**AN IOT-DRIVEN REAL TIME PARKING MANAGEMENT SYSTEM FOR ENCLOSED AREAS IN NAIROBI**

**Ann Wangari and Clifford Karani**

An Informatics and Computer Science Project Proposal Document Submitted to the School of Computing and Engineering Sciences in Partial Fulfillment of the Requirements for the Award of a Degree in Bachelor of Science in Informatics and Computer Science.

May 2024

**Declaration**

We declare that this project proposal has not been submitted to Strathmore University or any other University for the award of a Degree in Bachelor of Science in Informatics and Computer Science or any other Degree.

Student Name: Clifford Karani Koome

Admission Number:150489

Signature:

Date:

Student Name: Ann Wangari Kimani

Admission Number:150171

Signature:

Date:

Supervisor Name: Tiberius Tabulu

Signature:

Date:

**Approval**

The project proposal of Clifford Karani Koome and Ann Wangari Kimani was reviewed and approved by the following:

1.Tiberius Tabulu – Supervisor

**Abstract**

In Nairobi, the capital city of Kenya, there is an ever-growing population, most of whom commuting is a part of their daily lives. This has therefore caused a surge in the number of vehicles that maneuver in and around the capital city. Moreover, there is a limited number of parking spaces in Nairobi, especially within the CBD (Central Business District). There are parking space management systems currently being implemented within Nairobi such as through parking officers employed by the county government of Nairobi and automated ticketing parking systems.

The inefficiency of traditional parking management systems in closed areas, such as office buildings, is a persistent problem faced by drivers and owners of parking lots alike. Drivers also waste valuable time searching for empty spaces due to a lack of real-time data informing them of availability of a parking space and the location of the parking lot, as well as the parking space.

This project tackles this challenge by proposing a smart parking system powered by the Internet of Things (IoT). Sensors embedded in each parking space detect vehicle presence and transmit real-time data to a central hub. A web application displays this information, guiding drivers directly to available spots. Parking managers could also oversee the activities of their parking lot using the web application and make changes to their parking lot if necessary.

This IoT solution offers significant benefits. Real-time data ensures accurate parking availability, dramatically reducing wasted time and minimizing congestion within the closed parking lot. Additionally, the system can integrate with access control, streamlining entry and exit procedures. This solution also offers efficient management solutions to parking managers for parking lots.

By improving parking efficiency and user experience, this project presents a compelling solution for closed-area parking management.

**Table of Contents**

[**List of Figures** vi](#_Toc165745852)

[**List of Abbreviations** vi](#_Toc165745853)

[**Chapter One: Introduction** 1](#_Toc165745854)

[**1.1: Background** 1](#_Toc165745855)

[**1.2 Problem Statement** 2](#_Toc165745856)

[**1.3: General Aim** 2](#_Toc165745857)

[**1.3.1: Specific Objectives** 2](#_Toc165745858)

[**1.3.2: Research Questions** 3](#_Toc165745859)

[**1.4: Justification** 3](#_Toc165745860)

[**1.5 Scope and Limitations** 3](#_Toc165745861)

[**1.5.1 Scope of project** 3](#_Toc165745862)

[**1.5.2 Limitations in the Project** 4](#_Toc165745863)

[**Chapter Two: Literature Review** 5](#_Toc165745864)

[**2.1 Introduction** 5](#_Toc165745865)

[**2.2 Challenges in Nairobi's Parking Management** 5](#_Toc165745866)

[**2.3 Related Systems** 6](#_Toc165745867)

[**2.3.1 Smart Parking Systems by Intercomp** 6](#_Toc165745868)

[**2.3.2 Fleximodo** 7](#_Toc165745869)

[**2.3.3 Paytech** 9](#_Toc165745870)

[**2.4 Gaps in existing Application** 10](#_Toc165745871)

[**2.4.1: Smart Parking Systems by Intercomp** 10](#_Toc165745872)

[**2.4.2: Fleximodo** 11](#_Toc165745873)

[**2.4.3: Paytech** 11](#_Toc165745874)

[**2.5 Conceptual Framework** 12](#_Toc165745875)

[**Chapter Three: Development Methodology** 14](#_Toc165745876)

[**3.1 Introduction** 14](#_Toc165745877)

[**3.2 Software Development Methodology** 14](#_Toc165745878)

[**3.2.1 Requirements Engineering** 15](#_Toc165745879)

[**3.2.2 System Design** 16](#_Toc165745880)

[**3.2.3 Coding and Development** 16](#_Toc165745881)

[**3.2.4 Testing** 16](#_Toc165745882)

[**3.2.5 Documentation** 17](#_Toc165745883)

[**3.2.6 Deployment** 17](#_Toc165745884)

[**3.4 System Design** 18](#_Toc165745885)

[**3.4.1 Use case Diagram** 18](#_Toc165745886)

[**3.4.2 Class Diagram** 18](#_Toc165745887)

[**3.4.3 Entity Relationship diagram** 19](#_Toc165745888)

[**3.4.4 Database Schema** 19](#_Toc165745889)

[**3.5 System Development Tools and Techniques** 19](#_Toc165745890)

[**3.5.1 Integrated Development Environments (IDEs)** 19](#_Toc165745891)

[**3.5.2 Version control System (VCS)** 20](#_Toc165745892)

[**3.5.3 Code Review Tools** 20](#_Toc165745893)

[**3.5.4 Use Case Modeling** 20](#_Toc165745894)

[**3.5.5 Prototyping** 20](#_Toc165745895)

[**3.5.6 Scrum Framework** 20](#_Toc165745896)

[**3.6 Deliverables** 20](#_Toc165745897)

[**3.6.1 System Documentation** 20](#_Toc165745898)

[**3.6.2 IOT devices configurations** 21](#_Toc165745899)

[**References** 22](#_Toc165745900)

[**Appendix** 24](#_Toc165745901)

[**Appendix A1: Time Schedule** 24](#_Toc165745902)

# **List of Figures**

Figure 2.1: Smart Parking System by Intercomp website………………………………...6

Figure 2.2: Some of the Smart Parking System by Intercomp's features…………………7

Figure 2.3: Fleximodo’s Website………………………………………………………….8

Figure 2.4: Some of Fleximodo's features………………………………………………...8

Figure 2.5: Fleximodo's Management System……………………………………………9

Figure 2.6: Conceptual framework……………………………………………………….12

Figure 3.1: Agile Development Method……………………………………………….…15

Figure A1: Time Schedule for the development of the project…………………….….....24

# **List of Abbreviations**

**CBD-** Central Business District

**GUI-** Graphical User Interface

**IDE –** Integrated Development Environment

**IoT –** Internet of Things

**VCS** - Version control Systems

# **Chapter One: Introduction**

## **1.1: Background**

Nairobi, Kenya's capital city, is experiencing a period of vibrant growth. The population has surged in recent years, with projections suggesting a dramatic rise in the coming decade (Mairura, 2010). This influx of people brings exciting opportunities for economic development and a diversified workforce. However, it also creates challenges for urban infrastructure, particularly in areas like transportation (Nyamai & Schramm, 2023).

One major challenge is the strain on public transportation systems. As the population grows, the existing infrastructure struggles to keep pace, leading to overcrowding and unreliable service. This, in turn, has led many residents to seek alternative commuting options. One consequence is a significant rise in personal vehicle ownership, driven by the convenience and speed it offers. However, this has led to a surge in traffic congestion, particularly within the Central Business District (CBD). (Nyamai & Schramm, 2023)

This traffic congestion creates a critical bottleneck in the CBD, where a large portion of Nairobi's workforce is concentrated in office buildings. The limited availability of parking spaces in or near these buildings creates a constant struggle for drivers searching for a spot This wasted time searching for parking not only frustrates drivers but also further contributes to traffic congestion, creating a vicious cycle. (Omollo W. O., 2020)

The current parking management system in Nairobi relies on a manual process involving county executive officials who designate parking spaces, allocate a time limit for parking spaces, issue parking permits, collect fees and tow vehicles that break parking regulations. While this system functions in a basic manner, it is labor-intensive and requires a large workforce, which is costly to maintain. (Nairobi City County Transport Act no 3 of 2020).

Existing automated ticketing systems used in some locations like malls and airports offer a partial solution (Paytech, Solutions/paytech.com, 2021). However, their high infrastructure costs, involving individual tickets, ticketing machines, barriers, and payment systems for each parking lot, make them impractical for widespread implementation.

## **1.2 Problem Statement**

Due to the lack of real-time data on parking availability, drivers in closed parking areas frequently experience frustration and waste time looking for a parking space. Underutilised spaces due to manual or partially automated systems' inability to accurately track parking occupancy. Particularly in enclosed spaces such as malls and airports, this condition worsens traffic congestion as cars spend a significant amount of time searching for parking. Additionally, the county government is burdened by the challenges of its current parking system, as manual systems require a large workforce, resulting in higher operational costs.

The aim of this project is to develop a smarter and more efficient parking management solution to maximize available parking space, enhance user experience, and potentially generate additional revenue for the county government.

## **1.3: General Aim**

The objective of this project is to identify challenges with the current parking systems, analyze methods of management of current parking systems and develop an effective and efficient smart parking system which allows residents of Nairobi to locate a nearby parking lot, find a parking space with ease and to pay online and parking managers to manage their parking lots remotely without depending on a large workforce to oversee the activities of their parking lot by using the smart parking system website and compare the developed smart parking system with current smart parking systems.

### **1.3.1: Specific Objectives**

1. To review the importance of efficient smart parking systems, their constraints, and metrics of utilization.
2. To investigate the challenges with the current parking systems currently in Nairobi.
3. To analyze the effectiveness of the management methods of current parking systems in Nairobi.
4. To develop a smart parking system that integrates physical IoT components with a user-friendly website.
5. To test the integration of hardware and software components of the smart parking system.

### **1.3.2: Research Questions**

1. What are the constraints and metrics of utilization in efficient smart parking systems, and how do they impact overall parking management efficiency?
2. What are the challenges of current smart parking systems in Nairobi?
3. How effective are the current parking management systems in Nairobi?
4. How will an effective and efficient smart parking system be developed?
5. How will the hardware and software components of the smart parking system be tested?

## **1.4: Justification**

In Nairobi, there are many people who own vehicles and work in areas in and around the CBD which is often congested and is difficult to find parking spaces. As a result, many vehicle owners spend a lot of crucial time looking for parking spaces. In addition, the county government of Nairobi would not have to employ as many county officials in charge of parking spaces as parking management would rely less on manual labor, but not eliminate the need for them. The county government of Nairobi would therefore save on county expenditure as manual labor for parking management would be drastically reduced. A smart parking system would significantly reduce the amount of time spent looking for a parking space as it would direct vehicle owners to the nearest parking lot and to an available parking space while allowing them to pay online using the smart parking system website. Furthermore, parking lot managers could oversee their parking lots remotely using the smart parking system website thus reducing the dependence on several parking attendants to supervise the activities of the parking lots.

## 

## **1.5 Scope and Limitations**

### **1.5.1 Scope of project**

The project will aim at developing a system that will manage parking spaces within Nairobi mainly focusing in areas where parking managers or organizations own both public and private closed parking spaces and will be able to manage their respective parking spaces through the system. However, if the implementation is successful, the scope may be expanded to fit other cities and towns within the country as well as open area parking spaces. The system will aim to efficiently automate the payment processes and management of the parking spaces.

### **1.5.2 Limitations in the Project**

The proposed smart parking system using an IoT network offers a promising solution, but some limitations require consideration.

Sensor range and reliability may be affected by improper parking, large vehicles, or malfunctioning equipment. Network connectivity issues could lead to data delays and impact real-time accuracy. Security measures are crucial to protect user privacy and prevent unauthorized access to the system. The initial investment in sensors, a central hub, and web app development needs to be weighed against long-term cost savings from improved efficiency. User adoption of the web app is vital, and factors like user awareness and web app usability need to be addressed. Finally, integrating the system with existing parking management systems or access control mechanisms might pose compatibility challenges.

Addressing these limitations through redundancy measures, robust network solutions, strong cybersecurity protocols, cost-benefit analysis, user education campaigns, and ensuring compatibility will be crucial for the project's success.

# **Chapter Two: Literature Review**

## **2.1 Introduction**

This chapter critically analyzes existing research on parking management systems, aiming to identify effective solutions that address the challenges faced in Nairobi County. By examining research conducted by scholars worldwide, we can identify best practices and potential gaps in knowledge specifically relevant to the needs of Nairobi's vehicle owners.

## **2.2 Challenges in Nairobi's Parking Management**

The current parking management system in Nairobi County, while functional, faces limitations that hinder its effectiveness. These limitations contribute to inefficiencies and a negative user experience for both the county government and drivers.

Firstly, the system's reliance on manual patrolling and ticket issuance necessitates a large workforce. This translates to high operational costs for the county government, straining its resources. Secondly, the manual approach is susceptible to human error and potential corruption. Drivers may exploit weaknesses in monitoring to avoid paying parking fees, leading to lost revenue for the county.

Furthermore, the lack of real-time information for drivers about available parking spaces creates a significant challenge. Drivers waste precious time searching for a spot, contributing to traffic congestion throughout the city. There is inefficient search behavior, which accounts for over 30% of traffic congestion (Omollo W. O., 2020). Finally, the current system struggles to accurately track vacant spaces, potentially leading to missed opportunities to utilize available parking efficiently. This creates the perception of limited parking options, further frustrating drivers and hindering optimal use of parking resources.

These limitations highlight the need for a modernized parking management system that promotes efficiency and improves the user experience for drivers in Nairobi County. Such a system would not only optimize parking space utilization but also generate additional revenue through improved management and reduced opportunities for loss.

There are digital management systems in some closed area parking spaces. These are evident in some malls or corporate organizations. These systems face challenges where vacant spaces are not accurately recorded and may lead to congestion as well as drivers are not aware of available parking spots before going to the parking area which may lead to them wasting time if there are no spaces available which ultimately leads to inefficiency and more traffic.

## **2.3 Related Systems**

### **2.3.1 Smart Parking Systems by Intercomp**

The Smart Parking System by Intercomp is a Venezia, Italy based smart parking solutions agency that utilizes an innovative approach for the effective and comprehensive management of on-street parking in European cities. They utilize a sensor as an input, which provides real time information about the occupancy of a parking space. A central server receives the input data through radio communication. The website also guarantees user data security in their storage of data on their servers. They also use cameras for surveillance and a web application for their smart parking solution.

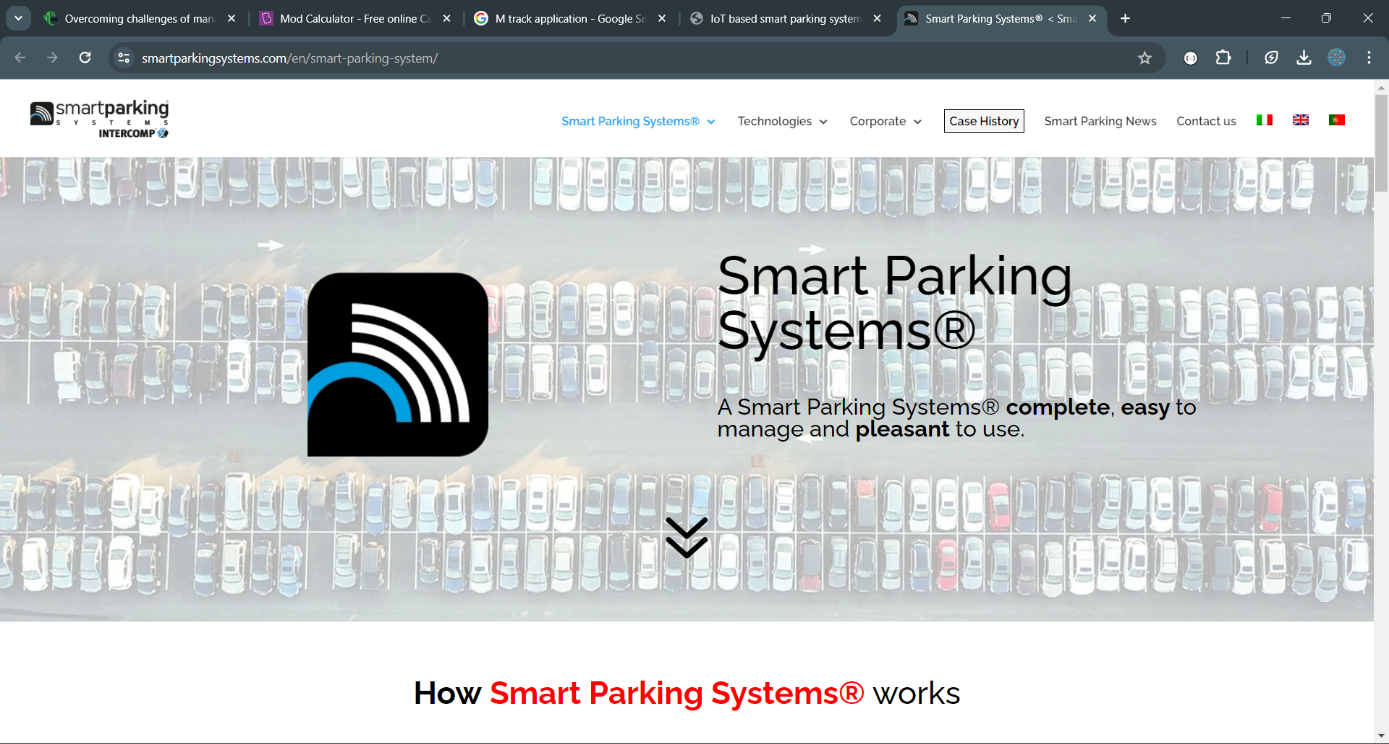


Figure 2.1: Smart Parking System by Intercomp : <https://smartparkingsystems.com/en/smart-parking-system/>

**A screenshot of a computer

Description automatically generated**

Figure 2.2.: Some of Smart Parking System by Intercomp’s features : <https://smartparkingsystems.com/en/smart-parking-system/>

### **2.3.2 Fleximodo**

Fleximodo utilizes secure authentication protocols to identify vehicles by utilizing an IoT permit card, which in even used in outside parking spaces. They use a triple sensing approach by using IoT PS Mini parking sensors to verify that a parking space has been occupied. They have a license plate recognition system which is used to accurately verify the vehicle occupying a parking space at a particular time based on their license plate number. There is real time monitoring of sensors in parking spaces using a web application that notifies the administrator when there is an issue with the sensors. Their parking sensors are also remotely controlled using an Over-The-Air transmission feature, thus saving time and resources.

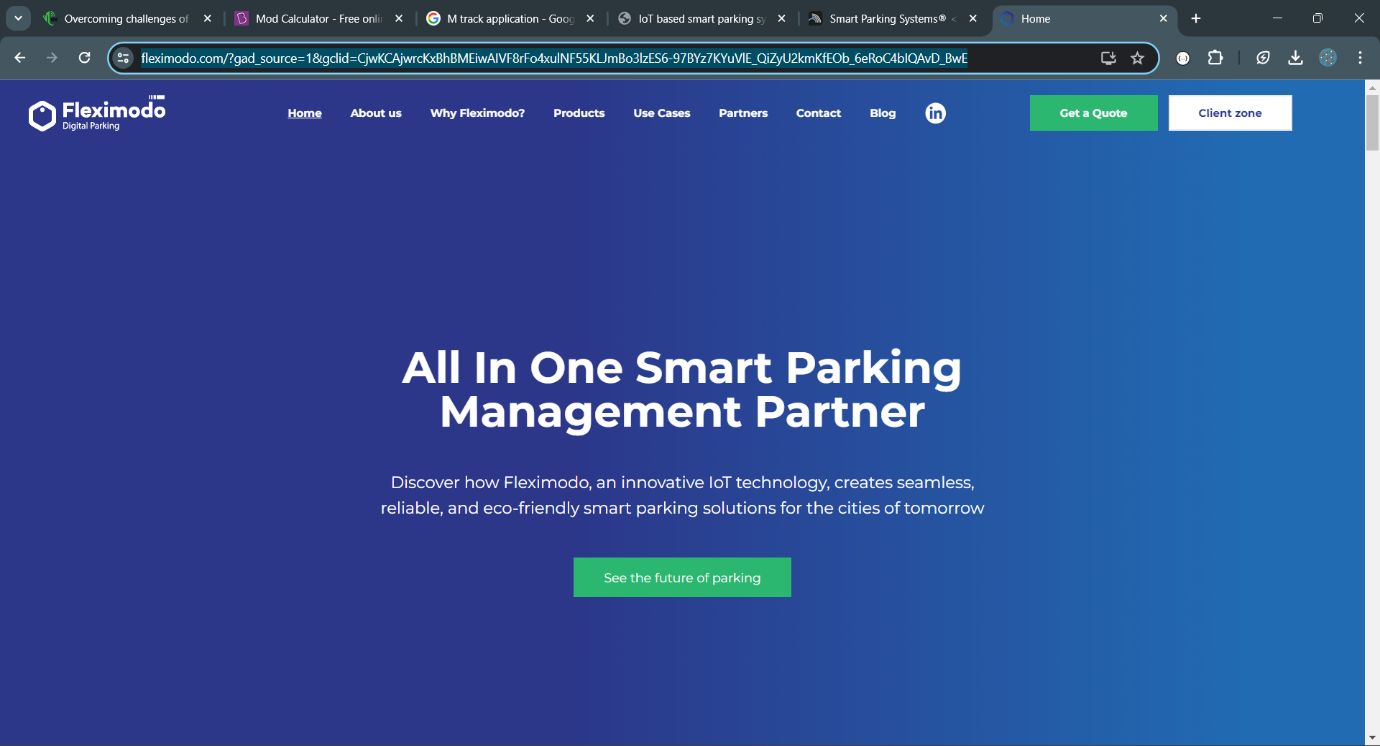


Figure 2.3.: Fleximodo’s Website: <https://www.fleximodo.com/>

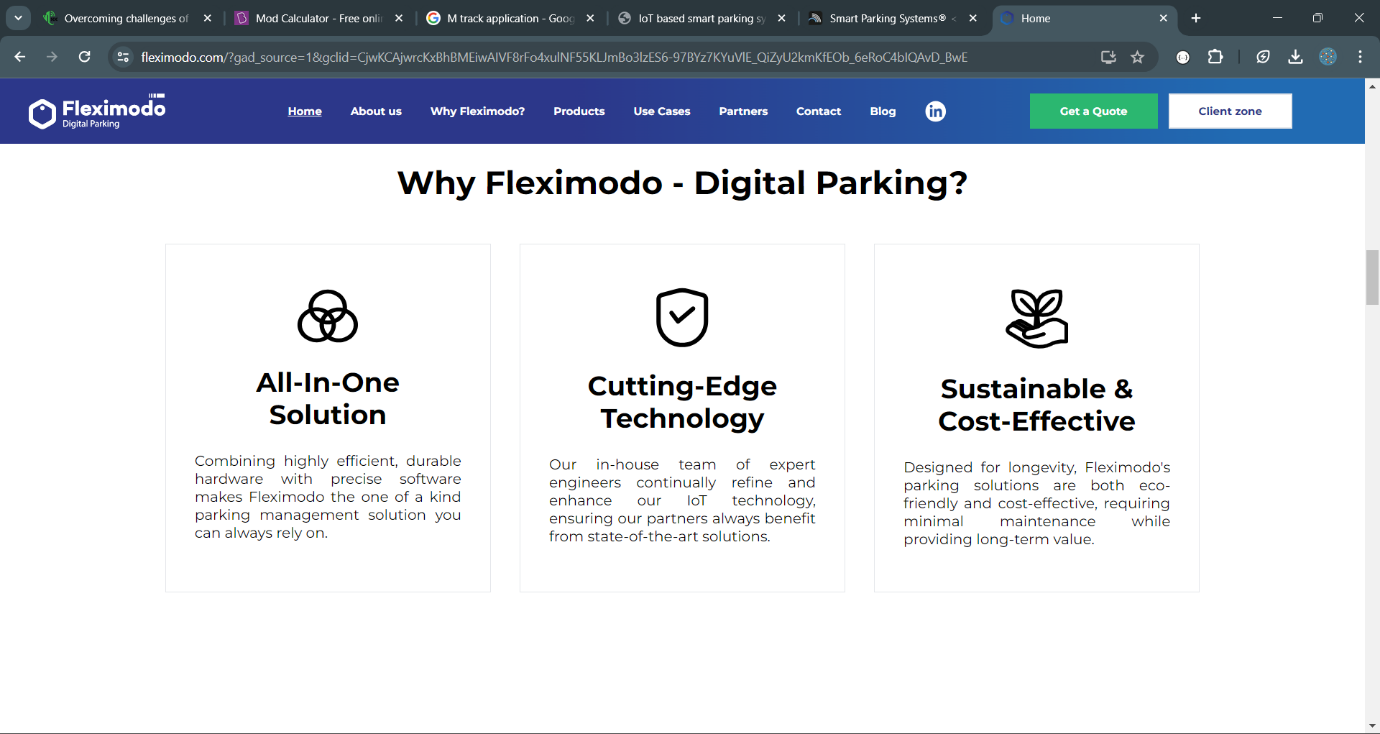
****

Figure 2.4: Some of Fleximodo's features: <https://www.fleximodo.com/why-fleximodo>

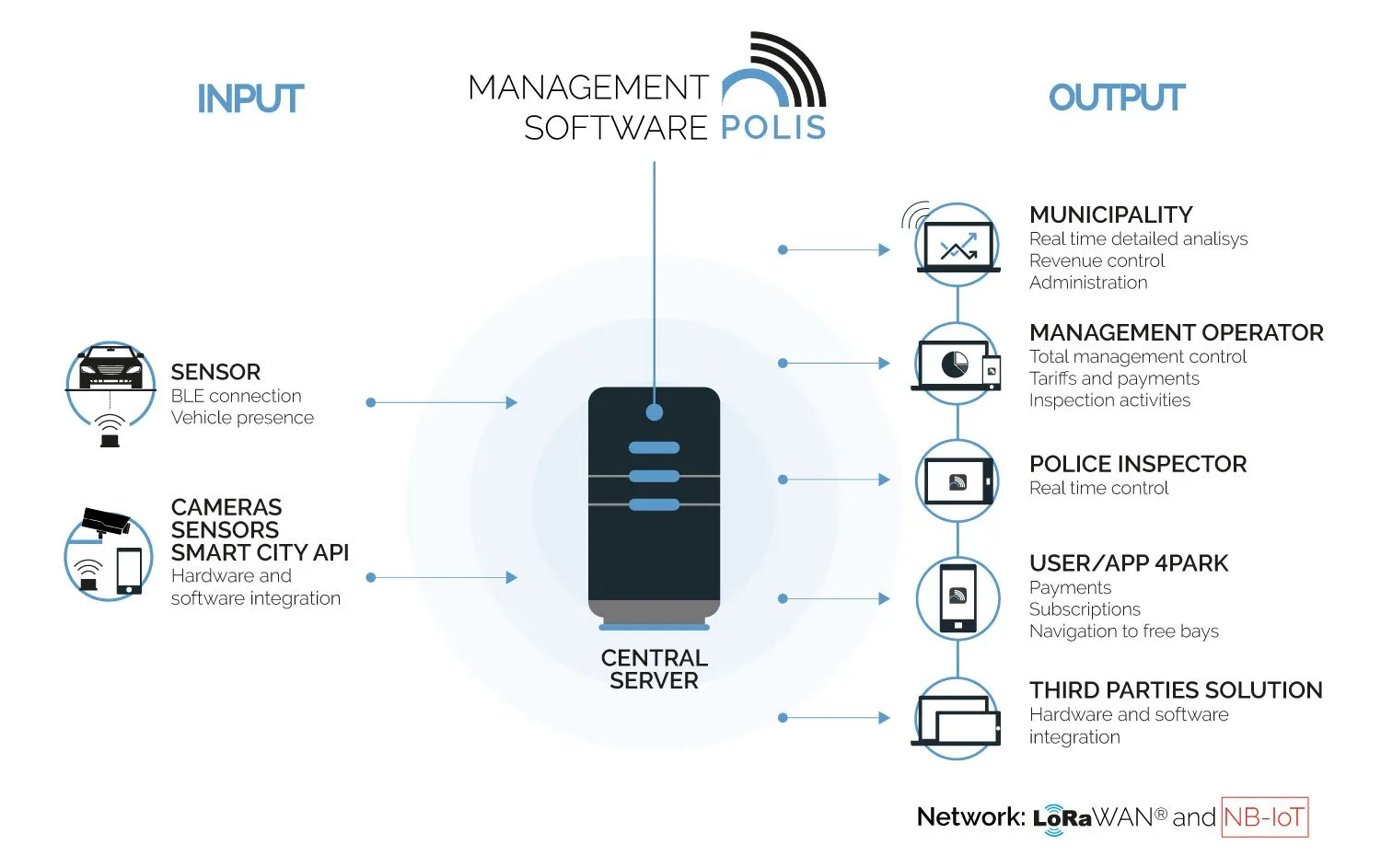


Figure 2.5: Fleximodo's Management System: <https://www.fleximodo.com/>

### **2.3.3 Paytech**

Paytech is a Nairobi-based parking, payment, and revenue-collecting service provider. They offer smart parking solutions, automated fare services and efficient energy solutions for sustainable facilities. They offer short term physical parking tickets using ticket vending machines and are checked using ticket verification machines, provide access cards, ticketless entry or exit by using their website or web app and license plate recognition. They also use payment machines particularly for users who opt to pay for parking by cash. They also implement an Automated Fare Collection System which integrates with their Access Control System which ensures that only ticketed or access-granted drivers can access the parking lots that supports this system by using barriers. (Paytech, Solutions/paytech.com, 2021)

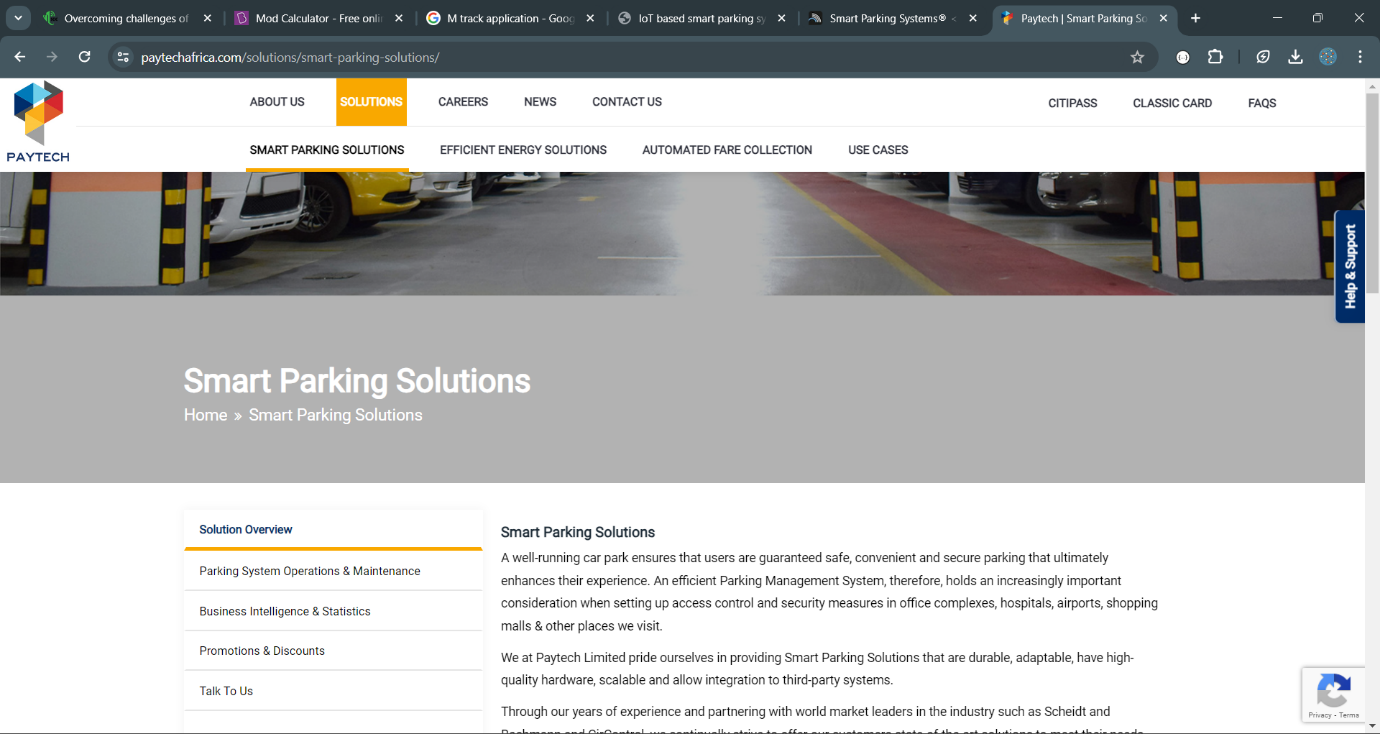


Figure 2.6: Paytech's Website: <https://paytechafrica.com/>

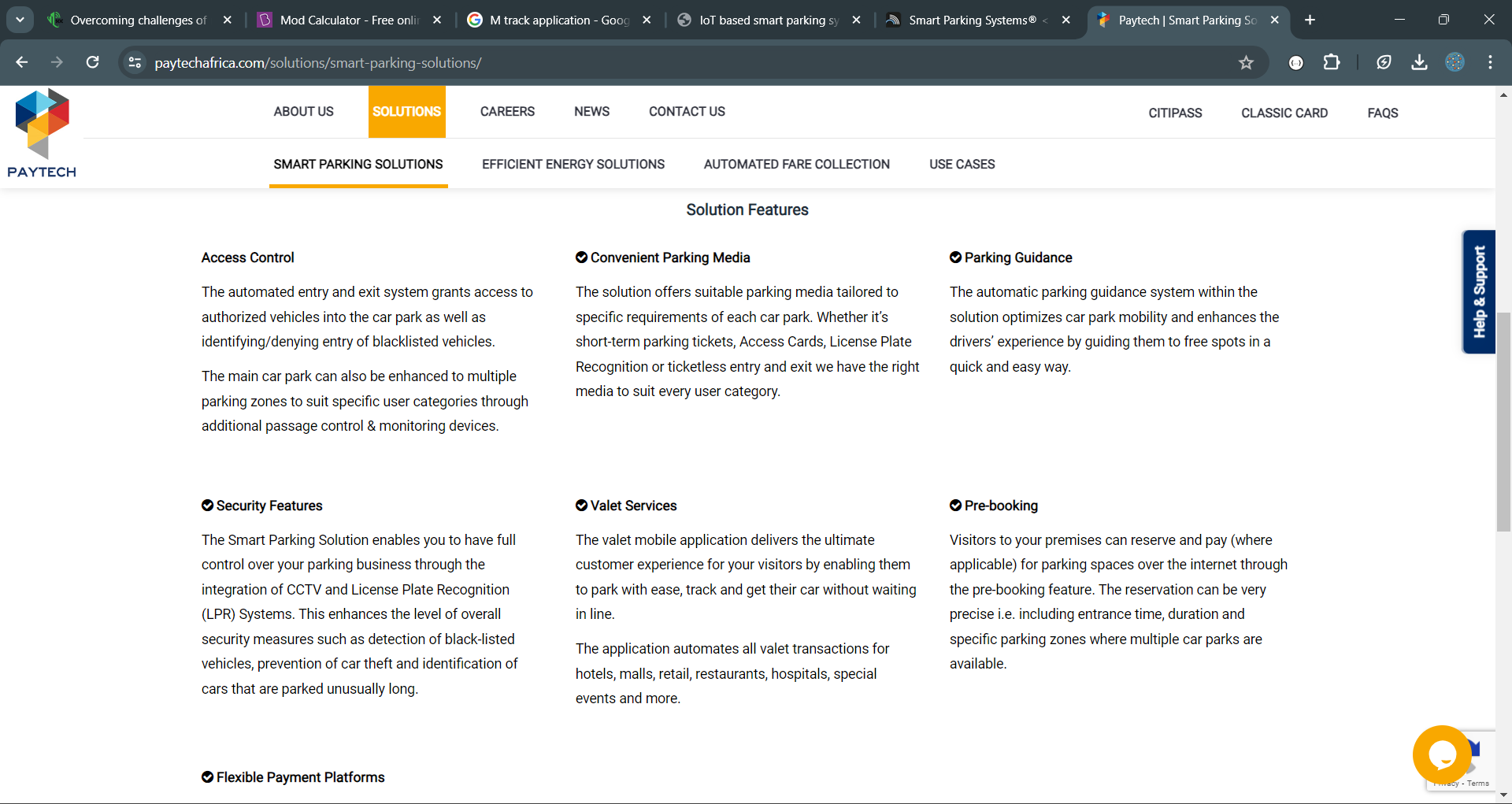


Figure 2.7 : Some of paytech's features: <https://paytechafrica.com/solutions/smart-parking-solutions/>

## **2.4 Gaps in existing Application**

### **2.4.1: Smart Parking Systems by Intercomp**

Smart Parking Systems by Intercomp limits its users to on-street parking, particularly European users who live in cities that on-street parking is the major type of parking space that is being utilized (S.p.A, n.d.). Their system ignores other types of parking spaces, and users in cities in countries that are outside of Europe, especially African countries. This project aims to provide IoT-based, smart parking solutions to users in Nairobi, Kenya, particularly in closed parking spaces.

### **2.4.2: Fleximodo**

Fleximodo implements three IoT PS Mini parking sensors in a single parking space (fleximodo.com, Why fleximodo?, n.d.), which although it promotes accountability, is expensive as three sensors are purchased for one parking space and becomes costly as compared to when one sensor is purchased for several parking spaces. In this project, only one ultrasonic sensor will be used together with one Arduino microcontroller module per parking space which is cost effective and parking spaces would be verified if they are occupied using one sensor.

### **2.4.3: Paytech**

Paytech uses ticket vending machines, ticket verification machines, barriers, payment machines, as well as physical parking tickets (Paytech, Solutions/paytech.com, 2021). A ticket vending machine is Ksh320,000 on Aliexpress.com, n.d., a handheld ticket verification machine costs Ksh13,094 on IndiaMart.com, n.d., an automatic parking barrier is Ksh230,000 minimum on Techyshop Kenya, 2023 and a parking payment machine is Ksh927,430.00 on madeinchina.com not inclusive of delivery as some machines are sourced from abroad. Evidently, these machines are expensive to purchase and maintain, and if one machine breaks down, there is a delay in offering parking services or a delay in payment which is also time-consuming. This is comparable to the cost of an Arduino microcontroller which is about Kshs3000 and an ultrasonic sensor of about Kshs800 both on Jumia.

## **2.5 Conceptual Framework**

Figure 2.5.1: Conceptual framework

IoT devices such as Arduino Microcontroller and ultrasonic sensors

Gateway

(MQTT Broker)

Web application

M-pesa payment, Map, Data visualisation

choose parking lot,

get directions.

pay

view history.

remove, add parking lots.

ban users.

add, remove parking managers.

view stats of all lots

add, remove spaces.

free spaces

view stats of their own lot

**2.5.1: Data Collection with Enhanced Security**

The system relies on a network of strategically placed ultrasonic sensors. These sensors act as the system's eyes, detecting vehicle presence in each parking space and collecting real-time data on occupancy. An Arduino controller serves as the local data collection unit. This microcontroller gathers data from the sensors and securely transmits it to the central hub using the MQTT broker as a secure communication channel. The MQTT broker encrypts this data, ensuring its safe transmission to the web portal. This focus on data security safeguards sensitive information and ensures the integrity of the parking space data.

**2.5.2: Centralized Data Processing and Management**

The web portal serves as the system's brain. It receives encrypted data from the MQTT broker, decrypts it, and processes the information on parking space availability. This processed data is then stored and visualized within the web portal, providing a clear and centralized overview of parking space occupancy. Parking managers have access to a dedicated interface within the web portal. This interface empowers them to manage their assigned parking areas. They can add new parking spaces within their designated zone, modify existing ones, or remove them as needed, keeping the system's database up to date. Additionally, parking managers can access real-time parking occupancy data for their assigned area. This real-time data allows them to optimize parking space allocation and ensure efficient utilization within their zone.

**2.5.3: Centralized Control and System Monitoring**

A dedicated admin interface within the web portal provides comprehensive control over the entire system. The admin plays a crucial role, assigning and managing different parking managers for specific areas, ensuring proper oversight and control. Furthermore, the admin can monitor overall system health and performance. This monitoring capability allows for proactive maintenance and troubleshooting, ensuring the system operates smoothly and efficiently.

# **Chapter Three: Development Methodology**

## **3.1 Introduction**

This chapter explores the methodology for software development and various aspects of system analysis and design. It also emphasizes the tools and techniques to be employed in application development, as well as the anticipated system deliverables upon completion of the development process.

## **3.2 Software Development Methodology**

A software development methodology is a framework that defines a structured process for planning, developing, deploying, and maintaining software applications. It outlines a set of practices, tools, and techniques that guide the development lifecycle and ensure the project is delivered efficiently and effectively.

In this project we will use Scrum framework, which is an of Agile development methodology. This methodology is suitable for our project since it prioritizes iterative development cycles with frequent user feedback loops. This allows for continuous adaptation throughout the development process and ensures the final product closely aligns with user needs. In this project, user needs include real-time parking information access and a user-friendly interface.

Additionally, the project might encounter unforeseen challenges or require adjustments based on user feedback. Scrum’s iterative nature allows for flexibility in adapting to changing requirements.

Furthermore, by breaking down the project into smaller, manageable tasks within each development sprint, Agile reduces overall project risk. This allows for early identification and mitigation of potential issues throughout the development process.

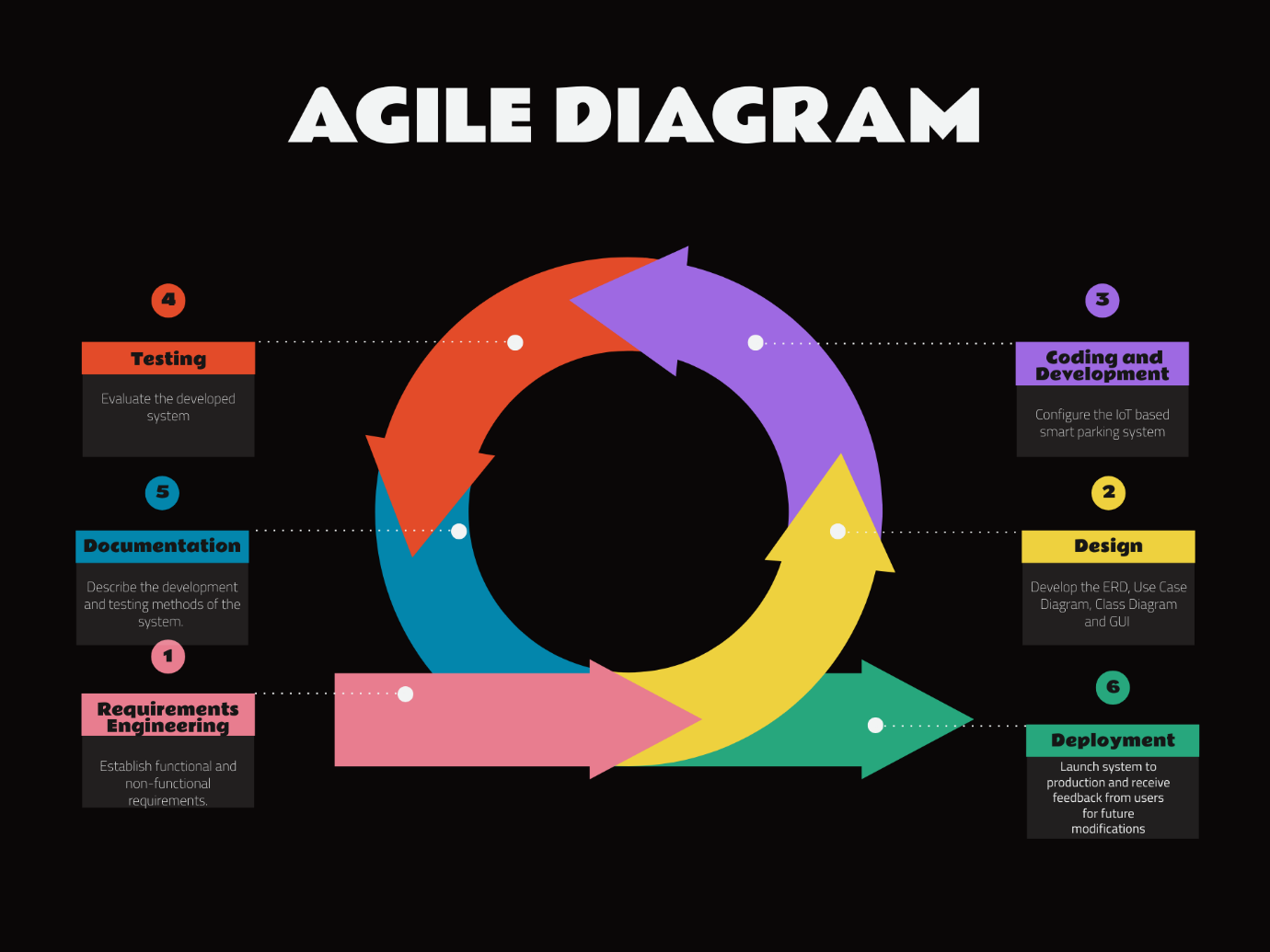


Figure 3.2.1: Agile Development Method, which Scrum framework is a part of

## **3.2.1 Requirements Engineering**

This initial stage involves gathering and documenting the specific functionalities and characteristics (how the system should behave) the system must possess. This will be achieved through a combination of techniques.

Firstly, there will be use of existing document analysis. Reviewing any existing documentation related to parking procedures or maps will provide valuable context for understanding current parking management practices.

Secondly, we will also come up with the functional and non-functional requirements. This will be done by brainstorming and online research on existing systems to be able to well define them.

By combining these techniques, a clear and concise set of requirements will be established that guide the development process. This ensures the final system addresses the needs of all stakeholders.

### **3.2.2 System Design**

With a clear understanding of the requirements, the system design stage will follow.This stage involves translating requirements into a technical blueprint for the system.

A use case diagram will be created to illustrate how different user groups (drivers and administrators) will interact with the system. This will visually represent the functionalities each user group can perform. For example, drivers searching for parking, administrators adding new parking spaces.

A class diagram and an Entity relationship diagram will be created as well. These diagrams will define the various components that make up the system and how they interact with each other. For example, a class will represent the sensor, another will represent the data processing module, and another will represent the user interface.

The Graphical User Interface (GUI) will be designed which will be used in the web application. This will create a better picture of the various functionalities of the system and for each user. It will also guide when reaching the coding and development.

### **3.2.3 Coding and Development**

This stage is where the actual coding and development of the system takes place. The functionalities identified in the requirements engineering stage will be broken down into smaller, more manageable tasks. These tasks will be grouped into short-term development cycles called sprints that both team members will work with.

In these sprints the focus will be using appropriate programming languages and tools to develop the functionalities for each sprint. This will involve coding functionalities like sensor data collection, data processing, and user interface development of the web portal.

### **3.2.4 Testing**

Unit testing will be conducted to ensure individual parts of the code function as intended and process integrate the newly developed functionalities with existing parts of the system to ensure everything works together seamlessly. This helps identify and fix bugs early in the development.

Various testing tools such as unit testing, integration testing will be used to ensure each module of the system is well integrated with each other and works seamlessly and performance testing to ensure quick and efficient functioning of the system and stability as well as scalability.

### **3.2.5 Documentation**

A comprehensive document will be maintained outlining the functional and non-functional requirements of the system. This serves as a reference point for the entire development team. The system design diagrams (use case diagrams and class diagrams) will be documented to illustrate the technical architecture of the system. A system user manual will be developed. The manual will provide clear instructions on how to use the system's functionalities effectively.

### **3.2.6 Deployment**

Deployment stage marks the culmination of the development process, where the system transitions from a development environment to a real-world setting. The system will be configured for the chosen deployment environment, ensuring compatibility and proper operation within that environment. Rigorous testing will be conducted in this deployed state to verify the system functions as intended and integrate seamlessly with any existing infrastructure.

**3.3 Software Requirements Analysis**

Software Requirements Analysis is the initial phase of the software development lifecycle (SDLC) that focuses on gathering, documenting, and analyzing the specific needs and functionalities a software system should possess. This analysis plays a critical role in ensuring the final software product meets the expectations of its users and stakeholders.

**3.3.1 System Narrative**

This section is a written account that describes the flow of events and interactions within the system. It essentially tells the story of how the system works from the perspective of the user and the system itself.

In Nairobi, finding a parking spot can be a daily challenge. To solve this problem, we set out to create an IOT-driven real time parking management system for enclosed areas in Nairobi that would make parking easier and more efficient for everyone.

Our system relies on a network of ultrasonic sensors placed in each parking space. These sensors detect the presence of vehicles and send real-time data to an Arduino controller, which acts as the local data collection unit. The Arduino controller securely transmits this data to a central hub using an MQTT broker, ensuring the safety and integrity of the data.

The central hub, a web portal, receives the encrypted data from the MQTT broker, decrypts it, and processes the information on parking space availability. This processed data is then stored and visualized within the web portal, providing users with a clear overview of parking space occupancy.

Parking managers play a crucial role in our system. They have access to a dedicated interface within the web portal, allowing them to manage their assigned parking areas. They can add, modify, or remove parking spaces as needed, ensuring that the system's database is always up to date. Additionally, parking managers can access real-time parking occupancy data for their assigned area, enabling them to optimize parking space allocation and ensure efficient utilization.

At the heart of the system is the admin interface, which provides comprehensive control over the entire system. The admin can assign and manage parking managers for specific areas, monitor overall system health and performance, and ensure proper oversight and control.

## **3.4 System Design**

This section outlines the system's architecture, elements, modules, interfaces, and data in order to meet the given specifications. It provides a blueprint that describes the structure, behaviour, and interactions between the system and users as well as other systems.

### **3.4.1 Use case Diagram**

A use case diagram will be used to visualize the actors in the project such as a driver, a parking manager and administrator and their respective functions, together with the scope of the project. This is also a key facilitator for gathering and assessing requirements that will be implemented in the system and depicting how a user would interact with the proposed smart parking solution.

### **3.4.2 Class Diagram**

There will be a class diagram which will be used to illustrate a system’s structure, its various classes, and objects in the classes as well as their attributes. It would also help in segmenting the systems into modules that can be manipulated with ease.

### **3.4.3 Entity Relationship diagram**

An Entity-Relationship Diagram (ERD) acts as a visual blueprint, depicting the system's data model. It showcases the key entities (parking spaces, sensors, users) and how they interact with each other. This clear representation offers several advantages: it fosters smooth communication between developers and stakeholders, aids in identifying data requirements early on, and helps prevent inconsistencies in the system's logic. Furthermore, the ERD serves as a foundation for designing the database schema, ensuring the database effectively stores and manages the information crucial for the Smart Parking System to function efficiently.

### **3.4.4 Database Schema**

There will be various tables in the database. Firstly, a parking space table. It will have a unique identifier for the parking space, description of the parking space location, current occupancy of the space and sensors ID to uniquely identify the sensors.

There will be a sensor table that will have a unique identifier for the sensor and key referencing the Parking Space this sensor is installed in.

Also, a user table to identify all types of users with their credentials and type of user.

The sensor table will have a relationship with the parking space table through the unique identifier of the sensor. Supposing a sensor is at same location as other sensors but on different floors, they will be labelled accordingly.

## **3.5 System Development Tools and Techniques**

This section outlines tools and techniques to be used in the process of designing, building, testing, and deploying software systems. These tools and techniques help to manage the complexity of system development and ensure that the final product meets the desired requirements.

### **3.5.1** **Integrated Development Environments (IDEs)**

There will be various IDEs used. We will use visual studio code when designing the GUI and functionalities of the web portal. We will also use Arduino IDE to program the Arduino controller that will manage the sensors. These IDEs were chosen since they have inbuilt tools that aid in writing and testing of program code.

### **3.5.2 Version control System (VCS)**

The main VCS that will be used is git, which is a tool used for source code management such as to help in management of changes to source code, track versions, and collaborate effectively. This VCS was chosen since it is easy to use and familiar.

### **3.5.3 Code Review Tools**

The code review tool to be used is GitHub that will facilitate code reviews, allowing developers to collaborate, identify issues, and improve code quality. This was chosen since it is easy to use, and we are familiar with its functionalities.

### **3.5.4 Use Case Modeling**

Use case modeling will be used to define the various interactions between users (parking lot managers, drivers, admin) and the system. This technique is chosen because it helps to identify and understand the different types of users and their goals, which is crucial for designing a system that meets their needs effectively.

### **3.5.5 Prototyping**

Prototyping will be used to create a simplified version of the system that demonstrates key features and interactions. This will help in detecting issues early in the development process that might come up later, allowing for iterative improvements and ensuring that the final system meets user expectations.

### **3.5.6 Scrum Framework**

Scrum framework, which is an Agile development methodology, will be used to manage the development process. Agile is chosen for its iterative and incremental approach, which allows for flexibility in responding to changing requirements and priorities. This approach also promotes collaboration and communication.

## **3.6 Deliverables**

### **3.6.1 System Documentation**

This focuses on System Documentation, including detailed documentation of the system requirements, design, and architecture. User manuals and guides, as well as progress reports, risk assessments, mitigation strategies, and lessons learned will also be included in the system documentation. This module will provide a comprehensive reference for developers, testers, and stakeholders, outlining the system's functionality and how it meets user requirements.

### **3.6.2 IOT devices configurations**

This involves configuring IoT devices such as sensors and controllers. This module will provide instructions for setting up and connecting IoT devices to the smart parking system, ensuring they are properly configured and integrated into the system.

# **References**

Aliexpress.com. (n.d.). *Magic Box Consumer Electronics.com*. Retrieved from aliexpress.com: https://www.aliexpress.com/i/1005006261830923.html

fleximodo.com. (2021, November 03). *Why fleximodo?* Retrieved from fleximodo.com: https://www.fleximodo.com/why-fleximodo

fleximodo.com. (n.d.). *Why fleximodo?* Retrieved from fleximodo.com: https://www.fleximodo.com/why-fleximodo

IndiaMart.com. (n.d.). *Parking Ticket Machine*. Retrieved from indiamart.com: https://www.indiamart.com/proddetail/parking-validation-machine-3991983648.html

Kenya, T. (2023, October 26). *Automatic Gate and Parking Barriers*. Retrieved from techyshop.com: https://www.techyshop.co.ke/automatic-gate-and-parking-barriers/

madeinchina.com. (n.d.). *Payment Kiosk*. Retrieved from madeinchina.com: https://jutaigateopener.en.made-in-china.com/product/lxmUIDQYRNhL/China-Customized-Logo-Smart-Parking-Payment-Machine-for-Sale.html

Mairura, O. (2010). Nairobi-emerging metropolitan region: Development planning and management opportunities and challenges. *ISOCARP Congress* (pp. 3-8). Nairobi: University of Nairobi.

Nairobi City County Transport Act no 3 of 2020, part II S.5 (c) (d) part VII S.49 (1) (2) S.53(1) S.57(2) S.58(1).

Nyamai, D. N., & Schramm, S. (2023). Accessibility ,mobility, and spatial justice in Nairobi, Kenya. *Journal of Urban Affairs*, 369-371.

Omollo, W. (2020). Provision of car parking space in the residential neighborhoods:. *International Journal of Human Capital in Urban Management*, 18.

Omollo, W. O. (2020). Provision of car parking space in the residential neighborhoods: A development control challenge in urban areas. *International Journal of Human Capital in Urban Management*, 4-9.

Paytech. (2021, November 03). *Smart Parking Solutions*. Retrieved from paytechafrica.com: https://paytechafrica.com/solutions/smart-parking-solutions/

paytechafrica.com. (2021, November 03). *Solutions/paytech.com*. Retrieved from paytechafrica.com: Smart Parking Solutions - Paytech. (2021, November 3). Paytech. https://paytechafrica.com/solutions/smart-parking-solutions/

S.p.A, I. (n.d.). *Home, smartparkingsystem.com*. Retrieved from smartparkingsystem.com: Home. (n.d.). Fleximodo. https://www.fleximodo.com/?gad\_source=1&gclid=CjwKCAjwrcKxBhBMEiwAIVF8rFo4xulNF55KLJmBo3lzES6-97BYz7KYuVlE\_QiZyU2kmKfEOb\_6eRoC4bIQAvD\_BwE

## **Appendix**

### **Appendix A1: Time Schedule**

Requirements specification

Design

Coding and Development of both Software and IoT components

Integrate and test.

Documentation: Writing Manual

May 6th

May 15th

May 19th

June 16th

June30th

May 13th

May 24th

Figure A1: Time Schedule for the development of the project