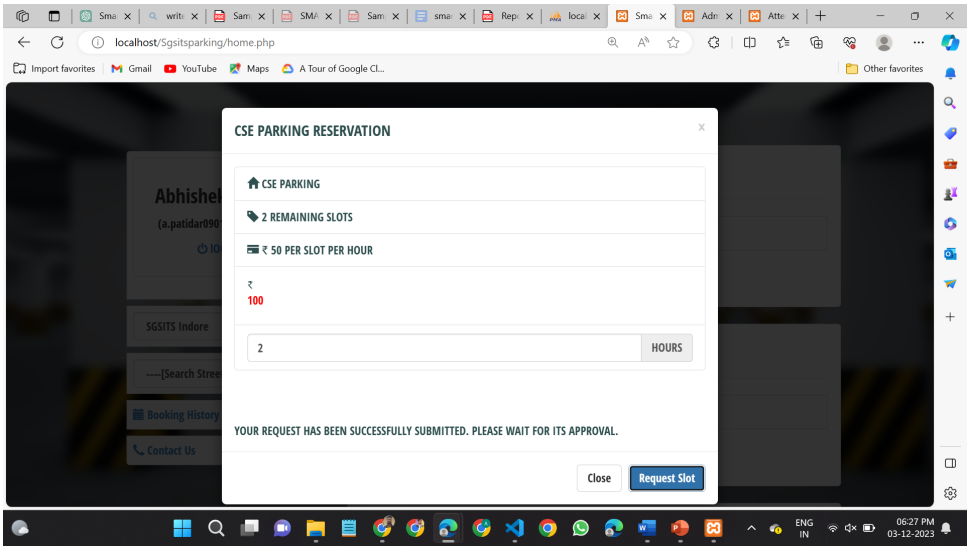

Figure 6.6: Request Submitted

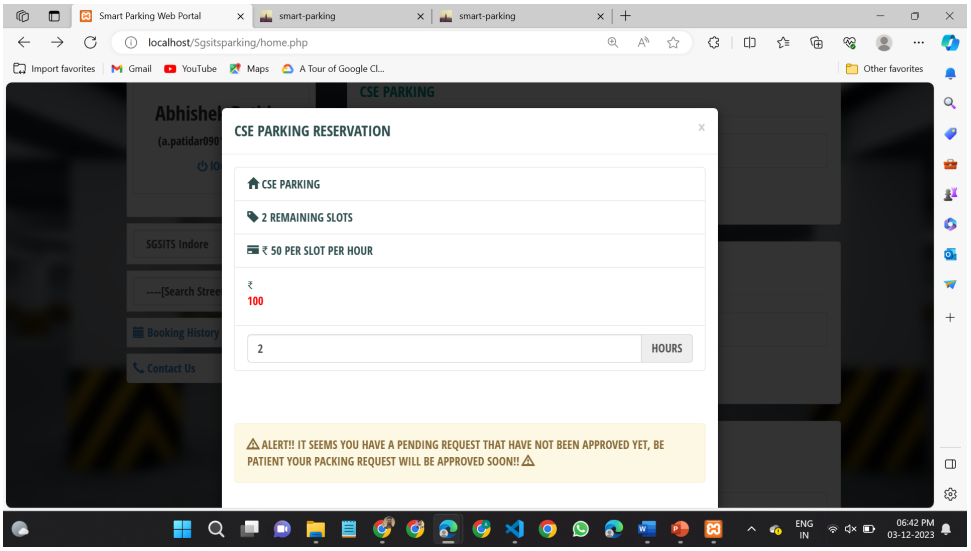

Figure 6.7: Already Requested Slot

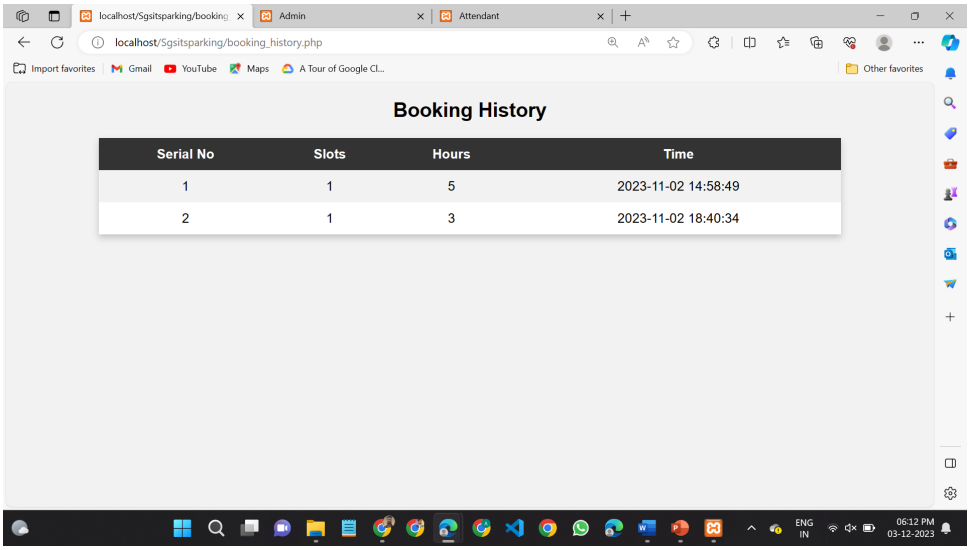

Figure 6.8: Booking History

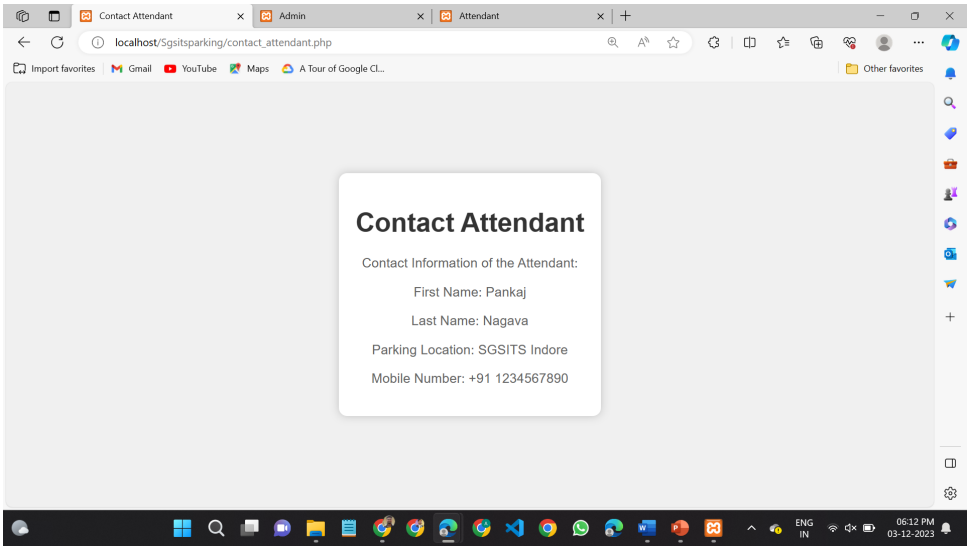

Figure 6.9: Contact Page

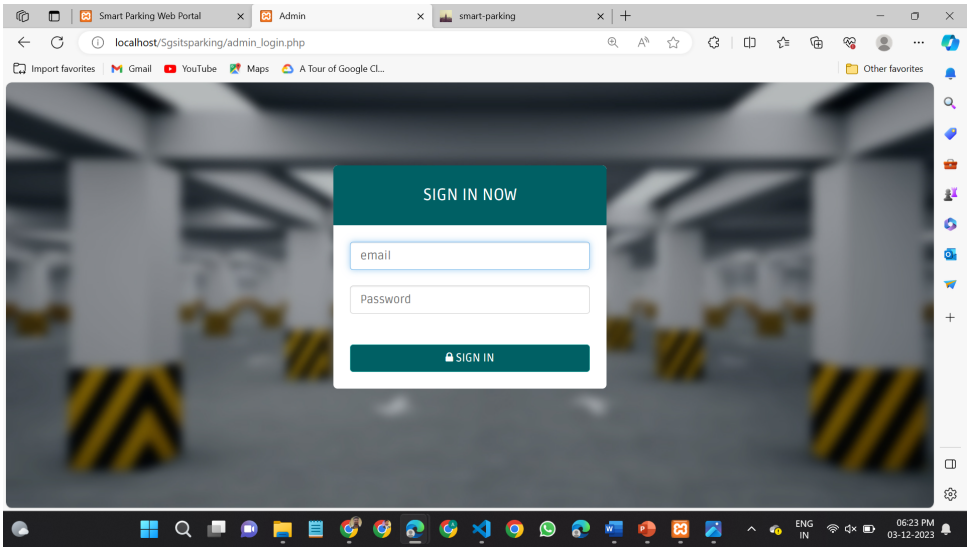

Figure 6.10: Admin or Attendant Login

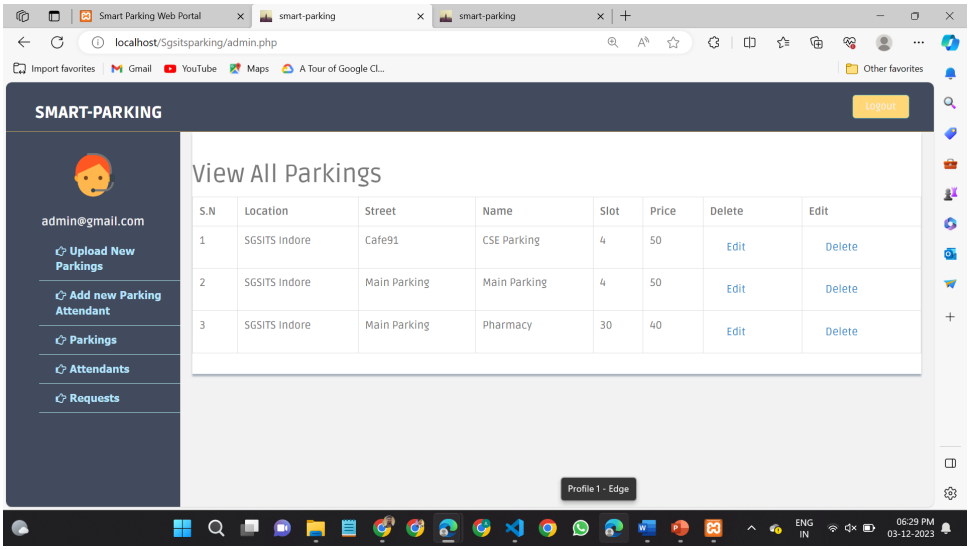

Figure 6.11: Admin Interface

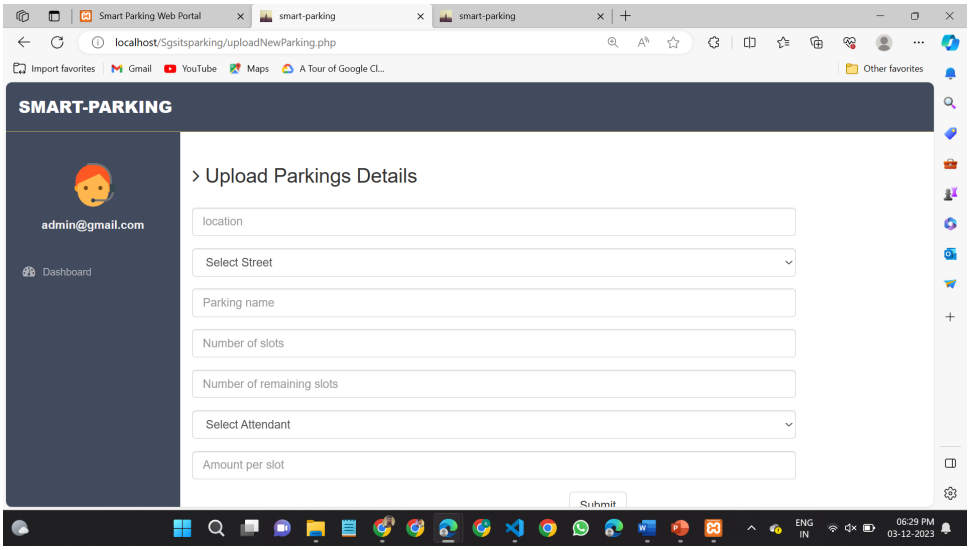

Figure 6.12: Upload New Parking

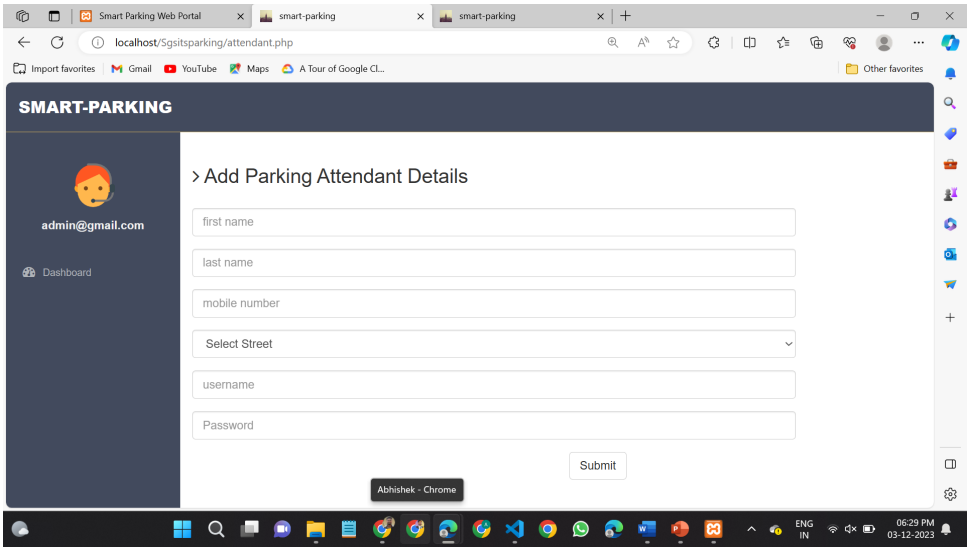

Figure 6.13: Add New Parking Attendant

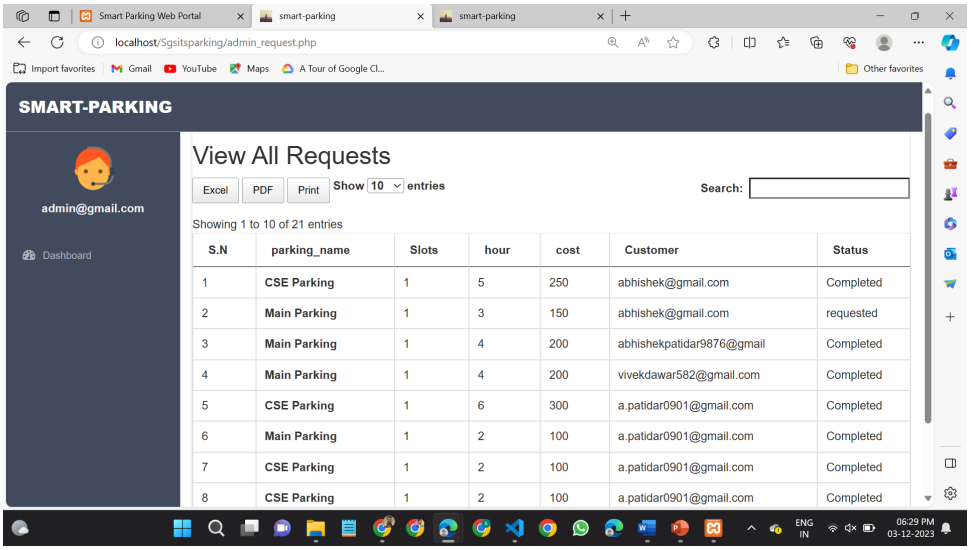

Figure 6.14: View, Search, or Download Requests

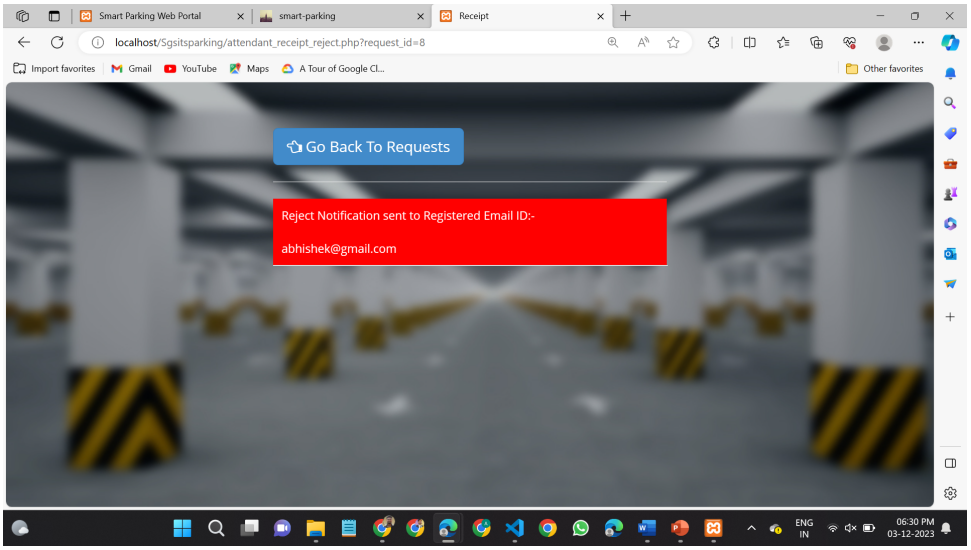

Figure 6.15: Rejected Request

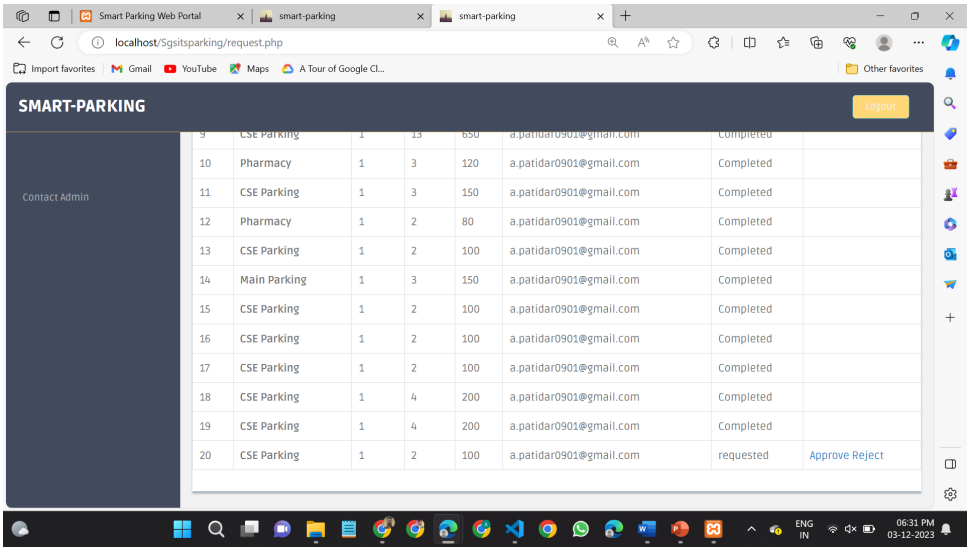

Figure 6.16: Approve or Reject Request by Attendant

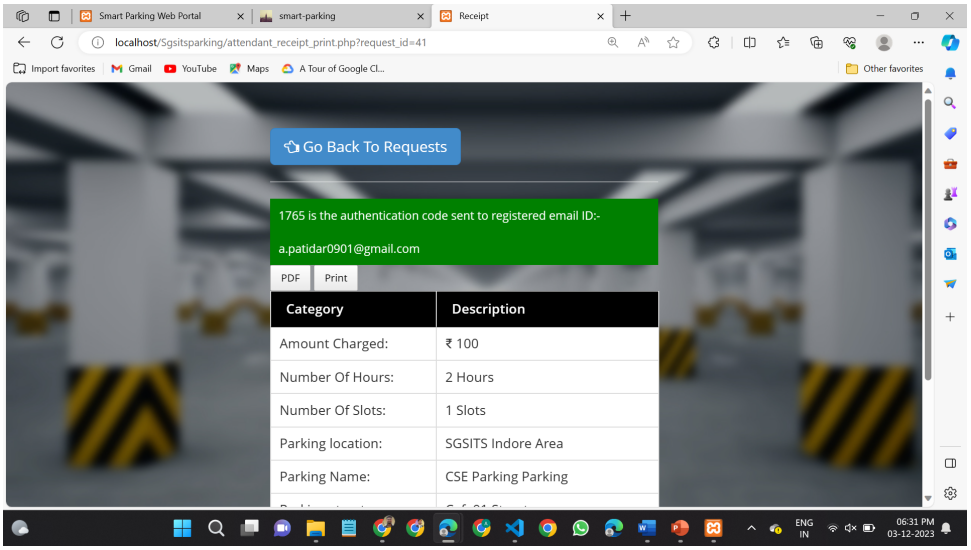

Figure 6.17: Approved Request and Code with Notification Sent

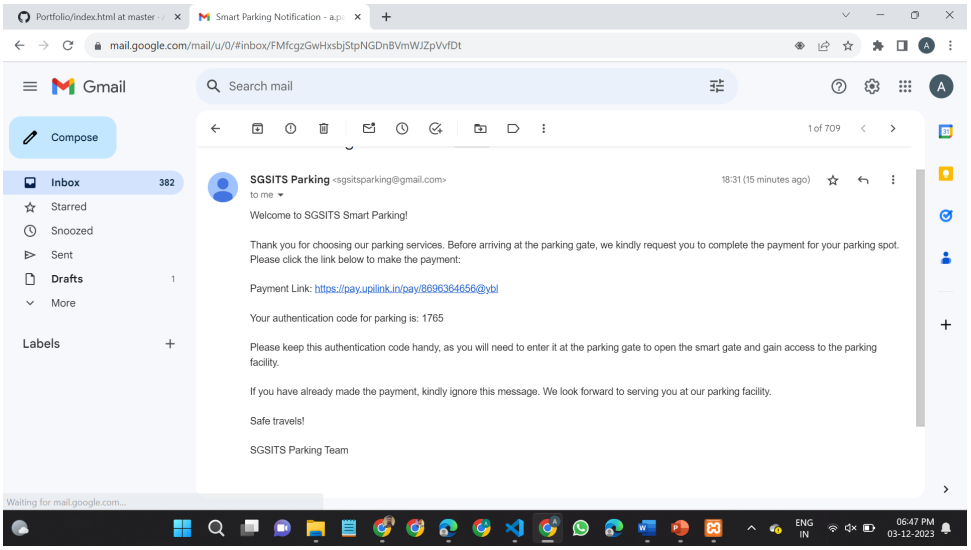

Figure 6.18: Email Notification

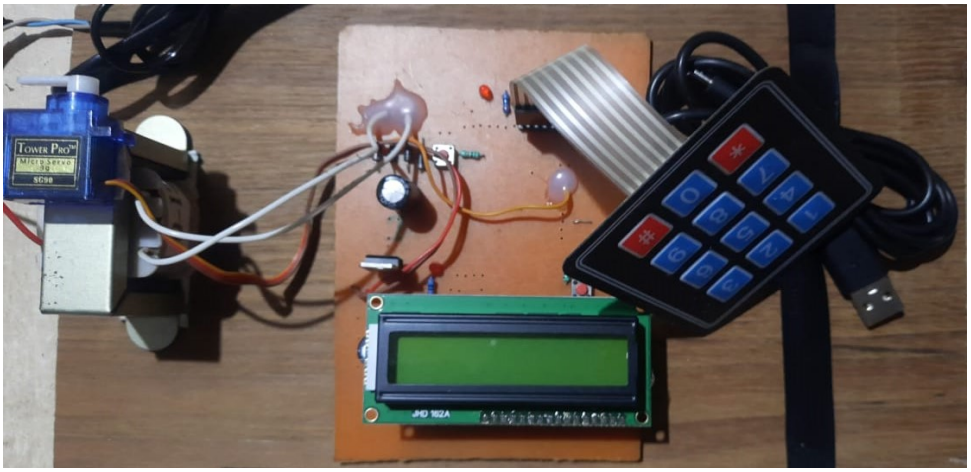

Figure 6.19: *IoT Model*

Chapter 7

TESTING AND RESULT

This chapter includes testing which refers to the process of evaluating the quality, functionality, and performance of a system. It helps to identify and rectify any defects or issues that could impact the overall success of the project.

7.1 Test Cases

Test Case - 1

Test Case: Add new parking

Test Statement:

Verify that a new parking space can be successfully added to the smart parking management system.

Expected Result:

1. Upon providing valid information (parking name, location, capacity), a new parking space should be added to the system.
2. The system should display a confirmation message upon successful addition.

Actual Result:

After entering valid information, a new parking space is added, and a confirmation message is displayed.

Pass/Fail:

Pass

Test Case - 2

Test Case: Add new Attendant

Test Statement:

Verify that a new attendant can be successfully added to the smart parking management system.

Expected Result:

1. Upon providing valid information (attendant name, contact details), a new attendant should be added to the system.
2. The system should display a confirmation message upon successful addition.

Actual Result:

After entering valid information, a new attendant is added, and a confirmation message is displayed.

Pass/Fail:

Pass

Test Case - 3

Test Case: Details of parking spaces

Test Statement:

Verify that the system can display accurate details of all parking spaces.

Expected Result:

1. The system should show a list of all parking spaces with their respective details (name, location, capacity).
2. The details should be accurate and up-to-date.

Actual Result:

The system displays a list of parking spaces with accurate details.

Pass/Fail:

Pass

Test Case - 4

Test Case: Details of attendants

Test Statement:

Verify that the system can display accurate details of all attendants.

Expected Result:

1. The system should show a list of all attendants with their respective details (name, contact details).
2. The details should be accurate and up-to-date.

Actual Result:

The system displays a list of attendants with accurate details.

Pass/Fail:

Pass

Test Case - 5

Test Case: Vehicle Authentication

Test Statement:

Verify that the vehicle authentication process works correctly by entering the code at the gate and opening the servo motor.

Expected Result:

1. Upon entering a valid code at the gate, the system should authenticate the vehicle and open the servo motor.
2. If the code is invalid, the system should deny access and not open the servo motor.

Actual Result:

After entering a valid code, the system authenticates the vehicle, and the servo motor opens.

Pass/Fail:

Pass

Test Case - 6

Test Case: Real-time update of slots available

Test Statement:

Verify that the system provides real-time updates on the available parking slots.

Expected Result:

1. The system should continuously update and display the current number of available parking slots.
2. The updates should be instantaneous and accurate.

Actual Result:

The system provides real-time updates on available parking slots.

Pass/Fail:

Pass

Test Case - 7

Test Case: Book slot

Test Statement:

Verify that a user can successfully book a parking slot using the provided code.

Expected Result:

1. Upon entering a valid booking code, the system should reserve a parking slot for the user.
2. The system should display a confirmation message upon successful booking.

Actual Result:

After entering a valid booking code, the system reserves a parking slot, and a confirmation message is displayed.

Pass/Fail:

Pass

Test Case - 8

Test Case: List of requests

Test Statement:

Verify that the system displays a list of pending requests for parking slots.

Expected Result:

1. The system should show a list of requests with details (user name, booking code, requested time).
2. The list should be accurate and up-to-date.

Actual Result:

The system displays a list of pending requests with accurate details.

Pass/Fail:

Pass

Test Case - 9

Test Case: Approve or reject Requests

Test Statement:

Verify that the system allows attendants to approve or reject pending requests.

Expected Result:

1. Attendants should be able to approve a request, leading to the reservation of a parking slot.
2. Attendants should be able to reject a request, notifying the user about the rejection.

Actual Result:

Attendants can successfully approve or reject pending requests.

Pass/Fail:

Pass

Test Case - 10

Test Case: Print Receipt

Test Statement:

Verify that the system can generate and print a receipt for a booked parking slot.

Expected Result:

1. After successful booking, the system should generate a receipt with details (user name, booking code, parking space, time).
2. The printed receipt should be accurate and contain all relevant information.

Actual Result:

The system generates and prints a receipt with accurate details after successful booking.

Pass/Fail:

Pass

Test Case - 11

Test Case: Send Notification

Test Statement:

Verify that the system sends notifications for successful bookings and rejections.

Expected Result:

1. Users should receive notifications for successful bookings with relevant details.
2. Users should receive notifications for rejected requests with appropriate information.

Actual Result:

Users receive notifications for successful bookings and rejections with accurate information.

Pass/Fail:

Pass

Chapter 8

Conclusion

This chapter presents the conclusion of the work done in the analysis and design phase. In Section 8.1 and 8.2 the conclusion of the proposed approach is discussed. Section 8.3 explains the limitations of the proposed approach. The Future Scope of the proposed approach is described in Section 8.4.

8.1 Features

The Smart Parking Project revolutionizes traditional parking systems by seamlessly integrating advanced technology for enhanced efficiency and user convenience. With a user-friendly interface, individuals can securely log in and utilize a powerful search feature to find real-time information on available parking spaces. Users can easily book parking slots, track their history, and enjoy a streamlined experience. On the administrative side, the project provides dedicated tools for efficient space management, allowing administrators to add new spaces effortlessly and attendants to review and respond to user requests based on availability and criteria. This comprehensive solution aims to redefine and elevate the parking experience for both users and administrators.

8.2 Benefits of the project

1. **Efficient Space Utilization:** The Smart Parking Project optimizes parking by providing real-time information on available slots, reducing congestion and environmental impact.

2. **User Convenience:** The intuitive mobile app enables drivers to easily find, view, and book parking spaces, ensuring a seamless user experience.

3. **Administrative Streamlining:** The project streamlines administrative tasks, allowing efficient handling of user requests and maintenance of comprehensive booking history.

4. **Enhanced Communication:** Features like a contact page and notifications promote effective communication between users and attendants, facilitating assistance and updates.

5. **Real-Time Slot Updates:** The system ensures real-time updates on parking slot availability, providing users with the latest information for an efficient parking experience.

8.3 Limitations of Proposed Approach

The proposed smart parking system, while innovative and efficient, has certain limitations that should be considered:

- **Dependency on Internet Connectivity:** The system relies on a continuous and stable internet connection for seamless operations. Disruptions in connectivity may affect performance and user experience.
- **Vulnerability to Power Failures:** In the event of power outages, the system's functionality may be compromised, leading to potential disruptions in user access and authentication processes.
- **Dependency on External Services:** Some functionalities, such as email notifications, rely on external services. Any issues with these services could impact the system's ability to send notifications promptly.

8.3.1 Future Enhancements

While the current implementation of the smart parking system is robust, there are opportunities for future enhancements to further elevate its capabilities and user experience:

- **Integration of Occupancy Sensors:** Implementing occupancy sensors in parking slots would enable real-time tracking of slot availability, providing users with more accurate and up-to-date information. This enhancement could enhance overall system efficiency.
- **Incorporation of Mobile Payment Options:** Introducing mobile payment options within the application would offer users a convenient and cashless method for paying parking fees. Integrating popular mobile payment services can enhance the payment process.
- **Smart Notification Customization:** Enhancing the notification system to allow users to customize their notification preferences can contribute to a more personalized user experience. Users could choose to receive notifications through various channels or tailor the content according to their preferences.

Chapter 9

References

1 INRIX Research. url:

- https://inrix.com/press-releases/parking-pain-us/.

2 Reinventing Parking. url:

- https://www.reinventingparking.org/2013/10/is-30-of-traffic-actually-searching-for.html.

3 Singapore smart parking management system. url:

- https://www.parking.sg/.

4 Smart-parking management algorithms in smart city. Nature Journal. url:

- https://www.nature.com/articles/s41598-022-10076-4.

5 The design and implementation of a smart-parking system for Helsinki Area. url:

- https://www.hel.fi/static/kanslia/Innovaatiorahasto/charles-thesis-formated-covered.pdf

6 The Problem With Manual Parking System. url:

- https://rentcubo.com/the-problem-with-manual-parking-system/