SUP'COM
Higher School of Communication of Tunis

جامعة قرطاج
Université de Carthage

# Smart Garbage Monitoring System


*Elaborated by:*

Rihab JERBI
rihab.jerbi@supcom.tn

Bedis MANSAR
bedis.mansar@supcom.tn

*Supervisor:*

Dr. Mohamed Béchaa KAÂNICHE
medbecha.kaaniche@supcom.tn


Academic year: 2022 - 2023

# Contents

# List of Figures

# List of Tables

# 1 General overview

The natural environment plays a quintessential part of our lives. Therefore, keeping it safe is a mandatory duty in order to live on a comfortable and healthy planet. However, the environment keeps facing issues such as pollution which have a noticeable impact on the ecosystem. Large cities are the biggest victims of this phenomenon. The rising population in these cities means more human waste that becomes harder and harder to manage. The traditional way of manually monitoring the garbage level has not been efficient in many cities around the world and has not yielded good results as the waste keeps piling up, and thus, becoming harmful to the environment. For that reason, it is necessary to develop a smart garbage monitoring system utilizing Cloud of Things technologies to notify the users of the the level of trash in each garbage can and optimize the path to clean the city. This project aims to provide the following functionalities:

- Detect and display the level of waste in garbage cans.
- Alert the user when the level detected exceeds the threshold set up beforehand.
- Provide the best path to clear all the garbage cans using geographic coordinates.

# 2 Target clients

The clients that will benefit with this solution will be the people responsible in the city for cleaning and gathering the garbage: the municipality. This project will aid them immensely in their task of monitoring the waste by facilitating the process and reducing the resources otherwise spent on manually doing the task. This project helps only in detecting and monitoring the waste found in garbage cans and not the waste thrown on the street.

# 3 Components

**Arduino UNO R3:** Arduino is a microcontroller board providing multiple digital and analog input/output pins to connect the different electronic components. Arduino can be programmed to run scripts to acquire data from the sensors, transmit the data and control actuators. Arduino is also cheap and provides the required functionalities for this project.

| | |
|---|---|
| **Digital I/O pins** | 14 |
| **Analog input pins** | 6 |
| **I/O voltage** | 5V |
| **Microcontroller** | ATmega328p |
| **Weight** | 5g |
| **Clock speed** | 16 MHz |
| **SRAM** | 2KB |

Table 1: Arduino Uno 3 Specifications

**Ultrasonic sensor:** An ultrasonic sensor is an electronic component that can measure the distance of a body using ultrasonic sound waves. When pulses are sent to the object, they are reflected, creating an echo that is received by the sensor, making it possible to measure

the distance by calculating the time lapses between the sending and receiving of the ultrasonic pulse.

| | |
|---|---|
| **Voltage** | DC 5V |
| **Current** | 15 mA |
| **Min range** | 2cm |
| **Max range** | 400cm |

Table 2: Ultrasonic sensor HC-SR04 Specifications

**Resistors:** Resistors will be used to limit the current's intensity, and thus protecting the sensors.

**Cables:** To link between the different components of the circuit, multiple wires will be used.

**ESP8266 Wifi module:** An ESP8266 Wifi module enables the arduino card to connect to the cloud server and database using Wifi.

**NEO6MV2 GPS Module:** A NEO6MV2 GPS Module is a global positioning system module that provides longitude and latitudee of its position.

**Batteries:** Batteries will be used to power-up the electronic card.

# 4 Technologies

To achieve the different parts of this project, multiple technologies and frameworks will be implemented:

- **Backend:**
  - **MQTT:** MQTT is a lightweight messaging protocol for communication between the different Internet of Things components. This protocol requires a MQTT broker which will be the intermediary between the communication of two devices to ensure the decoupling of the two devices. In this case, we will be using Mosquitto as the MQTT broker.
  - **MongoDB:** MongoDB is a NoSQL document-oriented database used to store all the data in this project such as data captured by the sensors and data of users of the application.
  - **Node-Red:** Node-Red is an open source tool used in Internet of Things projects to manage the data flow of the sensors.
- **Middleware:**
  - **Jakarta Entreprise Edition:** Jakarta Enterprise edition is a Java-based framework for developing API's for enabling communication between different and numerous applications. Jakarta's main selling point is the high level of security it provides that is not possible with other Java frameworks.
  - **WildFly:** Wildfly is a Java Entreprise Edition lightweight application server designed by Red Hat providing all the necessary functionalities to run a Java web application.

- **Frontend:**
  - **Flutter:** Flutter is an open source software development kit created by google. Crafting multi-platform, natively compiled mobile applications is possible with Flutter; hence, it will be the framework used in this project for developing the mobile application.

# 5 Architecture

Figure 1 demonstrates the architecture of the Smart Garbage Monitoring System:

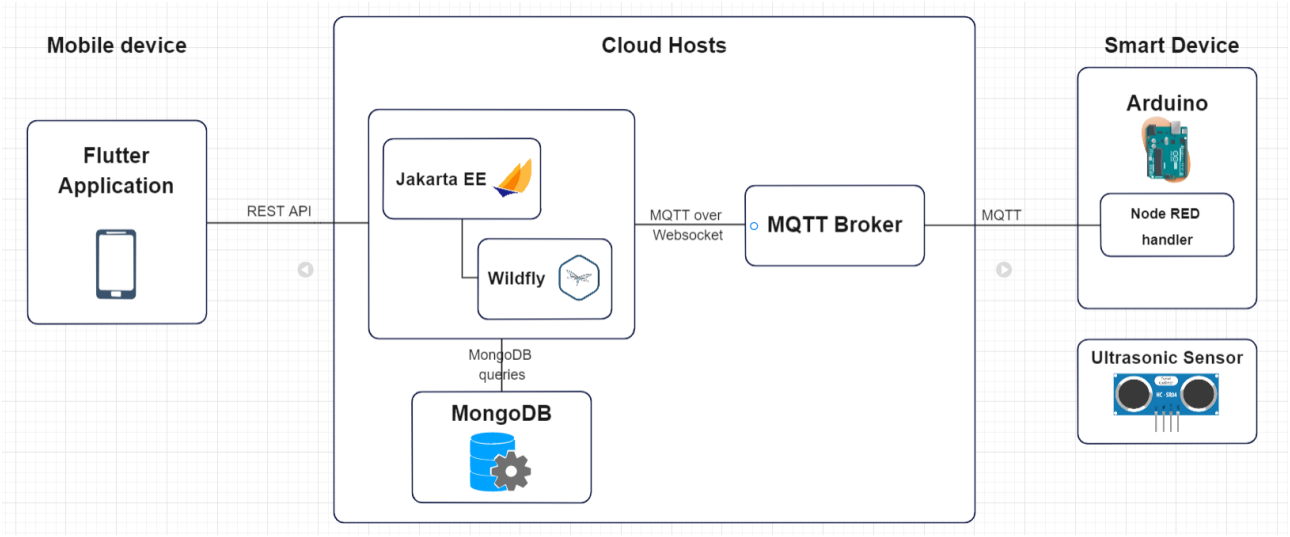


Figure 1: Smart garbage monitoring system architecture

In this project, Cloud of Things technologies will be used. Therefore, the middleware server, the database and the MQTT Broker will all be hosted on cloud.

# 6 Project deliverables

By the end of this project, these requirements must be delivered:

- **Source code:** The source code for the different parts of the project will be stored and delivered using Github.
- **Technical documentation:** A README.md file in GitHub detailing all the needed steps to launch the application.
- **Design document:** A document that describes the architecture of the application and its functionalities through different diagrams.
- **Demo:** A video in mp4 format that contains a demonstration of the proposed solution.

# 7 Constraints

The realization of all parts of the Smart garbage monitoring system must be achieved no later than 8 January 2023.

# 8 Deployment diagram

A deployment diagram is an UML diagram for visualizing the hardware components and devices, the links of communication between these different components and and the software files on that hardware. Figure 2 shows the deployment diagram for the smart garbage monitoring system:

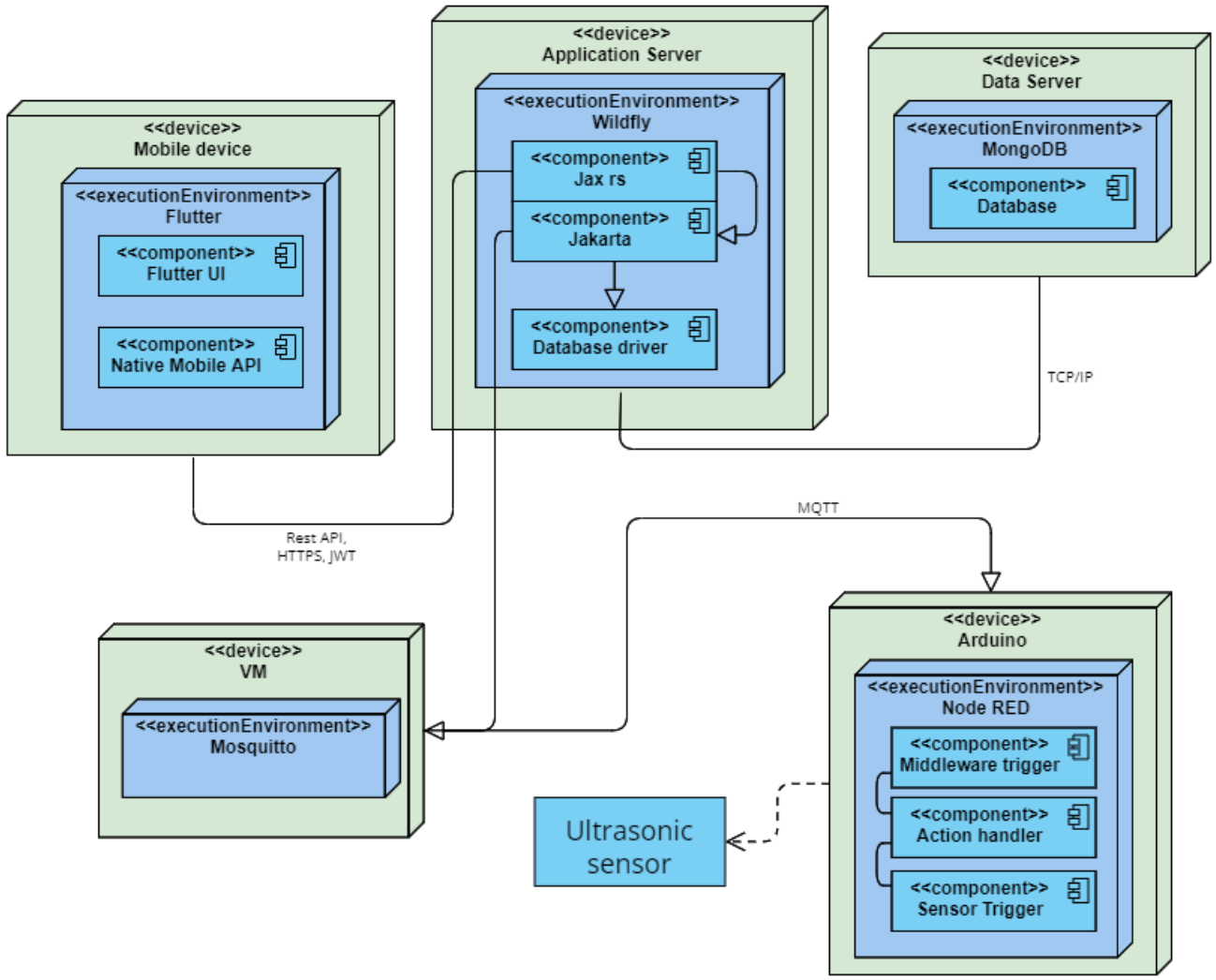


Figure 2: Smart garbage monitoring system deployment diagram

# 9 Marketing study

## 9.1 BMC

The figure below highlights the Business Model Canvas for the Smart garbage monitoring system:

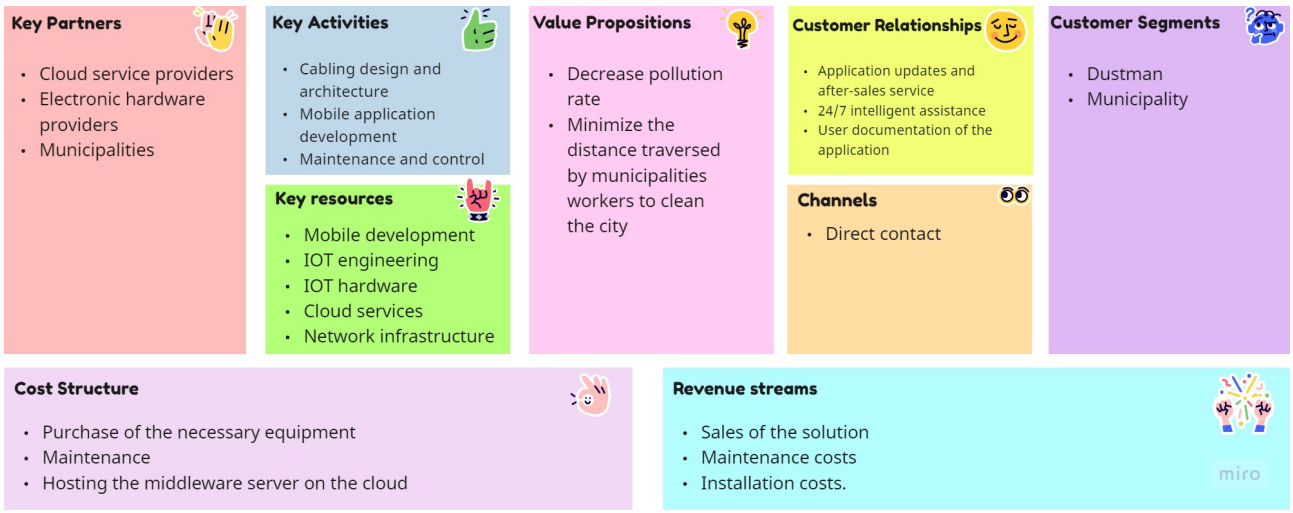


Figure 3: Smart garbage monitoring system business model canvas

## 9.2 4C

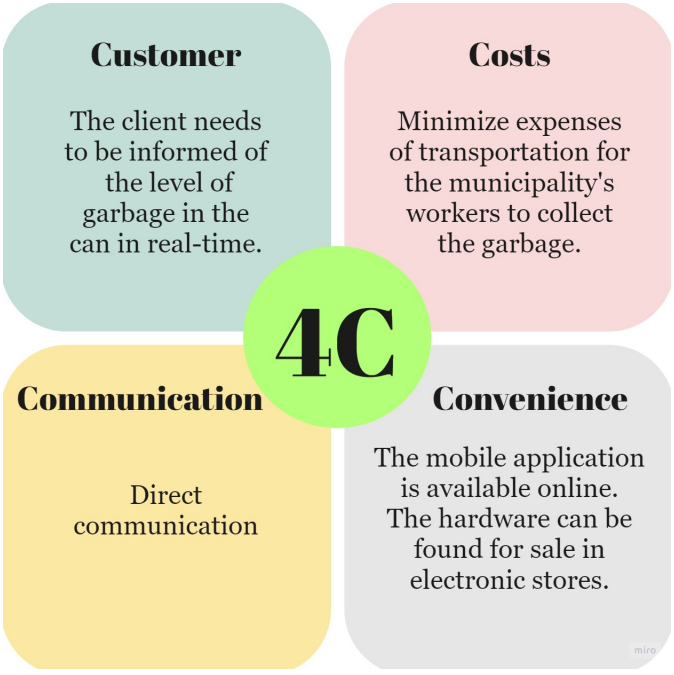


Figure 4: Smart garbage monitoring system 4C

Figure 4 illustrates the 4C model of the Smart garbage monitoring system composed of 4 elements:

- **Customer:** It is essential for the service to be in real-time to satisfy the client's demands.
- **Costs:** The solution must save costs of the municipality's task to clean the city.
- **Communication:** Direct communication.
- **Convenience:** The components for the solution such as the mobile application and the hardware components should available in most stores.