SUP'COM
Higher School of Communication of Tunis

UNIVERSITÉ DE CARTHAGE

ÉCOLE SUPÉRIEURE DE COMMUNICATIONS DE TUNIS

Scope of Statement

Smart Parking

Realised by :

Tasnim Ben Hamed and Mohamed Khalil Drira

Supervised by :

Dr. Mohamed Becha Kaaniche

Academic year : 2023/2024

Contents

Introduction 1
1 Objectives 1
2 Functionalities 1
3 Components 2
4 Functional Architecture 3
5 Constraints 4
6 Methodology 4
7 Deliverables 5
8 Business Marketing Study 6
8.1 SWOT Analysis 6
8.2 4P Marketing Mix 6

List of Figures

1 System's Functional architecture 4
2 SWOT Analysis 6
3 4P Marketing Mix 6

Introduction

In this project, we address the growing challenges of urban parking management brought about by the rising number of vehicles in large cities. With the increasing demand for parking spaces, securing a spot has become a hassle, contributing to traffic congestion and raising concerns about vehicle security.

To tackle these issues, we propose a smart parking solution that leverages the power of the Internet of Things (IoT) to transform the way parking is managed.Using Our solution uses IoT-enabled sensors to monitor real-time parking availability, providing drivers with instant updates and reducing search times, while also helping parking owners better manage and monitor their facilities.

To further enhance sustainability, we will integrate solar panels to power lighting, reducing reliance on traditional energy sources and minimizing operational costs.

By combining IoT technology with sustainable energy solutions, our project offers a comprehensive approach to addressing parking challenges in modern cities, enhancing efficiency, security, and environmental responsibility.

1 Objectives

The primary objective of this project is to develop a smart parking system that simplifies the process of finding available parking spaces, thereby saving time for drivers and reducing traffic congestion.
Additionally, the system aims to automate parking monitoring for parking facility owners, enabling more efficient management of parking spaces and optimizing energy usage.

2 Functionalities

Our smart parking system will offer a range of functionalities designed to enhance user experience, optimize parking facility management, and promote energy efficiency. Key functionalities include:

- **Real-Time Parking Availability Monitoring**: IoT-enabled sensors will monitor each parking space and provide real-time data on availability.
- **Automated Access Control**: The system will control access to the parking facility based on the user's license plate. Only authorized vehicles will be permitted entry, improving security and automating the management system.
- **Parking Spot Reservation System**: Drivers will be able to reserve the closest parking spots available in advance through the mobile app.
- **Real-Time Monitoring of solar energy generation**:
  Real-time monitoring of solar energy generation, battery levels, and power consumption optimizes energy use, while automated lighting adjustments based on ambient conditions enhance efficiency.
- **Parking Facility Management Dashboard**: A web-based dashboard will allow parking owners to view data on occupancy, access logs, and solar energy usage in real time.

- **User-Friendly Mobile Application**: A mobile app will provide drivers with access to real-time parking availability and reservation options.

3 Components

- **Raspberry Pi**: Central processor for the system, controlling sensors, LEDs, and other components.
- **Pi Camera**: Captures images or videos for monitoring parking spaces and security.
- **Sensors**:
  - **Photoresistors**: Measure ambient light to adjust parking lighting for energy savings.
  - **Ultrasonic Sensor**: Detects the presence of cars at entry and exit points of the parking area.
  - **Motion Sensor**: Detects movement to activate lights or security features.
- **Resistors**: Limit current to protect LEDs, sensors, and other components.
- **LEDs**: Provide visual signals for parking spot availability and system status.
- **Servomotor SG90**: Controls parking gates based on sensor data (e.g., for authorized vehicles).
- **Solar Panel (5W-10W, 12V)**: Generates renewable power for the system.
- **PWM Charge Controller (5A, 12V)**: Manages charging from the solar panel to the battery, preventing overcharging.
- **Battery (12V, 2Ah)**: Stores solar energy for use during low-light conditions.
- **Buck Converter (12V to 5V)**: Converts battery output to 5V for the Raspberry Pi and other components.
- **Voltage and Current Sensors (INA219)**: Monitor energy flow for system performance analysis.
- **Relay Module**: Controls power to components like LEDs, optimizing energy usage.
- **Connecting Wires and Terminals**

4 Functional Architecture

To ensure the functionalities outlined earlier in this project, we have chosen the following architecture :

- **Front-End - Progressive Web Application (PWA):**
  This approach allows for cross-platform accessibility without needing separate native applications for each platform. PWAs are fast, responsive, and accessible directly from a web browser, making them ideal for users looking to access real-time parking data.

- **Middleware - Jakarta EE and WildFly:**

  - **Jakarta EE:** It serves as the middleware for the system, providing a server-oriented platform for developing and executing distributed applications. It manages business logic and API interactions, ensuring secure and efficient data flow between the front-end and back-end.
  - **WildFly:** As a Jakarta EE-compliant application server, WildFly hosts and runs the Jakarta EE components.

- **Back-End :**

  - **MongoDB:** MongoDB is used as the primary database for storing high-volume data such as user profiles, parking availability, and sensor readings.
  - **MQTT Broker (Mosquitto):** Mosquitto, an MQTT broker, manages data exchanges between IoT devices and the cloud. MQTT is a lightweight, publish-subscribe protocol ideal for IoT applications, allowing sensors to send real-time data to the cloud with minimal network usage.
  - **Node-RED:** Node-RED is a programming tool for wiring together hardware devices, APIs, and online services. It provides a flow-based interface to process and manage data from sensors, trigger actions, and relay information to other parts of the system.

- **Protocols:**

  - **REST API:** In this system, REST APIs facilitate the exchange of data between the mobile app/dashboard and the server, enabling users to view real-time information and interact with the parking system.
  - **MQTT:** In this system, MQTT allows sensors and devices to send data to the cloud efficiently, enabling timely updates on parking availability and environmental monitoring.

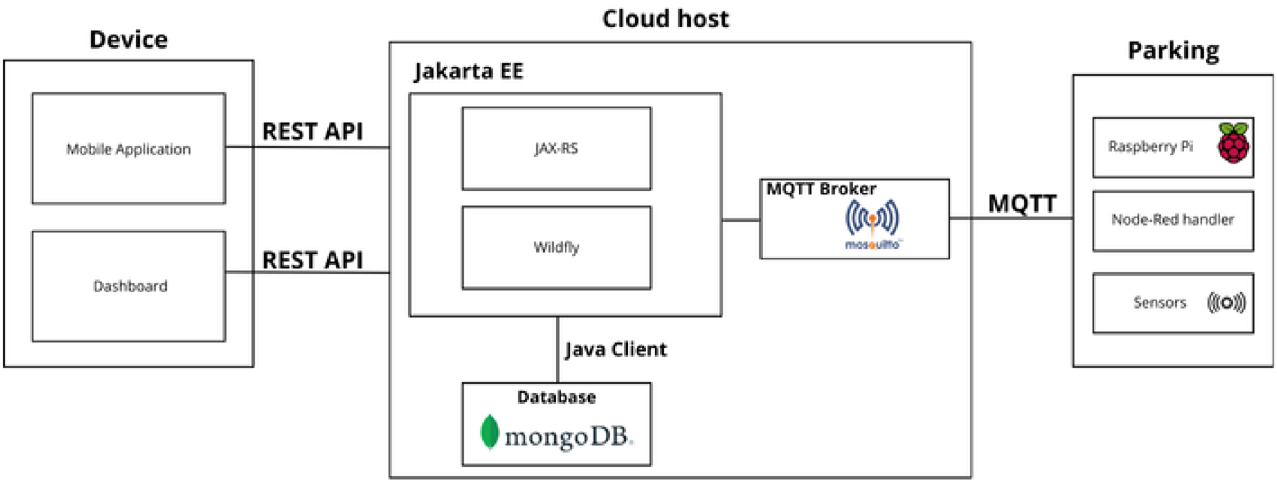

Figure 1: System's Functional architecture

5 Constraints

The Smart Parking System must be fully implemented, including all required functionalities, by January 10, 2025.

6 Methodology

This project will follow Extreme Programming (XP), an Agile software development framework known for delivering high-quality software and promoting team satisfaction. XP is particularly suited to projects like ours, where flexibility is crucial to accommodate frequent, last-minute changes.

Principles

XP shares core principles with other Agile methodologies but emphasizes them in an ”extreme” manner, including:

- Responsiveness to evolving customer needs.
- Collaborative teamwork, exemplified through Pair Programming.
- Commitment to high-quality work.
- Emphasis on early, rigorous testing.

Process

The XP framework involves five continuous phases, each iteratively advancing the project toward completion. These phases are summarized in Table 6.

Phase	**Description**
Planning	Define user stories and desired outcomes. Requirements are gathered, and stories are estimated to create a release plan with iterative phases. "Spikes" are used for stories requiring further research.
Design	Focus on simplicity to create a logical, structured system design, avoiding unnecessary complexity.
Coding	Implement code following XP practices, including coding standards, pair programming, continuous integration, and collective code ownership.
Testing	Continuous testing includes unit tests for features and acceptance tests to ensure alignment with requirements.
Listening	Continuous communication keeps all team members aligned with project goals and requirements.

Table 1: The Five Phases of the XP Process

7 Deliverables

At the end of the project, the following items will be delivered:

- Smart Parking Mobile App.
- Comprehensive Source Code Repository.
- Functional Smart Parking Prototype.
- Detailed Solution Report.

8 Business Marketing Study

8.1 SWOT Analysis

he SWOT Analysis below provides a strategic overview of our smart parking solution by identifying its core **Strengths**, **Weaknesses**, **Opportunities**, and **Threats**. This analysis highlights the key advantages that our solution offers, the challenges it may face, potential growth areas, and external factors that could impact its success.

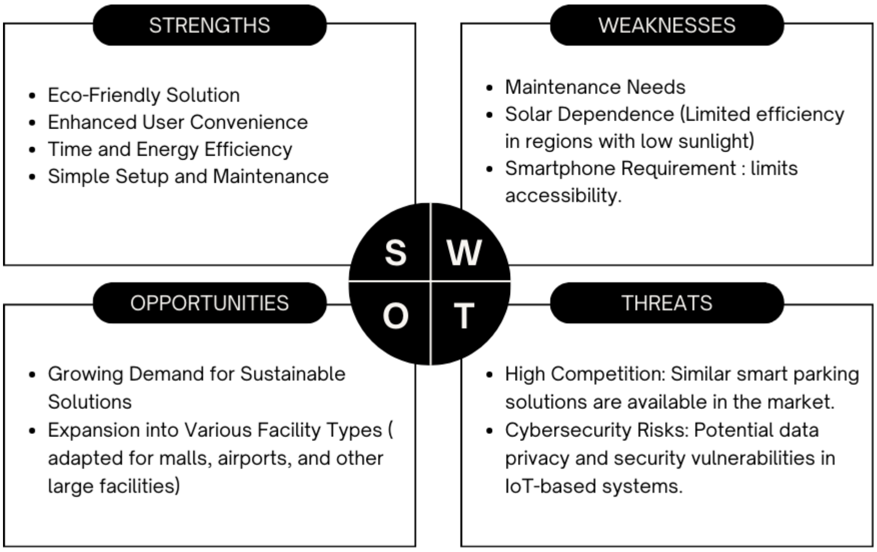
STRENGTHS
• Eco-Friendly Solution
• Enhanced User Convenience
• Time and Energy Efficiency
• Simple Setup and Maintenance
WEAKNESSES
• Maintenance Needs
• Solar Dependence (Limited efficiency in regions with low sunlight)
• Smartphone Requirement : limits accessibility.
S
W
O
T
OPPORTUNITIES
• Growing Demand for Sustainable Solutions
• Expansion into Various Facility Types (adapted for malls, airports, and other large facilities)
THREATS
• High Competition: Similar smart parking solutions are available in the market.
• Cybersecurity Risks: Potential data privacy and security vulnerabilities in IoT-based systems.

Figure 2: SWOT Analysis

8.2 4P Marketing Mix

The 4P Marketing Mix below outlines the key strategies for bringing our smart parking solution to market. It highlights our **Product** features, **Pricing** structure, **Placement** channels, and **Promotion** methods to ensure effective market penetration and customer reach.

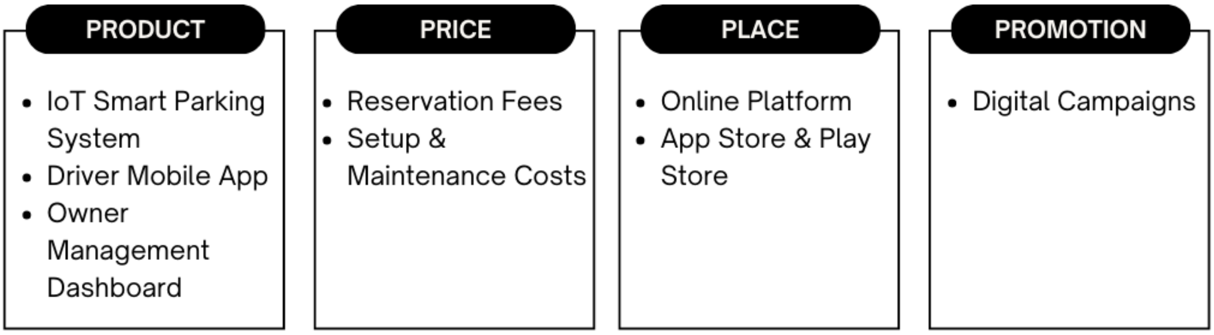

Figure 3: 4P Marketing Mix