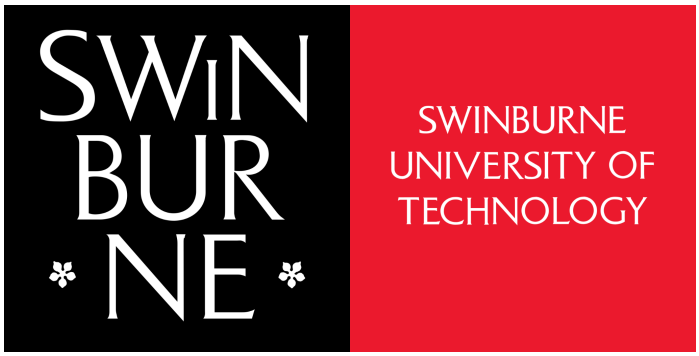
SWINBURNE UNIVERSITY OF TECHNOLOGY

Hawthorn Campus

BACHELOR OF COMPUTER SCIENCE

Semester 1, 2025

SWE30011 - IoT Programming



Smart Bin Management System Using IoT and Cloud Integration

Tutorial Day & Time: Friday 10:30 AM

Word Count:

Group Members:

- Vo Dang Khoi Nguyen (105241532)
- Samuel Tiong Zhe Ng (102776565)
- Alexander Rigato (105338913)
- Fraser Terry (105418857)

Table of Contents

1. Introduction

1.1) Topic Background & Motivation

Nowadays waste bins in many public and private spaces are often unmanaged which cause waste to overflow, hygiene issues and many potential safety hazards. Manual monitoring the waste level of the bins is very inefficient and often neglected since In most of the places in the world bins are only emptied on a fixed schedule and because of that many bins often overflow with trash and waste before its being empty making the whole progress emptying bins more inefficient.

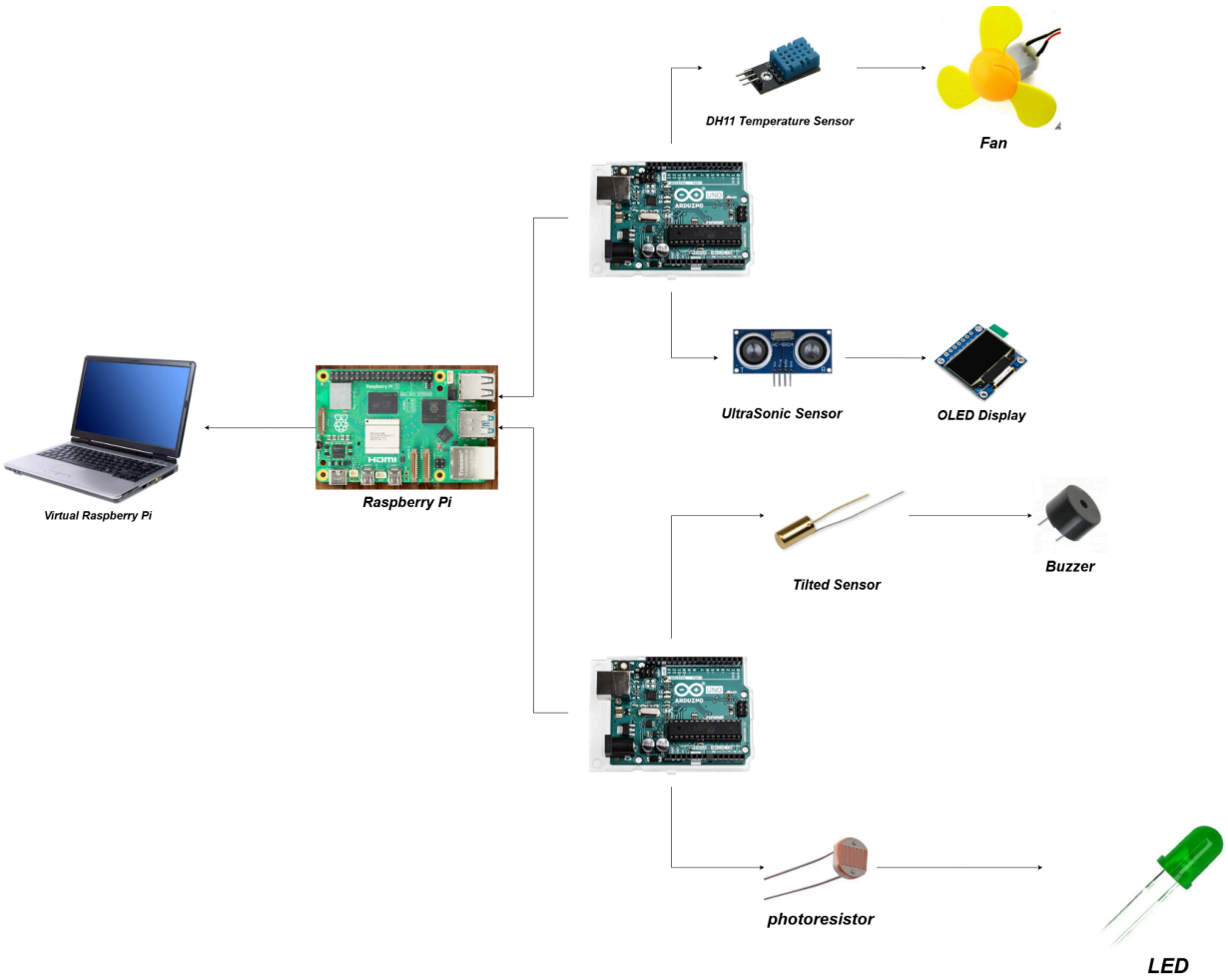
This project was motivated by the need to create a better way to manage public and private waste bins by creating a smarter, safer and more effective bin system by implementing IoT technology. Our team's goal is to develop a smart bin that allows the bin to detect if the bin is full or not but also monitoring unusual conditions inside the bin like overheating by trash that caught on fire by a fire source like a cigarette, matches or being tampered with, it also detect whenever the bin lid is open or when it's tilted and alarm the user about it. With the help of different sensors and actuators connected to a microcontroller and cloud platform to create a smart bin that will improve the waste management as well as safety through fire and tilt detection and finally improve the user with interaction like button controller lid. Our team was inspired to build this system as a practical real world application.

1.2) Proposed System Overview

The IoT smart bin system is a solution designed to solve common problems happening to public and private bins. The smart bin was built using an Arduino and Raspberry Pi that have multiple sensors and actuators connected to them. There will be an Ultrasonic sensor that will measure the fill level of the bin and indicate its current status by using an OLED screen. A temperature sensor to check for internal heat to detect the risk of waste being on fire and activate the ventilation fan. A led that comes with a buzzer that allows the user to detect whenever the bin lid is open. Finally to improve security, we also add a tilt sensor to detect if the bin has been moved or knocked over which will trigger a buzzer alert. The collected data from the sensors will be sent over to the Arduino and to the Raspberry Pi for processing and can be monitored remotely through a connected cloud platform.

2. Conceptual Design

2.1) System Description



The Smart Bin is an Internet of Thing solution which is designed to help with improving waste management through automation. By using a variety of different sensors, actuators, with 2 microcontroller devices as Arduino and 2 computers to provide a responsive system allowing the user to monitor and control the smart bin. The primary functionality of the system is to detect the waste fill level, monitor internal temperature, detect open lid, and raise alert in case of tilted bin and all of this data will be displayed on a web interface.

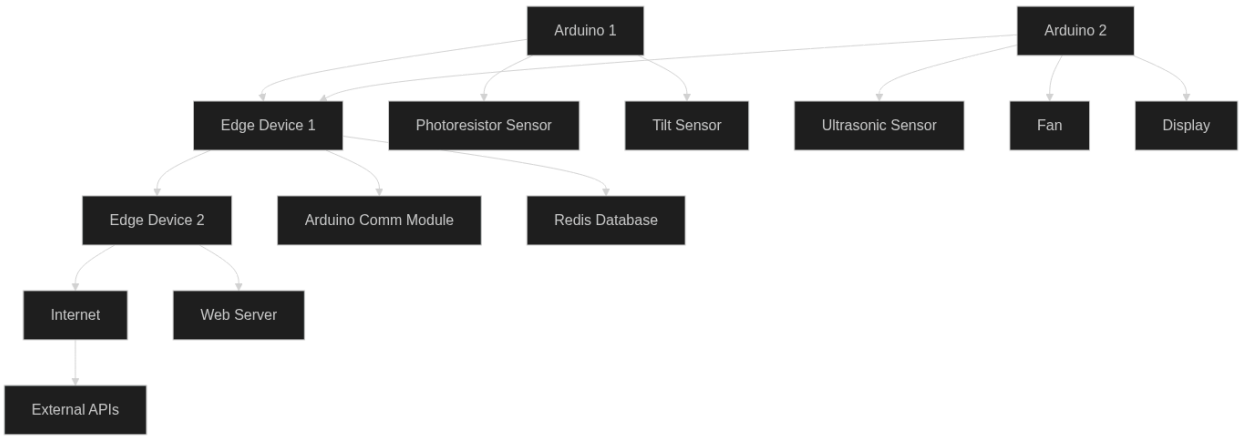
All of the sensor and actuator data are being sent and managed through 2 Arduino microcontrollers, and which run different logic to control different behaviour of the bin based on the reading of each sensor. The web interface is built to support the smart bin visibility by displaying different status on the web and it can also be used by facility managers.

The system is ideal for use in public places such as malls, schools, offices and smart cities where it would work more efficiently in waste management. By using lid detection, monitoring bin status to detect unusual conditions, the smart bin system improves the

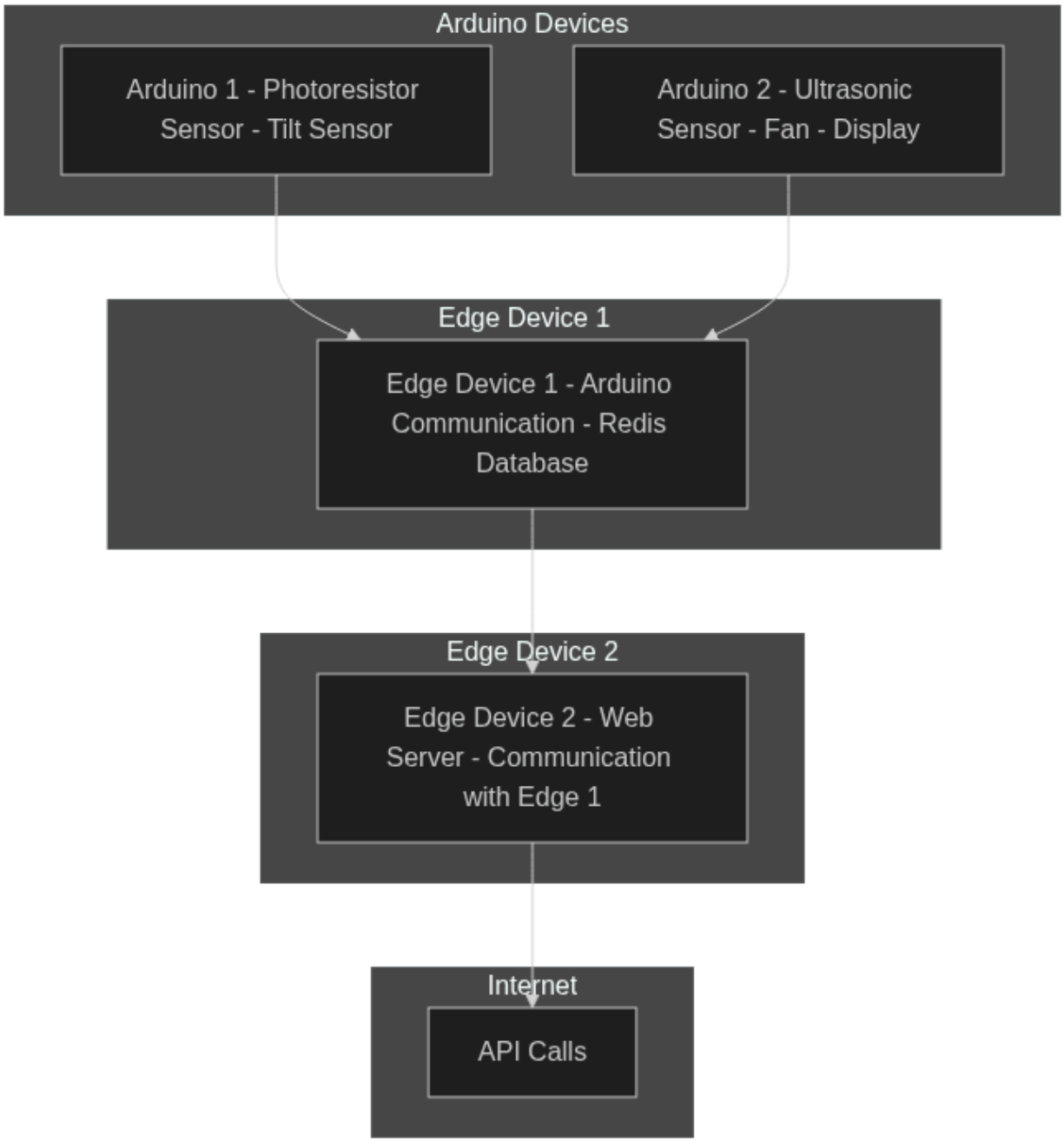
user convenience and supports time management for efficiency and provides safety alerts.

2.2) Diagram

UML Diagram



Block Diagram



3. Task Breakdown

Each person's responsibilities for the project were divided based on each member's strength. By doing this, it helps our team work rate in developing the IoT project more efficiently. The table below provides details about different tasks assigned to each member.

Team member	Assigned Tasks	Description of Work Completed
Vo Dang Khoi Nguyen	Arduino programming, testing, debugging, Sensors & Actuators wiring	Wrote and tested code for ultrasonic sensor and temperature sensor and actuators like OLED screen, fan motor and fixed logic issues
Samuel Tiong Zhe Ng	Arduino programming, testing, debugging, Sensors & actuators wiring	Wrote and tested code for Tilt ball sensor and Photoresistor and actuators like LED and buzzer.
Alexander Rigato	frontend UI programming, debugging, communication between database and backend	Wrote and tested frontend UI and responsiveness. Created a backend system to facilitate user input into the database and vice versa.
Fraser Terry	Raspberry Pi serial communication between arduino. Raspberry pi communication to redis communication.	Wrote and tested code for physical raspberry pi, to interact with the redis database. As well as read in serial serial communication from arduino to Raspberry pi.
All Members	Assembly different prototype of both hardware and software for testing and presentation preparation	Collaborated on creating the entire project system by assembling hardware like arduino, sensors, actuators with software like database, edge server and cloud. Debugging the system as well as preparing for the demonstration.

4. Implementation

4.1) IoT Architecture

The Smart Bin Management System employs a hybrid IoT architecture integrating edge computing and cloud services to maximize efficiency, reliability, and scalability. At the edge layer, multiple Arduino microcontrollers serve as primary sensor data collectors, handling real-time sensing and actuation. Data is transmitted via serial communication to a Raspberry Pi acting as a local edge server, performing preliminary data processing and filtering to reduce network congestion and latency.

The processed data is then securely transmitted to a cloud platform where centralized storage, and visualisation occur. This layered architecture ensures rapid response to

local events such as fire or tilt detection (except the motor controls). The system's modular design also supports seamless expansion and integration with future smart city infrastructures.

4.2) Sensing and Actuation Process

Sensors Include:

- **UltraSonic Sensor:**
  - Used to measure the distance between the waste level and the bin lid and estimate the fill percentage of the bin.
- **DHT11 Sensor:**
  - Used to read the current temperature inside of the bin to check if there is a potential fire caused by matches or cigarettes, if the temperature exceeds over the threshold then it will send a signal to the fan.
- **Tilt Sensor:**
  - Detect any unauthorized movement if it's being move away from its original position or checking if the bin is tilted or not
- **Photoresistor sensor**:
  - Use to detect if the bin lid is currently close or open

Actuators include:

- **Fan:**
  - If the current temperature of the DHT11 sensor exceeds the threshold it will turn on the fan for ventilation or whenever user cloud interface, allowing the user to turn on or off the fan on the website.
- **OLED screen:**
  - The OLED screen will be used as an actuator for the ultrasonic sensor and the DHT11 sensor as well. It will display the percentage filling level of the bin and the current temperature inside the pin to let the user know.
- **Buzzer:**
  - The buzzer will be used to alert whenever the tilt sensor is activated, alarming the bin has been moved or tilted.
- **LED:**
  - The Led will be used with the photoresistor sensor, the led will light up if the photoresistor can not detect light or else it will turn off when there is light.

4.3) Edge computing logic

To enable communication from the 2 Arduinos to raspberry pi and then to redis database, a python script was created using the “Thonny ide” on the raspberry pi. The script engages serial communication over 2 of its ports “ttyACM1” and “ttyACM0”, which is used to read the live data from the sensors using serial read. This live data is then stored in redis by importing the redis library. Additionally sensor data was printed to the terminal to help us evaluate and debug our smart bin system. We also contained a conditional rule for the fan actuator to turn on when the python script read in a temp over 30 degrees from the temp sensor.

4.4) Communication Protocol

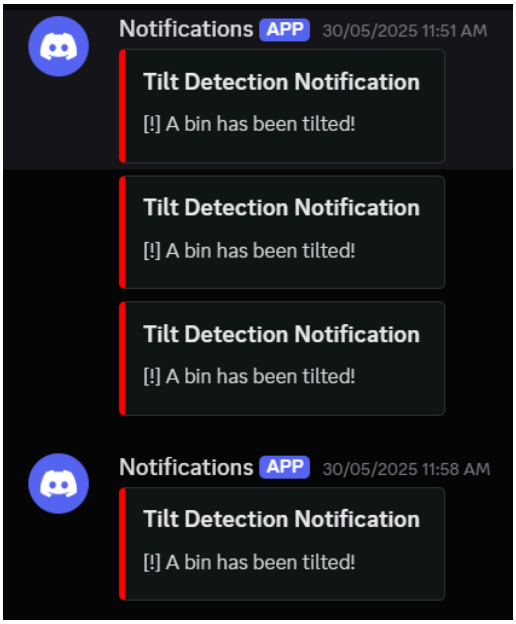
For the communication within our system we used a variety of different communication protocols. For the communication between the 2 Arduinos and the raspberry pi we used serial communication imported using the pySerial library, with ser1 and ser2 being our serial port variables. We set up our serial communication data to have a key with a corresponding variable, so that we could identify which sensor was transmitting, for example having T: 25 for the temp sensor sensing 25 degrees of temperature.
The data stream was communicated over 9600 baud, using the serial read keyword to check for information, and split accordingly.

Another form of communication that we used was the communication to the local redis database. The python script connects to the local redis server, over address 127.0.0.1 to send sensor data that was received from serial. The data is sent using key value pairs form with a sensor data name, and then the data itself. Additionally fan control commands are read from the Redis key “fanstate” every second and then sent back to the arduino, checking for a condition.

Both of the communication protocols allow our smart bin system to work with real time data, and adapt to changing conditions of our environment, helping to make it a sophisticated smart bin system.

4.5) External API Integration

To enhance real-time alerting, the system integrates with Discord Webhook API for notification delivery. Upon detection of bin tilt, sensor data triggers automated alerts sent directly to a designated Discord channel. This integration facilitates instant communication with facility managers and maintenance teams, enabling rapid response and coordinated action.
Utilizing Discord Webhook provides a cost-effective, scalable, and user-friendly platform for system notifications.



4.6) Cloud Computing Platform

To make use of further enhancements, we used cloud computing to host both the frontend and API of our IoT application. In doing so, we are able to put the frontend closer to the end user, making it easier for them to access it anywhere. This will provide a scalable and flexible solution to handle any increase in traffic load from device interactions especially if this project is deployed in a real world scenario. This setup also allows for efficient resource management and ensures that all systems remain reliable and performant even as the number of connected devices increases.

5. User Manual

The Smart Bin IoT project is designed to monitor the current state of the bin such as the waste level, temperature or check if the bin has been tampered with by using different sensors and actuators below here is the step on how to use and how does the system work:

5.1) System startup:

So in order to use the Smart Bin system the device needs to be connected such as the 2 arduino that contain the sensors and actuators needed to connect the physical Raspberry Pi and then it will be connected to a power source such as a USB adapter or a charged battery pack. Once power on, the system will automatically connect to 2 computers that control the raspberry pie and it will also start initializing.

5.2) Monitoring and Control:

The Smart Bin system will continuously working in the background as long as it still receiving power and the user can interact with the bin in the following ways:

- **Trash Disposal:** Throw trash into the bin. The ultrasonic sensor will check the current distance between the level of trash and the sensor and convert it into percentage and display it on an OLED screen.
- **Temperature Respond:** If the internal temperature of the Bin is becoming too high (e.g. potentially fire from matches, cigarette) the fan will automatically turn on to reduce the temperature inside the bin and the current temperature will also be displayed on the OLED screen.
- **Tilts Detection:** If the bin is tilted or being tampered by someone or something with a buzzer will activate to alert.
- **Lid Detection:** by using the photoresistor sensor, it can detect whenever the bin lid is closed or not and display it by a led light.
- **Web Interface:** The web interface will display every data receive from the sensor such as:
    - Temperature °C
    - Fill level percentage % and line graph
    - Bin lid status: Open / Close
    - Tilted warning
    - Toggle Fan button: ON /OFF

5.3) Troubleshooting:

In order to make the system run smoothly it also need to be go through maintenance after a while:

- Empty the bin when full
- Make sure the sensors are clean and unobstructed
- Replace or recharge power supply

6. Limitations

6.1) Technical Constraints

There are several limitations for this IoT project. This section will go over them one by one.

The responsiveness for the motor is not instant when controlled from the website. As for now, the time required to turn on/off the motor from the website will take approximately 5-10 seconds. This is because the sensors are flooding the database with other data and in order for it to read data for controlling the motor, we've implemented a system where it updates every 5 seconds. Besides that, the time needed for the data to be sent

from the website to the database then to the arduino board will take some time as well, hence the not so responsive motor.
Besides that, the bin also doesn't possess an automatic lid opening system. This could be implemented for user feasibility.

6.2) Areas for improvement

While the current Smart Bin system demonstrates its function in waste management automation, there remain areas with potential for enhancement.

- **Improved Data Analytics:** Integration of advanced machine learning models could provide predictive insights, such as forecasting bin fill rates based on historical usage patterns, optimizing emptying schedules, and identifying maintenance needs before sensor failure.
- **Enhance Sensor Accuracy and Variety:** Incorporating higher precision sensors and additional types such as gas sensors could detect hazardous waste or odor levels, further improving public safety and environmental monitoring.
- **Real-Time Responsiveness:** Optimizing communication protocols and data handling to reduce latency would improve actuator response times, creating a more immediate feedback loop between user commands and bin operations.

Addressing these areas will solidify the Smart Bin System's position as an innovative, reliable, and scalable solution in smart waste management.

7. Resources

We used the following resources in creating our IoT assignment:

Guides/Tutorials

- Project guide provided in the school tutorial (COS30008 – IoT Programming)
- Arduino Official Documentation
- Raspberry Pi Documentation
- Redis Documentation
- Discord Webhook API Documentation

Software

- Docker
- Redis
- ArduinoIDE
- NUxT
- Thonny IDE

Hardware

- Arduino Microcontroller (UNO)
- Raspberry Pi
- Ultrasonic Sensor
- DHT11 Temperature and Humidity Sensor
- Tilt Ball Sensor
- Photoresistor Sensor
- OLED Screen
- LED and Buzzer
- Motor (with fan blades)

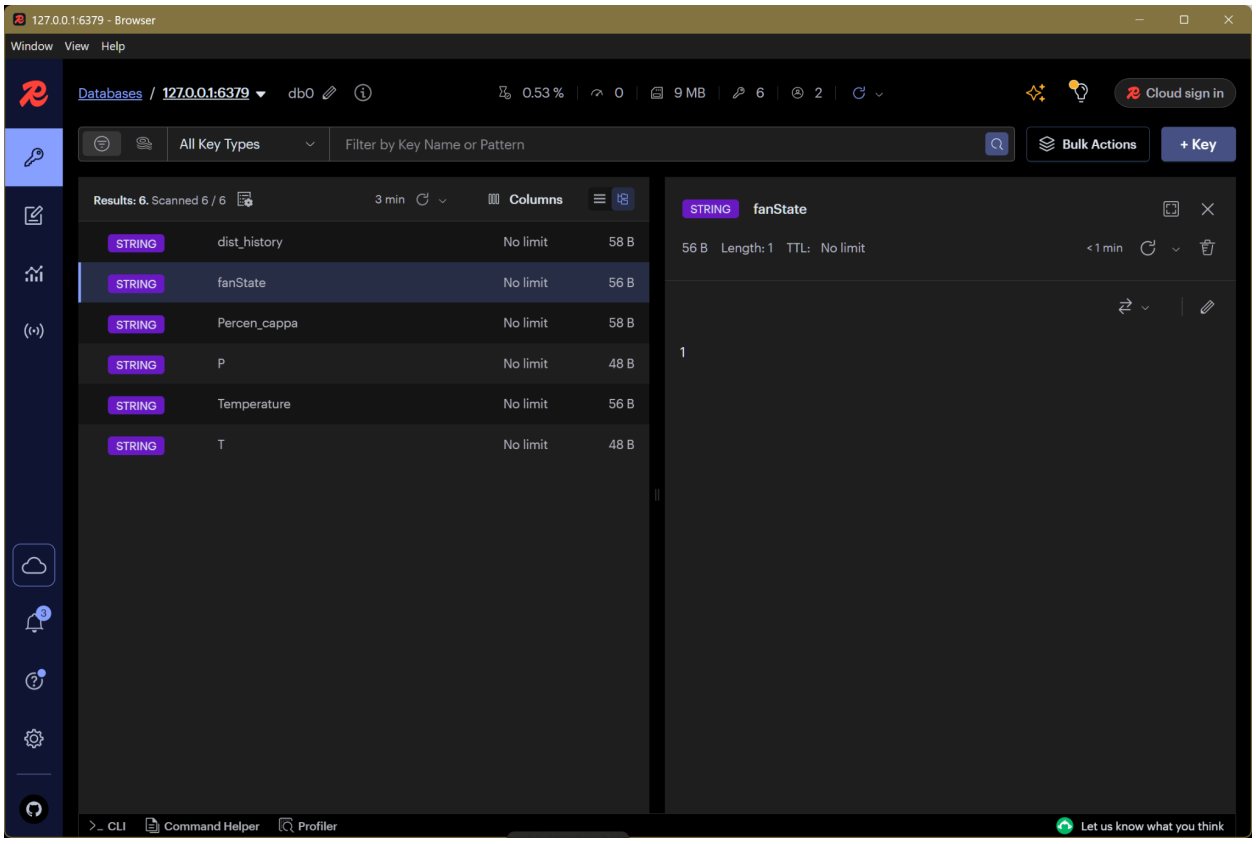
Miscellaneous Tools:

- Github
- Discord

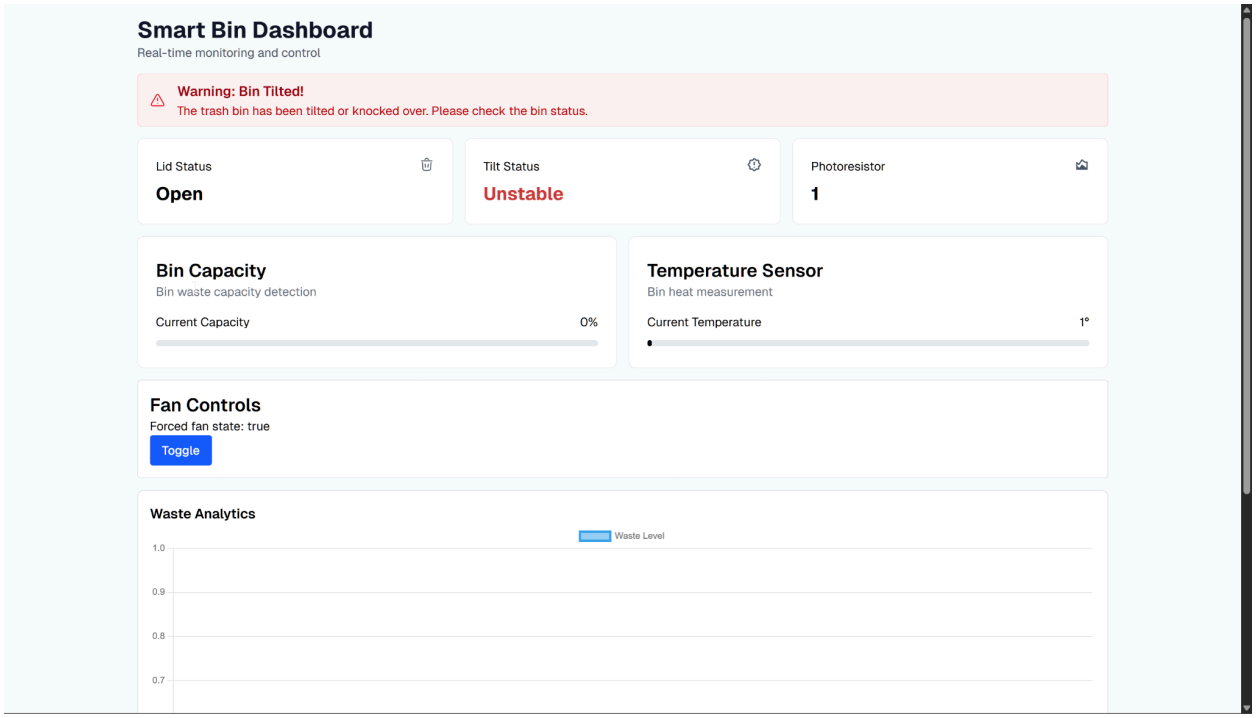
8. Appendix

The appendix includes supplementary materials that support the main content of the Smart Bin Management System Using IoT and Cloud Integration project.

DataBase:



Web Interface:


Smart Bin Dashboard
Real-time monitoring and control
Warning: Bin Tilted!
The trash bin has been tilted or knocked over. Please check the bin status.
Lid Status
Open
Tilt Status
Unstable
Photoresistor
1
Bin Capacity
Bin waste capacity detection
Current Capacity
0%
Temperature Sensor
Bin heat measurement
Current Temperature
1°
Fan Controls
Forced fan state: true
Toggle
Waste Analytics
Waste Level
1.0
0.9
0.8
0.7