PUSL3190 Computing Individual Project

Final Report

PARKING APPLICATION

Supervisor: Ms. Dulanjalee Wijesekara

Name: Rathnayake M Rathnayake
Plymouth Index Number: 10820945
Degree Program: Computer science

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Ms. Dulanjalee Wijesekara, for her unwavering support, guidance, and encouragement throughout the development of this project. Her invaluable insights and constructive feedback have been instrumental in shaping this project and helping me overcome various challenges along the way.

I am also grateful to my university for providing the resources and opportunities to undertake this project, which has significantly contributed to my personal and professional growth. I would like to thank my peers and colleagues for their support and for providing a collaborative environment that fostered creativity and innovation.

Additionally, I extend my appreciation to all the individuals and organizations who participated in this project, offering their time and expertise. Their contributions were vital to the successful completion of this project.

Finally, I would like to thank my family and friends for their constant encouragement and support, which motivated me to persevere and achieve my goals.

Thank you all for your invaluable contributions to this project.

Abstract

In Colombo, Sri Lanka, the rapid growth in population and vehicle ownership has created significant parking challenges, leading to traffic congestion, elevated fuel consumption, and increased pollution. The Parking App Project addresses these issues by providing a comprehensive, user-centric solution to streamline the parking experience. By leveraging technology and data-driven insights, the app aims to reduce the time spent searching for parking, protect users from scams, and optimize the utilization of parking facilities.

The app features two login forms: one for users (car owners) and one for employees (parking owners/ticket collectors). Users can register multiple vehicles, receive a unique user ID and QR code, and integrate payment methods for seamless transactions. Employees, including government-authorized parking managers and private property owners, can register their parking spaces, update availability in real-time, and manage parking operations through the app.

An admin panel oversees the entire system, ensuring secure management of user and employee data while maintaining privacy for sensitive information. The app provides real-time parking space updates, navigational directions to available spots, and secure electronic payments, enhancing convenience and safety for users.

This innovative solution not only alleviates parking-related issues but also promotes urban mobility and economic productivity. By integrating private parking spaces and offering tax benefits to owners, the app increases parking availability and creates a more inclusive, efficient ecosystem. The project employs Agile and RUP methodologies to ensure adaptable and structured development, aiming to transform Colombo into a sustainable, congestion-free city.

Contents

Chapter 01 Introduction 8
1.1 Introduction 8
1.2 Background 9
1.3 Problem statement 10
1.4 Objectives 11
1.4.1 Business objectives 11
1.5 Risk identified 13
1.6 Benefits of the solution 14
1.7 Requirement 15
1.7.1 Hardware requirement 15
1.7.2 Software requirement 16
1.7.3 Network requirement 17
1.8 Time frame 18
1.9 High Level Architecture Diagram 18
1.10 Chapter summary 19
Chapter 02 Literature Review 20
2.1 Chapter overview 20
2.2 Literature review 20
2.2.1 Introduction 20
2.2.2 Current State of Parking Systems in Colombo 21
2.2.3 Proposed Parking App Solution 22
2.2.4 Comparison with International Solutions 24
2.2.5 Future Research and Development 25
2.2.6 Conclusion 27
Chapter 03 Method Of Approach 28
3.1 Method Of Approach 28
3.2 Chapter Overview 28
3.3 Data collection method 28
3.4 Extraction Of Relevant Data 28
3.5 Data Validation And QA 29
3.6 Why Did I chose This Method 29
3.7 User Centric Design 30

3.7.1 Human-Centric Design 30
3.8 Integrating Agile and RUP Methodology for the Parking App Project 31
Chapter 04 System Analysis and Design 34
4.1 System Analysis And Design 34
4.1.1 functional requirement 34
4.1.2 non-functional requirement 35
4.2 UML Diagram 37
4.2.1 Use case diagram 37
4.2.2 Activity diagram 40
4.2.3 sequence diagram 41
4.2.3 class diagram 42
4.3 System high level architecture 42
4.3.1 Data Inputs 42
4.3.2 Data Processing 43
4.3.3 Data Output 44
4.4 System high level architecture 45
4.5 Method Of Data Gathering 46
4.5.1 Sample of data gathering 46
4.6 Beginning of the project 47
4.7 Primary Data Gathering 48
4.8 Secondary Data Gathering 48
4.9 System design 48
4.9.1 System user interfaces 48
4.9.1.1 User interfaces 49
4.9.1.2 Owner interfaces 50
4.9.1.3 Admin interfaces 51
4.9.2 Color selection 51
4.9.3 Heuristic Evaluation of User Interface 52
4.9.4 User Interface Evaluation using Schneiderman's Eight Golden Rules 52
Chapter 05 System implementation 53
5.1 Screenshot Of The Code 53
5.2 Packages that I Imported 58
5.3 Variables of the Code Implementation 59

5.3.1 General Variables: 59
5.3.2 OTP and SMS Variables: 59
5.3.3 Parking and QR Code Variables: 60
5.3.4 Session and Error Handling Variables: 60
Chapter 06 Testing and Results 60
6.1 Evaluation 60
6.2 Test cases 61
6.2.1 Admin interface test cases 61
6.2.2 owner interface testcases 62
6.2.3 User Interface Test Cases 62
6.3 Neumann architecture 63
Chapter 07 Future Directions and Conclusion 64
7.1 Project Conclusion 64
7.1.1 Design conclusion 64
7.1.2 Development conclusion 64
7.1.3 Lessons learned 65
7.2 Future Direction 65
7.2.1 what I Did 65
7.2.2 What I Recommend For the Future Implementation 66
7.2.3 What I needed to implement but I couldn't 66
7.3 Future Implementation 66
Chapter 08 Postmortem 67
8.1 How did I archived my objectives 67
8.2 Business objectives 68
8.2.1 Business opportunists 69
8.2.1.1 Strength and Weakness 69
Reference List 70
Appendices 72
Data Gathering method 72
Records of supervisory meetings 73
User Guide for my system 78

List of Tables

Table 1 Hardware requirement 15
Table 2 Software requirement 17
Table 3 Network requirement 17
Table 4 functional requirement 35
Table 5 non-functional requirement 36
Table 6 Admin interface test cases 62
Table 7 owner interface testcases 62
Table 8 User Interface Test Cases 63

List Of Figures

Figure 1 Time frame 18
Figure 2 High Level Architecture Diagram 18
Figure 3 Literature review 21
Figure 4 Use case diagram 38
Figure 5 Activity diagram 40
Figure 6 sequence diagram 41
Figure 7 class diagram 42
Figure 8 System high level architecture 45
Figure 9 User interfaces 49
Figure 10 Owner interfaces 50
Figure 11 Admin interfaces 51

Chapter 01 Introduction

1.1 Introduction

Colombo, the bustling capital of Sri Lanka, serves as the country's commercial and political hub, drawing in a daily influx of over 600,000 vehicles. This heavy vehicular presence underscores the city's significance but also highlights a critical infrastructural challenge: the acute shortage of parking spaces. The resulting parking crisis exacerbates traffic congestion, environmental pollution, and economic inefficiencies, significantly affecting the quality of urban life.

Existing parking systems in Colombo are outdated and predominantly manual, relying on cash transactions and minimal technology integration. These inefficiencies contribute to prolonged searches for parking, increased fuel consumption, and susceptibility to parking scams, further complicating the urban mobility landscape. The lack of a centralized, technology-driven solution

has led to underutilized parking spaces, unsafe parking conditions, and a negative overall experience for drivers.
To address these pressing issues, the Parking Hub Project aims to revolutionize parking management in Colombo through the development of a comprehensive mobile application. This app will leverage cloud-based technology and data-driven insights to streamline the parking process, enhance security, and optimize the utilization of available parking spaces. Key features include real-time parking availability, secure electronic payments, GPS navigation, and a robust user registration system.

The project's objectives are multifaceted, focusing on reducing traffic congestion, combating parking scams, improving user experience, and supporting economic efficiency. By providing real-time information and facilitating seamless transactions, the app will minimize the time and resources wasted on parking searches. Additionally, it will enhance safety and reliability, ensuring that only verified parking operators are included in the system.
The Parking Hub Project represents a significant step towards creating a more livable, efficient, and sustainable urban environment in Colombo. By digitizing and centralizing fragmented parking operations, the project seeks to transform the urban mobility experience, fostering a more productive and environmentally friendly city. Through thoughtful design and advanced technology, this initiative aims to address the root causes of Colombo's parking challenges, paving the way for a smarter and more connected urban future.

1.2 Background

Colombo, the dynamic capital of Sri Lanka, is the nation's epicenter of economic, political, and cultural activities. As the most attractive and valuable city in the country, it experiences an influx of over 600,000 vehicles daily, a statistic reported by the Ministry of Transport. This heavy vehicular presence underscores Colombo's significance but also highlights a critical infrastructural challenge: an acute shortage of parking spaces.

The rapid growth and urbanization of Colombo have not been matched by an equally robust development in parking facilities. The resulting imbalance between parking demand and supply leads to severe traffic congestion, prolonged searches for parking spaces, and increased fuel consumption. This situation exacerbates environmental pollution and diminishes urban efficiency, significantly affecting the quality of life for residents and visitors.

Current parking management systems in Colombo are outdated and predominantly manual, relying heavily on cash transactions and minimal technological integration. These inefficiencies contribute to prolonged searches for parking, increased fuel consumption, and a susceptibility to parking scams. Opportunistic behaviors, such as fraudulent parking attendants charging

exorbitant fees and providing unsafe parking conditions, further complicate the urban mobility landscape.

Additionally, the underutilization of available parking spaces is a common occurrence due to the lack of an effective system to ensure these resources are maximized. Spaces in prime locations often remain underused, while areas just slightly farther away are avoided due to safety concerns and poor management. This inefficient use of space reflects broader issues within urban planning and management in Colombo, highlighting the critical need for a systemic overhaul.

The introduction of a digital solution, such as the proposed Parking Hub mobile application, is envisioned to transform the landscape of parking in Colombo. By leveraging technology to create a real-time, user-friendly platform, the application aims to provide accurate information on parking availability, streamline payment processes, and secure transactions. This will not only improve the efficiency of parking management but will also tackle issues such as scams and space underutilization.

The proposed Parking Hub app aims to bridge the gap between existing inefficient parking management systems and the need for a modern, technology-driven solution. It will offer a comprehensive and integrated approach, connecting drivers, parking owners, and administrators without the need for complex IoT infrastructure. The app will enable automated booking, contactless entry/exit, digital payments, and centralized oversight of all parking operations through seamless software integration.

By focusing on software integration and mobile access without complex hardware, the Parking Hub app provides a feasible and scalable solution suited to Colombo's context. It offers a practical upgrade path for technology-averse parking owners and aims to drive adoption through better convenience, control, and transparency. The app's user-friendly mobile interface and administrative portal for managing operations and revenue analytics differentiate it from existing local parking solutions.

1.3 Problem statement

Colombo, the capital city of Sri Lanka, is the nation's commercial and political hub, attracting a daily influx of over 600,000 vehicles. This heavy vehicular presence underscores the city's significance but also highlights a critical infrastructural challenge: an acute shortage of parking spaces. The imbalance between parking demand and supply leads to severe traffic congestion, prolonged searches for parking, increased fuel consumption, and heightened environmental pollution.

The current parking management systems in Colombo are outdated and predominantly manual, relying on cash transactions and minimal technology integration. These inefficiencies contribute to time-consuming parking searches, traffic congestion, and economic losses due to increased fuel consumption and reduced productivity. Furthermore, the absence of a centralized,

technology-driven solution has led to underutilized parking spaces, unsafe parking conditions, and a negative overall experience for drivers.

Moreover, the city is plagued by parking scams and opportunistic behaviors due to the lack of a reliable system for validating legitimate parking facilities. Drivers are often exploited by fraudulent parking attendants, resulting in inflated parking fees and safety concerns. The inefficient utilization of existing parking spaces, coupled with the absence of a centralized management system, exacerbates these issues, compromising urban mobility and the quality of life for residents and visitors.

These parking challenges have broader implications, including increased traffic congestion, environmental pollution, and health hazards. According to statistics, a significant number of Sri Lankan citizens spend valuable time on the roads due to the lack of parking areas in Colombo. This situation leads to economic inefficiencies, with private sector losses attributed to increased fuel consumption and labor hours. Ultimately, these issues contribute to a decline in productivity and economic growth.

The root cause of these challenges is the absence of a proper parking facility system developed by the government or authorized parties. Without an effective system, managing the high demand for parking in Colombo remains an insurmountable task. Addressing these issues requires a comprehensive solution that leverages technology to streamline parking management, enhance security, and optimize the utilization of available parking spaces.

The Parking Hub Project aims to tackle these pressing issues by developing a comprehensive mobile application that revolutionizes parking management in Colombo. By providing real-time information on available parking spaces, implementing robust security features, and optimizing parking facilities through technological integration, the project seeks to enhance urban mobility, reduce environmental impact, and improve the overall urban experience.

1.4 Objectives

- Reduce Traffic Congestion
- Enhance Parking Space Utilization
- Improve User Experience
- Combat Parking Scams
- Support Economic Efficiency
- Optimize Fuel Efficiency

1.4.1 Business objectives

1. Reduce Traffic Congestion:

    - Implement a mobile application that provides real-time information on available parking spaces, reducing the time drivers spend searching for parking and alleviating traffic congestion in Colombo.

2. Optimize Parking Space Utilization:

    - Maximize the use of existing parking facilities by efficiently allocating spaces through a smart digital management system, increasing overall parking capacity without the need for extensive physical expansion.

3. Enhance User Experience:

    - Provide a seamless and user-friendly parking experience by integrating features such as secure electronic payments, GPS navigation, and real-time updates on parking availability.

4. Eliminate Parking Scams:

    - Incorporate robust verification and security features to ensure that only legitimate and verified parking operators are part of the system, protecting users from fraudulent activities and unsafe parking conditions.

5. Increase Economic Efficiency:

    - Reduce the economic impact of traffic congestion and parking inefficiencies by minimizing fuel consumption, decreasing the time spent on parking searches, and enhancing overall productivity for businesses and individuals.

6. Support Sustainable Urban Development:

    - Contribute to a cleaner environment by reducing vehicle emissions associated with prolonged idling and unnecessary driving, thereby supporting sustainable urban development initiatives.

7. Facilitate Private Property Registration:

    - Encourage private property owners to participate in the parking ecosystem by allowing them to register their spaces through the app, increasing the overall supply of parking spaces and providing additional revenue streams for property owners.

8. Centralize Parking Management:

- Develop an administrative portal for centralized oversight and management of parking operations, enabling data-driven decision-making, efficient resource allocation, and effective monitoring of parking facilities.

1.5 Risk identified

1. **Technical Risks:**

- System Downtime and Reliability: The application may experience downtime or technical glitches that can disrupt service availability and user experience.

- Integration Challenges: Difficulties in integrating the app with existing parking management systems, payment gateways, and third-party services such as Google Maps API.
- Data Security and Privacy: Risks of data breaches, unauthorized access, or mishandling of sensitive user information such as personal details and payment information.

2. **User Adoption Risks:**

- User Resistance to Change: Potential reluctance from users to adopt a new digital solution, especially among those who are accustomed to traditional parking methods.

- Usability Issues: The app might have a steep learning curve or user interface problems that could hinder user adoption and satisfaction.

3. **Financial Risks:**

- High Development and Maintenance Costs: Unexpected costs associated with developing, deploying, and maintaining the application, including server costs, updates, and support.

- Revenue Shortfall: Potential for lower-than-expected revenue generation if user adoption is slow or if parking fees collected through the app are insufficient to cover costs.

4. **Regulatory and Compliance Risks:**

- Regulatory Changes: Changes in local laws and regulations regarding parking management, data privacy, and digital transactions could impact the app's operations.

- Compliance Issues: Ensuring that the app complies with all relevant regulations and standards, such as data protection laws and payment security standards.

5. **Operational Risks:**

- Scalability Issues: Challenges in scaling the application to handle a growing number of users and parking facilities without compromising performance.

- Employee Training and Management: Ensuring that parking employees are properly trained to use the app and update parking availability in real-time.

6. **Market Risks:**

- Competitive Pressure: The emergence of competing parking solutions that could capture market share and reduce the app's user base.

- Market Acceptance: The app may fail to gain sufficient market traction if it does not meet user expectations or if there is insufficient awareness and marketing.

7. **Stakeholder Risks:**

- Stakeholder Buy-In: Difficulty in securing buy-in from key stakeholders, such as government authorities, private parking owners, and businesses.

- Partnership Dependence: Over-reliance on partnerships with private parking owners or third-party service providers, which could pose risks if these partnerships fail or are terminated.

8. **Environmental Risks:**

- External Factors: External factors such as economic downturns, natural disasters, or significant changes in urban infrastructure could impact the app's operations and user demand.

9 **Maintenance and Support Risks:**

- Ongoing Maintenance: The need for continuous updates, bug fixes, and feature enhancements to keep the app functional and competitive.

- Customer Support: Ensuring that there is adequate customer support to address user issues, feedback, and complaints promptly and effectively.

1.6 Benefits of the solution

The Parking Hub Project offers a comprehensive solution to the parking challenges in Colombo, bringing numerous benefits to the city's residents and visitors. By providing real-time

information on available parking spaces, the app reduces traffic congestion and enhances traffic flow, making commuting more efficient. The optimized utilization of parking facilities maximizes the city's parking capacity without requiring extensive physical expansion. Users enjoy a seamless and convenient parking experience, with secure electronic payments, GPS navigation, and real-time updates, significantly reducing stress and saving time. The app also enhances safety and security by verifying parking operators and protecting users from fraudulent activities. Economic efficiency is improved as the reduction in search time and fuel consumption translates to cost savings for drivers and businesses, boosting overall productivity. Additionally, the app contributes to environmental sustainability by decreasing vehicle emissions associated with prolonged idling. Private property owners can generate additional revenue by registering their spaces, increasing the overall supply of parking spaces. The centralized management portal enables data-driven decision-making and efficient monitoring of parking operations, while the secure payment system streamlines fee collection and improves revenue generation. The app builds community trust through its reliable and transparent service, promoting technological adoption and aligning with smart city initiatives. Designed to be scalable and adaptable, the Parking Hub app ensures future growth and addresses parking challenges in various urban contexts, ultimately enhancing urban mobility and contributing to a more livable, efficient, and sustainable Colombo.

1.7 Requirement

1.7.1 Hardware requirement

Requirement	Description
User Device	• smartphone with GPS capabilities and a camera for QR scanning. • Why: Essential for users to access the Parking Hub app, obtain real-time parking information, and verify parking using QR codes.
Employee Device	• Smartphone or tablet for managing parking spaces. • Why: Necessary for employees to update parking availability, manage parking spaces, and verify user entries/exits in real-time.

Table 1 Hardware requirement

- The hardware requirements for the Parking App Project ensure that both users and employees have the necessary tools to interact with the system effectively. Users need smartphones with GPS and cameras to access the app, find parking, and verify their reservations. Employees require smartphones or tablets to manage parking spaces, update availability, and verify user actions, thereby maintaining an organized and secure parking environment in Colombo.

1.7.2 Software requirement

Requirement	Description
Mobile Platform	• The app will be developed using Flutter for cross-platform support (iOS and Android). • Why: Ensures a consistent and high-quality user experience across different mobile operating systems with a single codebase.
Backend Infrastructure	• Cloud services such as Firebase for database management, user authentication, and hosting. • Why: Provides a scalable, secure, and real-time backend infrastructure to handle data storage, user management, and app hosting efficiently.
Payment Processing	• Secure integration with a reliable payment gateway (e.g., Stripe). • Why: Facilitates secure and seamless transactions, allowing users to pay for parking services electronically
Administrative Access	• Web-based portal accessible through standard web browsers (Chrome, Firefox, Safari, Edge). • Why: Enables administrators to manage and oversee the entire

	parking system, including user and employee management, through an intuitive and accessible interface.

Table 2 Software requirement

- These software requirements are crucial for ensuring the Parking App Project's success. Using Flutter for mobile development guarantees a consistent user experience across iOS and Android. Firebase provides a scalable and secure backend infrastructure. Integration with a payment gateway like Stripe ensures secure transactions. Finally, a web-based administrative portal allows efficient management of the entire system, ensuring smooth operation and oversight.

1.7.3 Network requirement

Requirement	Description
Internet Connection	• A continuous and reliable internet connection for users and parking employees. • Why: Ensures access to real-time data, including parking availability and transaction processing, enhancing the overall user experience and operational efficiency.
Network Security	• Secure network protocols for data transmission (e.g., HTTPS, SSL/TLS). • Why: Protects sensitive information from unauthorized access and ensures the integrity and confidentiality of data transmitted between devices and servers.

Table 3 Network requirement

- These network requirements are essential for ensuring the secure and efficient operation of the Parking App Project. A continuous and reliable internet connection enables real-time access to critical data for users and parking employees. Implementing secure network protocols like HTTPS and SSL/TLS protects data transmission, maintaining the integrity and confidentiality of user information and enhancing overall system security.

1.8 Time frame

Task	Start	End	Days
	02-11-23	13-04-24	164.0
Topic selection	02-11-23	05-11-23	4.0
Topic Discussion with Mentor	06-11-23	09-11-23	4.0
Research and Requirement Analysis	10-11-23	13-11-23	4.0
Making Notes	14-11-23	15-11-23	2.0
Proposal Preparation	16-11-23	18-11-23	3.0
System Design and Architecture	19-11-23	30-11-23	12.0
Front-end Development	01-12-23	29-12-23	29.0
Back-end Development	30-12-23	26-01-24	28.0
Database Design and Integration	27-01-24	10-02-24	15.0
Admin Panel Development	11-02-24	24-02-24	14.0
QR Code Integration	25-02-24	07-03-24	12.0
System Testing and Debugging	08-03-24	16-03-24	9.0
User Acceptance Testing	17-03-24	31-03-24	15.0
Final Documentation and Submission	01-04-24	13-04-24	13.0

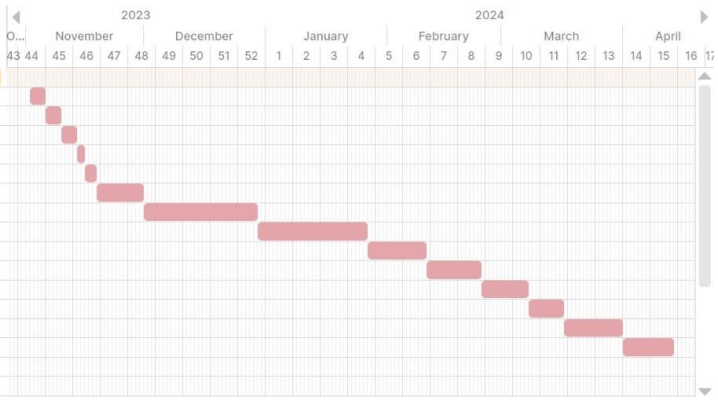


Figure 1 Time frame

1.9 High Level Architecture Diagram

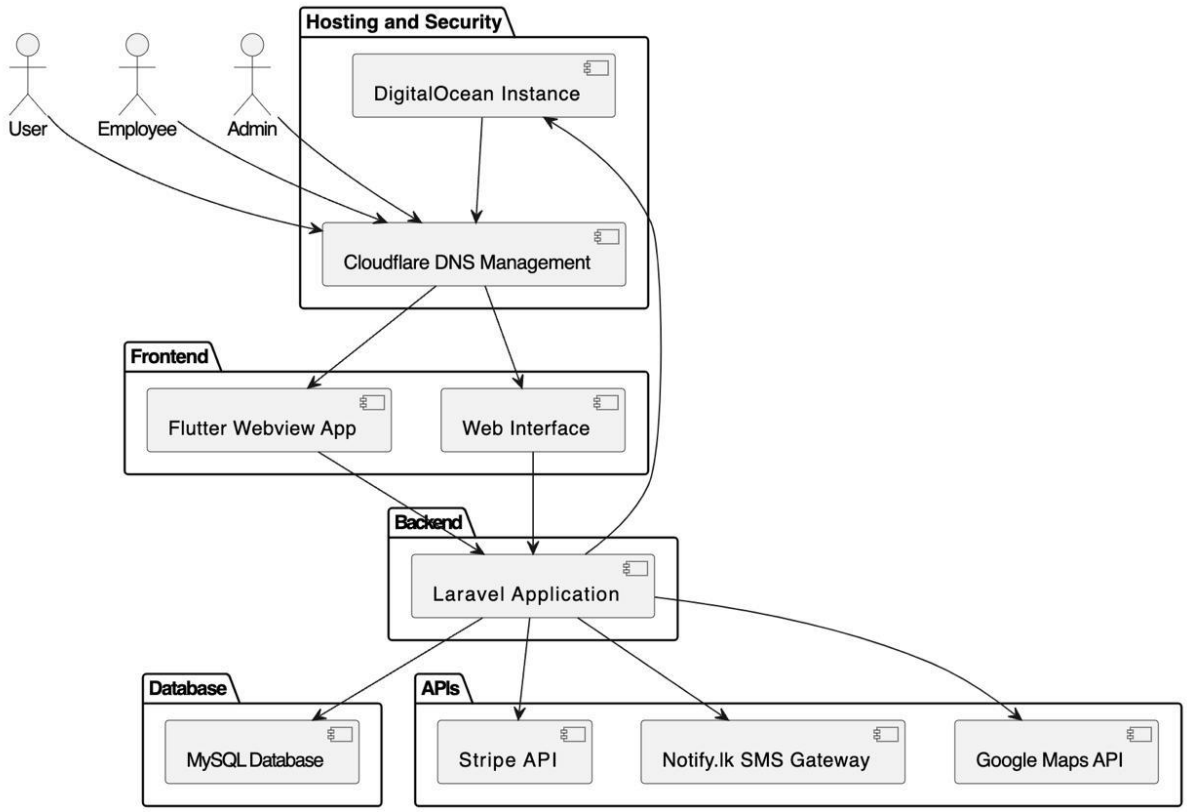


Figure 2 High Level Architecture Diagram

The Parking Hub system features a robust and efficient architecture. Hosting and Security are managed through Digital Ocean for scalable hosting and Cloudflare for secure, reliable access. Frontend components include a Flutter WebView App for cross-platform mobile access by users and employees, and a Web Interface for administrators to manage data and operations. The Backend is powered by a Laravel Application, which processes requests and handles business logic. The Database uses MySQL for secure storage of user data, parking information, and transactions. APIs integrated into the system include Stripe for payments, Notify.lk for OTP authentication, and Google Maps for location services. Users and employees interact with the app to access real-time parking data and manage spaces, while administrators use the web interface for oversight. This setup ensures a seamless, secure, and user-friendly parking management experience.

1.10 Chapter summary

Chapter 1 introduces the Parking Hub Project, aiming to address Colombo's parking challenges through a comprehensive mobile application. It outlines the city's infrastructural issues, including traffic congestion and parking scams, underscoring the need for a modern solution. The chapter details project objectives such as reducing congestion and enhancing user experience while highlighting identified risks and benefits. Additionally, it outlines hardware, software, and network requirements essential for implementation. The chapter concludes with a high-level architecture diagram illustrating the system's components. Overall, Chapter 1 provides a thorough introduction to the Parking Hub Project, setting the stage for subsequent chapters' exploration of development, implementation, and impact.

Chapter 02 Literature Review

2.1 Chapter overview

The literature review examines existing parking systems in Colombo and identifies a lack of comprehensive software-based solutions. It discusses previous research on parking technology, highlighting the limitations of IoT-based systems due to infrastructure requirements. The proposed parking app addresses these limitations by offering a cloud-based solution with minimal hardware needs. Key features include automated booking, digital payments, and centralized oversight. The review emphasizes the app's practicality, scalability, and alignment with Colombo's infrastructure constraints. It also discusses the importance of user acceptance, feature refinement, and revenue models.

2.2 Literature review

2.2.1 Introduction

The rapid urbanization and increasing vehicle population in Colombo, Sri Lanka, have exacerbated the city's parking challenges. Existing parking systems predominantly rely on outdated manual processes, physical tickets, and cash payments, leading to inefficiencies and user frustrations. The current methods fail to leverage modern technological advancements, resulting in prolonged searches for parking spaces, increased traffic congestion, and a higher risk of parking scams.

Research has proposed various technology-enabled parking systems, primarily focusing on Internet of Things (IoT) and hardware implementations. However, these solutions often involve significant infrastructure costs and complex deployment processes, making them impractical for large-scale implementation in Colombo. Comprehensive software-based platforms that offer end-to-end digital management of parking operations are notably absent in the local context.

To bridge this gap, the proposed parking app introduces an integrated cloud-based solution designed to connect drivers, parking owners, and administrators seamlessly. By avoiding the complexities and costs associated with IoT infrastructure, this app aims to provide automated booking, contactless entry and exit, digital payments, and centralized oversight of all parking operations through a user-friendly interface. The inclusion of features such as real-time occupancy updates by parking staff, private parking registration, and built-in tax management further distinguishes this solution.This literature review examines the limitations of Colombo's current parking systems, compares them with advanced international solutions, and outlines the proposed app's potential to transform parking management in the city. The review also explores future research directions to ensure the app's continued development and alignment with user needs and technological advancements. By focusing on software integration and usability, the proposed app aims to digitize and centralize fragmented parking operations, offering a scalable and practical solution tailored to Colombo's unique context.

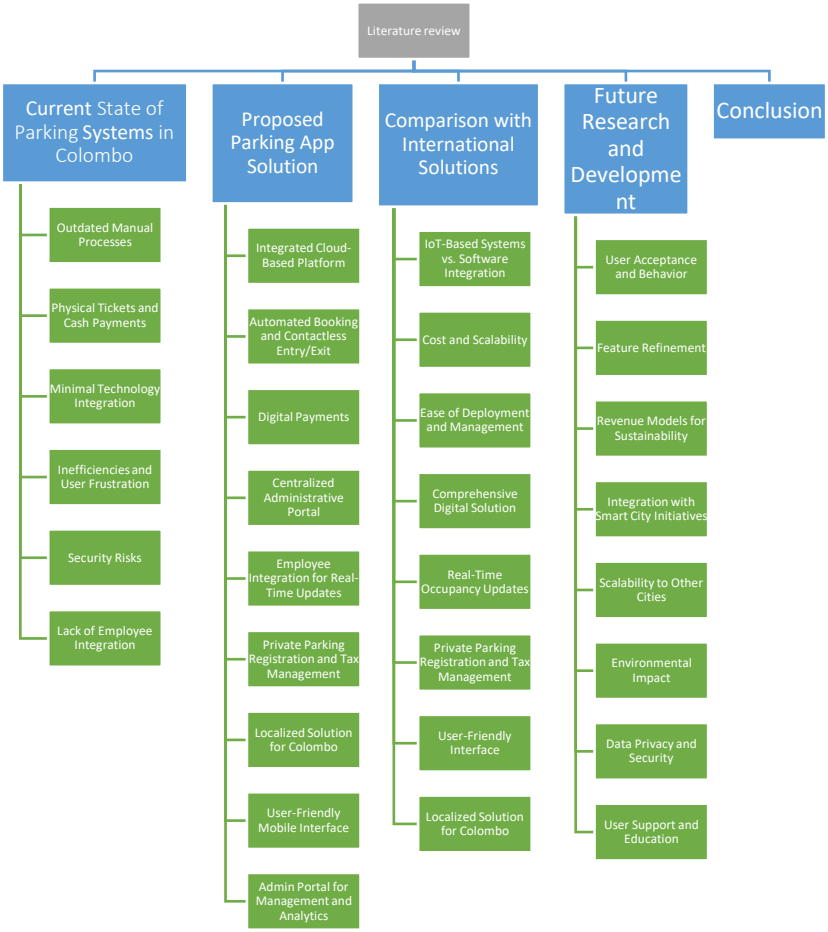


Figure 3 Literature review

- A high-level overview of the literature review structure for a project on parking systems in Colombo. The diagram is divided into five main sections: Current State of Parking Systems in Colombo, Proposed Parking App Solution, Comparison with International Solutions, Future Research and Development, and Conclusion. Each section includes detailed subtopics outlining key points and features of the proposed solution, highlighting the issues with current systems, and areas for future improvement and research.

2.2.2 Current State of Parking Systems in Colombo

The parking systems in Colombo are characterized by outdated manual processes, reliance on physical tickets, and cash payments. These methods have several significant limitations that contribute to the inefficiency of the current parking management system.

Outdated Manual Processes:

- The majority of parking operations in Colombo rely on manual processes, which include physically issuing tickets and manually recording parking space occupancy. This approach is not only time-consuming but also prone to human error. The lack of

automation means that parking operators must spend a considerable amount of time on tasks that could be streamlined with technology.

Physical Tickets and Cash Payments:

- Physical tickets and cash payments are the standard in Colombo's parking systems. This reliance on cash transactions increases the risk of fraud and revenue leakage, as cash handling can lead to discrepancies and theft. Additionally, the use of physical tickets is cumbersome for both users and operators, leading to longer wait times and inefficiencies.

Minimal Technology Integration:

- There is minimal integration of modern technology in Colombo's parking management. Without digital systems, there is no real-time information available on parking availability, leading to prolonged searches for parking spaces. Drivers often circle around looking for available spots, which exacerbates traffic congestion and increases fuel consumption.

Inefficiencies and User Frustration:

- The lack of a centralized digital system complicates the parking process, making it difficult for drivers to find available parking spaces quickly. This inefficiency not only frustrates users but also contributes to increased traffic congestion in the city. The absence of real-time data means that drivers cannot make informed decisions about where to park, leading to wasted time and resources.

Security Risks:

- The reliance on cash payments and physical tickets introduces several security risks. Without digital records, it is challenging to audit transactions and prevent revenue leakage. Moreover, the lack of secure payment options increases the risk of fraud and theft.

Lack of Employee Integration:

- Current parking systems do not incorporate parking employees into the digital framework. This means that parking staff cannot update parking availability in real-time, leading to inaccuracies in occupancy data. The inability to manage parking resources effectively further compounds the inefficiencies of the system.

2.2.3 Proposed Parking App Solution

The proposed parking app aims to revolutionize parking management in Colombo by addressing the inefficiencies of the current system through a comprehensive, software-based solution. This integrated cloud-based platform will connect drivers, parking owners, and administrators, eliminating the need for complex IoT infrastructure and offering a scalable, cost-effective alternative.

Integrated Cloud-Based Platform:

- At the core of the proposed solution is a cloud-based platform that enables seamless management of parking operations. This platform will provide real-time data on parking

availability, allowing drivers to find and reserve parking spaces efficiently. The cloud infrastructure ensures high availability, scalability, and reliability, making it suitable for Colombo's growing parking needs.

Automated Booking and Contactless Entry/Exit:

- One of the standout features of the app is its automated booking system, which allows users to reserve parking spaces in advance. This reduces the time spent searching for parking and alleviates traffic congestion. Additionally, the app supports contactless entry and exit through QR code scanning. Upon booking, users receive a unique QR code that can be scanned at the parking facility for seamless access, enhancing convenience and reducing physical contact.

Digital Payments:

- The app integrates secure digital payment options, allowing users to pay for parking services electronically. By partnering with reliable payment gateways such as Stripe, the app ensures fast and secure transactions. This shift from cash to digital payments minimizes the risks associated with cash handling, reduces operational inefficiencies, and provides a transparent transaction record for both users and parking operators.

Centralized Administrative Portal:

- A key differentiator of the proposed app is its centralized administrative portal. This web-based portal offers extensive control and oversight of parking operations. Administrators can manage user registrations, monitor parking availability, and analyze usage data through this portal. The data analytics capabilities provide valuable insights, helping administrators make informed decisions to optimize parking management and improve resource allocation.

Employee Integration for Real-Time Updates:

- Unlike many existing parking apps, the proposed solution incorporates parking employees into the system. Employees can update parking availability in real-time using the app, ensuring that the information provided to users is accurate and up-to-date. This integration streamlines the management of parking spaces, reduces errors associated with manual tracking, and enhances the overall efficiency of parking operations.

Private Parking Registration and Tax Management:

- The app also includes features for private property owners to register their parking spaces, increasing the overall supply of available parking in Colombo. By enabling crowdsourced sharing of private parking, the app can significantly expand parking options in a cost-effective manner. Additionally, built-in tax management features ensure that private parking owners comply with local regulations, simplifying the process for managing tax payments.

Localized Solution for Colombo:

- The proposed parking app is specifically designed to address the unique challenges faced by Colombo's parking system. It offers a practical, incremental upgrade path for technology-averse parking owners by focusing on software integration and mobile access

without requiring expensive hardware investments. This localized approach ensures that the solution is feasible, scalable, and well-suited to the city's infrastructure constraints.

User-Friendly Mobile Interface:

- The app will feature a user-friendly mobile interface designed to cater to drivers with varying levels of technological proficiency. The intuitive design makes it easy for users to navigate the app, book parking spaces, and make payments. By prioritizing usability, the app aims to drive adoption and ensure a positive user experience.

Admin Portal for Management and Analytics:

- The administrative portal provides a comprehensive suite of tools for managing operations and revenue analytics. Administrators have access to detailed reports on parking usage, financial transactions, and user behavior. These insights enable data-driven decision-making, helping administrators optimize operations and enhance service quality.

2.2.4 Comparison with International Solutions

IoT-Based Systems vs. Software Integration:

- Many international cities, especially in Europe and North America, have adopted advanced technological solutions to address their parking challenges. These cities often rely on Internet of Things (IoT) technologies, integrating sensors and cameras to monitor parking space occupancy and provide real-time data. Such systems offer high accuracy in tracking available spaces but come with significant upfront costs and require extensive infrastructure.

Cost and Scalability:

- IoT-based parking systems require substantial investment in hardware, such as sensors and cameras, as well as ongoing maintenance costs. These expenses can be prohibitive, especially in developing countries like Sri Lanka. Moreover, scaling IoT solutions to cover larger areas or additional parking facilities involves further significant investments.

The proposed app’s software-centric model, on the other hand, minimizes hardware dependencies and focuses on leveraging existing mobile devices and cloud infrastructure. This reduces initial setup costs and ongoing maintenance expenses. The cloud-based platform ensures that the system can easily scale to accommodate a growing number of users and parking facilities without substantial additional investments.

Ease of Deployment and Management:

- Implementing and managing IoT-based parking systems requires significant technical expertise and infrastructure adjustments, such as installing sensors and integrating them with existing systems. These complexities can pose challenges for cities with limited technical resources and infrastructure.

Comprehensive Digital Solution:

- While many international solutions focus on specific aspects of parking management, such as occupancy detection or automated payments, the proposed app offers a comprehensive digital solution. It covers the entire parking management process, including automated booking, contactless entry/exit, digital payments, and centralized oversight. This end-to-end digitization ensures a seamless and efficient user experience.

Real-Time Occupancy Updates:

- International IoT-based systems provide real-time data through sensors. However, the proposed app achieves similar functionality through real-time updates from parking employees. This method ensures accurate occupancy information without the need for expensive hardware, making it a practical alternative for Colombo.

Private Parking Registration and Tax Management:

- An innovative feature of the proposed app is the inclusion of private parking registration and built-in tax management. This allows private property owners to share their spaces, significantly increasing the overall parking supply in a cost-effective manner. Such features are less common in international IoT-based solutions, which typically focus on public parking facilities.

User-Friendly Interface:

- International parking systems often include mobile applications for user interaction, but the proposed app places a strong emphasis on a user-friendly interface tailored to drivers with varying levels of technological proficiency. This focus on usability aims to drive higher adoption rates and ensure a positive user experience.

Localized Solution for Colombo:

- The proposed app is specifically designed to address the unique challenges and infrastructure constraints of Colombo. While international solutions offer valuable insights, they may not always align with the local context. The localized approach of the proposed app ensures that it is well-suited to Colombo's specific needs, providing practical and incremental upgrades to the existing parking management system.

2.2.5 Future Research and Development

User Acceptance and Behavior:

- One of the key areas for future research is evaluating user acceptance of the proposed parking app. Understanding how drivers and parking operators perceive the app's usability, convenience, and security features is crucial for its success. Surveys, pilot studies, and user feedback sessions can provide valuable insights into user preferences and potential barriers to adoption. Analyzing user behavior and feedback will help identify areas for improvement and ensure the app meets the needs of its target audience.

Feature Refinement:

- Continuous improvement of the app's features based on user feedback and technological advancements is essential. Future research should focus on refining the app's

functionalities, such as enhancing the user interface, improving real-time data accuracy, and optimizing the booking and payment processes. Incorporating advanced features like dynamic pricing based on demand and predictive analytics for parking availability can further enhance the app's utility and user experience.

Revenue Models for Sustainability:

- Developing sustainable revenue models is critical for the long-term success of the parking app. Future research should explore various monetization strategies, such as subscription models, transaction fees, advertising partnerships, and collaborations with local businesses. Understanding the financial implications and potential revenue streams will help ensure the app's economic viability and provide a steady income for ongoing maintenance and development.

Integration with Smart City Initiatives:

- As Colombo evolves into a smart city, integrating the parking app with other smart city initiatives can amplify its impact. Future research should explore opportunities for collaboration with urban mobility projects, public transportation systems, and environmental monitoring programs. For example, integrating the app with public transit schedules and smart traffic management systems can create a more connected and efficient urban ecosystem. This holistic approach can enhance urban mobility, reduce traffic congestion, and improve the overall quality of life for residents.

Technological Advancements:

- Staying abreast of technological advancements is crucial for maintaining the app’s competitiveness and effectiveness. Research should investigate emerging technologies, such as machine learning for predictive analytics, blockchain for enhanced security and transparency, and augmented reality for improved navigation and user experience. Incorporating these innovations can enhance the app's capabilities, provide additional value to users, and ensure the app remains at the forefront of technological progress.

Scalability to Other Cities:

- While the app is designed specifically for Colombo, its principles and functionalities can be adapted to other cities facing similar parking challenges. Future research should examine the scalability and adaptability of the app to different urban contexts, including infrastructure requirements, cultural considerations, and regulatory environments. By developing a flexible and modular platform, the app can be customized to meet the needs of various cities, contributing to improved parking management and urban mobility on a larger scale.

Environmental Impact:

- Assessing the environmental impact of the proposed parking app is another important area for future research. By reducing the time spent searching for parking and alleviating traffic congestion, the app can potentially lower fuel consumption and emissions. Quantifying these environmental benefits and exploring additional features, such as promoting eco-friendly parking practices and integrating with electric vehicle charging

infrastructure, can further enhance the app's contribution to sustainable urban development.

Data Privacy and Security:

- Ensuring the privacy and security of user data is paramount. Future research should focus on enhancing data protection measures, such as advanced encryption techniques, secure data storage, and robust authentication protocols. Regular security audits and vulnerability assessments can help identify and address potential risks, ensuring the app remains secure against emerging threats. Additionally, transparent data privacy policies and user consent mechanisms will build trust and confidence among users.

User Support and Education:

- Providing excellent customer support and user education is essential for the app's success. Future research should explore effective strategies for user onboarding, training, and support. Developing comprehensive user guides, tutorials, and help centers can assist users in navigating the app's features and resolving any issues they may encounter. Implementing feedback loops and support channels, such as in-app chat, email, and helplines, will ensure users receive timely assistance and enhance their overall experience.

2.2.6 Conclusion

Colombo's parking system faces significant challenges due to outdated manual processes, cash payments, and minimal technology use. These inefficiencies cause prolonged searches for parking, increased traffic congestion, and a higher risk of scams. While IoT-based solutions are popular internationally, they are often impractical for Colombo due to their high costs and complexity.

The proposed parking app offers a practical alternative with a software-based solution that leverages cloud technology. This app enables automated booking, contactless entry/exit, digital payments, and centralized management without relying on expensive IoT infrastructure. Key features include real-time updates from parking employees, private parking registration, and integrated tax management.

This solution addresses immediate parking issues and provides a scalable, practical upgrade path. It enhances user convenience, operational efficiency, and control while ensuring security and user adoption. Future research should focus on user acceptance, feature refinement, sustainable revenue models, and integration with smart city initiatives. With continuous improvement, this app can significantly improve Colombo's parking system.

Chapter 03 Method Of Approach

3.1 Method Of Approach

3.2 Chapter Overview

- Our approach combines Agile Development and the Rational Unified Process (RUP) to ensure flexibility and structured management. Agile practices, including iterative sprints, daily stand-ups, and user story-driven development, facilitate quick adaptation and continuous stakeholder involvement. RUP complements this by dividing the project into four phases: Inception, Elaboration, Construction, and Transition, ensuring thorough documentation and risk assessment. This hybrid method allows us to address Colombo's parking challenges with a user-centric, scalable, and resilient solution.

3.3 Data collection method

- To gather data for the project on parking systems in Colombo, a detailed survey was conducted, and the responses were meticulously organized using Google Sheets. The survey aimed to capture a broad spectrum of insights and feedback from key stakeholders, including drivers, parking lot operators, and city administrators. It included questions on the frequency and nature of parking issues, the amount of time spent searching for parking, satisfaction with existing parking systems, and openness to adopting a new technology-driven solution. The survey was distributed through various channels such as email, social media, and local community groups to ensure a diverse and representative sample. The collected data was then analyzed and stored in Google Sheets, facilitating easy access and thorough analysis for informed decision-making in the project.

3.4 Extraction Of Relevant Data

- The extraction of relevant data from the Google Sheets involved several critical steps to ensure the accuracy and usability of the survey responses. First, the raw data was meticulously cleaned to remove any incomplete, inconsistent, or duplicate entries, ensuring a robust dataset for analysis. Next, the survey responses were categorized into key themes such as the frequency of parking issues, time spent searching for parking, user satisfaction with current parking systems, and willingness to adopt a new technology-driven solution. This categorization facilitated a structured analysis of the data. Filters were then applied to isolate responses based on demographics, such as frequent versus occasional drivers and different geographic areas within Colombo. This filtering process allowed for a detailed and nuanced understanding of parking challenges and user preferences. Finally, statistical analysis was conducted to quantify key metrics, such as the average time spent searching for parking and the percentage of users

experiencing frequent parking issues. This structured approach ensured that the most relevant and actionable insights were extracted from the survey data.

3.5 Data Validation And QA

Ensuring the accuracy and reliability of data is crucial in developing the parking application for Colombo. The data validation process begins with an initial review of survey responses and other collected data to remove incomplete, inconsistent, or duplicate entries. This step ensures that only valid data is used in subsequent analyses and development stages. The data is then normalized to maintain consistency in formats and units, facilitating easier analysis and integration into the system. Automated scripts and manual reviews are employed to perform validation checks, verifying the correctness and logical consistency of the data. These checks ensure that all mandatory fields are completed and that values fall within acceptable ranges.

Quality assurance (QA) is another vital component, encompassing both automated and manual testing to ensure the application's functionality and performance meet user expectations. Automated unit tests are written and executed to verify that individual components of the application function correctly. Integration tests are also conducted to ensure that different parts of the system work together seamlessly. Additionally, real-time data, such as parking availability updates, is continuously verified against expected outcomes to ensure accuracy and reliability. This comprehensive QA process helps identify and rectify issues early, ensuring a robust and reliable application.

By implementing thorough data validation and QA processes, the development team can ensure that the parking application for Colombo is both reliable and user-friendly, effectively addressing the city's parking challenges.

3.6 Why Did I chose This Method

The combined approach of Agile Development and the Rational Unified Process (RUP) was chosen for developing the parking application for Colombo due to its ability to balance flexibility and structure, essential for a complex urban project. Agile Development emphasizes iterative progress through short sprints, allowing the team to quickly adapt to changes and incorporate continuous feedback from stakeholders, including end-users, city planners, and administrators. This ensures that the application remains aligned with the evolving needs of its users. Meanwhile, RUP provides a disciplined, phase-driven framework that helps manage complexity by dividing the project lifecycle into clear stages: Inception, Elaboration, Construction, and Transition. This structured approach ensures thorough documentation, risk assessment, and quality assurance at every stage, reducing the likelihood of overlooked issues and ensuring comprehensive coverage of all project aspects. By integrating Agile’s adaptability with RUP’s rigor, this hybrid method offers a robust, scalable, and user-centric solution to effectively address Colombo’s parking challenges.

3.7 User Centric Design

User-centric design is at the core of developing the parking application for Colombo, ensuring that the final product meets the needs and expectations of its users. This approach focuses on creating an intuitive and accessible interface that provides a seamless experience for all users, including drivers, parking lot operators, and city administrators.

Understanding User Needs:
The design process begins with a thorough understanding of user needs, gathered through surveys, interviews, and observational studies. By engaging directly with potential users, we gain valuable insights into their pain points, preferences, and behaviors. This feedback informs the design decisions, ensuring the application addresses real-world challenges effectively.

Intuitive Interface:
An intuitive user interface (UI) is essential for user-centric design. The application features a clean, straightforward layout that makes it easy for users to navigate and perform tasks such as finding available parking spaces, making reservations, and processing payments. Icons, buttons, and menus are designed to be self-explanatory, reducing the learning curve for new users.

Personalized Experience:
The application incorporates personalized features to enhance user satisfaction. Drivers can save their favorite parking locations, receive real-time updates on parking availability, and set preferences for notifications. Parking lot operators can manage bookings, track occupancy, and view analytics tailored to their specific needs. This level of personalization ensures a more efficient and enjoyable user experience.

Accessibility:
Accessibility is a key consideration in user-centric design. The application is designed to be accessible to users with varying abilities, incorporating features such as voice commands, adjustable text sizes, and compatibility with screen readers. This inclusive approach ensures that all users can benefit from the application's functionalities.

Continuous Improvement:
User-centric design is an ongoing process. The application includes mechanisms for users to provide feedback directly through the app. This feedback is regularly reviewed and used to make iterative improvements to the application. Regular updates ensure that the app evolves in response to user needs and technological advancements, maintaining a high level of user satisfaction.

3.7.1 Human-Centric Design

Human-centric design is a fundamental approach in developing the parking application for Colombo, focusing on creating a system that prioritizes the needs, behaviors, and experiences of its users. This approach goes beyond mere functionality, aiming to create an intuitive and empathetic experience that resonates with users on a human level.

Empathy and User Understanding:
The design process begins with empathy, understanding the users' lives, challenges, and expectations. Through interviews, surveys, and user observation, we gather deep insights into the daily experiences of drivers, parking lot operators, and city administrators. This empathetic understanding informs every design decision, ensuring the application addresses the real, human needs of its users.

Inclusive and Accessible Design:
Human-centric design is inherently inclusive, ensuring that the application is accessible to all users, regardless of their physical abilities, technical expertise, or socioeconomic status. The application features adjustable text sizes, high-contrast modes, and compatibility with assistive technologies like screen readers and voice commands. This inclusivity ensures that everyone can benefit from the application's features.

Intuitive and Simple Interface:
The application's interface is designed to be intuitive and simple, minimizing the cognitive load on users. Clear, concise instructions, recognizable icons, and logical navigation paths ensure that users can easily understand and interact with the application. By reducing complexity, the application becomes more approachable and user-friendly.

Emotional Connection:
Human-centric design also focuses on creating an emotional connection with users. The application employs friendly, supportive language and positive reinforcement to enhance the user experience. Features like real-time parking availability updates and personalized notifications are designed to reduce stress and provide reassurance, making the user feel valued and supported.

Iterative Feedback and Improvement:
The design process is iterative, involving continuous feedback from users. Regular usability testing sessions and feedback loops allow users to share their experiences and suggest improvements. This ongoing dialogue ensures that the application evolves in line with user needs and preferences, fostering a sense of ownership and loyalty among users.

Ethical Considerations:
Human-centric design incorporates ethical considerations, ensuring that the application respects user privacy and data security. Transparent data policies and robust security measures protect user information, building trust and confidence in the application.

3.8 Integrating Agile and RUP Methodology for the Parking App Project

To develop the parking application for Colombo, I will integrate Agile development practices with the Rational Unified Process (RUP). This hybrid approach will allow for flexibility and continuous improvement while maintaining a structured, phase-driven development process. Over the course of six months, I will ensure that the project adapts to changes, meets user needs, and adheres to high standards of quality and documentation.

Agile Development Practices

Agile methodology is ideal for this project due to its emphasis on iterative development, continuous feedback, and adaptability. The key Agile practices I will implement include:

Sprint Planning and Reviews:

Sprint Planning:

At the start of each two-week sprint, I will define clear goals and tasks. This ensures focused progress and timely adjustments based on stakeholder feedback.
Sprint Reviews: At the end of each sprint, I will review the outcomes, gather feedback from stakeholders, and refine the approach for the next sprint.

Daily Stand-Ups:

Although I am working individually, I will conduct brief daily reviews to assess progress, identify challenges, and plan the day's work. This self-check ensures transparency, quick problem-solving, and keeps the project on track.

User Stories and Backlog Grooming:

User Stories:

I will create user-focused narratives to drive the development process, ensuring the app remains user-centric.

Backlog Grooming:

Regular refinement of the project backlog will prioritize essential features and tasks, ensuring the most valuable functionalities are developed first.

Rational Unified Process (RUP)

- RUP will provide a disciplined, phase-driven approach that complements Agile's flexibility. By dividing the project lifecycle into four distinct phases, I can manage complexity and ensure thorough documentation and risk assessment at every stage.

Inception Phase (Month 1):

Define the project's scope, business case, and boundaries.
Identify initial risks and secure stakeholder buy-in.
Deliverables: Project charter, initial requirements document, high-level project plan.

Elaboration Phase (Month 2):

Conduct a deeper analysis of requirements and system architecture.

Address high-risk elements through detailed design and prototype reviews.
Deliverables: Detailed requirements document, architecture design, prototype.

Construction Phase (Months 3-5):

Focus on developing the application through iterative cycles.
Integrate and test features as they are developed, aligning with architectural guidelines.
Deliverables: Incremental builds of the application, integrated features, tested modules.

Transition Phase (Month 6):

Transition the application from development to production.
Conduct final user testing, provide training, and ensure deployment support.
Deliverables: Final version of the application, user manuals, training materials, deployment plan.

Combining Agile and RUP for the Parking App Project

By combining Agile and RUP, I will ensure the project remains adaptable to changes while benefiting from structured planning and risk management. The project's timeline integrates both methodologies to maximize their strengths.

Early Sprints (Months 1-2):

Focus on developing core functionalities such as user registration, payment integration, and real-time updates.
Conduct detailed design and prototype reviews during the Elaboration phase to address high-risk elements.

Middle Sprints (Months 3-4):

Develop and integrate features iteratively, ensuring thorough testing and validation.
Regular feedback sessions with stakeholders to ensure alignment with user needs.

Late Sprints (Months 5-6):

Finalize features and conduct extensive user testing.
Prepare for deployment, including user training and documentation.
Key Agile Features in Each Phase

Inception Phase:

Initial sprint planning to define user stories and backlog.
Daily self-reviews to ensure alignment and address early challenges.

Elaboration Phase:

Sprints focused on detailed design and prototype development.
Regular sprint reviews to gather feedback and refine requirements.

Construction Phase:

Iterative development cycles with continuous integration and testing.
User stories drive development, ensuring user-centric features.

Transition Phase:

Sprints focused on final testing, user training, and deployment preparation.
Final sprint reviews to ensure all requirements are met and the application is ready for production.

Chapter 04 System Analysis and Design

4.1 System Analysis And Design

4.1.1 functional requirement

Priority	
Must-Have	• **User Registration:** A secure registration process for users that collectsessential information and creates a unique profile for each user. • **Vehicle Registration:** Capability for users to register multiple vehiclesunder their profile, with vehicle-specific information that aids in identification. • **Payments Integration:** A reliable digital payment system to facilitatecashless transactions with features such as QR code scanning for verifying payments. • **Employee Registration:** A system for parking employees and privateparking owners to register and manage their parking spaces. • **Admin Panel:** An administrative backend for overarching control, including user and employee management, transaction oversight, andsystem settings configuration.

Should-Have	• **Real-Time Parking Availability Updates:** A live update feature thatreflects current parking space occupancy, managed by parking employees. • **Automated Booking System:** An interface for users to book availableparking spaces in real-time.
Could-Have	• **Feedback System:** An optional feedback mechanism for users to ratetheir parking experience and for administrators to collect data on usersatisfaction. • **Revenue Analytics:** Advanced analytics for administrators to track theapp's revenue streams and understand financial metrics.
Won't-Have	• **Integration with Existing Municipal Systems:** Linking the app directlyto city traffic management systems is not planned for the initial roll-out. • **Complex IoT Integration:** IoT-based parking management, requiringextensive hardware like sensors or cameras, is not included due to budget and infrastructure considerations

Table 4 functional requirement

4.1.2 non-functional requirement

Priority	Requirement
Must-Have	• **Performance:** The system must handle multiple user requests simultaneously without lag, providing a response time of no more than two seconds for all actions. • **Reliability:** The system must have an uptime of 99.9%, ensuring availability during peak traffic hours. • **Security:** Protection of user data, including financial information and personal details, with compliance to data protection regulations.

	• **Usability:** An intuitive UI/UX for the mobile application, catering to non-technical users as well. • High system performance with fast loading times and minimal downtime. (SLA 99.9%)
Should-Have	• **Scalability:** The app should be scalable to accommodate the growing number of users and parking spaces without performance degradation. • **Compatibility:** The app should be compatible with both iOS and Android platforms and support various payment methods.
Could-Have	• **Localization:** Multi-language support, considering the linguistic diversity of Colombo. • **Customer Support Integration:** A chat or help desk feature for real-time assistance.

Table 5 non-functional requirement

4.2 UML Diagram

4.2.1 Use case diagram

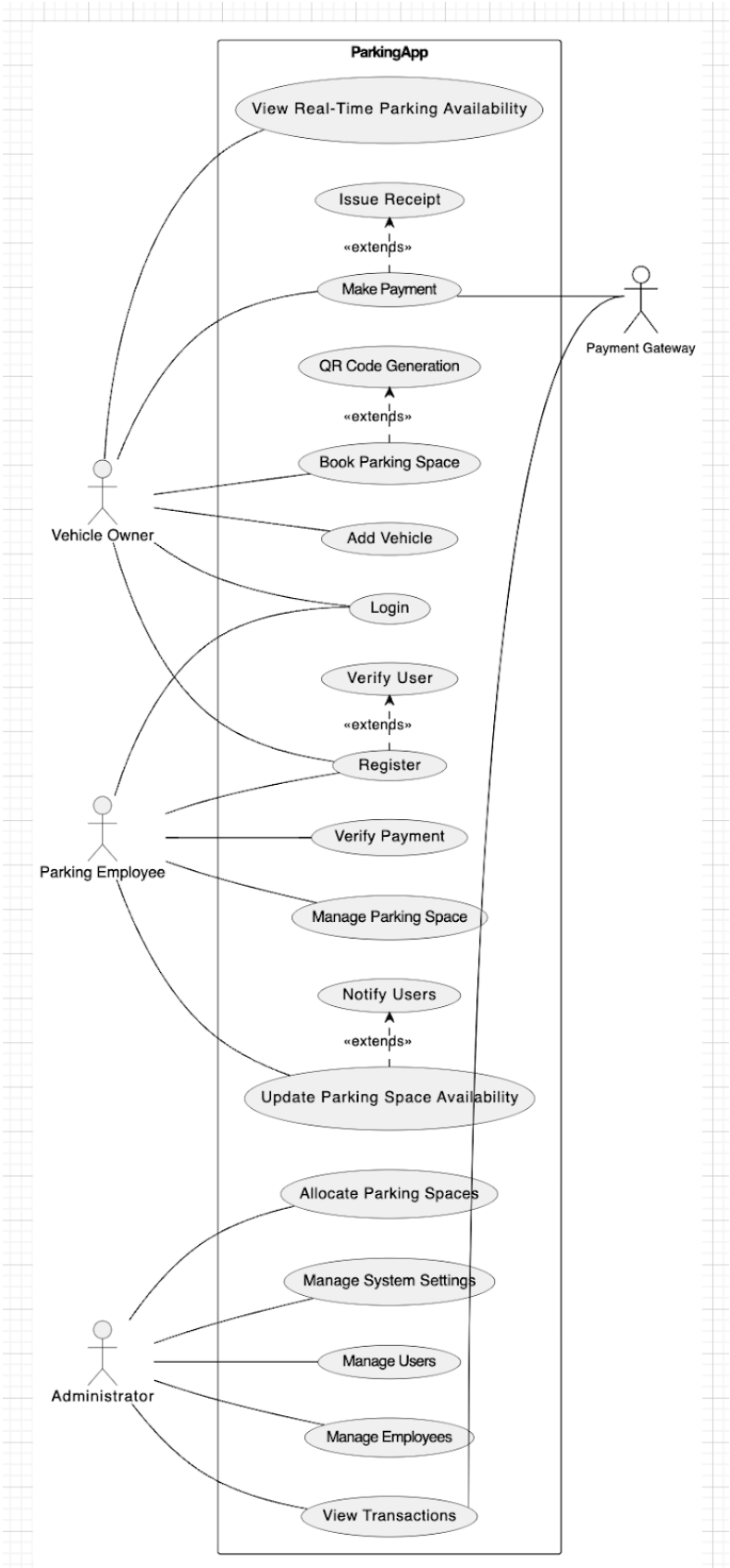


Figure 4 Use case diagram

- This use case diagram outlines the interactions and key functionalities of the Parking App, identifying the roles of various actors involved: Vehicle Owner, Parking Employee, Administrator, and an external Payment Gateway. The Vehicle Owner can view real-time parking availability, register and log in to the app, add vehicle details, book parking spaces, generate QR codes for booking verification, make payments, and receive receipts. The Parking Employee is responsible for verifying users and payments, managing parking spaces by updating availability, and notifying users about space availability. The Administrator oversees the entire system by allocating parking spaces, managing system settings, users, and employees, and viewing transaction details. The external Payment Gateway facilitates the processing of payments. This diagram ensures a comprehensive understanding of the system's operations, enhancing the efficiency and effectiveness of parking management in Colombo.

4.2.2 Activity diagram

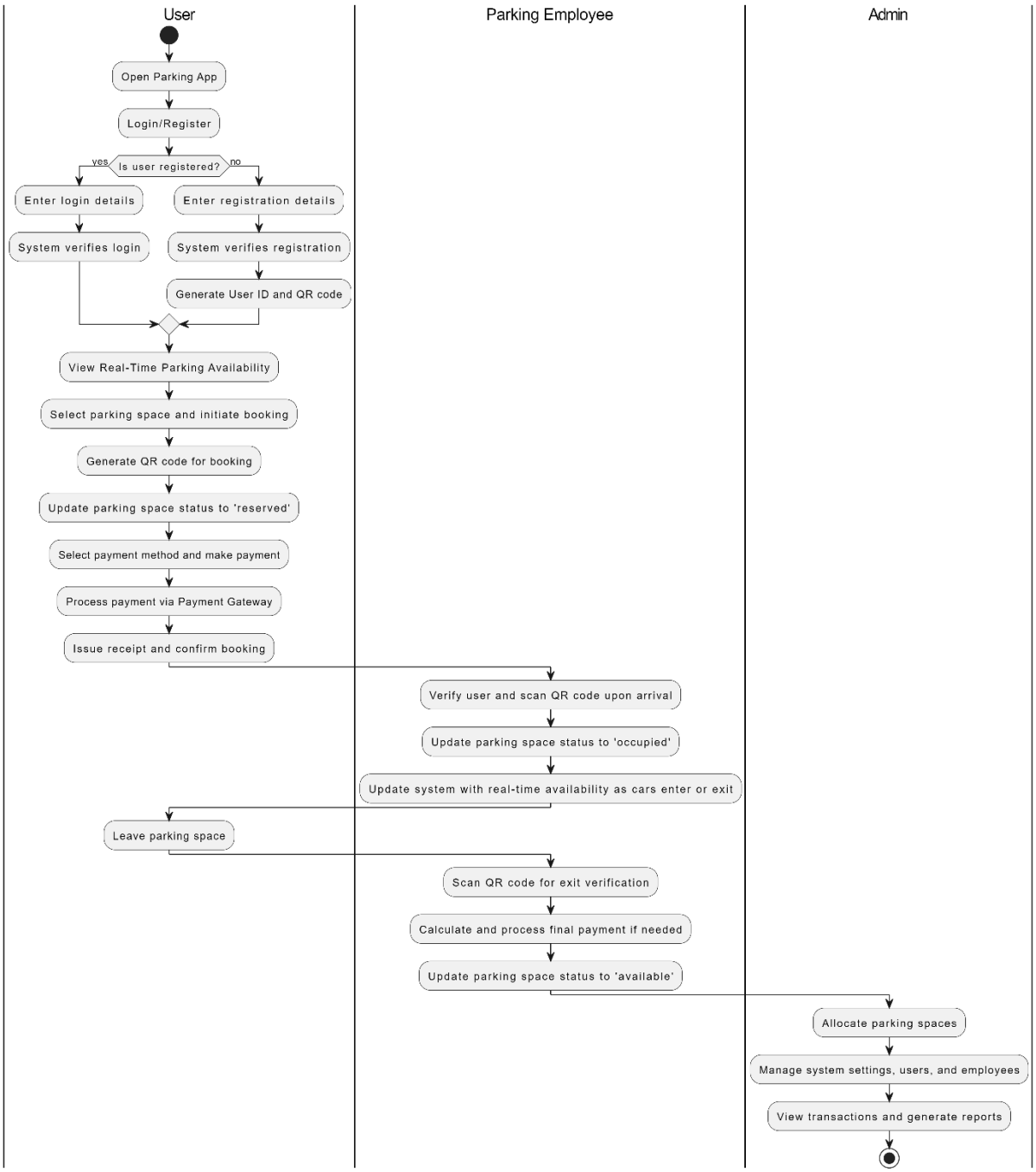


Figure 5 Activity diagram

- This activity diagram outlines the interactions within the parking app among Users, Parking Employees, and Admins. Users start by logging in or registering, then view real-time parking availability, select a parking space, make a booking, and complete the payment process. Parking Employees verify user entries by scanning QR codes, update

parking space statuses, and handle real-time availability as cars enter and exit. Admins manage overarching system settings, allocate parking spaces, oversee users and employees, and generate reports on transactions, ensuring the system operates smoothly and efficiently.

4.2.3 sequence diagram

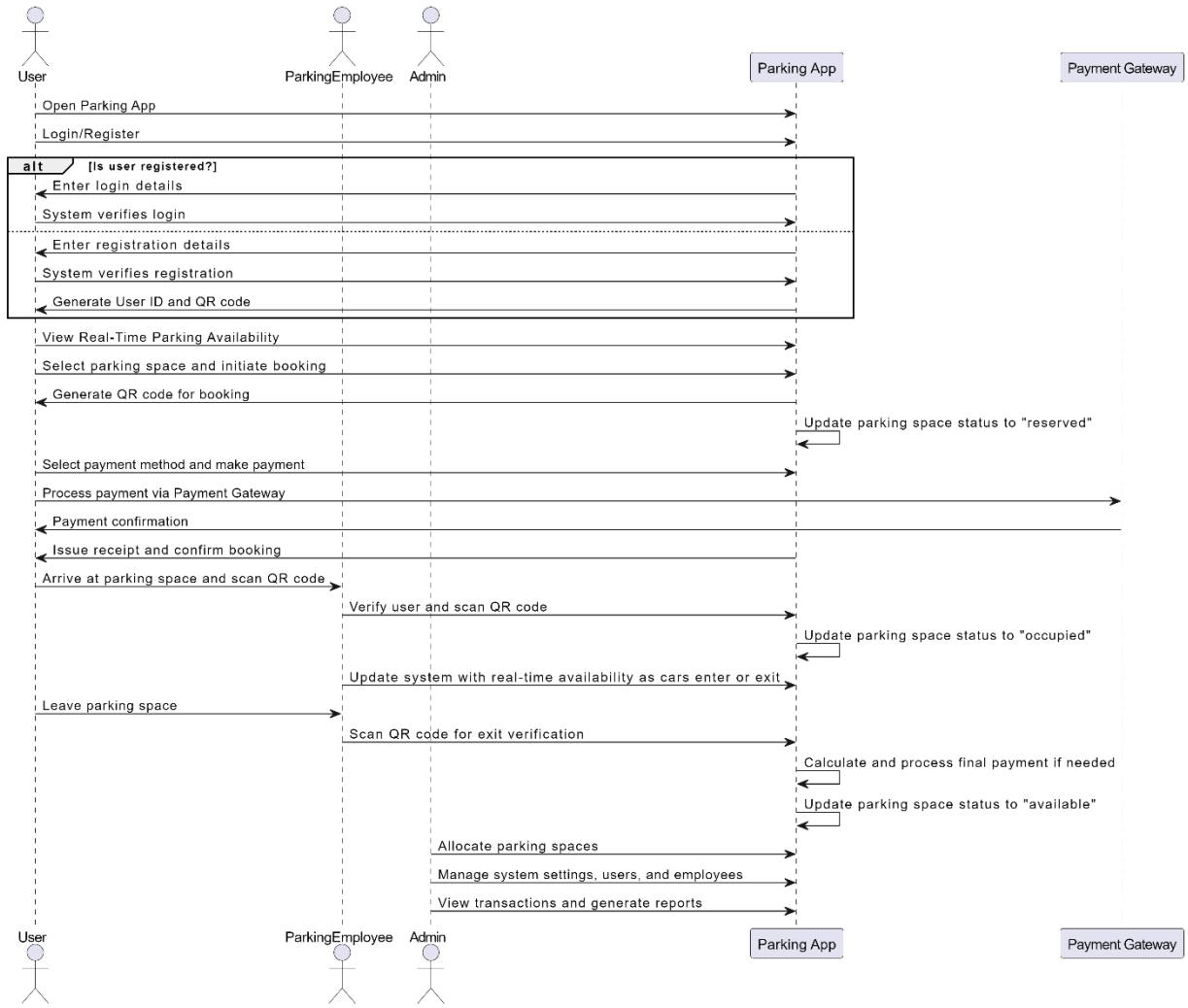


Figure 6 sequence diagram

The sequence diagram illustrates the interactions between a User, Parking Employee, Admin, Parking App, and Payment Gateway in a parking management system. The User initiates the process by opening the Parking App and either logging in or registering. Once logged in, the User views real-time parking availability, selects a parking space, and generates a QR code for booking. The User makes a payment, processed via the Payment Gateway, and receives a booking confirmation. Upon arrival at the parking space, the Parking Employee verifies the User's QR code and updates the system to reflect the parking space status. When leaving, the

User's QR code is scanned again, and the system updates the parking status to available. The Admin manages system settings, users, employees, and transactions, ensuring smooth operation and oversight.

4.2.3 class diagram

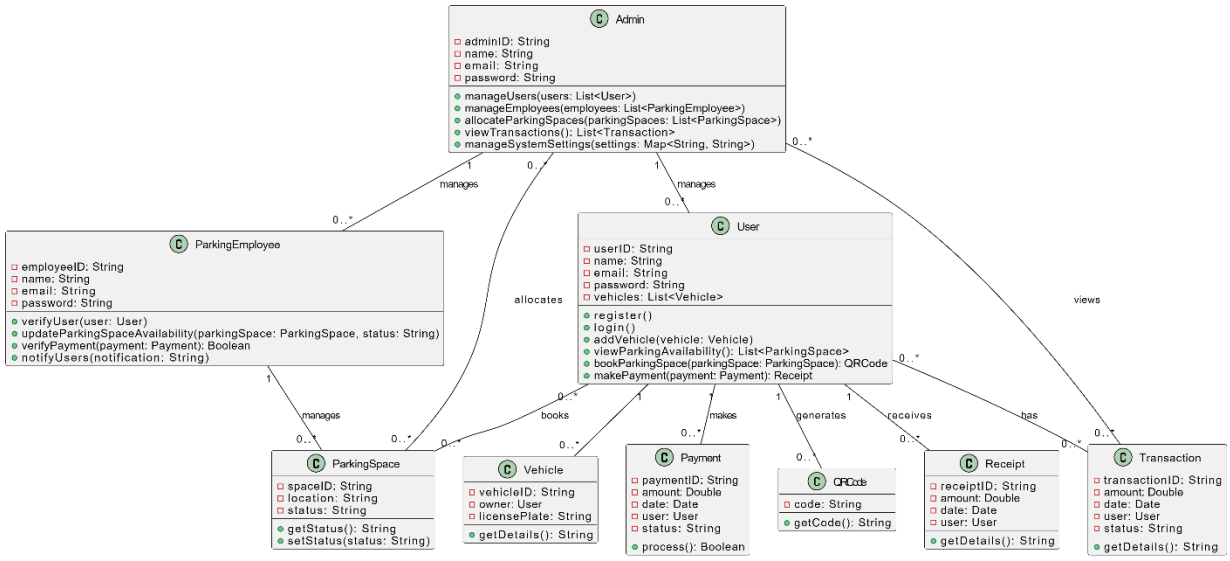


Figure 7 class diagram

- This class diagram illustrates the structure of the Parking Hub project, depicting relationships between key entities: User, ParkingEmployee, Admin, ParkingSpace, Vehicle, Payment, QRCode, Receipt, and Transaction. Users can register, login, view parking availability, book spaces, and make payments. ParkingEmployees manage parking spaces, verify users and payments, update availability, and notify users. Admins oversee system settings, manage users and employees, allocate spaces, and monitor transactions. The diagram highlights the interactions and responsibilities of each class, showcasing a cohesive system for efficient parking management.

4.3 System high level architecture

4.3.1 Data Inputs

User Input:

Registration and Login:
Users provide their personal details (name, email, password) when registering. During login, they enter their phone number and Enter the verification code.

Booking Details:
When booking a parking space, users select their desired location and input booking specifics.
Payment Information:
Users input their credit or debit card details to complete the payment process for their parking reservation.

Parking Employee Input:

Registration and Login:
Parking employees enter their personal information and details about their managed parking spaces during registration. They use their email and password to log in.

Parking Space Updates:
Employees regularly update the availability status of parking spaces as vehicles enter and exit.

Admin Input:

System Settings:
Admins configure and update system settings, such as parking rates and operational hours.

User and Employee Management:
Admins input and manage data related to users and parking employees, including approvals and role assignments.

4.3.2 Data Processing

User Authentication:

Verification:
Upon registration and login, the system verifies the entered credentials. If the user is new, the system generates a unique user ID and QR code.

Booking Management:
When a user selects a parking space, the system checks real-time availability, updates the status to 'reserved', and generates a QR code for the booking.

Payment Processing:

Transaction Handling:
The system processes payments through an integrated payment gateway, ensuring secure transactions.

Receipt Generation:
After successful payment, the system generates a digital receipt for the user.

Employee Management:

User Verification:
On-site, parking employees verify users by scanning the QR code and confirming the booking.

Availability Updates:
Employees update the system with the status of parking spaces (occupied or available) as vehicles enter and leave.

Administrative Controls:

System Monitoring:
Admins monitor system performance and user activities, ensuring smooth operations.

Data Analytics:
The system processes input data to provide insightful analytics and reports for decision-making.

4.3.3 Data Output

User Output:

Confirmation and QR Code:

Users receive a booking confirmation along with a QR code for their reservation.
Real-Time Updates: Users can view real-time availability of parking spaces and their booking status.

Digital Receipts:

After payment, users receive digital receipts for their transactions.

Parking Employee Output:

Verification Details:

Employees get real-time updates about user bookings and availability statuses, allowing them to manage the parking space effectively.

System Updates:

Employees update the system on the status of parking spaces, which reflects in real-time for users and admins.

Admin Output:

Reports and Analytics:

Admins have access to detailed reports and analytics on system usage, transactions, and user activities.

Management Dashboards:

Admin dashboards display real-time data and insights, aiding in effective management and decision-making.

4.4 System high level architecture

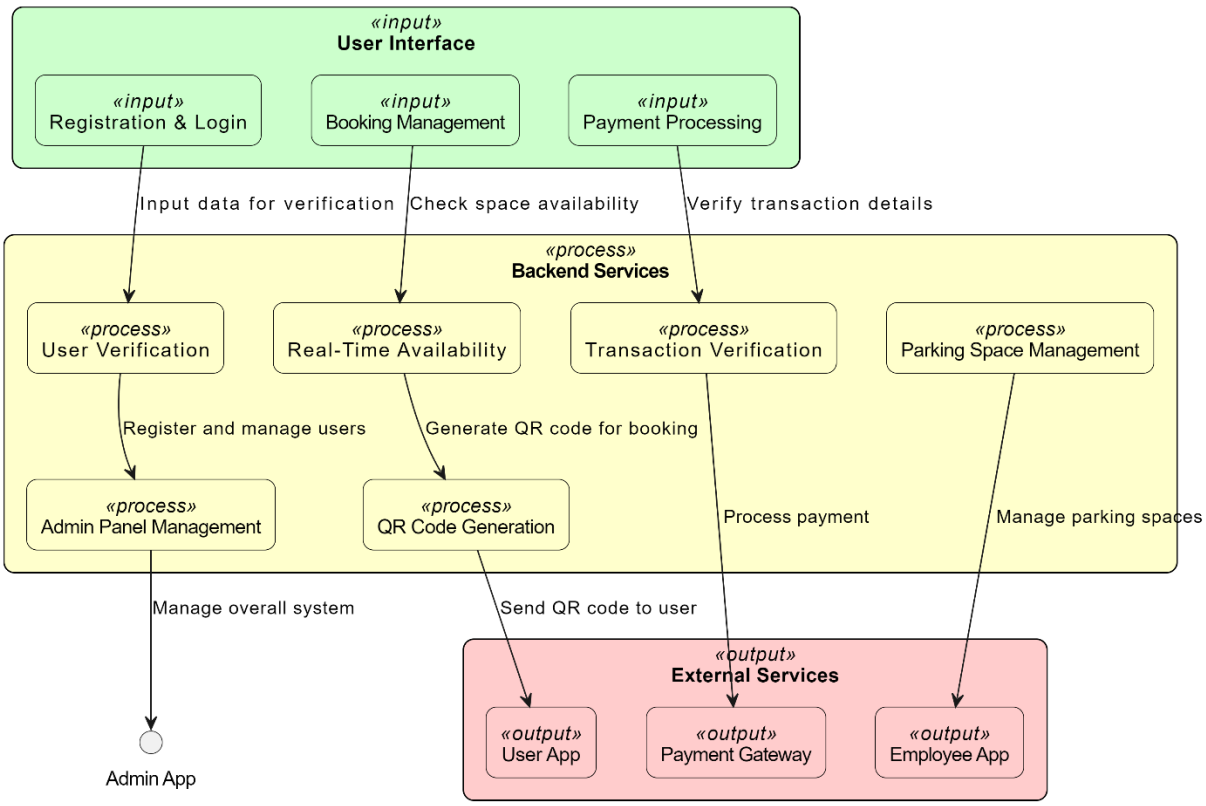


Figure 8 System high level architecture

- This high-level architecture diagram shows the parking app's structure. The User Interface includes Registration & Login, Booking Management, and Payment Processing. These inputs interact with Backend Services like User Verification, Real-Time Availability, Transaction Verification, QR Code Generation, Parking Space Management, and Admin Panel Management. The External Services layer outputs to the User App, Payment Gateway, and Employee App, ensuring smooth system integration and functionality. The admin app oversees overall system management.

4.5 Method Of Data Gathering

To ensure the Parking Hub Project is grounded in accurate and relevant data, a multi-faceted data-gathering approach will be employed. Online surveys via Google Forms will be conducted to capture perspectives from drivers, residents, business owners, and visitors in Colombo. These surveys will gather information about parking behaviors, challenges, payment preferences, and experiences with parking scams. Additionally, a thorough review of existing literature on Colombo's traffic and parking issues will be undertaken. This includes studying research papers, government reports, and academic articles that provide insights into traffic congestion patterns, parking space utilization, environmental impacts, and urban planning policies.

Field observations and primary research will further enrich the data collection process. On-site visits to various parking facilities will be conducted to assess space utilization and management practices. Interviews with key stakeholders, such as parking lot owners, city planners, and traffic police, will offer qualitative insights into practical challenges and potential solutions for parking in Colombo. As an individual undertaking this project, I will systematically compile and integrate data from these sources to identify key trends and insights. Data visualization tools will be used to create maps and charts illustrating parking demand and problem areas, ensuring the project remains responsive to emerging insights. This comprehensive approach aims to develop a robust and well-informed solution to Colombo's parking challenges, tailored to effectively meet user needs.

4.5.1 Sample of data gathering

To develop a user-centric parking application for Colombo, I employed a comprehensive data gathering strategy. As this is an individual project, I personally conducted each step to ensure the accuracy and relevance of the data collected. The methods included reading research papers, conducting interviews with Uber and PickMe drivers, and distributing surveys.

Literature Review
I began by reviewing several research papers on parking management systems, urban traffic congestion, and digital payment solutions. These academic sources provided valuable insights into the current technologies and methodologies used in parking systems globally. The literature review helped identify gaps in existing solutions and guided the feature set for the proposed application.

Interviews with Drivers
Understanding the practical challenges faced by daily commuters was crucial. I interviewed several Uber and PickMe drivers, gathering firsthand information about their experiences with parking in Colombo. These discussions highlighted common issues such as the lack of available parking spaces, the inconvenience of cash payments, and the frequent encounters with parking scams. The drivers also shared their preferences for potential solutions, which informed the design of user-friendly features in the app.

Surveys

To reach a broader audience, I designed and distributed a survey using Google Forms. The survey included questions on parking habits, problems encountered, and desired features in a parking app. It was shared via social media and community networks to ensure a diverse range of responses from different user demographics. The survey data provided quantitative insights that complemented the qualitative data from the interviews and literature review.

By integrating information from research papers, driver interviews, and survey responses, I was able to gather comprehensive data to inform the development of the parking application. This multi-method approach ensured that the application would effectively address the real-world challenges and needs of its users in Colombo.

4.6 Beginning of the project

The initiation of this project stemmed from a keen observation of the pervasive parking issues in Colombo, Sri Lanka's bustling capital city. As someone who frequently navigates the city, I experienced firsthand the challenges of finding parking spaces, dealing with cash payments, and encountering fraudulent parking attendants. Recognizing these issues, I saw an opportunity to leverage technology to create a more efficient and user-friendly parking solution.

Identifying the Problem
The first step involved identifying and clearly defining the problem. I conducted informal discussions with fellow city dwellers, including drivers and business owners, to understand their parking-related frustrations. Their feedback confirmed that the lack of available parking spaces, inefficient manual processes, and security concerns were significant pain points. These insights formed the foundation of my problem statement.

Preliminary Research
Next, I delved into preliminary research to explore existing solutions and technologies. I reviewed various academic papers on urban parking management, digital payment systems, and mobile application development. This research helped me understand the current state of parking solutions, both locally and globally, and identify gaps that my project could address. It also provided a theoretical framework that would guide the project's development.

Setting Objectives
With a clear understanding of the problem and potential solutions, I defined the project objectives. The primary goal was to develop a cloud-based parking application that would streamline the entire parking process for users in Colombo. Key objectives included reducing traffic congestion, eliminating fraudulent activities, and providing a secure, convenient way to book and pay for parking. These objectives were aligned with the insights gathered from my initial research and discussions.

Planning and Design
After setting the objectives, I moved on to planning and design. I outlined the key features of the application, such as real-time parking availability, digital payments, and a user-friendly interface. I also created initial sketches and wireframes to visualize the app's functionality and user flow.

This stage involved a lot of brainstorming and iterative design, ensuring that the solution would be both effective and user-friendly.

Starting this project involved a combination of identifying a pressing urban issue, conducting thorough preliminary research, and meticulously planning the application's features and objectives. This structured approach laid a solid foundation for the development phase, ensuring that the project would address real-world needs and offer tangible benefits to its users.

4.7 Primary Data Gathering

To gather primary data for the project, I conducted a comprehensive survey targeting various stakeholders, including everyday drivers and rideshare operators like Uber and PickMe drivers. The survey was designed using Google Forms and distributed through online platforms and social media to reach a broad audience. Key questions focused on their daily parking challenges, preferences for parking solutions, and experiences with current parking systems. Additionally, I conducted informal interviews with Uber and PickMe drivers, as they frequently navigate the city's parking landscape and could provide valuable insights into the most problematic areas and peak times for parking difficulties. This direct engagement with the target users ensured that the data collected was relevant, accurate, and reflective of the real-world issues faced by those who would ultimately benefit from the proposed parking solution.

4.8 Secondary Data Gathering

In addition to primary data collection, I engaged in secondary data gathering by reviewing relevant research papers and academic articles. These sources provided a broader context and deeper understanding of urban parking challenges and the effectiveness of various parking management solutions implemented in other cities. By analyzing existing literature, I could identify best practices, technological trends, and potential pitfalls to avoid. This secondary research helped to ground my project in established knowledge and ensured that my approach was informed by both practical insights from local stakeholders and theoretical frameworks from the academic community.

4.9 System design

4.9.1 System user interfaces

4.9.1.1 User interfaces

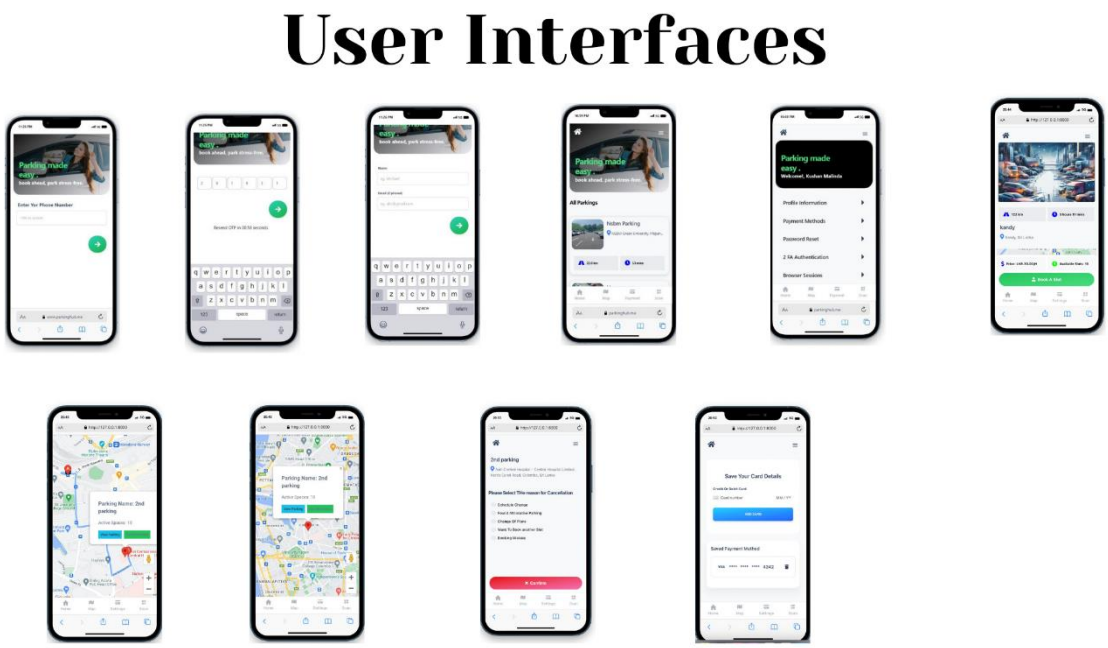


Figure 9 User interfaces

The user interfaces shown in the image depict various screens of the parking application, designed for ease of use and seamless navigation. Starting from user registration and OTP verification, the interfaces guide users through profile setup, parking space search, and booking processes. Users can view real-time parking availability on a map, select parking spots, and manage payments securely. Additional features include profile management, payment method addition, and transaction history review. The clean and intuitive design ensures that users can efficiently perform all necessary tasks related to parking without any hassle.

4.9.1.2 Owner interfaces

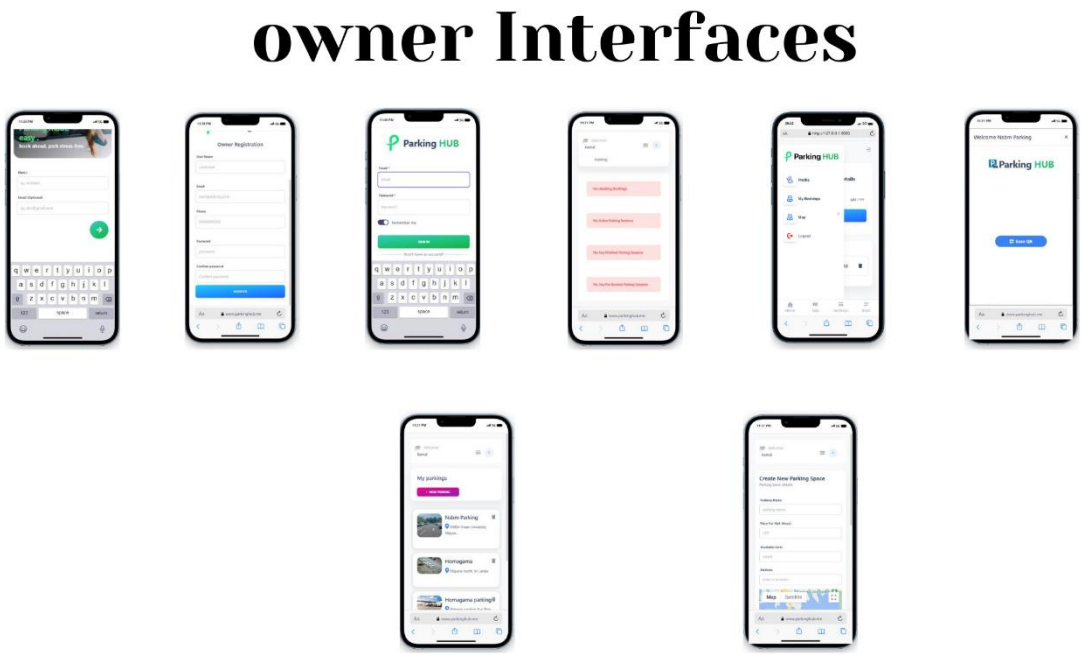


Figure 10 Owner interfaces

The Owner Interfaces for the Parking Hub app provide a comprehensive and user-friendly experience for parking space owners. The set of screens includes an initial login screen, a registration form for new owners, and an easy-to-navigate dashboard for managing parking spaces. The interface allows owners to add and update details about their parking spots, view a list of their properties, and create new parking listings complete with location data and availability. Essential actions like logging in and submitting information are streamlined through simple, clear forms and buttons, ensuring that owners can efficiently manage their parking facilities from anywhere.

4.9.1.3 Admin interfaces

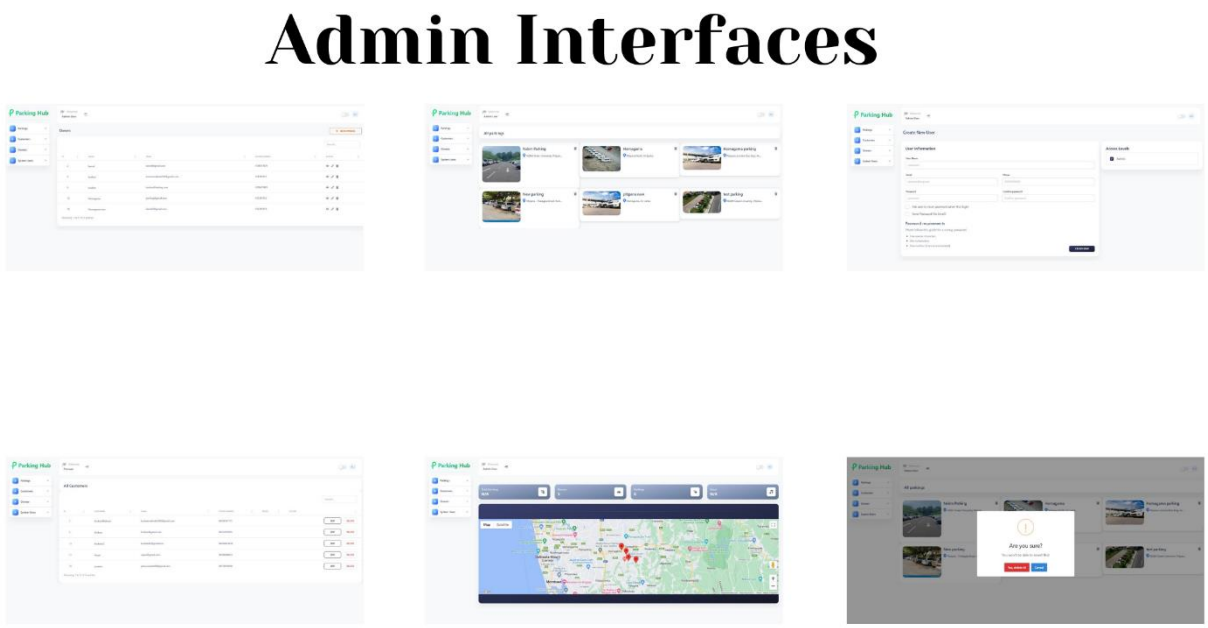


Figure 11 Admin interfaces

The Admin Interfaces for the Parking Hub app offer a robust and detailed dashboard for managing the entire parking system. These interfaces include a variety of functionalities that allow admins to oversee and interact with both parking spaces and user accounts effectively. Key features visible in the interfaces include a comprehensive list view of all parking spaces, a photo gallery for parking area verification, detailed forms for editing parking space information, and a map integration for geolocation tracking. Additionally, there are options for directly messaging users, which facilitates immediate communication and issue resolution. This design ensures that admins have all the necessary tools to manage operations smoothly and maintain a high standard of service across the platform.

4.9.2 Color selection

- The color scheme of the Parking Hub app has been meticulously chosen to enhance user interaction and aesthetic appeal while ensuring functionality and accessibility. The use of green highlights important actions like booking slots, encouraging user participation with its connotations of go-ahead and safety. Blue, symbolizing trust and reliability, is strategically employed for interactive elements such as links and direction buttons, helping to instill confidence in users as they navigate the app. Neutral backgrounds in white and light grey create a clean, uncluttered space that aids in readability and minimizes visual fatigue, making the app user-friendly for longer periods. Accents of red are used for critical actions like cancellation, drawing attention quickly and effectively due to its association with alerts and stopping. Text in black and dark grey ensures high readability against lighter backgrounds. This thoughtful color palette not only enriches

the visual experience but also supports the usability of the app, making it an intuitive and efficient tool for managing parking needs.

4.9.3 Heuristic Evaluation of User Interface

Heuristic evaluation is a method used to identify usability problems in a user interface (UI) by assessing it against recognized usability principles. For the Parking Hub project, a selection of 3-5 evaluators with usability expertise is made to independently review the interface using Nielsen's ten usability heuristics. These evaluators check aspects such as system status visibility, the match between the system and the real world, and user control and freedom, among others. After the evaluation, all evaluators convene to discuss their findings, each presenting the issues identified, the heuristic violated, and suggested improvements.

Once the findings are consolidated, issues are prioritized based on their severity and impact on user experience, focusing on addressing critical problems immediately. A detailed report of all usability issues and proposed recommendations is then prepared and presented to the development team for implementation. Following the implementation of changes, a re-evaluation is conducted to ensure that the modifications have effectively resolved the issues without introducing new problems, maintaining a cycle of continuous improvement to enhance the usability and user satisfaction of the app.

4.9.4 User Interface Evaluation using Schneiderman’s Eight Golden Rules

1. Strive for Consistency
   - Consistent sequences of actions should be maintained in similar situations; consistent commands and terminology should be used throughout the app. This reduces the learning curve and confusion, enhancing the user's confidence in using the app.

2. Enable Frequent Users to Use Shortcuts
   - As the frequency of use increases, so should the user's ability to perform tasks quickly. For the Parking Hub, implementing keyboard shortcuts, favorite parking spots, and customizable dashboard features can cater to power users.

3. Offer Informative Feedback
   - For every operator action, there should be some system feedback. For example, when a parking slot is booked, a clear confirmation message should appear. Feedback must be immediate and informative to keep the user well-informed.

4. Design Dialog to Yield Closure
   - Sequences of actions should be organized into groups with a beginning, middle, and end. The process of booking a parking space, for instance, should guide the user from selection to payment to confirmation, each step clearly marked and acknowledged.

5. Offer Error Prevention and Simple Error Handling

- The system should either prevent errors from occurring or make it easy to correct them when they do happen. For instance, before finalizing a parking booking, the app could provide an overview screen allowing users to review and modify their entries easily.

6. Permit Easy Reversal of Actions

- This feature supports exploratory learning by allowing users to undo actions without fearing permanent consequences. For example, users should be able to cancel a parking reservation with a single tap and without undergoing a complex process.

7. Support Internal Locus of Control

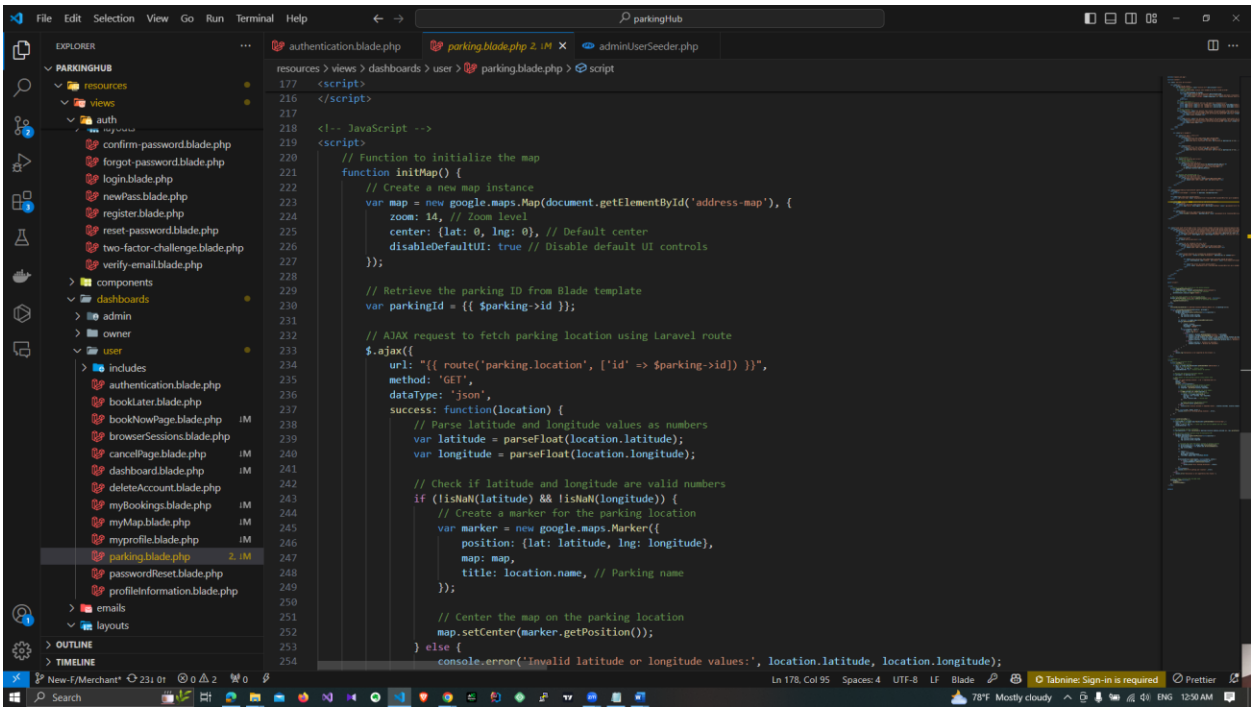
- Users should feel in control of the system and not feel controlled by it. By enabling users to initiate and control actions, the app should empower users, like choosing alternative parking suggestions or modifying booking times as per their convenience.

8. Reduce Short-Term Memory Load

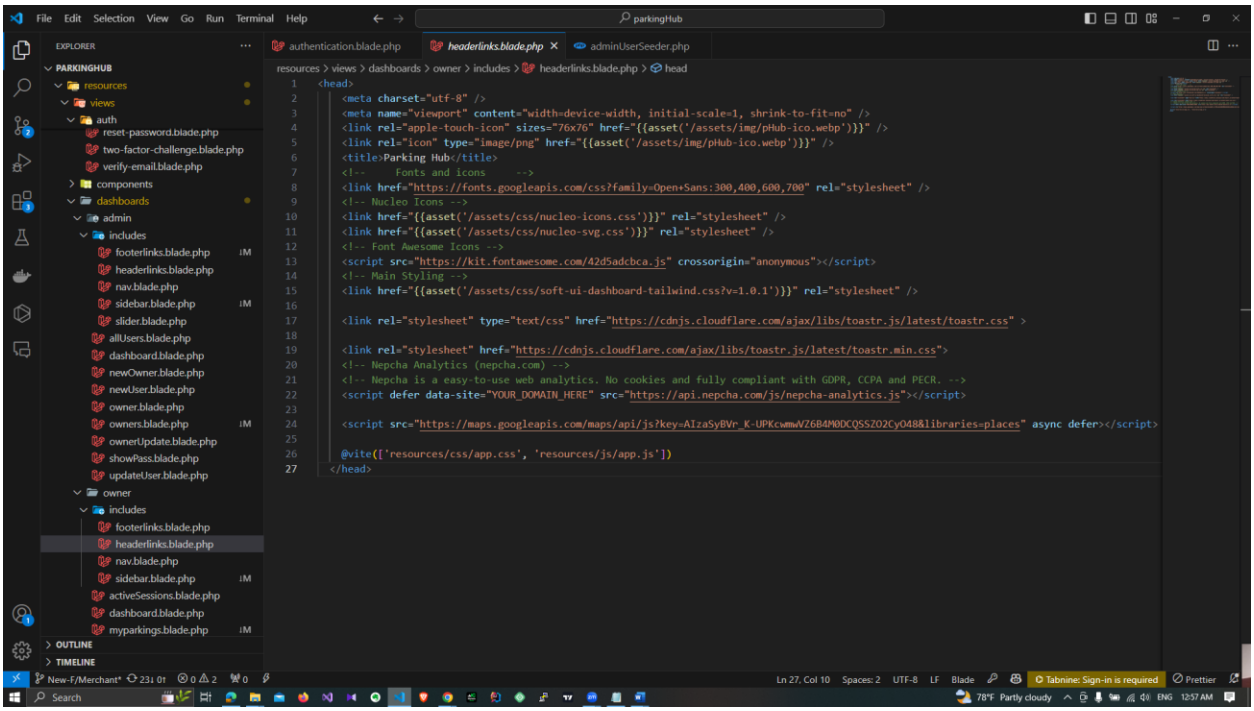
- The human mind can only hold a limited amount of information in short-term memory. Therefore, the app's interface should not require users to remember information from one part of the dialogue to another. Information necessary for a task at hand, such as availability and prices of parking slots, should be visible or easily retrievable when needed.

Chapter 05 System implementation

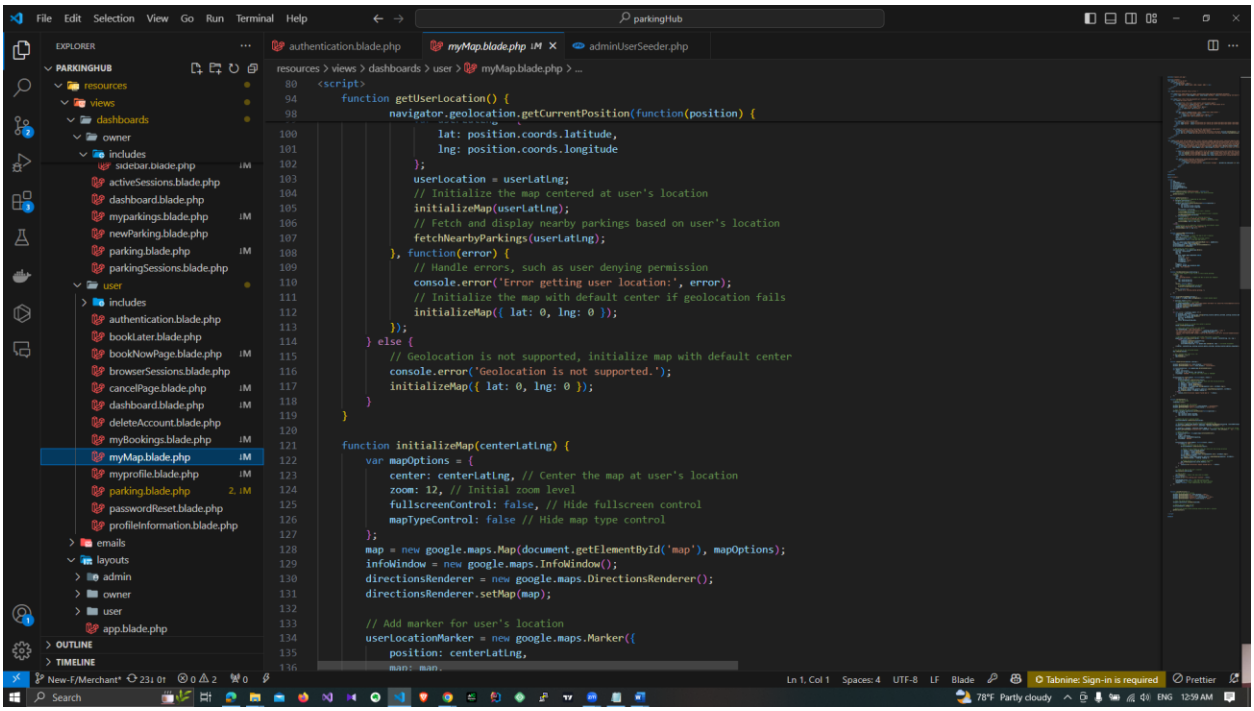
5.1 Screenshot Of The Code



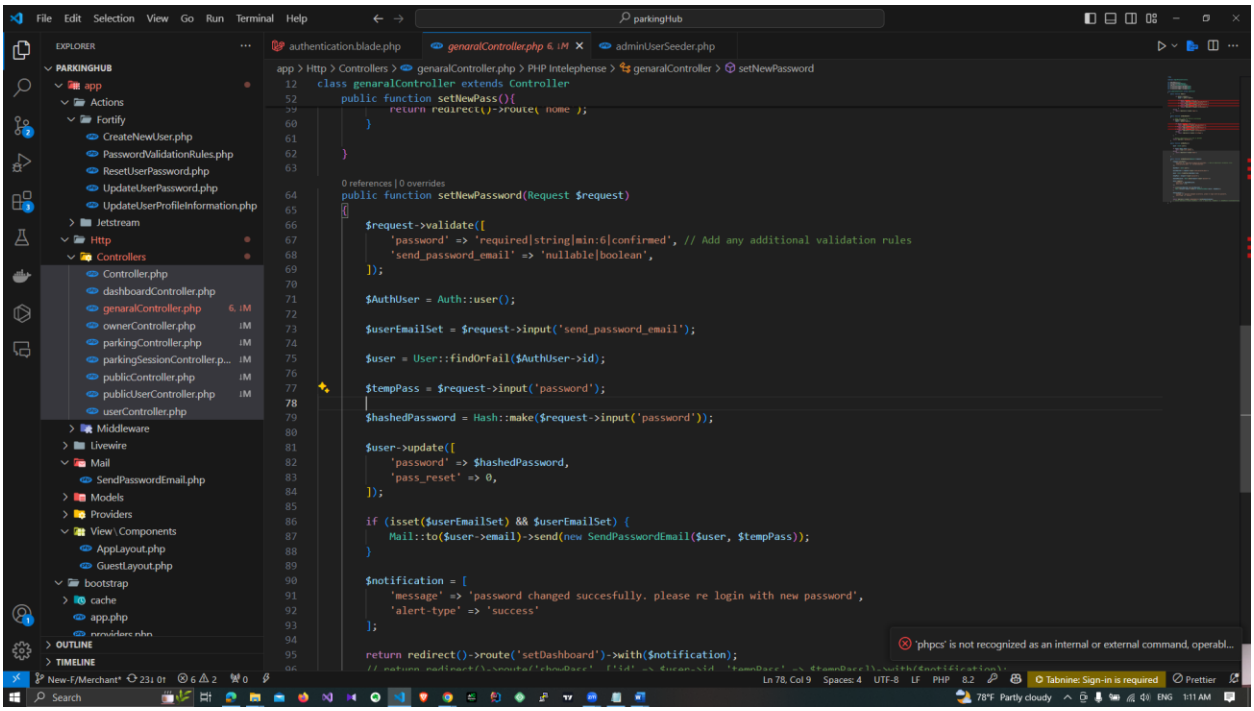
- The provided code is part of a Blade template from a Laravel application designed for the Parking Hub app, which focuses on enhancing user interaction with dynamic parking information and interactive elements. This template extends a base user layout and features a series of functionalities, including a carousel for displaying multiple images of a parking location, dynamically updated content displaying parking details such as distance and estimated time, and Google Maps integration for showing geographical information and calculating routes. Interactive elements within the template allow users to book parking slots immediately or for later times, facilitated through visually appealing buttons and modal forms. The use of AJAX calls for fetching location data and JavaScript functions for map initialization and distance calculations ensures a responsive and informative user experience, highlighting the integration of advanced web technologies to provide real-time, user-centric data and interactive capabilities in the Parking Hub app.



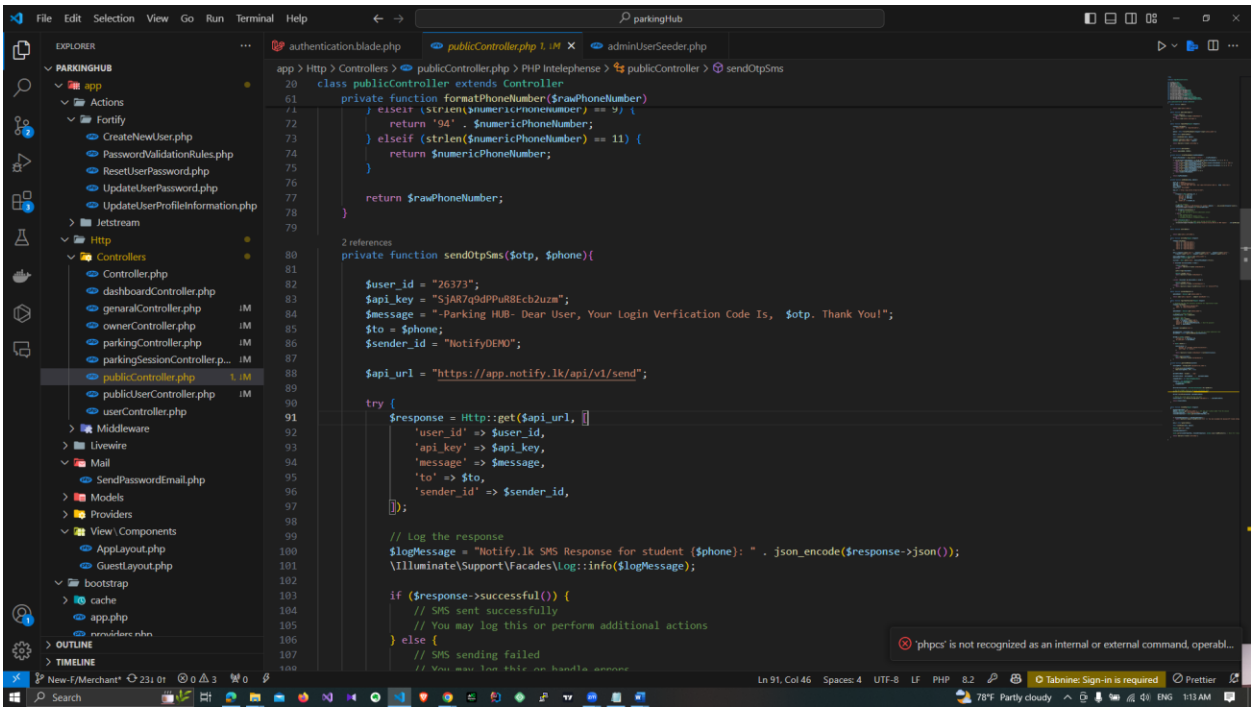
- The HTML `<head>` section of the "Parking Hub" application is meticulously designed to optimize both functionality and aesthetics. It sets character encoding and mobile-responsive settings, includes icons for brand consistency, and specifies the page title. External resources such as Google Fonts for typography, Nucleo and Font Awesome icons for visual elements, and Tailwind CSS for styling are incorporated to enhance the interface. Additionally, the script for Google Maps API facilitates interactive mapping features crucial for the app's navigation functionalities. Privacy-compliant Nepcha Analytics for tracking user interactions and Vite for resource optimization underscore the application’s focus on efficient performance and user privacy. This setup demonstrates a strategic integration of technologies to ensure a smooth and engaging user experience.



- The provided script is part of a web application that integrates Google Maps to display a user's current location and nearby parking options. The script initializes a map centered on the user's position, which is obtained through the browser's geolocation API. Users can interact with the map to view details about nearby parking spaces, including distance and estimated travel time. The script also includes functionality to fetch parking data from a server, display it on the map, and offer navigation directions to a selected parking spot. Additionally, there are interactive elements like a QR code modal and real-time guidance to the chosen parking, enhancing user engagement and providing a comprehensive parking solution. The script handles user interactions for starting navigation, updating positions, and canceling navigation, ensuring a dynamic and responsive user experience.



- The PHP code from the `genaralController` in a Laravel application manages user authentication and redirection based on user roles. Functions within the controller determine user access levels and direct users to appropriate dashboards (Admin, Owner, or User) or to the login page if unauthenticated. It includes methods to handle password resets, validating and updating the user's password in the database, and optionally sending an email with the new password. This controller is pivotal for maintaining secure user sessions and ensuring users are directed to relevant parts of the application according to their roles and authentication status.



- The `publicController` in a Laravel application manages user authentication and registration processes. It handles user logins, OTP (One-Time Password) verification, and the registration of new users with unique QR codes for identification. The controller offers methods for sending OTPs via SMS, verifying them, and creating user accounts with associated QR codes, ensuring secure and efficient user management. Additionally, it includes functionality to resend OTPs, enforcing limits on resend attempts to enhance security. This comprehensive approach streamlines user access and identity verification within the application.

5.2 Packages that I Imported

Spatie:
This package is often employed for its robust tools for handling common backend functionalities such as permissions and activity logging. In the context of a parking app, Spatie helps manage user roles and permissions efficiently, ensuring that access is controlled and data is securely handled. It streamlines the administrative aspects by providing a comprehensive and flexible way to handle authorization without extensive custom coding.

Jetstream:
Integrated into the Laravel framework, Jetstream provides a starting point for new projects with features like login, registration, email verification, two-factor authentication, session management, and optional team management. For a parking app, Jetstream accelerates development by handling most of the boilerplate user management code, allowing the developer to focus on core features. It enhances user experience by ensuring that the authentication process is smooth and secure.

Stripe:
This package is critical for handling online payments securely and efficiently. By integrating Stripe, the parking app can process parking fees, making transactions seamless and straightforward for users. Stripe's comprehensive API supports a wide range of payment options, providing flexibility and ease of use for app customers. This integration not only facilitates financial transactions but also builds trust with users through its reliable and secure payment gateway.

Tailwind CSS:
A utility-first CSS framework used to style the application. Tailwind CSS allows for rapid UI development with its low-level utility classes that can be composed to build complex designs directly in the HTML markup. For the parking app, Tailwind CSS provides the flexibility to customize the appearance of the app without writing extensive custom CSS, making the UI design process more efficient and responsive to various device sizes.

5.3 Variables of the Code Implementation

5.3.1 General Variables:

1. $user - Represents the authenticated user, frequently used to access user-related data and manage session-based actions.
2. $request - An instance of Illuminate\Http\Request, used to handle incoming HTTP requests and retrieve data from forms.
3. $otp - Stores the One-Time Password generated for verification purposes.
4. $phone - Holds the user's phone number, formatted or raw, for OTP verification or communication purposes.
5. $customer - Represents a user in the context of verifying OTP or handling customer-specific data.
6. $phoneNumber - Specifically used to hold a formatted phone number for registration or OTP verification.
7. $sessionPhoneNumber - Retrieved from the session to ensure it's used consistently across requests.
8. $sessionOtp - The OTP stored in the session for comparison during verification.

5.3.2 OTP and SMS Variables:

1. $user_id and $api_key - Used to authenticate with the SMS service provider.
2. $message - Contains the message content sent via SMS.
3. $to - The recipient's phone number for the SMS.
4. $sender_id - Identifier for the sender in the SMS communication.
5. $api_url - URL of the API used for sending SMS.

5.3.3 Parking and QR Code Variables:

1. $parking - Represents a parking entity in various contexts, such as storing or updating parking information.
2. $parkings - A collection of parking data, used when multiple parking records are retrieved.
3. $qrCodeContent and $qrCodePath - Used in QR code generation; content typically includes unique identifiers, and path represents the file location.
4. $validatedData - Holds data validated through the request, ensuring it meets application requirements before use.
5. $imagePath - Used to store the path of an image file, particularly for profile pictures or gallery images related to parking.
6. $randomPassword - Stores a randomly generated password for user registration.

5.3.4 Session and Error Handling Variables:

1. $notification - Commonly used to pass success or error messages back to the user interface after operations.
2. $resendAttemptsKey and $resendAttemptCount - Manage the number of times an OTP can be resent to a user within a specific time frame.
3. $image - Used in loops to handle multiple image files uploaded for parking galleries.

Chapter 06 Testing and Results

6.1 Evaluation

Functionality

- Role-Based Access Control: The application effectively manages different user roles, such as Admin, Owner, and User, directing each to appropriate dashboards. This ensures that users have access only to features pertinent to their roles, enhancing usability and security.
- OTP Verification: The OTP system for user verification appears robust, enhancing security by verifying user identities before granting access. This system also supports SMS functionalities, indicating good integration with external services.
- QR Code Generation: QR code functionality for user identification offers a modern approach to user verification and quick access, adding a layer of convenience and security.

Usability

- User Interface Design: The project utilizes Tailwind CSS, which likely provides a clean, responsive design. However, the effectiveness of the UI in terms of user navigation, clarity, and overall user experience should be assessed through user testing.
- Accessibility: Consider evaluating the accessibility of the application to ensure it is usable by people with a wide range of disabilities. This includes visual, auditory, physical, speech, cognitive, and neurological disabilities.

Performance

- Load Testing: Conducting load testing to understand how the application performs under various loads is crucial, especially for systems that handle multiple user interactions and real-time data processing like parking management systems.
- Response Time: Measuring the response times for various actions within the application, particularly those that involve backend processing like OTP verification and parking searches, will help identify any bottlenecks.

Security

- Data Encryption and Protection: Ensure that sensitive data, especially user passwords and personal information, are encrypted both in transit and at rest. The use of HTTPS, secure cookies, and proper database encryption practices should be verified.
- Vulnerability Assessment: Regular security assessments to identify potential vulnerabilities in the application, including SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF), are recommended.

User Satisfaction

- Feedback Collection: Implementing a system to collect user feedback directly within the application can provide insights into user satisfaction and areas for improvement.
- Surveys and User Testing: Conducting periodic surveys and user testing sessions will help gather qualitative data on user satisfaction and usability, guiding future development.

6.2 Test cases

6.2.1 Admin interface test cases

Test Case ID	Title	Objective	Steps	Expected Result
TC1	Add New User	Verify that the admin can add a new user	1. Navigate to User Management 2. Click "Create New User" 3. Enter valid user details and submit	User is added successfully, confirmation message displayed
TC2	View Parking List	Ensure that the admin can view a list of all parkings	1. Navigate to the Parking List section	All registered parkings are displayed correctly
TC3	Edit Parking Information	Verify that the admin can edit	1. Select a parking to edit	Changes are saved correctly, confirmation

		parking information	2. Modify the details and save changes	message displayed
TC4	Delete User	Ensure the admin can delete a user	1. Navigate to User Management 2. Select a user and initiate the delete action	User is deleted successfully, confirmation message displayed

Table 6 Admin interface test cases

6.2.2 owner interface testcases

Test Case ID	Title	Objective	Steps	Expected Result
TC1	Register as New Owner	Test the registration process for a new parking owner	1. Access the owner registration interface 2. Fill in the required information and submit	Owner account is created, owner is redirected to their dashboard
TC2	Add New Parking Space	Verify that the owner can add a new parking space	1. Navigate to "Create New Parking Space" 2. Enter details for the parking space and submit	New parking is added, visible in the owner's parking list
TC3	Edit Parking Details	Ensure that parking details can be edited	1. Select a parking space to edit 2. Change the details and save	Parking details are updated correctly with a success notification

Table 7 owner interface testcases

6.2.3 User Interface Test Cases

Test Case ID	Title	Objective	Steps	Expected Result
TC1	User Login	Verify the user login process including OTP verification	1. Enter the mobile number and request OTP	User is logged in successfully and redirected to the dashboard

			2. Enter the received OTP and verify	
TC2	View Available Parking	Check if the user can view available parking spots	1. Navigate to the parking list on the user interface	All available parkings are listed with details like distance and price
TC3	Book a Parking Spot	Verify the parking spot booking functionality	1. Select a parking spot 2. Confirm the booking	Parking spot is booked successfully, confirmation is displayed
TC4	Update Profile	Ensure users can update their profile information	1. Go to the profile settings 2. Update information and save changes	Profile is updated successfully with a confirmation message

Table 8 User Interface Test Cases

6.3 Neumann architecture

Control Unit: The Control Unit acts as the brain of the operation in the parking app's server backend. It is responsible for interpreting user commands and directing the flow of operations within the application. This includes managing user inputs such as parking searches, reservations, and cancellations. It ensures that data flows correctly between the application's interface, its database, and integrated services like payment gateways and mapping services. By orchestrating these processes, the Control Unit maintains order and efficiency within the application's operations.

Arithmetic Logic Unit (ALU): The ALU is crucial for processing all data-related tasks within the application. This includes performing calculations needed to determine parking fees, which may vary by time and location. It also manipulates data concerning parking availability as users make reservations or changes. Moreover, the ALU handles logic operations that ensure system integrity and rule enforcement, such as verifying available spaces and user credentials, making it a cornerstone for reliable app functionality.

Memory: In this architecture, Memory serves a dual purpose, storing both the application’s data and its operational instructions. This includes personal data of users, such as registration details and payment information, alongside data on parking spaces like location specifics, availability, and pricing. Memory also holds transactional records and the actual program code that runs the backend logic and interfaces. By keeping this information readily accessible and well-organized, the system ensures smooth and efficient performance.

Input/Output Systems: The Input/Output components are integral to the interaction between the user and the system. Inputs are received primarily through users' mobile devices, incorporating forms, selection inputs, and GPS data for location tracking. Outputs are also directed to these devices, displaying app interfaces, parking options, and route directions, as well as sending notifications via SMS or push notifications. These systems not only facilitate user communication and interaction but also ensure that feedback is timely and relevant, enhancing user experience.

Chapter 07 Future Directions and Conclusion

7.1 Project Conclusion

The development and impending deployment of the Parking App Project in Colombo are poised to make significant strides in addressing the city's critical parking and traffic congestion issues. By integrating real-time parking data, the application promises to substantially reduce the time drivers spend searching for parking spaces, thereby easing traffic flow and enhancing overall urban mobility. This reduction in search times and traffic congestion not only facilitates a smoother commuting experience but also contributes to environmental benefits by lowering vehicle emissions. Furthermore, the app's robust security measures are designed to protect users from prevalent parking scams, ensuring a safer and more reliable parking experience. Overall, this project stands as a key innovation in urban infrastructure management, with the potential to transform Colombo into a more livable, efficient, and sustainable city.

7.1.1 Design conclusion

The design of the Parking App Project in Colombo effectively addresses the multifaceted challenges of parking management and urban congestion. By integrating advanced technological solutions and user-centric design principles, the application not only optimizes parking space utilization but also enhances the overall parking experience for users. With features like real-time availability updates, secure payment options, and intuitive navigation, the app significantly reduces the time and stress associated with finding parking in a bustling metropolis. Moreover, the inclusion of robust security measures safeguards users against scams, instilling trust and reliability. As a result, this innovative application is set to revolutionize urban mobility in Colombo, promoting economic efficiency, reducing environmental impact, and improving daily life for its residents. The successful implementation of this design is a testament to the potential of digital solutions in transforming urban landscapes, making it a model for other cities facing similar challenges.

7.1.2 Development conclusion

The development phase of the Parking App Project in Colombo has successfully concluded, marking a significant milestone in the quest to modernize and streamline urban parking management. Throughout this phase, the project team has adhered to a rigorous, iterative development process, incorporating feedback from multiple stakeholders to ensure that the application meets the diverse needs of its users. The app's architecture, built on a robust, scalable

platform, promises to handle the demands of a growing urban population and adapt to future enhancements. By leveraging cutting-edge technologies and agile development practices, the team has created an application that not only improves parking efficiency but also significantly reduces traffic congestion in the city. This development has paved the way for a smoother transition to deployment and operational phases, setting the stage for a transformative impact on Colombo's urban mobility and parking systems. As the project moves forward, the foundations laid during this phase will be crucial in achieving sustained success and user satisfaction.

7.1.3 Lessons learned

Throughout the development of the parking application project for Colombo, undertaken individually as part of my university curriculum, I gained invaluable experience across several domains. Managing the project solo enhanced my skills in planning, organizing, and executing complex tasks, with time management and resource allocation proving particularly insightful. The technical aspect involved implementing mobile application development, cloud-based solutions, and integrating APIs like Google Maps, which significantly boosted my technical proficiency. Problem-solving and adaptability were tested through unforeseen challenges, teaching me to manage uncertainty effectively. Engaging directly with potential app users deepened my understanding of user-centered design, crucial for aligning the product with user needs. Additionally, navigating regulatory and compliance issues provided a solid grounding in legal aspects of tech development. Data analysis for decision-making was pivotal, guiding feature adjustments and enhancements, while developing scalable solutions prepared me for future projects. Implementing robust security measures underscored the importance of protecting user data. Presenting the project outcomes to my university supervisor enhanced my public speaking and presentation skills, proving essential for professional development.

7.2 Future Direction

7.2.1 what I Did

For my university project, I developed a parking application specifically designed for the city of Colombo. This initiative aimed to address several critical issues plaguing urban mobility in the city, including the inefficient use of parking spaces, vulnerability to parking scams, and the severe traffic congestion caused by the inadequate availability of parking. The application leverages technology to provide real-time information on parking availability, streamline the parking process, and enhance overall urban efficiency. By integrating features such as mobile payments, GPS-based navigation, and user-friendly interfaces, the project seeks to improve the parking experience for drivers, reduce environmental impact through decreased idle driving, and boost economic productivity by cutting down the time wasted in search of parking. Additionally, the app includes robust security measures to protect users from potential scams and ensure the safety of their data. This project not only fulfills academic requirements but also aims to contribute meaningfully to solving real-world problems in urban planning and mobility in Colombo.

7.2.2 What I Recommend For the Future Implementation

For the future implementation of the parking application project in Colombo, several enhancements are recommended to further improve its efficiency, security, and user experience. Firstly, integrating Internet of Things (IoT) devices without the need for human oversight could streamline operations significantly. This would involve the use of automated systems such as smart gates, where users scan a QR code to enter and exit, reducing manual intervention and increasing the speed of transactions.

Additionally, advancing the mobile application for both Android and iOS platforms is crucial. This will ensure a broader accessibility and enhance user interaction, making the app more robust and user-friendly across different devices. Implementing machine learning algorithms to analyze parking management data is also recommended. Such technology could predict traffic congestion patterns, helping to optimize traffic flow and reduce gridlock by suggesting the best times and locations for parking.

The introduction of IoT technology with smart gate systems would not only automate the process but also enhance security and efficiency. Users scanning a QR code upon entry and exit would allow for precise tracking of parking space occupancy in real time, improving the management of available spaces and potentially reducing unnecessary traffic within parking areas. These recommendations aim to leverage cutting-edge technology to address urban mobility challenges, making parking management more efficient and transforming Colombo into a smarter, more connected city.

7.2.3 What I needed to implement but I couldn’t

In the development of the parking application project, there were ambitious plans to expand the app's functionality to include an iOS version and a call option directly integrated into the application for users and owners. These features were aimed at enhancing accessibility and improving communication between users and parking space owners. Additionally, the intention was to allow users to add multiple vehicles to the system, providing greater flexibility and convenience.
However, as a university student with a limited time frame and resources for the project, implementing these advanced features proved to be unfeasible. The constraints of academic deadlines and the availability of resources necessary for such developments posed significant challenges, leading to the prioritization of core functionalities over these additional features. In the future, with more time and possibly greater resources, these valuable expansions could be revisited to further improve the app’s utility and user experience.

7.3 Future Implementation

Looking ahead to future implementations of the parking application project, several ambitious enhancements are planned to further elevate the functionality and user experience. Key among these is the development of an iOS version of the app, which will broaden its accessibility to users on Apple devices. Additionally, integrating a call option within the app will facilitate direct communication between users and parking space owners, improving service responsiveness and

user convenience by allowing immediate resolution of issues and inquiries directly through the app.

Another significant enhancement involves the integration of Internet of Things (IoT) technology, specifically through the deployment of an IoT-enabled smart gate system. Users will interact with these gates by scanning a QR code upon entering and exiting parking facilities, allowing for automated and secure access. This system will not only streamline the parking process but also enhance security and operational efficiency by providing real-time data on parking space occupancy.

These enhancements, however, will require careful planning, additional resources, and extended development time beyond what was available in the initial phase. With the right support and a focused effort, these features can significantly improve the app's utility and make it a more versatile tool for addressing urban mobility challenges in a smart and connected city environment.

Chapter 08 Postmortem

8.1 How did I archived my objectives

1. Reducing Traffic Congestion: By providing real-time information on parking availability, the application likely helped diminish the time drivers spend searching for parking. This feature would decrease overall traffic congestion by reducing the number of vehicles circling the city, thereby easing traffic flow in Colombo. If implemented effectively, this functionality would significantly contribute to alleviating urban traffic bottlenecks.

2. Enhancing Parking Space Utilization: The app's capability to display real-time parking availability and facilitate advanced bookings ensures that parking spaces are utilized more efficiently. This allows users to find and reserve parking in less frequented areas, balancing the distribution of parked vehicles across the city. By maximizing the use of existing parking infrastructure, the project likely improved the overall efficiency of urban parking resources.

3. Improving User Experience: With features such as GPS navigation, secure electronic payments, and instant access to parking space availability, the application is designed to enhance the overall user experience. These elements not only make parking more convenient but also safer and more reliable, thereby elevating user satisfaction and reducing the stress associated with parking in a busy city.

4. Combatting Parking Scams: By integrating a robust verification system for parking operators and offering secure payment methods, the application helps protect users from common parking scams. This increases the trustworthiness and reliability of the parking services available through the app, providing a safer and more transparent parking environment.

5. Supporting Economic Efficiency: The efficient management of parking spaces and the reduction in time spent searching for parking help decrease fuel consumption and reduce time wasted during commutes. This not only saves money for users but also enhances productivity, contributing positively to the local economy.

8.2 Business objectives

1. Reduce Traffic Congestion: Implement a mobile application that provides real-time information on available parking spaces, thereby reducing the time drivers spend searching for parking. This objective aims to alleviate traffic congestion in Colombo by minimizing unnecessary driving and idling on the roads.

2. Optimize Parking Space Utilization: Maximize the use of existing parking facilities by efficiently allocating spaces through a smart digital management system. This objective seeks to increase overall parking capacity without the need for extensive physical expansion, making better use of the existing infrastructure.

3. Enhance User Experience: Provide a seamless and user-friendly parking experience by integrating features such as secure electronic payments, GPS navigation, and real-time updates on parking availability. This objective focuses on making the parking process as convenient and stress-free as possible for users.

4. Eliminate Parking Scams: Incorporate robust verification and security features to ensure that only legitimate and verified parking operators are part of the system. This objective is designed to protect users from fraudulent activities and unsafe parking conditions, thereby enhancing the reliability of parking services.

5. Increase Economic Efficiency: Reduce the economic impact of traffic congestion and parking inefficiencies by minimizing fuel consumption and decreasing the time spent on parking searches. This objective also aims to enhance overall productivity for businesses and individuals by improving the efficiency of the urban transport system.

6. Support Sustainable Urban Development: Contribute to a cleaner environment by reducing vehicle emissions associated with prolonged idling and unnecessary driving. This objective aligns with broader environmental goals and supports sustainable urban development initiatives.

7. Facilitate Private Property Registration: Encourage private property owners to participate in the parking ecosystem by allowing them to register their spaces through the app. This increases the overall supply of parking spaces and provides additional revenue streams for property owners.

8. Centralize Parking Management: Develop an administrative portal for centralized oversight and management of parking operations. This enables data-driven decision-making, efficient resource allocation, and effective monitoring of parking facilities.

8.2.1 Business opportunists

The Parking Hub project in Colombo opens up several significant business opportunities that capitalize on technological innovation and urban infrastructure improvements. First, there's a substantial market potential in providing a digital solution to a city plagued by parking inefficiencies and traffic congestion. The app can generate revenue through service fees for bookings, subscriptions for premium features, or through partnerships with local businesses and parking facilities. Additionally, there is an opportunity to expand services to other cities facing similar challenges, scaling the solution regionally or even globally.

Private property owners also present a key opportunity. By integrating these stakeholders into the app's ecosystem, the project can tap into underutilized spaces, turning private properties into profitable parking areas. This not only expands the parking network but also provides property owners with a steady income stream.

Advertising and data analytics offer further avenues for monetization. Businesses can advertise on the app, targeting local drivers, while the data collected from users and parking transactions can provide valuable insights into urban mobility patterns, potentially sold to city planners and developers.

Furthermore, by aligning with smart city initiatives, the project can secure government support and funding, positioning itself as a critical component in broader urban planning and sustainability projects. This strategic alignment not only enhances the project’s visibility and viability but also strengthens its potential for integration into larger infrastructural or environmental initiatives.

8.2.1.1 Strength and Weakness

The business objectives of the Parking Hub project possess notable strengths and weaknesses. On the positive side, the objective of reducing traffic congestion directly addresses a major urban issue, improving quality of life and reducing environmental impact. Optimizing parking space utilization maximizes existing resources without requiring new infrastructure investments, offering immediate benefits. Enhancing user experience with a user-friendly app increases customer satisfaction and loyalty, encouraging widespread adoption. Eliminating parking scams by enhancing security builds trust among users, encouraging reliance on the app. Increasing economic efficiency reduces fuel costs and time wasted, providing clear economic benefits. Supporting sustainable urban development aligns with global sustainability trends and can attract government and environmental support. Facilitating private property registration increases parking supply without new construction, providing additional revenue streams for property owners. Centralizing parking management provides comprehensive control, leading to efficient resource allocation and decision-making.

However, these objectives also have inherent weaknesses. Reducing traffic congestion depends heavily on widespread user adoption and consistent real-time data, which can be challenging initially. Optimizing parking space utilization requires comprehensive data integration and cooperation from all parking operators, which might be difficult to coordinate. Enhancing user experience necessitates continuous updates and user feedback, which can be resource-intensive. Implementing robust security measures to eliminate parking scams can be complex and costly. Demonstrating the economic benefits of increased efficiency requires extensive data collection and analysis. The environmental benefits of supporting sustainable urban development might take time to become evident, and initial technology investments may appear contradictory to immediate sustainability goals. Encouraging private property owners to register their spaces may require significant incentives and outreach. Developing and maintaining a centralized management system involves significant initial investment and ongoing maintenance efforts, which could strain finances in the early stages

Reference List

1. Buitleir, D. de (2023) 'What is a parking management system? (explainer)', Wayleadr Blog. Available at: https://wayleadr.com/blog/parking-management-system/ (Accessed: 20 November 2023).
2. Choudhary, R., Sinha, A.S., Jaiswal, K., and Chandra, A. (2023) 'An IoT-based Smart Parking System', Papers with Code. Available at: https://paperswithcode.com/paper/an-iot-based-smart-parking-system (Accessed: 21 November 2023).
3. Colombo Municipal Council (no date) 'Colombo to have structured parking with the introduction of solar powered parking meters', Colombo Municipal Council. Available at: https://www.colombo.mc.gov.lk/news-more.php?id=30 (Accessed: 20 November 2023).
4. Construction and application evaluation of the traffic ... (no date) IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9115109/ (Accessed: 19 November 2023).
5. Councile, C.M. (2023) 'Driving into Colombo’s money spinning car parks! - news features', Daily Mirror. Available at: https://www.dailymirror.lk/news-features/Driving-into-Colombos-money-spinning-car-parks/131-260089 (Accessed: 20 November 2023).
6. Dammulla, R. and Mudunkotuwa, R. (2022) 'Analysis on the road traffic congestion in Colombo Metropolitan Area', CINEC Academic Journal. Available at: https://caj.sljol.info/articles/10.4038/caj.v5i1.70 (Accessed: 19 November 2023).
7. Does traffic congestion slow the economy? (no date) ResearchGate. Available at: https://www.researchgate.net/publication/258154937_Does_Traffic_Congestion_Slow_the_Economy (Accessed: 20 November 2023).
8. Eco Park: A low-cost and sustainable approach for smart ... (no date) IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/10235241/ (Accessed: 19 November 2023).
9. Effect of road traffic congestion on stress at work: Evidence from the employees working in metropolitan areas of Colombo, Sri Lanka (no date) SSRN. Available at:

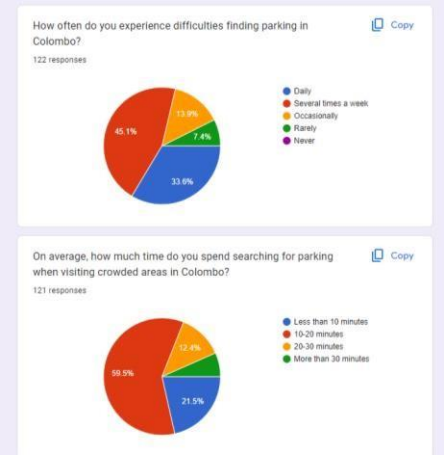
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3853900 (Accessed: 19 November 2023).
10. Find out detailed information (2017) ParkMeApp. Available at: https://www.parkmeapp.com/blog/findoutdetailedinformation/ (Accessed: 20 November 2023).
11. Gaetano, R., Pipino, C., and Pagano, C. (2023) 'An Overview of Urban Mobility: Revolutionizing with Innovative Smart Parking Systems', Sustainability, 15(17), p. 13174. Available at: https://doi.org/10.3390/su151713174 (Accessed: 21 November 2023).
12. IOT based vehicle parking model with mobile apps software (no date) IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9652765/ (Accessed: 19 November 2023).
13. Kanishka Werawella presents solutions to traffic congestion in Sri Lanka: Institute of Policy Studies Sri Lanka (2020) Institute of Policy Studies Sri Lanka. Available at: https://www.ips.lk/kanishka-werawella-presents-solutions-to-traffic-congestion-in-sri-lanka/ (Accessed: 20 November 2023).
14. O'Connell, E., Olariu, S., and Khalid, M. (2021) 'Smart Parking Systems: Reviewing the Literature, Architecture, and Ways Forward', Smart Cities, 4(2), pp. 623-642. Available at: https://doi.org/10.3390/smartcities4020032 (Accessed: 19 November 2023).
15. Park indicator: Book your parking spot (no date) IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8697378 (Accessed: 19 November 2023).
16. Parking fees in Colombo: To pay or not to pay? (2020) Talking Economics. Available at: https://www.ips.lk/talkingeconomics/2020/01/21/parking-fees-in-colombo-to-pay-or-not-to-pay/ (Accessed: 20 November 2023).
17. Parking management system comprehensive guide (no date) Workero. Available at: https://www.workero.com/parking-management-system-guide/ (Accessed: 20 November 2023).
18. Research on parking app choice behavior based on MNL (2021) Travel Behaviour and Society. Available at: https://www.sciencedirect.com/science/article/abs/pii/S2214367X21000697 (Accessed: 19 November 2023).
19. Smart parking and charging system for E ... (no date) IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/10157661 (Accessed: 19 November 2023).
20. Smart parking system (S-park) (no date) IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/9350429/ (Accessed: 19 November 2023).
21. Smart parking system for commercial stretch in Cities (no date) IEEE Conference. Available at: https://ieeexplore.ieee.org/document/8286588 (Accessed: 19 November 2023).
22. VFT Holdings (Pvt) Ltd (no date) 'ZK Parking Management System Solutions', VFT Holdings. Available at: https://vftholdings.lk/ZK_Parking_Management_solution_sri_lanka.php (Accessed: 20 November 2023).
23. Weerasinghe, T.D., Karunarathna, D.I.M. and Subashini, B.L.C. (2021) 'Effect of road traffic congestion on stress at work: Evidence from the employees working in metropolitan areas of Colombo, Sri Lanka', SSRN. Available at:

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3853900 (Accessed: 19 November 2023).

24. 6 steps to build a parking app (no date) Cleveroad Inc. Available at: https://www.cleveroad.com/blog/parking-app-development/ (Accessed: 20 November 2023).

Appendices

Data Gathering method



Records of supervisory meetings

UNIVERSITY OF PLYMOUTH

NSBM GREEN UNIVERSITY TOWN

PUSL3190 Computing Individual Project
Student Progression Report
[Student Copy]

01. Student Name: R.M.K.M. Rathnayake
02. Plymouth Index Number: 10820945
03. Degree Program: Computer Science BSc (Hons)
04. Supervisor Name: Mrs. Dulanjalee Wijesekara
05. Project Title: Parking application

Meeting Number	Meeting 01	Meeting 02	Meeting 03	Meeting 04	Meeting 05	Meeting 06	Meeting 07
Date	16/10	20/10	05/03	14/03			
Student Signature							
Supervisor Signature							

Meeting Number	Meeting 08	Meeting 09	Meeting 10	Meeting 11	Meeting 12	Meeting 13	Meeting 14
Date							
Student Signature							
Supervisor Signature							

UNIVERSITY OF PLYMOUTH | NSBM GREEN UNIVERSITY TOWN

Final Year Project – Supervisory meeting minutes

Meeting No: 4

Date : 2024/03/14

Project Title : Parking App

Name of the Student : R.M.T.M. Rathnayake

Students ID : 10820945

Name of the Supervisor : Dulanjalee Wijesekara

Items discussed:

Discussed about changes to chapter 01 and topics to be discussed in literature review

Items to be completed before the next supervisory meeting:

To draft the literature review

14/03/2024

Supervisor (Signature & Date)

Instructions to the supervisor: **Do not sign** if the above boxes are blank.

UNIVERSITY OF PLYMOUTH | NSBM GREEN UNIVERSITY TOWN

Final Year Project – Supervisory meeting minutes

Meeting No: 1

Date : 2022/10/16

Project Title : Parking App

Name of the Student : R.M.K.M. Rathnayake

Students ID : 10820945

Name of the Supervisor : Dulanjalee Wijesekara

Items discussed:

Discussion conducted on topic list student had

Items to be completed before the next supervisory meeting:

To finalise one topic out of that.

16/10/2023

Supervisor (Signature & Date)

Instructions to the supervisor: **Do not sign** if the above boxes are blank.

UNIVERSITY OF PLYMOUTH | NSBM GREEN UNIVERSITY TOWN

Final Year Project – Supervisory meeting minutes

Meeting No: 2

Date : 2024/02/05

Project Title : Parking APP

Name of the Student : R.M.K.M. Rathnayake

Students ID : 10820445

Name of the Supervisor : Dulanjalie wijesekara

Items discussed:

Discussed about the selected topic and required contents for chapter 01.

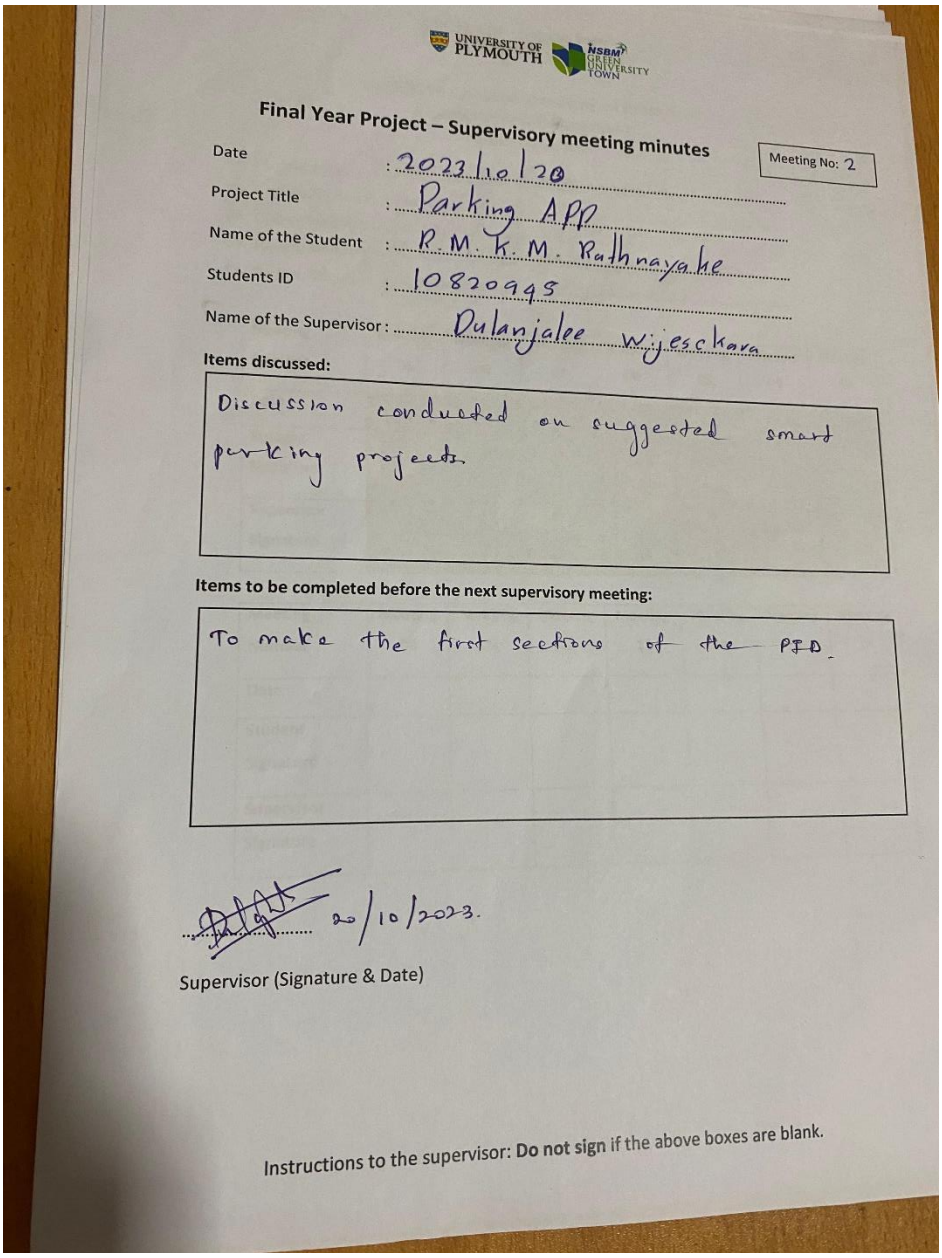
Items to be completed before the next supervisory meeting:

To complete chapter 01.

05/03/2024

Supervisor (Signature & Date)

Instructions to the supervisor: **Do not sign** if the above boxes are blank.



UNIVERSITY OF PLYMOUTH NSBM GREEN UNIVERSITY TOWN

Final Year Project – Supervisory meeting minutes

Meeting No: 2

Date : 2023/10/20

Project Title : Parking APP

Name of the Student : R.M.K.M. Rathnayake

Students ID : 10820945

Name of the Supervisor : Dulanjalee Wijeskara

Items discussed:

Discussion conducted on suggested smart parking projects.

Items to be completed before the next supervisory meeting:

To make the first sections of the PID.

20/10/2023.

Supervisor (Signature & Date)

Instructions to the supervisor: **Do not sign** if the above boxes are blank.

User Guide for my system

User Access

Via Web:

1. Visit parkinghub.me in your web browser.
2. You will be prompted to enter your mobile number.
3. Follow the on-screen instructions to log in or register for a new user account.

Via App (Android):

1. Download the APK file from the provided source.
2. Before installing, ensure that you have allowed installation of apps from unknown sources in your device settings.
3. Install the APK file on your Android device.
4. Open the app and follow the prompts to log in or register for a new user account.

Owner Access

Via Web:

1. Go to parkinghub.me/login in your web browser.
2. Enter your login details if you already have an owner account.
3. If you are a new user, click on the register button to create a new owner account.

Via App (Android):

1. Download the APK file from the provided source.
2. Before installing, ensure that you have allowed installation of apps from unknown sources in your device settings.
3. Install the APK file on your Android device.
4. Open the app and enter your login details if you already have an owner account.
5. If you are a new user, click on the register button to create a new owner account.

Admin Access

Via Web:

1. Visit parkinghub.me/login in your web browser.
2. Log in using your admin account credentials.

Note:

- For security reasons, always ensure that you are accessing ParkingHub through official channels.
- Make sure to keep your login credentials secure and do not share them with anyone.
- Contact support if you encounter any issues or have questions regarding your ParkingHub account.

Code installing via git

Prerequisites:
Git: Ensure Git is installed on your system. You can download it from Git's official website.
Composer: Composer is a dependency manager for PHP. You can download and install it from Composer's official website.
PHP: Laravel requires PHP. Make sure PHP is installed on your system. You can download it from PHP's official website.
Step 1: Clone the Repository
Open your terminal or command prompt and navigate to the directory where you want to clone the Laravel app. Then, execute the following command:

```
git clone https://github.com/kushanmali/parkingHub.git
```

Step 2: Install Dependencies
Navigate into the cloned directory using the terminal:

```
cd repository_name
```

Once inside the directory, run the following command to install Laravel dependencies:

```
composer install
```

Step 3: Create Environment File

Duplicate the .env.example file and rename it to .env. This file contains environment-specific settings like database credentials and application keys.

```
cp .env.example .env
```

Step 4: Generate Application Key
Run the following command to generate a unique application key:

```
php artisan key:generate
```

Step 5: Set Up Database
Open the .env file and configure your database connection settings according to your local environment.

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=your_database_name
DB_USERNAME=your_database_username
DB_PASSWORD=your_database_password
```

Step 6: Run Migrations

Execute the following command to run the database migrations and create the necessary tables in your database:

```
php artisan migrate –seed
```

Step 7: Compile Frontend Assets
Now, compile the frontend assets using npm. If the project uses Laravel Mix, you'll typically use npm run dev:

```
npm run dev
```

Step 8: Serve the Application
Start the Laravel development server:

```
php artisan serve
```

Install via zip

This guide will help you install and run the ParkingHub application locally on your machine. Before proceeding, ensure you have the following prerequisites installed:

Composer
PHP
XAMPP or any other local server environment

Imagick extension
MySQL
Installation Steps
1. Create MySQL Database
Create a MySQL database for the ParkingHub application. You can use your preferred MySQL management tool to create the database.

2. Extract Zip File
Extract the contents of the provided zip file to your desired folder on your local machine.

3. Run Commands
Open your terminal or command prompt and navigate to the folder where you extracted the zip file.

Run the following commands:

bash
Copy code
```
npm run dev
php artisan serve
```
4. Configure Environment Variables
In the root folder of the ParkingHub application, locate the .env file. Open it with a text editor and update the database connection details with your MySQL credentials.

Example:

plaintext
Copy code
```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=your_database_name
DB_USERNAME=your_database_username
DB_PASSWORD=your_database_password
```
5. Start Local Server
After configuring the .env file, open your web browser and navigate to the URL provided by the local server. This will typically be http://localhost:8000 if you're using Laravel's built-in server.

6. Access ParkingHub
Once the local server is running and you've navigated to the correct URL, ParkingHub should be accessible through your web browser.

Note:
Make sure to keep your database credentials secure.
If you encounter any issues during installation, refer to the Laravel documentation or seek assistance from the community.
Ensure you have all the required PHP extensions and server configurations set up properly.

PUSL3190 Computing Individual Project

Interim Report

PARKING APPLICATION

Supervisor: Ms. Dulanjalee Wijesekara

Name: Rathnayake M Rathnayake
Plymouth Index Number: 10820945
Degree Program: Computer Science

Table Of Content

Chapter 01 Introduction.....2
1.1 Introduction.....2
1.2 Problem Definition.....2
1.3 Project Objectives.....4
Chapter 02 System Analysis.....5
2.1 Facts Gathering Techniques.....5
2.2 Existing System.....5
2.3 Overall Use case diagram.....6
2.3.1 Overall Use case diagram: Elaboration.....7
2.4 Drawbacks of the existing system.....7
Chapter 03 Requirements Specification.....8
3.1 Functional Requirements.....8
3.1.1 Must-Haves:.....8
3.1.2 Should-Haves:.....9
3.1.3 Could-Haves:.....9
3.1.4 Won't-Haves:.....9
3.2 Non-Functional Requirements.....9
3.2.1 Must-Haves:.....9
3.2.2 Should-Haves:.....10
3.2.4 Could-Haves:.....10
3.3 Assumptions.....10
3.4 Out of Scope Items.....10
3.5 Hardware / Software Requirements.....10
3.5.1 Hardware:.....10
3.5.2 Software:.....10
3.6 Networking Requirements (Optional).....10
Chapter 04 Feasibility Study.....11
4.1 Operational Feasibility.....11
4.2 Technical Feasibility.....11
4.2.1 Technology Stack:.....11
4.2.3 Integration:.....11
4.2.4 Security:.....12
4.2.5 Performance and Scalability:.....12
4.3 Vendor Management:.....12
4.4 Legal and Regulatory Feasibility:.....12
Chapter 05 System Architecture.....13
5.1 Class Diagram of Proposed System.....13
5.2 ER Diagram.....15
5.3 High-level Architecture Diagram.....16
5.4 Networking Diagram.....18
5.4.1 Networking Diagram: Elaboration
5.4.1.1 Virtual Private Cloud (VPC):.....18
5.4.1.2 Public Route Table:.....19

5.4.1.3 Public Subnet:..... 19
5.4.1.4 EC2 Instance:..... 19
5.4.1.5 EC2 Security Group:..... 19
5.4.1.6 Private Subnets (1 & 2):..... 19
5.4.1.7 MySQL RDS Database:..... 19
5.4.1.8 RDS Security Group:..... 19
5.4.1.9 SSH Client:..... 19
5.4.1.10 Private Route Table:..... 19
5.4.1.11 App Users and Parking Officer:..... 20
Chapter 06 Development Tools and Technologies..... 20
6.1 Programming Languages and Tools.....20
6.2 Third Party Components and Libraries.....21
07 Discussion..... 21
7.1 Summary of the Report..... 22
7.2 Challenges Faced..... 22
7.3 Future Plans / Upcoming Work..... 22
References..... 23

Chapter 01 Introduction

1.1 Introduction

As urban centers across the globe grapple with the complexities of population growth and increasing vehicle ownership, parking emerges as a critical challenge. Colombo, the vibrant capital of Sri Lanka, is no stranger to this phenomenon. The city's rapid development has fueled a demand for parking that far exceeds the existing infrastructure, leading to multifaceted problems impacting traffic, the environment, and even the safety of its residents.

A lack of sufficient parking spaces leaves drivers endlessly circling congested areas, wasting precious time and fuel while contributing to gridlock. Traffic congestion chokes the city, leading to longer commutes, economic losses due to delayed deliveries and lost productivity, and a surge in harmful air pollution. To make matters worse, Colombo's parking landscape lacks a cohesive management system, opening the door for scams and fraudulent practices that exploit drivers and undermine trust.

1.2 Problem Definition

The parking app project confronts a critical issue in Colombo: the acute shortage of parking spaces and a disjointed parking management system. This situation forces drivers into a challenging and often detrimental scenario, characterized by several key problems:

Wasted Time and Resources

Colombo sees over half a million vehicles traversing its boundaries daily, as reported by the Ministry of Transport. This staggering number lays bare the stark imbalance between the parking demand and the limited supply. The ramifications of this mismatch are profound:

- **Increased Traffic Congestion:** The hunt for parking spaces leads to vehicles circulating for longer periods, contributing significantly to traffic congestion.
- **Elevated Fuel Consumption:** The extended search for parking not only wastes time but also increases fuel expenditure, thereby elevating the environmental footprint.
- **Diminished Urban Efficiency:** The cumulative effect of prolonged parking searches undermines the city's efficiency, impacting both economic activity and the quality of urban life.

Vulnerability to Scams

The absence of a reliable system for validating legitimate parking facilities leaves drivers susceptible to scams. This problem manifests in several ways:

- **Fraudulent Parking Attendants:** Drivers encounter individuals who demand inflated parking fees without offering the security of a regulated space.
- **Safety Concerns:** The lack of verified parking spots can lead to vehicles being parked in unsecured areas, heightening the risk of theft or damage.

Inefficient Use of Space

Many existing parking facilities, especially within the private sector, suffer from inefficient utilization. This inefficiency stems largely from:

- **Lack of Technological Integration:** A significant number of parking lots do not leverage technology to optimize their operation, leading to suboptimal use of available spaces.
- **Absence of a Centralized Management System:** Without a unified system to manage and allocate parking resources, both public and private lots cannot effectively cater to the demand.

Negative Overall Experience

The collective impact of these issues severely compromises the parking experience in Colombo. For many, what should be a simple component of urban travel becomes a significant source of anxiety, financial burden, and lost time.

Quantitative Research Imperative

To tackle these issues effectively, a detailed quantitative analysis is imperative. This research should aim to:

- **Measure the Average Time Spent on Parking Searches:** By understanding how long drivers typically spend looking for parking, we can quantify the inefficiency of the current system.
- **Assess the Economic Impact of Traffic Delays:** Investigating the broader economic repercussions of parking-induced traffic congestion will highlight the urgency of finding a solution.
- Gauge the Prevalence of Parking Scams: Collecting data on the frequency and nature of scams will help tailor the app's security features to protect users effectively.

This comprehensive understanding of the problem's scope will form a robust foundation for the proposed parking app solution, designed to alleviate these pressing concerns and enhance the overall urban mobility in Colombo.

1.3 Project Objectives

The primary goal of developing this parking application is to address the pressing parking issues in Colombo, Sri Lanka. This project has been designed with specific, measurable objectives to significantly enhance the urban parking experience, contributing to the overall well-being and efficiency of the city. Below are the detailed objectives:

- **Reduce Time Spent in Searching for Parking:** One of the most critical objectives of the parking app is to minimize the time drivers spend looking for available parking spaces. By leveraging real-time data, the app will provide instant information about the location and availability of parking spots, particularly in private parking areas. This will not only save valuable time for the users but also alleviate traffic congestion caused by vehicles circling around in search of parking.

- **Increase Accessibility to Parking Information:** The app is envisioned to serve as a comprehensive repository of information regarding private parking facilities across Colombo. This will include details such as location, pricing, and hours of operation. By aggregating this data in a user-friendly interface, the app will empower drivers to make informed decisions about where to park, based on convenience, cost, and proximity to their destination.

- **Minimize Fraudulent Parking Practices:** Parking fraud and scams are prevalent issues that this project aims to tackle head-on. Through a stringent verification process for parking spaces and attendants, coupled with a secure, digital payment system, the app will significantly diminish the risk of fraudulent activities. This initiative will protect users from scams and unauthorized parking charges, fostering a safer and more trustworthy parking environment.

- **Enhance the Overall Parking Experience:** Beyond the functional improvements, this app seeks to revolutionize the parking experience in Colombo. By mitigating the challenges related to finding parking, ensuring the availability of reliable parking

information, and securing the parking process, the app will make parking a hassle-free component of urban mobility. The anticipated outcomes include improved road safety by reducing the likelihood of accidents due to erratic parking, decreased environmental pollution through less idling and emissions, and bolstered economic health by improving access to businesses and reducing lost productivity.

By achieving these objectives, the parking app will not only solve the immediate practical concerns associated with parking but also contribute to a broader vision of a more livable, efficient, and sustainable Colombo. This initiative represents a significant step forward in the city's journey towards smart urban mobility, enhancing the quality of life for its residents and visitors alike.

Chapter 02 System Analysis

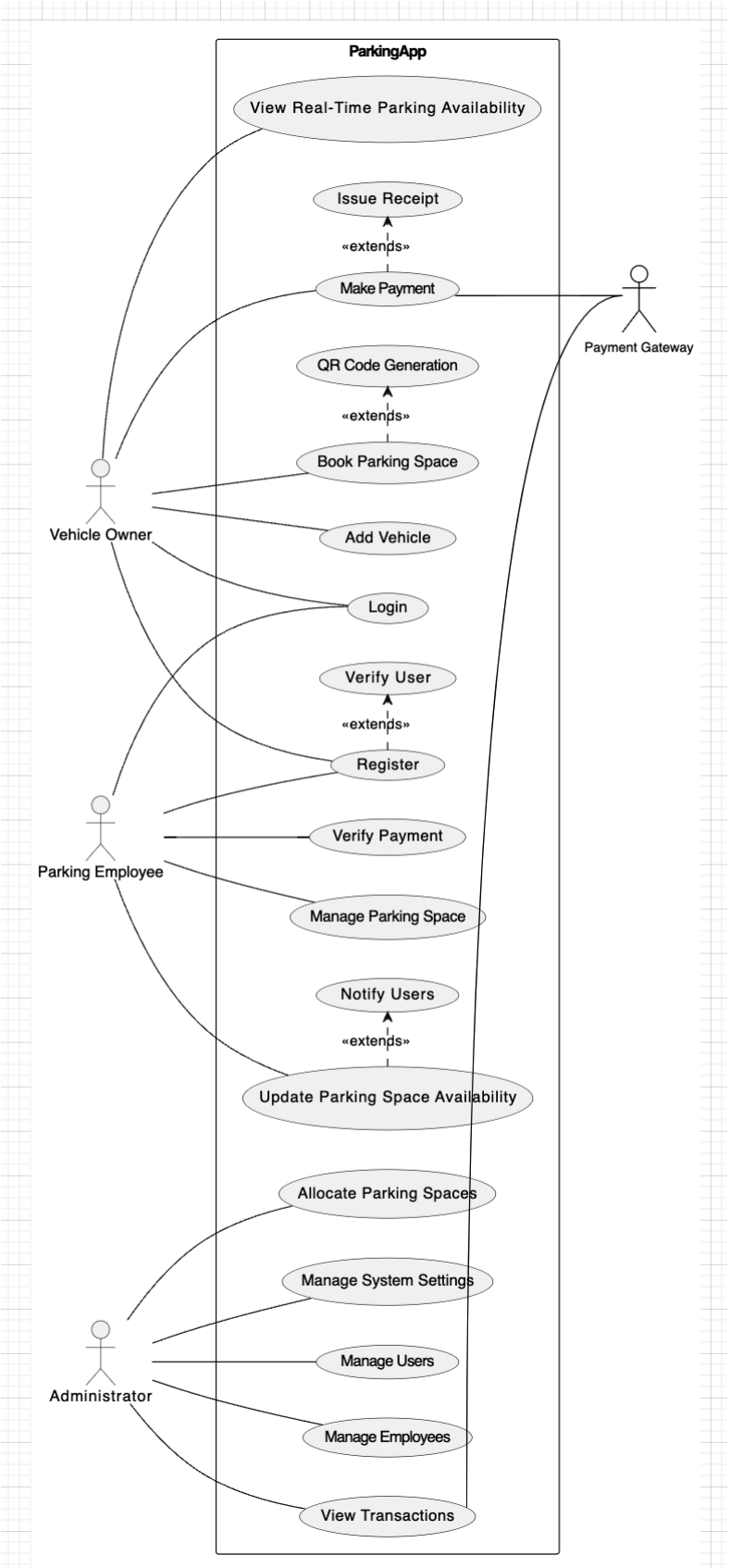
2.1 Facts Gathering Techniques

To thoroughly understand the parking challenges in Colombo, various fact-gathering techniques were employed. These included surveys and interviews with city residents, observations of parking patterns in high-density areas, and consultation with transport and urban planning experts. Additionally, a review of existing literature and case studies from similar urban settings provided insights into effective parking management strategies. The research also involved analyzing data from traffic congestion studies to pinpoint peak times and locations most affected by inadequate parking facilities.

2.2 Existing System

The current parking system in Colombo relies heavily on manual operations with limited technological intervention. It encompasses a mix of government-operated and private parking spaces, primarily managed through physical tickets or tokens and cash transactions. This system lacks real-time information about parking availability, leading to inefficient use of existing spaces. Additionally, there is minimal oversight of parking operations, which contributes to the prevalence of scams and unauthorized parking solutions.

2.3 Overall Use case diagram



2.3.1 Overall Use case diagram: Elaboration

The proposed parking app for Colombo, as visualized in the use case diagram, is designed to alleviate the traffic congestion caused by the critical shortage of parking space and inefficient parking management. It does so by digitizing the entire parking process, thereby creating a seamless and integrated system that serves vehicle owners, parking employees, and administrators. The use case diagram is the blueprint of the system's functionality from which we can deduce its implications on the transportation woes of Colombo.

For the Vehicle Owner, the app serves as a gateway to locate and secure parking spots with ease, bypassing the frustration of manual searches that exacerbate traffic snarls. The 'View Real-Time Parking Availability' use case directly addresses the problem of vehicles crowding the streets in search of parking, as over 500,000 vehicles commute through Colombo daily. Real-time visibility of parking space availability is crucial in this scenario, as it minimizes the time spent on the road, reduces fuel consumption, and lowers emissions, all of which are outlined as key objectives.

The 'Make Payment' and 'Issue Receipt' use cases are central to the elimination of cash transactions, reducing the reliance on outdated manual processes, and curbing the instances of parking scams and ransom seekers, which are major concerns mentioned in the problem statement. By integrating a secure 'Payment Gateway', the system ensures that transactions are transparent and fraud is minimized. QR Code Generation' further tightens this security, providing a unique identifier for each transaction that can be verified by parking employees and also serves as a contactless entry/exit mechanism.

For the Parking Employee, the app transforms their role into one of oversight and management rather than manual ticketing. 'Manage Parking Space' and 'Update Parking Space Availability' directly impact the utilization of the parking spaces. By updating availability in real-time, parking employees contribute to a dynamic system that can respond instantly to changes, thus optimizing the use of existing parking infrastructure.

The System Administrator role focuses on maintaining the integrity and efficiency of the entire operation. The 'Manage Users' and 'Manage Employees' use cases ensure that only verified individuals are allowed access to the system, which addresses security concerns. 'Manage System Settings' and 'View Transactions' provide the administrators with the tools to oversee the operational and financial aspects of the app, thus ensuring that the system runs smoothly and meets its objectives.

2.4 Drawbacks of the existing system

- **Severe Shortage of Parking Spaces:** The fundamental issue is the overwhelming demand for parking compared to the woefully inadequate supply. This directly leads to the search for parking becoming a significant contributor to traffic congestion and wasted time for drivers.

- **Lack of Centralized Information:** Without a single, reliable source of parking availability data, drivers must rely on guesswork and inefficiently cruise for an open spot. This adds to the time spent searching, wasted fuel, and the overall negative experience.
- **Systemic Inefficiencies**: Underutilized parking spaces and inefficient lot management highlight suboptimal use of existing resources. This stems from the fragmented nature of the current system and lack of technological solutions to optimize space usage.
- **Opportunistic Fraud:** The system's unregulated and chaotic nature leaves drivers open to scams and exploitation. Fraudulent parking attendants or exorbitant pricing take advantage of the lack of transparency and oversight.
- **Poor User Experience:** The combined impact of these drawbacks makes parking a constant source of stress and frustration in Colombo instead of the functional convenience it should be.

Relating Drawbacks to Problems Defined in Section 1.2

1. **Wasted Time and Resources:** The shortage of spaces and lack of information directly correlate to drivers burning fuel and losing productive time while searching for available parking.
2. **Vulnerability to Scams:** The problem of illegitimate parking schemes ties directly into the lack of a centralized, regulated system that leaves drivers no reference point to validate parking fees or attendants.
3. **Inefficient Use of Space:** Ineffective management and limited technology contribute to existing spaces not being used to maximum potential. This further exacerbates the parking shortage.
4. **Negative Overall Experience:** Each drawback plays into the larger issue of making parking a stressful, unpredictable, and costly affair, fundamentally harming the urban transportation experience.

Chapter 03 Requirements Specification

3.1 Functional Requirements

3.1.1 Must-Haves:

- User Registration: A secure registration process for users that collects essential information and creates a unique profile for each user.
- Vehicle Registration: Capability for users to register multiple vehicles under their profile, with vehicle-specific information that aids in identification.

- Payments Integration: A reliable digital payment system to facilitate cashless transactions with features such as QR code scanning for verifying payments.
- Employee Registration: A system for parking employees and private parking owners to register and manage their parking spaces.
- Admin Panel: An administrative backend for overarching control, including user and employee management, transaction oversight, and system settings configuration.

3.1.2 Should-Haves:

- Real-Time Parking Availability Updates: A live update feature that reflects current parking space occupancy, managed by parking employees.
- Automated Booking System: An interface for users to book available parking spaces in real-time.

3.1.3 Could-Haves:

- Feedback System: An optional feedback mechanism for users to rate their parking experience and for administrators to collect data on user satisfaction.
- Revenue Analytics: Advanced analytics for administrators to track the app's revenue streams and understand financial metrics.

3.1.4 Won’t-Haves:

- Integration with Existing Municipal Systems: Linking the app directly to city traffic management systems is not planned for the initial roll-out.
- Complex IoT Integration: IoT-based parking management, requiring extensive hardware like sensors or cameras, is not included due to budget and infrastructure considerations.

3.2 Non-Functional Requirements

3.2.1 Must-Haves:

- Performance: The system must handle multiple user requests simultaneously without lag, providing a response time of no more than two seconds for all actions.
- Reliability: The system must have an uptime of 99.9%, ensuring availability during peak traffic hours.
- Security: Protection of user data, including financial information and personal details, with compliance to data protection regulations.
- Usability: An intuitive UI/UX for the mobile application, catering to non-technical users as well.

3.2.2 Should-Haves:

- Scalability: The app should be scalable to accommodate the growing number of users and parking spaces without performance degradation.
- Compatibility: The app should be compatible with both iOS and Android platforms and support various payment methods.

3.2.4 Could-Haves:

- Localization: Multi-language support, considering the linguistic diversity of Colombo.
- Customer Support Integration: A chat or help desk feature for real-time assistance.

3.3 Assumptions

- Users and parking employees have access to smartphones with internet connectivity.
- The GPS data provided by users for parking location is accurate and reliable.
- Parking employees will diligently update the parking space availability.
- All financial transactions are secured and monitored for fraudulent activities.

3.4 Out of Scope Items

- On-premises hardware installations for parking management are excluded.
- Integration with other smart city solutions is not considered in the initial release.
- Multi-vehicle tracking or advanced telematics features are not included.

3.5 Hardware / Software Requirements

3.5.1 Hardware:

- User Device: Smartphone with GPS and camera for QR scanning.
- Employee Device: Smartphone or tablet for managing parking spaces.

3.5.2 Software:

- Mobile Platform: The app will be developed using Flutter for cross-platform support.
- Backend Infrastructure: Cloud services such as Firebase for database, authentication, and hosting services.
- Payment Processing: Secure integration with a reliable payment gateway.
- Administrative Access: Web-based portal accessible through standard web browsers.

3.6 Networking Requirements (Optional)

- A continuous and reliable internet connection for users and parking employees to access real-time data and perform transactions.
- Secure network protocols for data transmission to prevent unauthorized access.

The MoSCoW method applied in this requirement specification provides a clear hierarchy of requirements, ensuring that the parking app development stays focused on the essential features that address the core problems identified in Colombo's current parking management scenario. The delineation of must-haves, should-haves, and could-haves also assists in prioritizing the development process and allocating resources efficiently. With these specifications, the project is poised to deliver a solution that not only satisfies the immediate parking concerns in Colombo but also lays the groundwork for a scalable and robust system for future enhancements.

Chapter 04 Feasibility Study

4.1 Operational Feasibility

The operational feasibility of the parking app in Colombo examines whether the proposed solution fits within the existing operational structure and the city's transportation management strategy. This assessment considers the readiness of users and employees to adopt a new system, the cultural shift required, and the app's potential to meet its objectives effectively.

The strong dependence on manual processes and the prevalence of parking scams indicate a market ready for disruption and willing to adopt new technologies. The app's operational feasibility is enhanced by its alignment with the city's need to alleviate traffic congestion, which is a significant issue during peak hours. The user-friendly nature of the app, coupled with incentives to encourage adoption, such as reduced parking fees or preferred spots for app users, can significantly increase its operational uptake.

Training programs will be necessary to familiarize both users and parking employees with the app's functionality. The operational shift from a traditional to a digital system may initially face resistance; thus, change management strategies will be essential to ensure a smooth transition.

4.2 Technical Feasibility

4.2.1 Technology Stack:

The technology stack chosen, including Flutter for front-end development, Laravel for the backend, and AWS cloud services, aligns well with the app's requirements. These technologies are robust, scalable, and have been proven to work effectively in similar applications worldwide.

4.2.3 Integration:

The app is designed to function independently, minimizing the complexity of integrating with existing systems. However, the use of APIs can allow for future integration with municipal traffic systems if needed.

4.2.4 Security:

The AWS cloud provides a secure infrastructure that complies with various compliance standards. Cloudflare's inclusion for DDoS protection further strengthens the app's security posture, ensuring that user data and transactions are secure.

4.2.5 Performance and Scalability:

AWS's scalable infrastructure ensures that the app can handle a growing number of users and data. The use of a CDN through Cloudflare will enhance the app's performance by reducing latency and ensuring quick load times for users accessing the service across Colombo.

4.3 Vendor Management:

Contracts with AWS and Cloudflare need careful management to prevent over-reliance on these services, avoiding potential vendor lock-in issues.

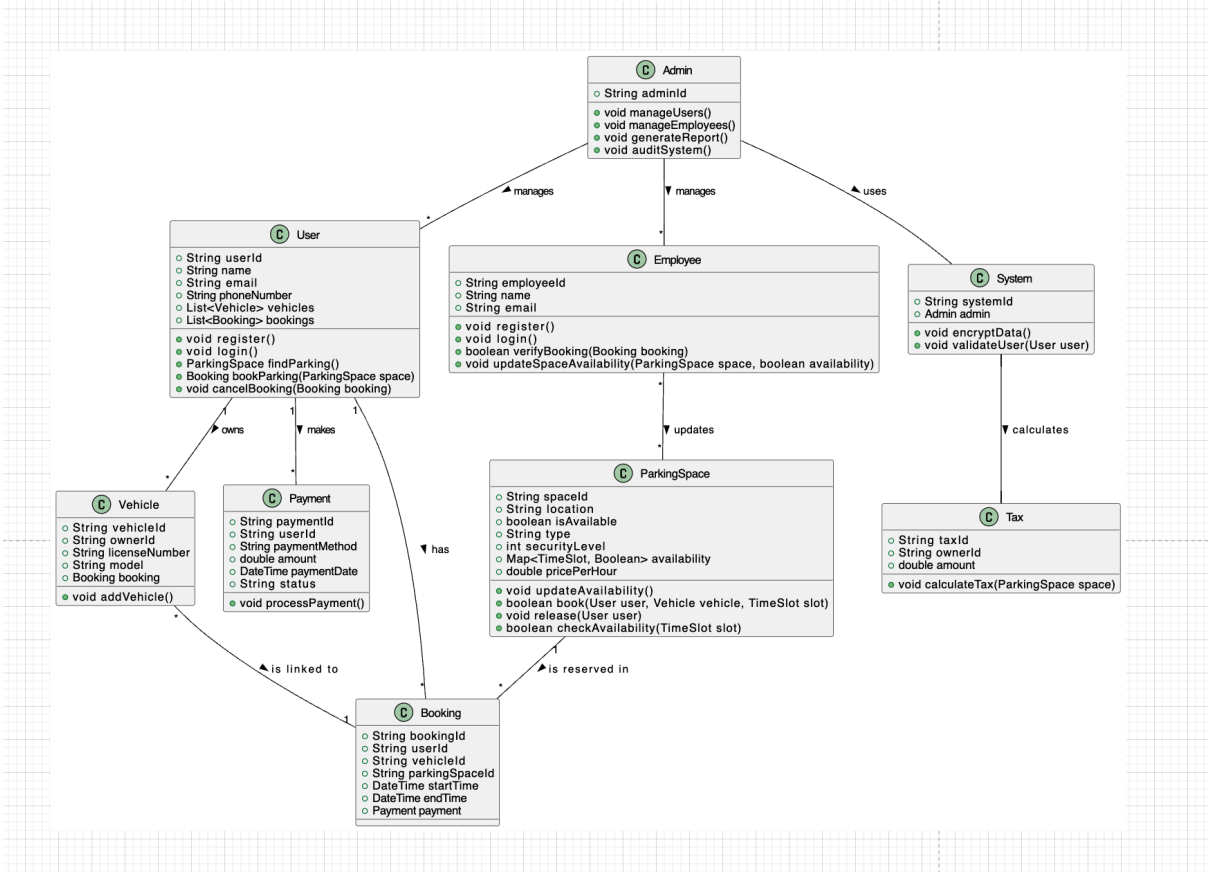
4.4 Legal and Regulatory Feasibility:

Compliance with data protection laws such as the General Data Protection Regulation (GDPR) is crucial since the app will handle personal and financial information. AWS provides compliance with various certifications that bolster the app's legal standing.

In conclusion, the feasibility study underscores the parking app's potential to provide a practical solution to Colombo's parking woes. By leveraging modern technologies and taking a user-centric approach, the app can deliver operational efficiency, security, and scalability. The outlined budget illustrates a viable economic model with a clear path to ROI, contingent on the careful management of resources and timelines. The proposed system promises to deliver an integrated solution that not only simplifies parking management but also supports broader objectives of improving urban mobility and productivity.

Chapter 05 System Architecture

5.1 Class Diagram of Proposed System



The proposed system is a cloud-based parking management system designed to address the shortcomings of existing parking systems in Colombo, Sri Lanka. The system is designed to be user-friendly and comprehensive, incorporating features for users, employees, and administrators.

Breakdown of the classes involved in the system:

- **Admin** class: This class manages the overall system, including user and employee data, generating reports, and auditing the system.
- **User** class: This class represents a registered user of the app. It has attributes such as user ID, name, email, phone number, and a list of vehicles and bookings. It allows users to register, login, find parking, book parking spaces, and view booking history.
- **Employee** class: This class represents a parking employee responsible for managing parking spaces. It has attributes such as employee ID, system ID, and name. It allows employees to register, login, update space availability, and process payments.
- **Vehicle** class: This class represents a vehicle registered by the user. It has attributes such as vehicle ID, license number, model, owner ID, and booking information.

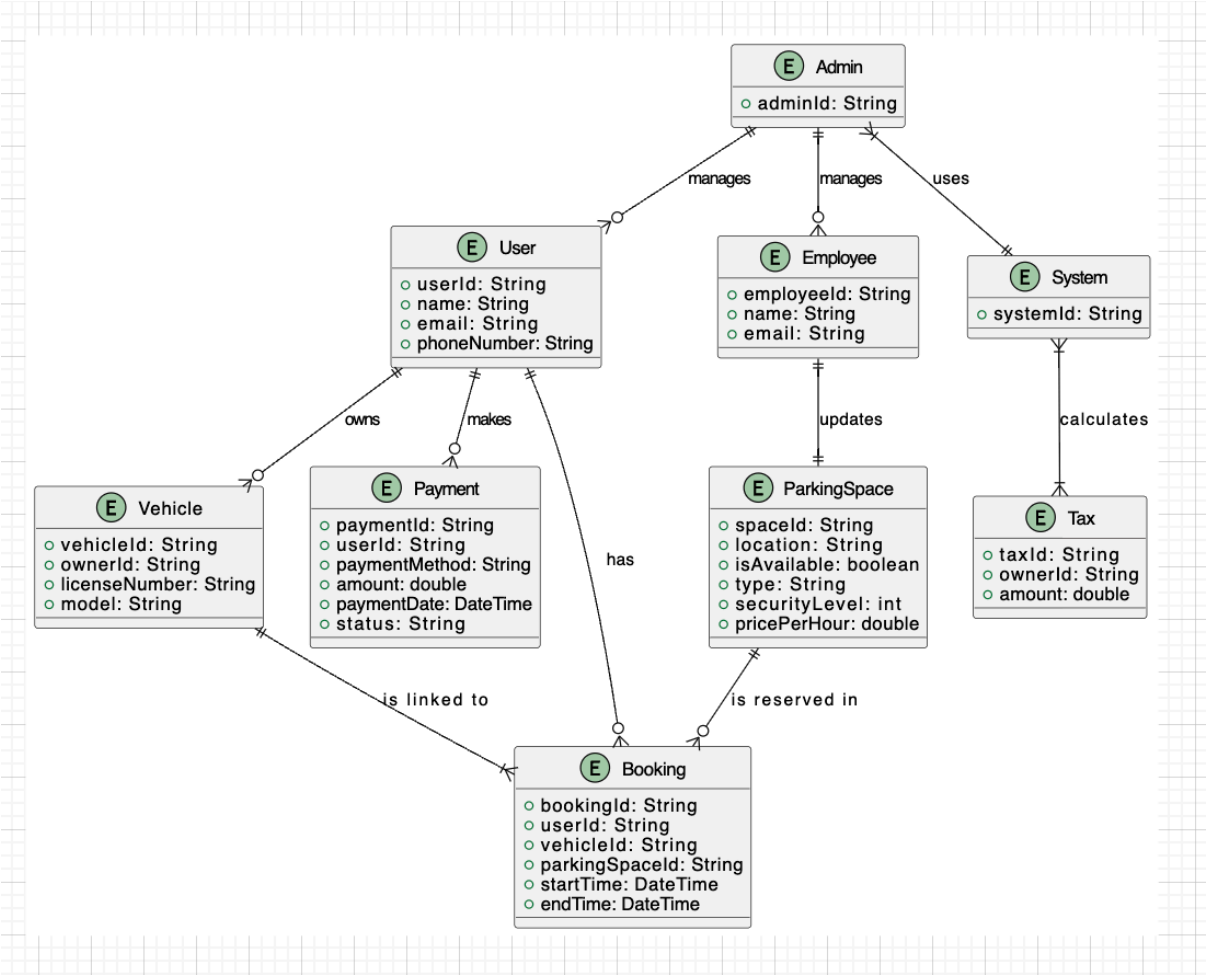
- **Parking Space** class: This class represents a parking space within the system. It has attributes such as space ID, location, availability, type (e.g., car, motorbike), security level, and a map of time slots with availability.
- **Booking** class: This class represents a parking reservation made by a user. It has attributes such as booking ID, user ID, vehicle ID, parking space ID, start time, and end time.
- **CPayment** class: This class handles payment processing within the system. It likely has attributes such as payment ID, payment method, amount, and date/time of payment.
- **Tax** class: This class might be used to manage any taxes associated with parking fees or private parking registration. It could have attributes such as tax ID, tax amount, and owner ID.

The relationships between these classes are as follows:

- **Admin** manages **Users** and **Employees**.
- **Users** can register **Vehicles** and make **Bookings** for **Parking Spaces**.
- **Employees** update the availability of **Parking Spaces** and process payments through **CPayment**.
- **Vehicles** are linked to **Bookings** through the **User**.
- **Parking Spaces** are linked to **Bookings** through the **Booking** class.
- **Bookings** might be associated with a **CPayment** class depending on the implementation.
- **Parking Spaces** might be owned by someone and a **Tax** class might handle any ownership fees.

Overall, the class diagram outlines a comprehensive parking management system that leverages software integration to address the challenges of Colombo's parking landscape. It aims to provide a user-friendly and efficient solution for all parties involved.

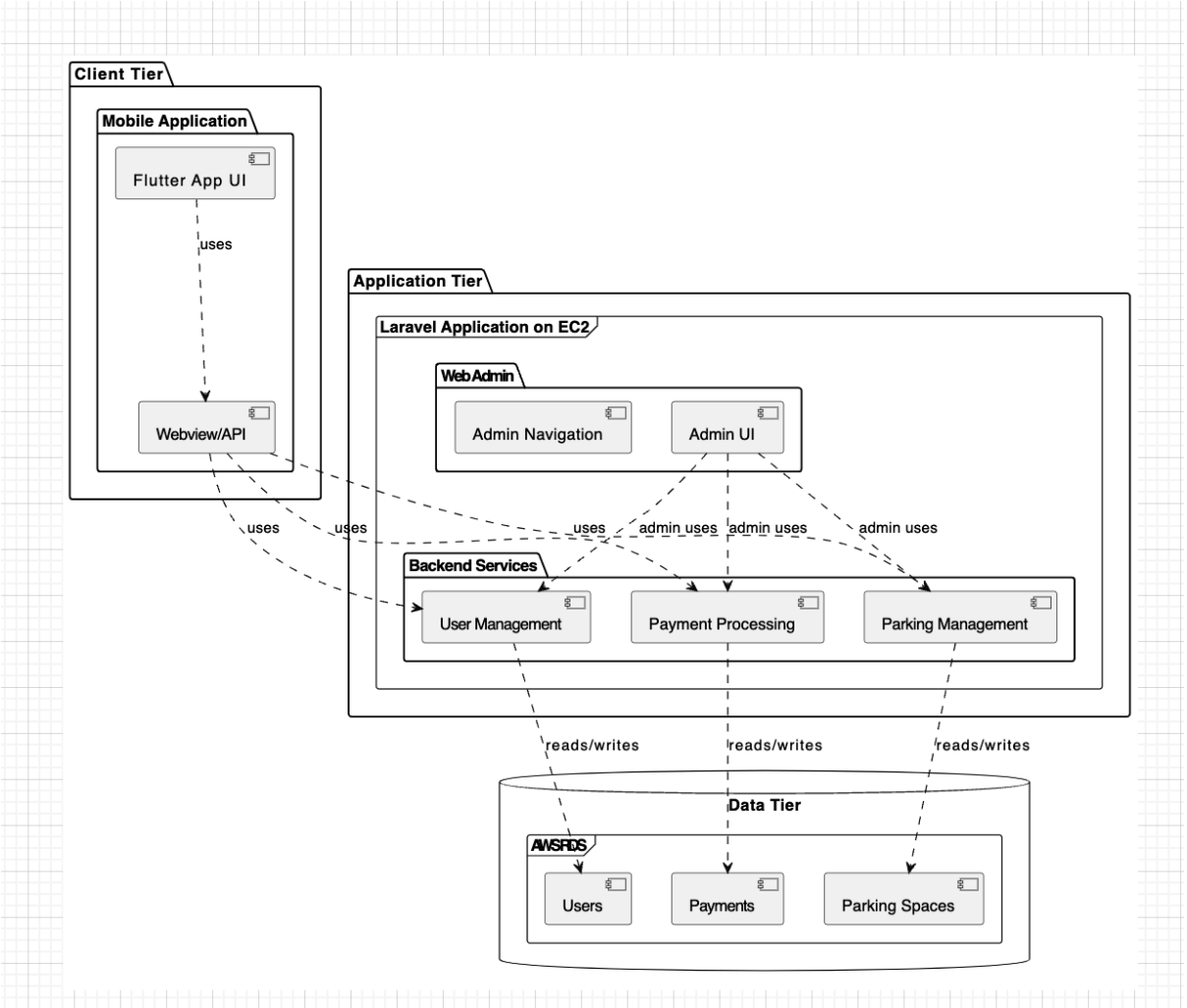
5.2 ER Diagram



Note:

- Laravel Eloquent ORM will be used to define models for each entity and their relationships.
- Eloquent ORM facilitates the transition from ER diagrams to actual database implementation while taking care of underlying normalization rules.
- With Eloquent Active Record implementation, it is also straightforward to handle CRUD operations and database relationships in an object-oriented manner.

5.3 High-level Architecture Diagram



Three-tier architecture diagram for a parking management system. This architecture is divided into three distinct layers:

5.3.1. Client Tier:

- Mobile Application: This represents the front-end through which users interact with the system. It consists of:
    - Flutter App UI: The user interface built using Flutter, a UI toolkit for building natively compiled applications for mobile, web, and desktop from a single codebase.
    - Webview/API: The method by which the mobile application communicates with the backend services. This could be through a webview displaying web content directly in the app, or through an API that sends and receives data.

5.3.2. Application Tier:

- **Laravel Application on EC2:** This is the server-side environment where the application logic is processed. It's hosted on an EC2 (Elastic Compute Cloud) instance, which is a part of Amazon Web Services (AWS). Inside this application tier, there are further subdivisions:

    - **Web Admin:** A sub-system for administrative purposes, which includes an interface for admin navigation and administrative tasks.

    - **Backend Services:** This is where the core functionalities of the parking management system reside, which includes:
        - User Management: Handles user-related operations such as registration, authentication, and user profile management.
        - Payment Processing: Manages financial transactions, such as parking fees and payments.
        - Parking Management: Responsible for managing parking space inventory, booking, and occupancy.

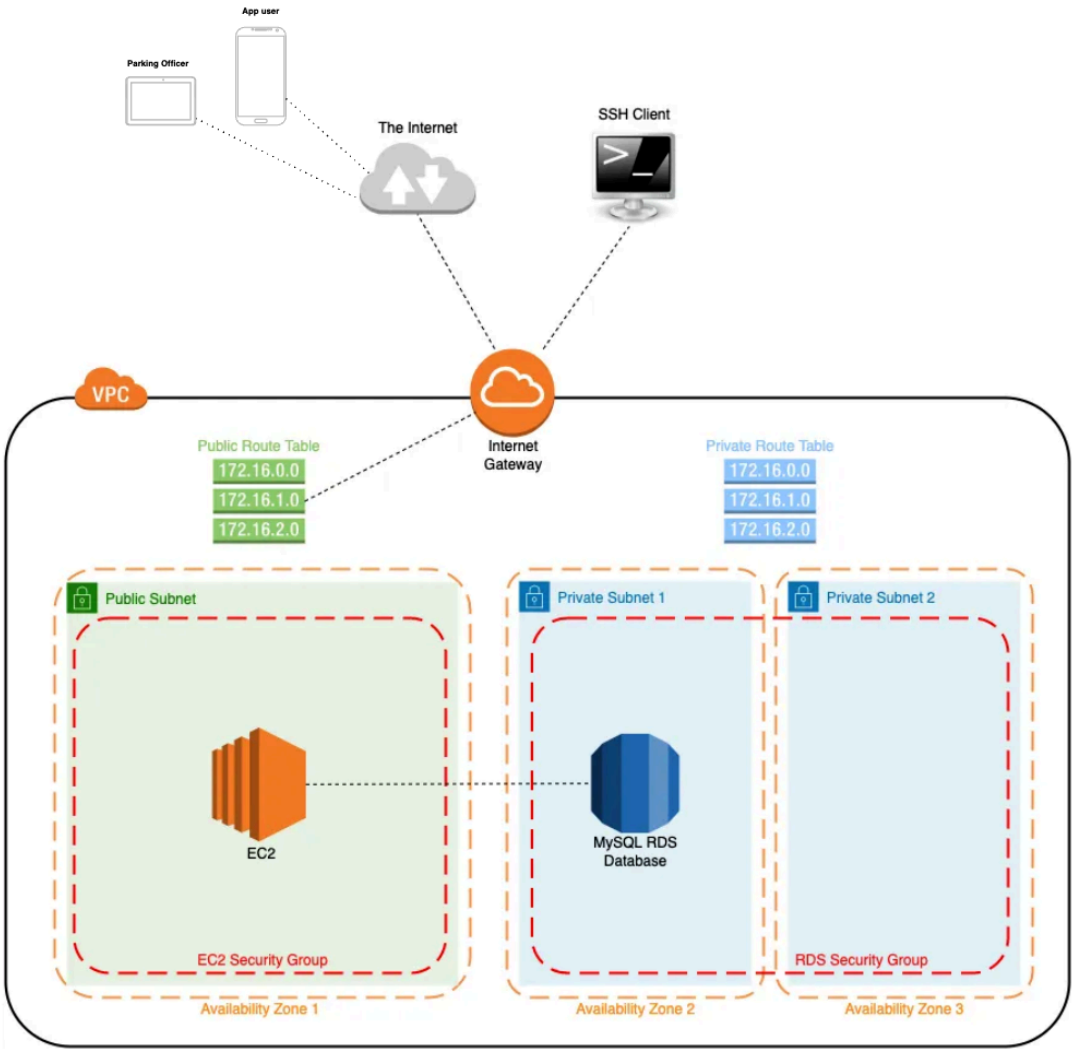
5.3.3. Data Tier:

- AWS RDS (Relational Database Service): This is the database layer where all the data is stored. It uses AWS RDS, a managed relational database service that supports multiple database engines. The data entities include:
    - **Users:** Stores user information, including personal details, vehicle information, and transaction history.
    - **Payments:** Manages records of all payments processed within the system.
    - **Parking Spaces:** Contains information about parking space availability, locations, bookings, and other related data.

Connections:

- The dashed lines indicate the interactions between different components. For instance:
    - The Mobile Application uses the Webview/API to interact with the Backend Services.
    - The Backend Services, in turn, interact with the Database to read and write data.
    - The Admin UI uses Backend Services to manage the system.

5.4 Networking Diagram



5.4.1 Networking Diagram: Elaboration

5.4.1.1 Virtual Private Cloud (VPC):

Represents an isolated section of the cloud provider's network where you can launch resources in a defined virtual network.
Internet Gateway:

This component allows communication between the VPC and the internet. It is attached to the public route table, suggesting that the subnet associated with this route table will have internet access.

5.4.1.2 Public Route Table:

Contains routes for network traffic within the public subnet, directing outbound traffic to the internet gateway. The CIDR blocks shown are the example IP address ranges associated with the subnets.

5.4.1.3 Public Subnet:

Hosts the resources that need to be accessible from the internet. In this case, an EC2 (Elastic Compute Cloud) instance resides here, which might be serving the application or web server.

5.4.1.4 EC2 Instance:

A virtual server where applications are deployed. This is within an availability zone, which is a distinct location within a region that is engineered to be isolated from failures in other availability zones.

5.4.1.5 EC2 Security Group:

Acts as a virtual firewall for the EC2 instance to control inbound and outbound traffic.

5.4.1.6 Private Subnets (1 & 2):

These are isolated from direct access to the internet, used for backend systems like databases. They are in separate availability zones (AZ), which provides high availability by replicating resources across different, physically separate data centers.

5.4.1.7 MySQL RDS Database:

A managed relational database service that handles routine database tasks such as patching, backup, recovery, failure detection, and repair.

5.4.1.8 RDS Security Group:

Similar to the EC2 Security Group, it controls the access to the RDS instance, ensuring that only authorized traffic can reach the database.

5.4.1.9 SSH Client:

Represents an external administrator's computer that connects to the EC2 instance using SSH (Secure Shell) for secure logins and command execution.

5.4.1.10 Private Route Table:

Routes the traffic within the private subnets and does not allow direct access to the internet. This route table would typically have routes for internal traffic and might also have a route to a NAT (Network Address Translation) Gateway or instance for outbound internet access if needed.

5.4.1.11 App Users and Parking Officer:

These are external entities, likely using a mobile or web application, which communicate with the EC2 instance over the internet.

Chapter 06 Development Tools and Technologies

The development of the parking app for addressing Colombo's vehicle parking issues requires a flexible yet structured approach. To this end, Agile Development, guided by the Rational Unified Process (RUP), has been chosen as the primary methodology. Agile methodologies promote adaptive planning, evolutionary development, early delivery, and continual improvement, encouraging rapid and flexible response to change. The RUP framework complements this by providing a disciplined approach to assigning tasks and responsibilities within a development organization. Its four phases: Inception, Elaboration, Construction, and Transition, ensure that the project is broken down into manageable modules, with each phase having specific goals and deliverables.

This approach facilitates dealing with the complexities of developing a software solution that integrates various technologies and third-party components. The iterative nature of RUP within an Agile framework allows for continuous feedback and iteration, ensuring the final product is closely aligned with user needs and the dynamic requirements of Colombo's parking infrastructure.

6.1 Programming Languages and Tools

The parking app project utilizes a modern technology stack to ensure high performance, reliability, and ease of use for both developers and end-users. The core technologies chosen for this project include:

- **Flutter with Dart:** Flutter, powered by the Dart programming language, is selected for mobile application development due to its cross-platform capabilities, allowing for a single codebase to generate apps for both Android and iOS devices. Its rich set of fully-customizable widgets enables the creation of native interfaces in minutes, making it a perfect choice for developing a user-friendly mobile app for parking management.

- Laravel 10.X: For the backend services, Laravel 10.X, a PHP web application framework, is employed. Known for its elegant syntax, Laravel makes development tasks such as authentication, routing, sessions, and caching straightforward, thereby speeding up the development process. Laravel's ecosystem also provides robust security features, which are critical for protecting user data in the parking app.

- **Tailwind CSS:** Tailwind CSS is used for styling the web components. It's a utility-first CSS framework packed with classes like flex, pt-4, text-center, and

rotate-90 that can be composed to build any design directly in the markup. Tailwind's approach minimizes the time spent styling the dashboard, making it a fast and efficient choice for creating a visually appealing user interface.

- **SoftUI Dashboard:** For the administrative panel, SoftUI Dashboard, a Bootstrap 5 based template, is integrated with Laravel. It offers a sophisticated design with a focus on usability and simplicity, facilitating the management of parking app data, including user and employee registrations, parking slot availability, and transactions.

6.2 Third Party Components and Libraries

To enhance functionality and ensure robustness, the following third-party components and libraries are integrated:

- **Google Map Navigation:** Essential for providing users with real-time directions to parking locations. The integration of Google Maps API allows the app to offer detailed maps, route planning, and location tracking, significantly enhancing the user experience by making it easier to find parking spaces in busy areas of Colombo.

- **Let's Encrypt SSL:** To ensure secure data transmission, Let's Encrypt SSL certificates are used. This free, automated, and open certificate authority (CA) provides digital certificates necessary for secure HTTPS (SSL/TLS) connections, safeguarding user data during transmission.

- **AWS Services:** The app's infrastructure is hosted on AWS, employing services such as EC2 for scalable computing capacity and RDS for MySQL databases. AWS EC2 instances host the backend services, ensuring high availability and scalability. The MySQL database on AWS RDS provides a reliable and secure data storage solution, crucial for managing the large volumes of data generated by the app.

These development tools and technologies have been carefully selected to ensure that the parking app is reliable, secure, user-friendly, and scalable. By leveraging the strengths of each component, the project aims to deliver a comprehensive solution that addresses the parking challenges in Colombo effectively.

07 Discussion

The interim report for our Sri Lanka parking app project has thoroughly detailed our analysis of Colombo's parking challenges, the feasibility of the proposed solution, and the planned system design. Key highlights include:

- **Problem Statement:** Concrete identification of Colombo's parking crisis stemming from a lack of spaces, inefficient management, and security vulnerabilities.

- **Objectives:** Establishing clear goals to reduce search times, increase information accessibility, minimize fraudulent practices, and ultimately create a better user experience.
- **Requirements Analysis:** Outlining 'Must-haves', 'Should-haves', and 'Could-haves', while prioritizing core functionalities and considering potential expansions.
- **Feasibility Study:** Analyzing operational, technical, and legal feasibility, demonstrating the project's viability and outlining strategies for user adoption.
- **System Architecture:** Visualizing the solution through design diagrams (Class Diagram, ER Diagram, and high-level Architecture Diagram), ensuring proper relationships and scalable structure.
- **Development Tools and Technologies:** Selecting a powerful tech stack including Flutter, Laravel, Tailwind, and integrating crucial third-party components for maps and security.

7.1 Summary of the Report

This interim report indicates that the parking app project is progressing well, having solidified the foundational analysis and design. The team has demonstrated a thorough understanding of Colombo's parking issues and is committed to developing a solution that addresses these challenges head-on.

7.2 Challenges Faced

Naturally, challenges have arisen during the project's early stages. These include:

- **Data Gathering:** Collecting reliable, quantitative data on parking patterns and scam prevalence was crucial but required significant primary research efforts.
- **Balancing Prioritization:** Striking a balance between core 'Must-have' features and desirable 'Should-have' or 'Could-have' enhancements to stay within budget and on schedule.
- **Legacy Infrastructure:** While the app is initially a standalone solution, future plans may consider integration with existing city systems, which poses potential complexity challenges.

7.3 Future Plans / Upcoming Work

The next phase focuses on the tangible development and implementation of the app, with significant milestones including:

- **Prototyping:** Creating an interactive prototype to allow for early user feedback and testing concepts such as map integration and UI flow.
- **Database Design:** Finalizing the database schema based on the ER Diagram, ensuring efficient data storage and retrieval for optimal app performance.

- **Security Implementation:** Implementing robust security measures throughout the development process to protect user information and transactions.
- **Scalability Planning:** While the focus is on immediate need, outlining a strategy for how the app will grow with Colombo, requiring future infrastructure and architecture considerations.
- **Deployment Timeline:** Establishing deployment phases, pilot programs, and a robust launch plan.

References

Eco Park: A low-cost and sustainable approach for smart ... - IEEE xplore. Available at: https://ieeexplore.ieee.org/document/10235241/ (Accessed: 19 November 2023).

Smart parking and charging system for E - IEEE xplore. Available at: https://ieeexplore.ieee.org/document/10157661 (Accessed: 19 November 2023).

Smart parking system for commercial stretch in Cities | IEEE conference ... Available at: https://ieeexplore.ieee.org/document/8286588 (Accessed: 19 November 2023).

IOT based vehicle parking model with mobile apps software | IEEE ... Available at: https://ieeexplore.ieee.org/document/9652765/ (Accessed: 19 November 2023).

Smart parking system (S-park) - IEEE xplore. Available at: https://ieeexplore.ieee.org/document/9350429/ (Accessed: 19 November 2023).

Construction and application evaluation of the traffic ... - IEEE xplore. Available at: https://ieeexplore.ieee.org/document/9115109/ (Accessed: 19 November 2023)

Park indicator: Book your parking spot - IEEE xplore. Available at: https://ieeexplore.ieee.org/document/8697378 (Accessed: 19 November 2023).

City of White Plains Parking app: Case study of a smart ... - IEEE xplore. Available at: https://ieeexplore.ieee.org/document/7371494/ (Accessed: 19 November 2023).

Dammulla, R. and Mudunkotuwa, R. (2022) Analysis on the road traffic congestion in Colombo Metropolitan Area, CINEC Academic Journal. Available at: https://caj.sljol.info/articles/10.4038/caj.v5i1.70 (Accessed: 19 November 2023).

Weerasinghe, T.D., Karunarathna, D.I.M. and Subashini, B.L.C. (2021) Effect of road traffic congestion on stress at work: Evidence from the employees working in metropolitan areas of Colombo, Sri Lanka, SSRN. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3853900 (Accessed: 19 November 2023).

Author links open overlay panelXiaofei Ye a d et al. (2021) Research on parking app choice behavior based on MNL, Travel Behaviour and Society. Available at: https://www.sciencedirect.com/science/article/abs/pii/S2214367X21000697 (Accessed: 19 November 2023).

Parking fees in Colombo: To pay or not to pay?Talking Economics. Available at: https://www.ips.lk/talkingeconomics/2020/01/21/parking-fees-in-colombo-to-pay-ornot-to-pay/ (Accessed: 20 November 2023).

Car parking app development: Expert tips (no date) agilie.com. Available at: https://agilie.com/blog/car-parking-app-development-expert-tips (Accessed: 20 November 2023).

Buitleir, D. de (2023) What is a parking management system? (explainer), Wayleadr Blog. Available at: https://wayleadr.com/blog/parking-management-system/ (Accessed: 20 November 2023).

Parking management system comprehensive guide | workero. Available at: https://www.workero.com/parking-management-system-guide/ (Accessed: 20 November 2023).

Vftsolution.com (no date) ZK Parking Management System Solutions, VFT HOLDINGS (PVT) LTD |Parking Management System Sri Lanka. Available at: https://vftholdings.lk/ZK_Parking_Management_solution_sri_lanka.php (Accessed: 20 November 2023).

6 steps to build a parking app (no date) Cleveroad Inc. - Web and App development company. Available at: https://www.cleveroad.com/blog/parking-app-development/ (Accessed: 20 November 2023).

Find out detailed information (2017) ParkMeApp. Available at: https://www.parkmeapp.com/blog/findoutdetailedinformation/ (Accessed: 20 November 2023).

Does traffic congestion slow the economy? - researchgate. Available at: https://www.researchgate.net/publication/258154937_Does_Traffic_Congestion_Slow_the_Economy (Accessed: 20 November 2023).

Kanishka Werawella presents solutions to traffic congestion in Sri Lanka: Institute of Policy Studies sri lanka (2020) Institute of policy studies Sri Lanka | Institute of policy studies Sri Lanka. Available at: https://www.ips.lk/kanishka-werawellapresents-solutions-to-traffic-congestion-in-sri-lanka/ (Accessed: 20 November 2023).

Councile, C.M. COLOMBO TO HAVE STRUCTURED PARKING WITH THE INTRODUCTION OF SOLAR POWERED PARKING METERS, Colombo Municipal Council. Available at: https://www.colombo.mc.gov.lk/news-more.php?id=30 (Accessed: 20 November 2023).

Driving into Colombo's money spinning car parks! - news features: Daily Mirror (2023) News Features | Daily Mirror. Available at: https://www.dailymirror.lk/newsfeatures/Driving-into-Colombos-money-spinning-car-parks/131-260089 (Accessed: 20 November 2023).

PUSL3190 Computing Individual Project

Project Initiation document

PARKING APPLICATION

Supervisor: Ms. Dulanjalee Wijesekara

Name: Rathnayake M Rathnayake
Plymouth Index Number: 10820945
Degree Program: Computer science

01 Introduction

Colombo is the capital city of Sri lanka. And it is the most attractive and valuable city in the whole country. And it can be called the spine of Sri lanka. Because it has commercially and politically significant factors which are caused by the economic development of Sri Lanka.

So, the above aspects are the biggest reason people have greater attraction to Colombo city. And it will cause too many transporter issues people have to face.

According to the Ministry of Transport over 500,000 vehicles come from Colombo city to different places around the country. So, there is a critical shortage of parking space in the Colombo area. The reason is vehicles demanding more parking areas around the Colombo city, but government and private authorities can't bear the higher demand because of their less amount of parking areas. Although they have needful parking areas for the vehicles, but they can't manage these parking slots with an effective system. Such as people are not aware of all the parking facilities around the Colombo city and they get caught by fraudsters around the city.

And another major problem is there are a lot of parking scams and ransom seekers around the city. Although someone finds a parking place to park their vehicle, it might be a not as safe as public parking.

Above parking facility issues might be a huge concern among the citizens of those areas. Because these issues can lead to problems such as traffic congestion, environmental pollution and health hazards. In towns and cities convenience parking facilities are more vital to everyday life.

According to statistics which are given by researchers, a huge number of Sri Lankan citizens spent their valuable time on the roads. Due to the lack of parking areas in Colombo, the number of vehicles on the road is beefing up and it will take more time to reach desired destinations. And it is the result of paying more money to fuel the vehicles.

Private sector incur at least 10 percent of loss in their revenue due their fuel and men hours of their whole production. In the end it will cause a decline in productivity in the economy of the country.

These are general issues regarding that parking problem area in Colombo city. But there is another major problem. In the Colombo area, there is no proper parking facility system developed by the government or authorized party. It is the major problem in this area. If the government can design a proper and effective parking system around the city it will help to reduce the above-mentioned inconvenience circumstances faced by people. So, we are trying to cope with these difficulties and find the best and effective solutions to overcome these issues which have resulted in endangering people's day to day lifestyle.

02 Business case

Economic Benefits of Parking App

Propose parking app is very essential for many ways in Sri Lanka. It is most important for the economic sector in the country. This app provides the best solutions for problems which are faced by society. So, there are more economic benefits than we think.

1. It helps to reduce traffic congestion in peak time.
    - In weekdays Colombo traffic congestion is highly increasing. So, because of that many people face lot of inconvenience during their valuable time. Traffic congestion is a headache for people who work from urban areas. In their off time they must go residential areas. If vehicles can't find proper parking place to park their vehicle, they park vehicle where they want. So, that will lead to traffic jam in the urban areas. By our parking app vehicle owners can find proper and safe places to park their vehicles without any inconveniences. If there is not any road blocking in the street there will be reduces traffic jam in the roads.
2. It will help to increase productivity in the economy.
    - Traffic congestion is a critical point in the economy. Sometimes it will cause a crush down the economy in the country. How traffic congestion that much valuable in the economy. Here is that, when working people spend more time on the road which is arising from traffic congestion it will lead to reduced productivity in the economy. Because working hours are reduced due to traffic congestion. By using this app traffic congestion can minimize during rush hours in the day. So, there is no need to waste valuable time of employees to going to their works. Researchers are finding that if there is low traffic congestion in the country it will lead to higher productivity in the economy.
3. This app led to reduced accidents on the roads during the rush time.
    - During the past few years road accidents rates have been rapidly increasing. That is a major problem in the economy and society of the country. In some ways traffic congestion is the reason for increasing that rate. If vehicles can't find a proper place to park their vehicles, they are roaming around the city for looking a

better place. So, that kind of unexamined driving will cause road accidents. Using our parking app will help to reduce road accidents which are caused from traffic congestion.

4. Through this app the government can collect fees from parking areas without being caught by scammers.
    - Many government owned parking areas are controlled by the private sector. But in most places is full of corruption. These private owners did not pay actual income to the government. And they are cover up real income which are generated from these parking area. Because of that state loose most of its income from these parking areas by caught frauds. And also vehicles owners are caught by that kind of ransom seekers in fake parking areas. So, that parking app is very useful for both government and people to do their work rightfully.
5. Parking area owners can generate income from this app without any hesitation.

03 project objectives

Objectives means what are the desired outcomes what we must achieved in our work. So, every work has a final objective. And also, in our parking app there will be specific objectives which will help to overcome some of the issues people are facing during there lifestyle.

We have plans to develop this parking app around Colombo city. Because Colombo is the capital of Sri Lanka, and it is busiest city in the country. During the peak hours of weekdays the city is facing worst traffic jam on the roads. And it will cost more than we imagined.

So, our main objective is to reduce traffic congestion by developing an effective parking app around the city. And it will help drivers to find the best location to park their vehicle and do not be part of the traffic congestion.

If vehicle has proper parking place to park their vehicle, they don’t need to park their vehicle Indiscriminately. And also, it will help to reduce accident on the roads.

And there is more objective we are planning to achieve through this parking app.

- To reduce scammers and ransom seekers from fake parking spaces.

There are a lot of vehicle drivers who get caught to scammer in the parking areas. They did not provide any kind of parking ticket or legally valued document to identify they are ticketers which are serving in parking areas.

Some other scammers also cheat the vehicle owners by promising to provide them with parking spaces. Because of that kind of incidents many people lost their own money to park their own vehicles. Furthermore, in some unsafe and unprotected parking areas, people get robbed and they bear many losses.

Those are the major problems which the people faced caused by fake parking areas. If we can use parking app it will help to resolve all the problems which are cruel to vehicle owners. Because this parking app can identify the most trusted parking areas around Colombo city. And this app only registered verified vehicle park owners in the Colombo area. And parking app users are also registered into the system. Because of this system there will be no any kind of scamming or cheating for users or parking area owners.

- To reduce cost which is caused by traffic congestions. Etc (Fuel cost, Labor hour cost, Accident)

Nowadays traffic congestion is a major problem in the highly populated cities in the world. High levels of traffic congestion cause many problems in society. Such as high level of traffic congestion causes a high level of fuel consumption in peak time in the day. And also it will cause environmental pollution.

Furthermore, traffic congestion causes road accidents. And it will cost it to more than ever we think.

Finally, the major problem of increasing traffic congestion in the road is scarcity of parking spaces around the city. If vehicles can’t find a proper place to park their vehicle, they park where they want and no matter what the outcomes arise from such careless vehicle parking.

So, through this parking app we are encouraged to reduce traffic congestion by locating available parking places for users during the peak times. If vehicle owners can find proper parking places without any kind of inconvenience they don’t want to park their vehicle Indiscriminately.

- To help to increase productivity of the country through effective transport. Etc (without traffic jam during the weekdays)
- To get effective utilization of parking areas around the Colombo city (some government owns parking slots are operating under insufficient system)

04 Literature Review

Several studies have proposed technology-based solutions to address parking challenges in urban areas like Colombo. Mobile apps are a popular approach, allowing drivers to check parking availability and make reservations. However, most parking apps rely on complex sensor networks and infrastructure changes, which can be expensive and difficult to implement on a large scale.

The proposed parking app system aims to improve parking convenience through a mobile platform while avoiding extensive infrastructure costs. It utilizes a straightforward registration system for drivers and parking employees, with key features like unique user IDs, vehicle profiles, and integrated payment processing. This allows drivers to easily find and pay for available parking spots, and employees to manage bookings.

A central advantage of the proposed system is that it does not require installation of sensors or cameras for space occupancy detection. Instead, it relies on parking employees updating availability information in real-time as cars enter and exit. This makes deployment faster and cheaper compared to sensor-based systems. While some studies have explored sensor less parking apps, they focus on public street parking whereas this system is designed for managed lots.

Additionally, the parking app incorporates a registration process for allowing private property owners to share their spaces, unlike apps tied to specific lots. Enabling crowdsourced sharing of private parking opens up more spaces. The ability to pay taxes also incentivizes property owners to participate. This can significantly increase parking supply in a cost-effective manner.

In terms of technology, the parking app uses common platforms like SQL databases for low-cost and widespread implementation . Competing systems relying on proprietary or complex technology may present interoperability issues and vendor lock-in effects. The simplicity of the proposed design allows scaling to meet demand.

The proposed system also prioritizes security by verifying driver and parking employee information during registration. Some studies note security risks with freely accessible public parking apps. Restricting access and using QR code scans for entry/exit enhances protection against misuse. This is important for user adoption.

While every parking solution has tradeoffs, the proposed app system balances convenience, security and costs. It provides the core functionality of availability information and reservations without expensive sensors. The decentralized and crowdsourced nature offers more flexibility than fixed infrastructure-based systems. With thoughtful design considering user needs and adoption barriers, this parking app system represents an efficient and practical approach to improving urban parking.

05 Method of Approach

Based on the parking application project proposal, i have decided to choose the Agile software development methodology. Agile promotes a iterative approach with continuous

feedback and improvements through the development lifecycle. It is appropriate for this project as the requirements may expand over time and the methodology allows incorporating changes swiftly with comparison to other traditional sequential approaches like waterfall. Agile delivers working software early through incremental builds. This make it easy to continuously refine and add features based on real user feedback instead of frozen requirements. As an individual developer, Agile also helps scope the work by breaking it down into manageable iterations.
Some key advantages of using Agile are:

- Flexibility to change requirements: As I develop the app, new ideas or improvements may come into view Agile allows me to easily make these changes at any stage rather than follow a rigid plan.

- Faster time-to-market: I can release an initial version sooner and then continuously improve it in response to real user feedback. No need to wait until every planned feature is built.

- Better quality: With very short development cycles and constant user testing, defects can be identified and fixed immediately leading to higher quality software.

- Enhanced collaboration: Agile emphasizes cross-functional collaboration. As an individual developer, I can leverage forums and communities to get feedback and input on my parking app.

- Coding: During each sprint, I rapidly develop working software to deliver the selected user stories as per best coding practices. I will use test-driven development to ensure quality.

Testing and Review: At the end of each sprint, I will rigorously test the working software, fix defects quickly, demonstrate it to sample users, and obtain their feedback. Reviews help to verify I am on track to solve real user problems, not just building features. Feedback may lead to new user stories being added to the backlog for future sprints.

I plan to release the app on a smaller scale and gather wide customer feedback using surveys, interviews and monitoring usage metrics. This will further help refine and verify product-market fit before wide public rollout.
To enable quick progress through sprints, I plan to leverage various technical enablers:

- Use Flutter framework for cross-platform support
- Firebase integration for cloud data storage and authentication
- GitHub for version control and code sharing
- Automated testing frameworks

I also plan to actively use developer forums and communities throughout for queries, insights, troubleshooting etc rather than take on everything alone.
Adopting Agile self-management rather than working to a pre-defined blueprint requires discipline. Pitfalls I need to be mindful of include:

- Scope creep from attempting too many new features
- Letting coding take precedence over testing

I also plan buffer periods between sprints to account for any overflow or delays. Ongoing customer validation is key rather than over-indexing on initial feedback.
In summary, Agile provides the right framework for driving this project iteratively at a self-managed pace. The focus is on building releasable software early through a collaborative approach centered on continuous testing and customer engagement. Disciplined backlog tracking and sprint execution will be key for progressing effectively. The end goal is to deliver a parking application with validated product-market fit in a quick and adaptable manner.

06 Initial project plan

To start, I will develop the mobile app for drivers and employees. I want to focus first on the backend and database because that is the most complex part.

First, I will build the driver side of the app. This will include payment processing, maps integration, QR code generation, and more. I am using Flutter to develop the app and Firebase for the database. Functionality is most important so the design will be simple for now.

Next, I will develop the employee side of the app. This will allow parking lot staff to verify payments, check in vehicles, and update availability. I will connect this to the driver side so parking fees can be calculated.

After the main app is working, I will create an admin website. This will let me manage user accounts and parking data. I want the admin site to control the whole system.

Once everything is functioning, I will improve the design to be intuitive and easy to use. I want even teenagers to quickly understand the app without confusion.

This covers the main pieces - the driver app, employee app, admin panel, and design. By breaking things down into smaller steps, I can take on the huge task of building this parking solution.

07 Risk Analysis

- Errors in the computer code. These can be very hard to fix. If I make mistakes writing the code, the app might not work right.

- Hacking. Since the app uses private information like bank details, someone could try to steal this. This would be a big problem.
- Issues with the GPS and maps. If the API key doesn't work or the navigation isn't accurate, the app won't be useful. This is risky because the whole app relies on the mapping feature.
- If I get sick or hurt, I might not be able to finish building the app on time.
- If my computer or phone breaks, I won't be able to work on the app.