

PRAKTIKUM UJIAN AKHIR SEMESTER

PERANCANGAN SMART TRASH : SISTEM TEMPAT SAMPAH OTOMATIS BERBASIS ESP32

Cagita Dian A'yunin, Tubagus Rangga Gilang Yanuari, Fredlina Devhania Kholishah
cagitadian@student.ub.ac.id, ranggagilang@student.ub.ac.id, devhania88@student.ub.ac.id
233140707111025, 233140707111032, 233140707111035
Fakultas Vokasi, Universitas Brawijaya

Abstrak

Praktikum ini bertujuan untuk merancang dan membangun sistem tempat sampah otomatis berbasis mikrokontroler ESP32 dengan pemrograman melalui Arduino IDE yang hasil datanya dapat diakses melalui website yang telah dibangun. Sistem ini merupakan bentuk penerapan teknologi *Internet of Things (IoT)* dalam kehidupan sehari-hari yang sederhana, praktikum ini bertujuan untuk meningkatkan efisiensi dan kebersihan dalam pengelolaan sampah rumah tangga maupun di lingkungan publik. Beberapa komponen utama yang digunakan dalam proyek ini meliputi sensor ultrasonik HC-SR04 yang berfungsi untuk mendeteksi keberadaan objek seperti tangan atau benda di depan tempat sampah, servo motor MG90S yang menggerakkan tutup tempat sampah secara otomatis, LED sebagai indikator visual saat tempat sampah hampir penuh. Proyek ini dilaksanakan menggunakan perangkat keras fisik secara langsung. Tempat sampah dibuat dari bahan kardus bekas yang dibentuk agar mampu menampung komponen elektronik serta mendukung mekanisme kerja sistem secara keseluruhan. Mikrokontroler ESP32 digunakan sebagai pusat kendali yang mengoordinasikan semua input dari sensor dan mengatur respon aktuator. Hasil pengujian menunjukkan bahwa sistem dapat berjalan dengan baik; deteksi sensor bekerja secara responsif, servo motor membuka dan menutup tutup sesuai perintah, dan indikator LED aktif apabila sampah telah mencapai ambang batas maksimal tempat sampah. Praktikum ini memberikan pondasi yang kuat dalam memahami bagaimana perangkat fisik dapat dikendalikan melalui kode program untuk membentuk sistem dan dapat bermanfaat kedepannya di dunia nyata.

Kata Kunci—Internet of Things, ESP32, Tempat Sampah Otomatis, PlatformIO

Abstract

This practicum aims to design and build an automatic trash can system based on the ESP32 microcontroller with programming through the Arduino IDE whose data results can be accessed through the website that has been built. This system is a form of application of Internet of Things (IoT) technology in simple daily life, this practicum aims to improve efficiency and cleanliness in household waste management and in the public environment. Some of the main components used in this project include the HC-SR04 ultrasonic sensor which functions to detect the presence of objects such as hands or objects in front of the trash can, MG90S servo motor that moves the trash can lid automatically, LED as a visual indicator when the trash can is almost full, and a buzzer that will activate if the weight of the garbage exceeds a certain limit. This project was implemented using physical hardware directly. This practicum provides a strong foundation in understanding how physical devices can be controlled through program code to form a system and can be useful in the future in the real world. The ESP32 microcontroller is used as a control center that coordinates all inputs from sensors and regulates actuator responses. The test results show that the system can run well; the sensor detection works responsively, the servo motor opens and closes the lid according to the command, and the LED indicator is active when the garbage has reached the maximum threshold of the trash can. This practicum provides a strong foundation in understanding how physical devices can be controlled through program code to form a system and can be useful in the future in the real world.

Keywords— Internet of Things, ESP32, Automatic Trash Can, PlatformIO

1. Introduction (Pendahuluan)

Seiring dengan perkembangan teknologi, sistem otomatisasi semakin banyak diterapkan dalam kehidupan sehari-hari guna meningkatkan efisiensi dan kenyamanan. Salah satu bentuk penerapan teknologi tersebut adalah sistem tempat sampah otomatis yang mampu beroperasi tanpa perlu sentuhan langsung dari pengguna. Inovasi ini sangat bermanfaat dalam menjaga kebersihan dan mengurangi penyebaran kuman, terutama di ruang publik seperti sekolah, perkantoran, dan fasilitas umum lainnya. Praktikum ini mengusung konsep simulasi tempat sampah otomatis berbasis mikrokontroler ESP32 dengan memanfaatkan Arduino IDE. Sistem ini mengintegrasikan berbagai sensor dan aktuator, seperti sensor ultrasonik, LED, dan servo motor, yang seluruhnya dikendalikan melalui pemrograman C++ di dalam ESP32. Dengan praktikum ini, dapat menjadi langkah awal dalam memahami proses kerja sistem otomatis serta cara menghubungkan dan mengontrol berbagai komponen elektronik dalam satu kesatuan sistem berbasis Internet of Things (IoT).

1.1 Latar Belakang

Permasalahan kebersihan dan pengelolaan sampah masih menjadi isu penting, khususnya di tempat-tempat umum. Tempat sampah konvensional sering kali menimbulkan risiko penularan penyakit akibat sentuhan langsung, serta kurang efektif dalam memberikan notifikasi saat kapasitas penuh. Oleh karena itu, dibutuhkan sistem tempat sampah yang lebih modern dan efisien, yang dapat bekerja secara otomatis dan memberikan peringatan ketika volume atau berat sampah telah mencapai batas maksimum. Dengan memanfaatkan teknologi mikrokontroler seperti ESP32 serta sensor-sensor pendukung, sistem ini tidak hanya mampu membuka dan menutup tutup tempat sampah secara otomatis, tetapi juga mendeteksi apakah sampah sudah penuh atau belum. Platform Arduino IDE memberikan kemudahan dalam proses perancangan dan pengujian sistem tanpa perlu komponen fisik, sehingga sangat ideal untuk keperluan pembelajaran dan praktikum.

1.2 Tujuan Eksperimen

Eksperimen ini memiliki beberapa tujuan utama, yaitu:

1. Merancang dan mensimulasikan sistem tempat sampah otomatis berbasis ESP32 menggunakan Arduino IDE.
2. Mempelajari cara kerja dan pemrograman sensor ultrasonik HC-SR04, serta aktuator seperti servo motor.
3. Mengintegrasikan berbagai komponen elektronik dalam satu sistem yang mampu mendeteksi keberadaan pengguna dan berat sampah secara real-time.
4. Menampilkan indikator visual (LED) berdasarkan kondisi penuh atau tidaknya tempat sampah.
5. Memberikan pengalaman praktis dalam bidang Internet of Things (IoT), pemrograman mikrokontroler, dan sistem otomatisasi.

2. Methodology (Metodologi)

2.1 Tools & Materials (Alat dan Bahan)

Pada praktikum simulasi tempat sampah otomatis ini, alat dan bahan yang digunakan sepenuhnya berbasis virtual melalui platform Wokwi. Berikut adalah daftar komponen yang digunakan dalam perancangan sistem:

| No. | Komponen | Fungsi |
|-----|-----------------------------|---|
| 1. | ESP32 | Mikrokontroler utama sebagai pusat kendali sistem |
| 2. | Sensor Ultrasonik HC - SR04 | Mendeteksi keberadaan tangan/objek di depan tutup tempat sampah |
| 3. | Sensor Motor MG90S | Membuka dan menutup tutup tempat sampah secara otomatis |
| 4. | LED | Memberikan indikasi visual jika sampah hampir penuh |
| 5. | Kabel Jumper | Menghubungkan antar komponen |
| 6. | Website | web untuk merancang dan menjalankan sistem |

2.2 Implementation Steps (Langkah Implementasi)

Langkah-langkah implementasi praktikum tempat sampah otomatis adalah sebagai berikut:

1. Perancangan Rangkaian di Wokwi
 - Membuka platform Wokwi dan memilih proyek baru berbasis ESP32.
 - Menambahkan komponen-komponen utama seperti sensor HC-SR04, HX710, servo, LED, dan buzzer ke dalam workspace.
 - Menghubungkan kabel virtual antar komponen sesuai dengan diagram rangkaian yang telah dirancang.
2. Pemrograman Mikrokontroler ESP32
 - Menulis program menggunakan bahasa C++ yang berfungsi untuk:
 - Membaca data dari sensor ultrasonik dan HX710.
 - Mengatur gerakan servo berdasarkan deteksi tangan/objek.
 - Menyalakan LED dan buzzer berdasarkan nilai berat sampah.
 - Mengunggah dan menjalankan kode langsung di Wokwi.
3. Pengujian dan Simulasi
 - Menjalankan simulasi dan mengamati bagaimana sistem merespon terhadap objek yang mendekat dan penambahan berat virtual.
 - Memastikan bahwa sistem bekerja sesuai logika: servo membuka saat tangan terdeteksi, LED menyala saat berat mendekati batas, buzzer berbunyi saat penuh.
4. Evaluasi Hasil Simulasi
 - Mengevaluasi waktu respon sensor dan aktuator.
 - Menyesuaikan logika program jika ditemukan error atau ketidaksesuaian fungsi.
 - Menyimpan dan mendokumentasikan hasil simulasi sebagai laporan praktikum.

3. Result dan Discussion (Hasil dan Pembahasan)

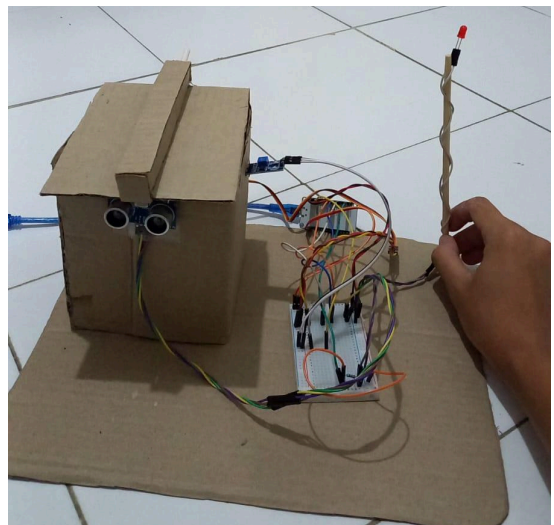
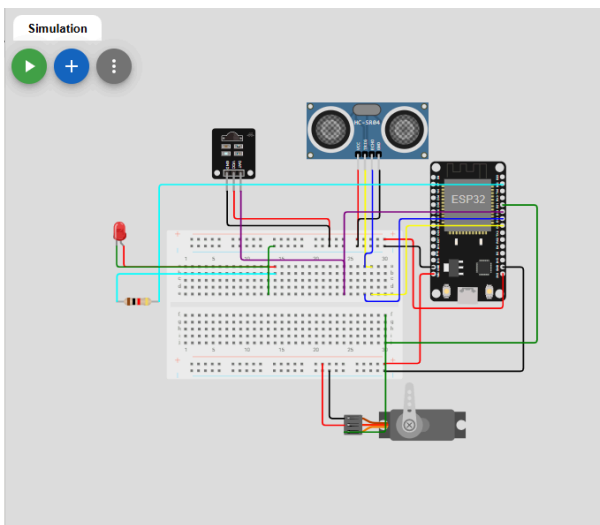
3.1 Experimental Result (Hasil Eksperimen)

1. Rangkaian Sistem yang Telah Dibuat

Sistem ini terdiri dari:

- ESP32 sebagai mikrokontroler utama,

- Sensor Ultrasonik HC-SR04 untuk mendeteksi jarak,
 - Sensor IR (Infrared) untuk mendeteksi keberadaan objek secara langsung,
 - Servo MG90S sebagai aktuator untuk membuka dan menutup penutup,
 - LED sebagai indikator visual berdasarkan deteksi IR,
 - Koneksi WiFi dan MQTT untuk mengirimkan data sensor ke broker MQTT
2. Pengujian Sensor dan Aktuator
- a. Sensor IR dan LED
- Ketika ada objek terdeteksi oleh sensor IR (`irValue == 0`), LED akan menyala.
 - Jika tidak ada objek (`irValue == 1`), LED akan mati.
 - Fungsi ini berjalan tanpa delay yang mengganggu kerja sensor lainnya.
- b. Sensor Ultrasonik dan Servo
- Jika jarak objek < 20 cm, servo akan berputar ke 90° sebagai simbol membuka tutup.
 - Jika objek menjauh > 20 cm, servo akan kembali ke posisi 0° (menutup).
 - Sudut servo telah dikalibrasi agar tidak mentok dan tidak menimbulkan getaran.
- c. Koneksi MQTT
- Modul berhasil dikoneksikan ke WiFi 2.4GHz.
 - Data jarak dan deteksi IR berhasil dikirim ke broker MQTT.
 - Hasil pada Serial Monitor menunjukkan output pengiriman MQTT dan deteksi sensor.
3. Troubleshooting yang Dihadapi
- Awalnya, servo dan LED tidak merespon saat kode menggunakan MQTT, karena WiFi belum dikonfigurasi.
 - Setelah library dan kredensial WiFi ditambahkan, koneksi berhasil dan data berhasil dikirim ke MQTT.
 - Serial Monitor sebelumnya menampilkan titik-titik (...) karena tidak tersambung ke jaringan atau baud rate tidak sesuai. Setelah diperbaiki, data tampil normal.



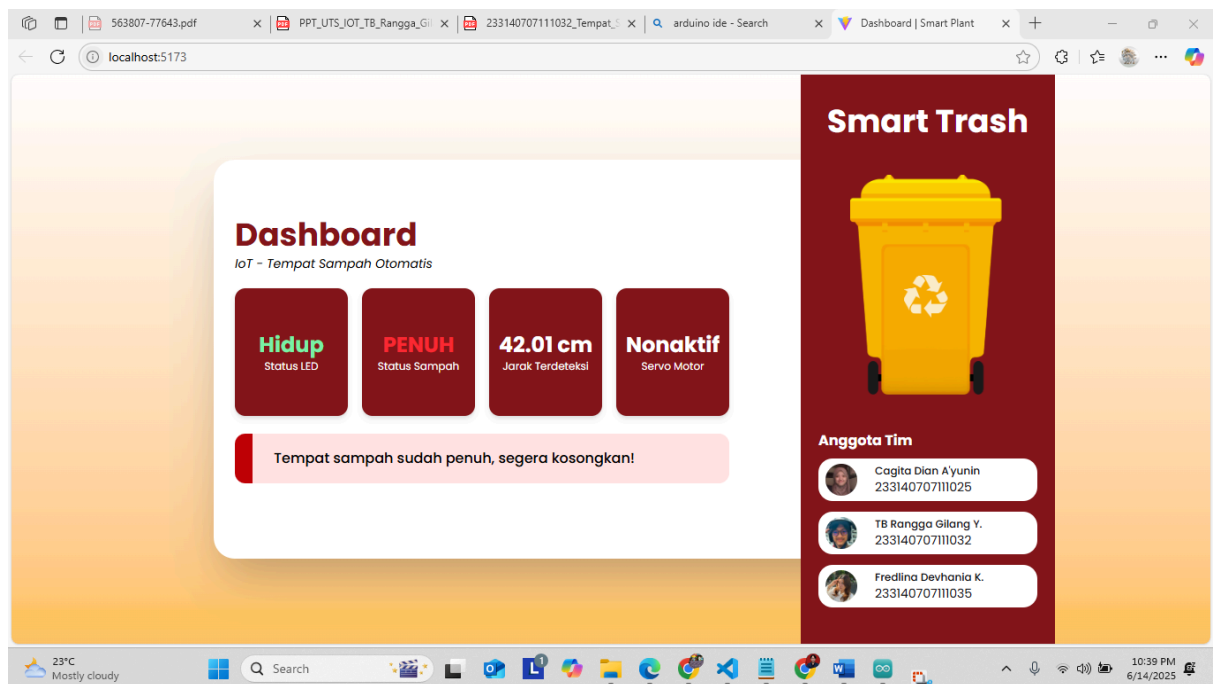
Berikut adalah screenshot hasil dari simulasi di Arduino IDE dan Webiste buatan kelompok kami:

```
97 // Publish data ke MQTT
98 client.publish("iot/trash/distance", String(distance).c_str());
99 client.publish("iot/trash/led", objectDetected ? "ON" : "OFF");
100 client.publish("iot/trash/servo", distance < 15 ? "AKTIF" : "NONAKTIF");
101 client.publish("iot/trash/status", objectDetected ? "PENUH" : "AMAN");
102
103 // Debug ke serial monitor
104 Serial.print("Jarak: ");
105 Serial.print(distance);
106 Serial.print(" cm | IR: ");
107 Serial.print(irValue);
108 Serial.print(" | Servo: ");
109 Serial.print(isOpen ? "Buka" : "Tutup");
110 Serial.print(" | LED: ");
111 Serial.print(objectDetected ? "ON" : "OFF");
112 Serial.print(" | Status: ");
113 Serial.println(objectDetected ? "PENUH" : "AMAN");
114
115 delay(1000); // Delay antar pembacaan
116 }
117
```

Output Serial Monitor X

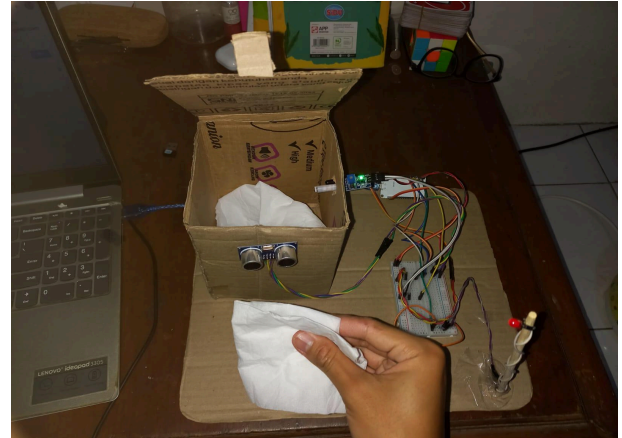
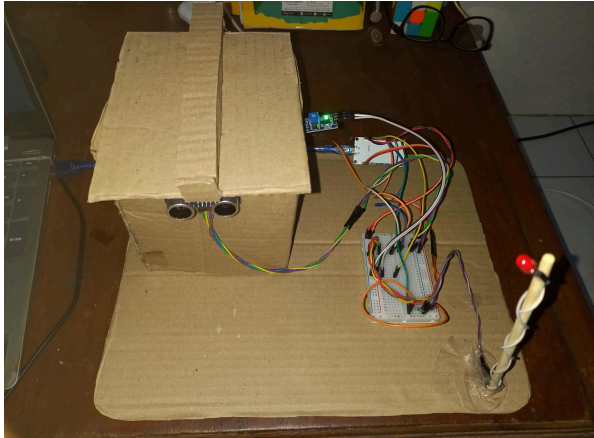
Message (Enter to send message to 'ESP32 Dev Module' on 'COM7')

Jarak: 29.55 cm | IR: 1 | Servo: Tutup | LED: OFF | Status: AMAN
Jarak: 41.77 cm | IR: 1 | Servo: Tutup | LED: OFF | Status: AMAN
Jarak: 29.55 cm | IR: 1 | Servo: Tutup | LED: OFF | Status: AMAN
Jarak: 29.55 cm | IR: 1 | Servo: Tutup | LED: OFF | Status: AMAN
Jarak: 10.91 cm | IR: 1 | Servo: Buka | LED: OFF | Status: AMAN



3.2 Pengujian Sensor Ultrasonik

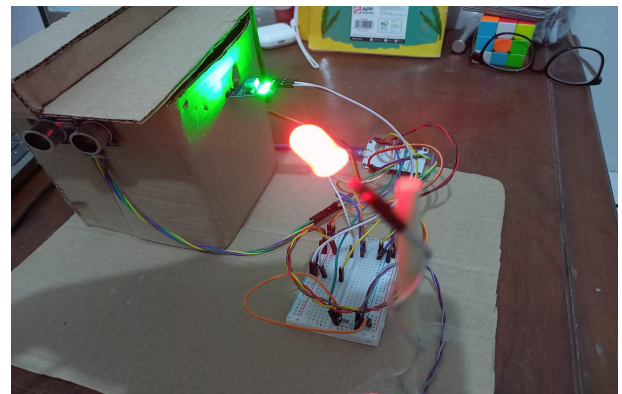
- Objek didekatkan ke sensor.
- Servo motor membuka tutup saat jarak < 20 cm.
- Setelah 3 detik, tutup kembali.



Mendekatkan Sampah pada sensor Ultrasonik yang kemudian tutup sampah akan terbuka dengan Servo Motor untuk penggerak tempat sampah lalu kita bisa memasukkan sampah ke dalam tempat sampah

4.2.2 Pengujian Sensor IR dan LED

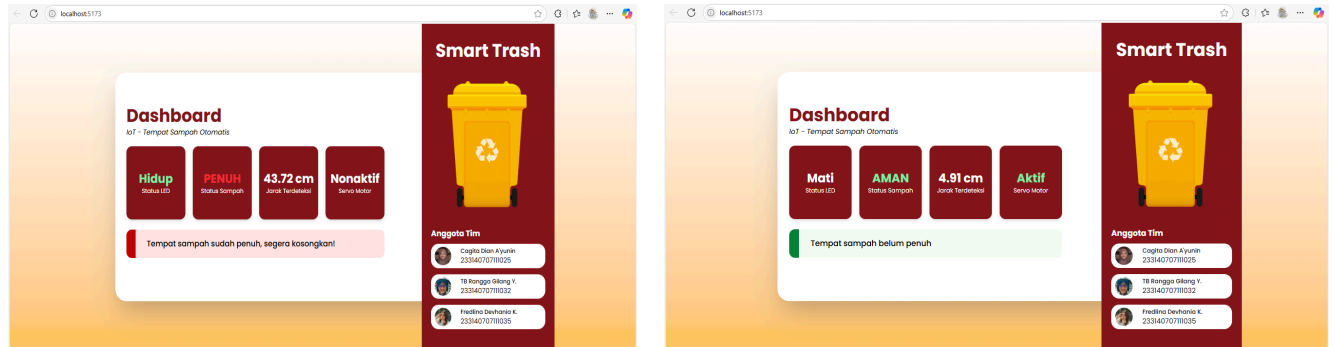
- Sensor IR mendeteksi objek (logika LOW).
- LED menyala menandakan tempat sampah penuh.
- LED mati jika tidak ada objek terdeteksi.



Ketika sudah memasukkan sampah kedalam tempat sampah, Kemudian sensor IR (infrared) kemudian akan mendeteksi apakah sampah sudah penuh atau belum dengan skema jika sampah sudah mendekati sensor atau sudah ada di depan sensor IR maka nanti LED akan menyala yang menandakan bahwa sampah telah penuh dan harus segera dibuang

4.2.3 Pengujian Koneksi MQTT

- ESP32 mengirim data ke broker HiveMQ.
- Data seperti jarak, status IR, dan status servo terkirim dengan lancar.
- Monitoring berhasil menggunakan MQTT tools



Kami juga menyediakan Website yang langsung terintegrasi dengan alat IOT Smart Trash dimana disini akan menampilkan pesan Penuh jika sampah sudah penuh dan menampilkan pesan Aman jika sampah masih kosong atau belum penuh, dan menampilkan informasi lain seperti jarak sampah pada sensor Ultrasonik, Status Servo, dan Status LED yang dimana ini menggunakan skema MQTT tools untuk bisa tersambung dan memudahkan dalam monitoring sampah.

4. Appendix (Lampiran)

4.1 Kode Program Esp32

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ESP32Servo.h>

// Pin Konfigurasi
#define TRIG_PIN 5
#define ECHO_PIN 18
#define LED_PIN 22
#define SERVO_PIN 21
#define IR_PIN 15

// WiFi dan MQTT
const char* ssid = "L3LORONG";
const char* password = "LantaiLorong";
const char* mqtt_server = "broker.hivemq.com";
```



```

WiFiClient espClient;
PubSubClient client(espClient);
Servo servo;
bool isOpen = false;

void setup_wifi() {
  Serial.println("Connecting to WiFi...");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nConnected to WiFi!");
}

void reconnect() {
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    if (client.connect("esp32TrashClient")) {
      Serial.println("Connected to MQTT");
    } else {
      Serial.print("Failed, rc=");
      Serial.print(client.state());
      delay(5000);
    }
  }
}

void setup() {
  Serial.begin(115200);

  // Setup pin
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  pinMode(LED_PIN, OUTPUT);
  pinMode(IR_PIN, INPUT);

  // Setup servo
  servo.setPeriodHertz(50);
  servo.attach(SERVO_PIN, 500, 2400);
  servo.write(0); // Tutup awal
  isOpen = false;
}

```



```

// Setup WiFi dan MQTT
setup_wifi();
client.setServer(mqtt_server, 1883);
}

void loop() {
    if (!client.connected()) {
        reconnect();
    }
    client.loop();

    // Baca sensor IR
    int irValue = digitalRead(IR_PIN);
    bool objectDetected = irValue == 0; // 0 artinya benda terdeteksi

    // Baca jarak dengan sensor ultrasonik
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    long duration = pulseIn(ECHO_PIN, HIGH);
    float distance = duration * 0.034 / 2;

    // Kontrol Servo
    if (distance < 20 && !isOpen) {
        servo.write(180); // Buka
        isOpen = true;
        delay(3000);
    } else if (distance >= 20 && isOpen) {
        servo.write(0); // Tutup
        isOpen = false;
    }

    // Kontrol LED dari IR sensor
    digitalWrite(LED_PIN, objectDetected ? HIGH : LOW);

    // Publish data ke MQTT
    client.publish("iot/trash/distance", String(distance).c_str());
    client.publish("iot/trash/led", objectDetected ? "ON" : "OFF");
    client.publish("iot/trash/servo", distance < 15 ? "AKTIF" :
"NONAKTIF");

```

```
    client.publish("iot/trash/status", objectDetected ? "PENUH" :  
"AMAN");  
  
    // Debug ke serial monitor  
    Serial.print("Jarak: ");  
    Serial.print(distance);  
    Serial.print(" cm | IR: ");  
    Serial.print(irValue);  
    Serial.print(" | Servo: ");  
    Serial.print(isOpen ? "Buka" : "Tutup");  
    Serial.print(" | LED: ");  
    Serial.print(objectDetected ? "ON" : "OFF");  
    Serial.print(" | Status: ");  
    Serial.println(objectDetected ? "PENUH" : "AMAN");  
  
    delay(1000); // Delay antar pembacaan  
}
```