Smart IoT Waste Bin

Sıla Hacıalioğlu – 20200808040

1. Introduction

1.1. Background and Motivation for Selecting the Project

The inspiration to embark on this project stemmed from encountering a smart trash bin prototype online. I found the concept both innovative and practical, which motivated me to create my own version. Rather than simply replicating the idea, I aimed to significantly enhance it by incorporating advanced features that would make the system more applicable to real-world scenarios. These enhancements include a smart lid control mechanism, remote monitoring of the bin's fullness level via the Arduino IoT Cloud, and real-time GPS-based location tracking. Additionally, the aesthetic design of the bin was inspired by the character WALL·E from the Disney movie, adding a playful and engaging touch to the overall appearance.

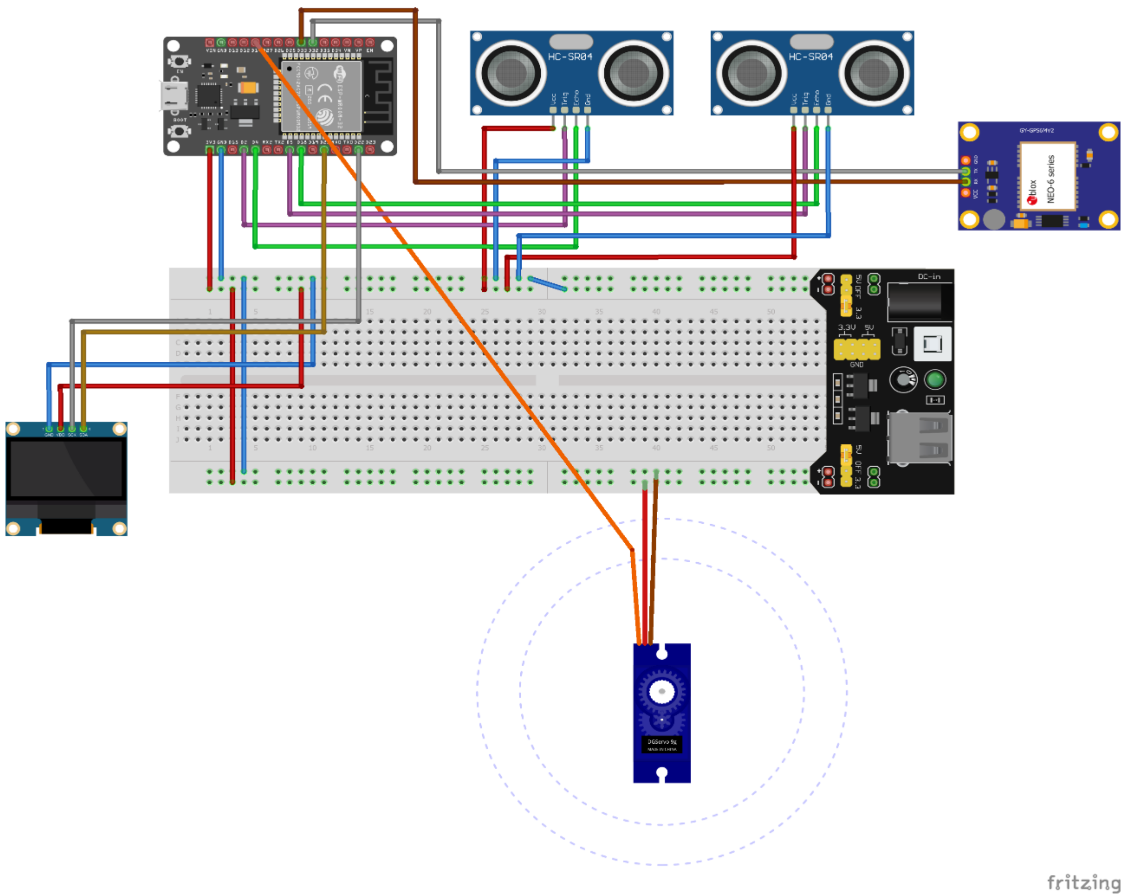
1.2. Problem Definition and Relevance

In modern cities, waste collection processes often face inefficiencies, especially due to the lack of real-time information about trash bin fullness. This can lead to some bins overflowing before being emptied. This project addresses that problem by enabling remote monitoring of trash levels and location data. Such a system can be particularly beneficial for municipalities and organizations involved in waste management, helping them optimize collection routes, reduce operational costs, and improve overall efficiency.

2. System Architecture

This smart trash bin system is built using the ESP32 microcontroller and integrates various components such as ultrasonic sensors, a servo motor, GPS module, OLED display, and Arduino IoT Cloud. The system is capable of detecting when a user's hand approaches to automatically open the lid, monitor the bin's fullness level, and track its real-time location. All this information is sent to the cloud where it can be monitored remotely. Notifications are triggered when the bin reaches a critical fullness level.

2.1. Architecture Diagram



2.2. Components Used

Component	Description
ESP32	Central microcontroller for data processing, Wi-Fi, and cloud communication
HC-SR04 Sensors	1 sensor for hand detection, 1 for trash level monitoring
Servo Motor	Controls the lid's open/close mechanism
OLED Display	Shows the real-time fill percentage
TinyGPSPlus Module	Provides latitude and longitude, shared with the cloud
Arduino IoT Cloud	Cloud platform for dashboard, notifications, and remote control

2.3. Functionality

Auto Lid Control: The lid opens automatically when a hand is detected within 4 cm of the sensor.
Fill Level Monitoring: Calculates how full the bin is based on the distance measured from the ultrasonic sensor.
Notification System: Sends a cloud alert when the bin reaches 70% capacity.
Real-Time Tracking: GPS module updates and sends current location to the cloud.
Display Output: OLED screen continuously updates to show current fill level.

3. Implementation Details

3.1. Hardware Setup

Hand Sensor: TRIG1 (Pin 5), ECHO1 (Pin 18)
Trash Sensor: TRIG2 (Pin 2), ECHO2 (Pin 4)
Servo: Pin 14
OLED (I2C): SDA (Pin 21), SCL (Pin 22)
GPS: RX (Pin 32), TX (Pin 33)

3.2. Sensor and Actuator Integration

- **Hand Detection Sensor**: Detects a hand within 4 cm to trigger the lid to open using the servo motor.
- **Trash Level Sensor**: Measures the distance from the sensor to the trash and converts this to a fullness percentage (0–100%).
- **Servo Motor**: Rotates to 0° to open the lid and to 90° to close it automatically or via remote control.
- **OLED Display**: Continuously updates to show the current fill level in percentage.
- **GPS Module**: Continuously parses GPS data and updates location to the cloud in real-time.

3.3. Communication Protocols Used

- Communication with the cloud is handled via the Arduino IoT Cloud library.
- The cloud uses **MQTT** protocol behind the scenes for efficient communication and control.
- Device variables such as fullness, lidControl, notify, and location are synced with the cloud automatically.

3.4. Cloud Services Used

- **Arduino IoT Cloud**:
  - Provides a user-friendly dashboard for real-time monitoring and remote control.

    - Displays:
        - Current trash fullness level
        - Real-time location via GPS
        - Lid status (open/closed)
    - Allows users to manually control the lid and receive alerts when the bin is nearly full.

3.5. Data Analytics Methods Applied

- **Sensor Data Averaging**:
    - A circular buffer stores the last 10 readings of the fullness level.
    - These values are averaged to minimize noise and spikes in sensor readings.
- **Threshold Notification**:
    - A notification is triggered when the averaged fullness level reaches 70% or higher.
    - The alert is reset when the level drops below 60%, preventing repetitive alerts.

3.6. Security Features Implemented

- **Cloud Authentication**:
    - The Arduino IoT Cloud requires device certificates and credentials stored securely in arduino_secrets.h.
- **Wi-Fi Credentials Protected**:
    - Sensitive information like SSID and password are not hardcoded in the main sketch; instead, it is placed in the arduino_secrets.h file, which is not uploaded to public repositories.
- **Limited Control Surface**:
    - Only essential variables (lidControl, notify, location, fullness) are exposed to the cloud, minimizing the risk of remote misuse.

4. Results and Analysis

- **Figure 1** displays the physical prototype of the smart trash bin, showcasing the integration of key components such as ultrasonic sensors, the servo motor for lid operation, an OLED display for showing fullness levels, and the ESP32 board as the central controller.
- **Figure 2** presents a front view of the trash bin, which bears a resemblance to the character *WALL·E* from the Disney movie. This similarity adds a playful and charming aesthetic to the design.
- **Figure 3** is a screenshot from the Arduino IoT Cloud Dashboard, where real-time data such as trash bin fullness, GPS location, and lid status are monitored and controlled remotely.

- **Figure 4** shows the mobile view of the same dashboard, highlighting the responsive and user-friendly interface for monitoring the bin from a smartphone or tablet.

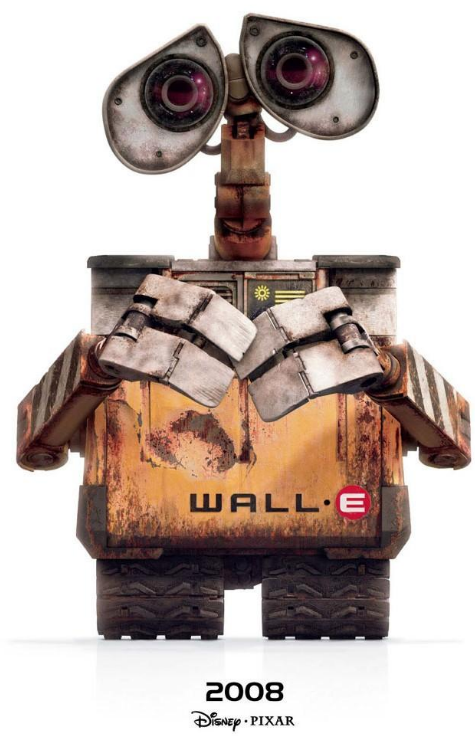


Figure 1 **Figure 2**

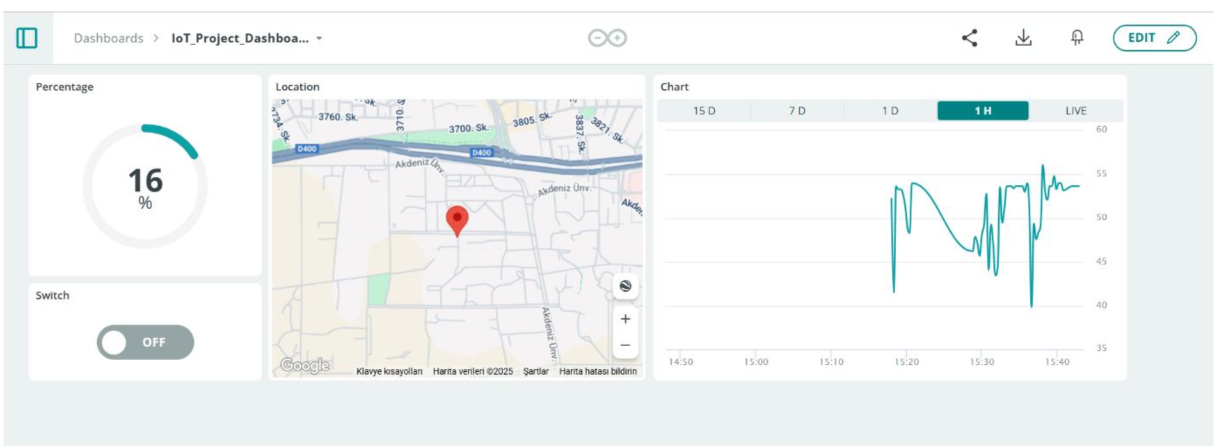


Figure 3

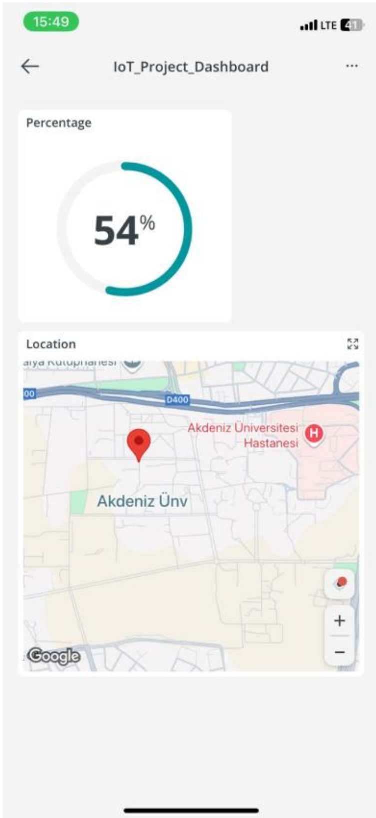


Figure 4

5. Interpretation of Results

Most components of the system functioned as expected. However, one key issue was encountered during development:

- **Power Interference Issue:**
  Initially, all sensors and the servo motor were powered directly from the ESP32's 3.3V and GND pins. This caused interference, particularly affecting the servo motor's operation and the ultrasonic sensors' readings. The root cause was identified as insufficient current supply and voltage instability when the servo motor activated.
- **Solution Implemented:**
  A separate external power supply was connected via the breadboard to power the servo motor independently. This resolved the interference and stabilized the readings from the sensors.

6. Conclusion and Future Work

Conclusion

- In this project, a functional prototype of a smart trash bin was successfully developed using the ESP32 microcontroller, ultrasonic sensors, a servo motor, an OLED display, and a GPS module. The system integrates with the Arduino IoT Cloud to provide real-time monitoring of the bin's fullness, lid control, and GPS location tracking. Additionally, a notification system was implemented to alert users when the bin is over 70% full. The design supports both automatic lid operation based on hand proximity and remote manual control via the cloud dashboard.
- This project demonstrates how IoT technology can be effectively applied to improve waste management efficiency and user convenience, making it suitable for use in smart cities, public areas, or business facilities.

Future Work

- **Multiple Bin Support and Tracking**: In future iterations, the system can be scaled to manage and monitor multiple trash bins simultaneously. This would involve assigning unique identifiers to each bin and displaying their individual statuses (e.g., fullness, location) on a centralized dashboard. This feature would be especially useful for municipalities and large facilities managing several waste containers across different locations.

- **Environmental Bins for Specific Waste Types**: An advanced version of the system could include smart environmental bins that are dedicated to collecting specific types of waste, such as plastics, metals, paper, or electronic waste. These bins could incorporate sensors to detect and sort materials and transmit data to the cloud for better waste categorization and recycling efficiency.

- **Geofencing and Route Optimization**: By combining GPS data from multiple bins, the system can be expanded to optimize collection routes based on bin locations and fullness levels, improving operational efficiency for waste management services.

7. References

- Arduino. (n.d.). *Arduino IoT Cloud Documentation*. Arduino Official Website. Retrieved from https://docs.arduino.cc/cloud/iot-cloud
- Hart, M. (n.d.). *TinyGPSPlus Library* [C++ library]. GitHub. https://github.com/mikalhart/TinyGPSPlus

- TechBuilder. (2022, November 12). *Smart Dustbin using ESP32 and Arduino IoT Cloud | IoT Project with GPS Location and OLED Display* [Video]. YouTube. https://www.youtube.com/watch?v=mHaeDoFOVX0
- Techiesms. (2023, February 19). *Smart Dustbin Using ESP32 and Arduino IoT Cloud - Real time Location, Fullness Notification* [Video]. YouTube. https://www.youtube.com/watch?v=h9g5ln8TCBI

8. GitHub Repository

https://github.com/silahacis/SmartTrashBin