

DAFTAR PUSTAKA

- Aldino De Lama, Unang Sunarya, & Atik Novianti (2016). *Deteksi Logam Pada Penggiling Batu berbasis SMS Gateway dan Mikrotokonroler*.
- Andry Andaru, (2018). *Pengertian Database Secara Umum.OSF Preprint*
- Afrizal Dwi Kusuma. (2019). *Penggunaan Telegram Bot Pada Telegram Messenger dengan Metode WEBHOOKS untuk Sistem Peminjaman Infrastruktur di UIN Maulana Malik Ibrahim Malang*.
- Benny Suhendar, Tb. Dedy Fuady, & Yoga Herdian. (2021). *Rancang Bangun Sistem Monitoring dan Controlling Suhu Ideal Tanaman Stoberi Berbasis Internet of Things (IoT)*.
- Chika Raichany, Tama, Husnul Khatimah & Purnama Putra, (2023). *Pelatihan dan Penyuluhan Tentang Pengolahan Sampah Organik dan Anorganik Berbasis Zero Wast*.
- Dewi Lusitas Hidayati Nurul. (2019). *Prototype smart home dengan modul nodemcu esp8266 berbasis internet of things (iot)*.Doctoral dissertation.
- Erni Setyaningsih, Dhidik Prastiyanto, & Suryono. (2017). *Penggunaan Sensor Photodiode sebagai Sistem Deteksi Api pada Wahana Terbang Vertical Take-Off Landing (VTOL)*.
- Guntoro. (2019). *Aplikasi Mamajemen Data dan Aplikasi Katalog Pemasaran Bisnis Properti Berbasis Android Menggunakan Firebase Realtime Database*.
- Irma Kurnia Juliany, Muhammad Salamuddin, & Yuni Kristina Dewi. (2018). *Perancangan Sistem Informasi E-Marketplace Bank Sampah Berbasis Web. Semnasteknomedia Online*.
- Kim, J D. (2007). Evaluation of pilot-scale in-vessel composting for food waste treatment. *Journal of Hazardous Materials*. Volume 154. Issues 103. Page 272-277.
- Mohamad Rizal (2011). *Analisis Pengelolaan Persampahan Perkotaan : Cipta karya Pekerjaan Umum Kabupaten Sigi, Sulawesi Tengah*. Volume 9 Nomor 1
 lei 2011.
- per, & Govinda Jibrillah Hasan. (2021). *Rancang Bangun Buka Tutup Otomatis Sampah Otomatis Berbasis Arduino*.



Purwanto dan Sulistyastuti. (1991). *Analisis Kebijakan dari Formula ke Implementasi Kebijakan*.

Romi Shaputra, Pamor Gunoto, & Muhammad Irsyam. (2019). *Krain Air Otomatis Pada Tempat Berwhudu Menggunakan Sensor Ultrasonik Berbasis Arduino Uno*.

Yuni Puspitawati, Mardwi Rahdriawan (2012). *Kajian Pengelolaan Sampah Berbasis Masyarakat dengan Konsep 3R (Reduce, Reuse, Recycle) di Kelurahan Larangan Kota Cirebon*.



LAMPIRAN

Lampiran 1 Program penentuan setiap jenis sampah

```

PemilahSampah_arduino | Arduino 1.8.19
File Edit Sketch Tools Help
PemilahSampah_arduino
#include <Servo.h>
#include <DHT.h>

#define infraredPin1 2 // Pin digital untuk sensor infrared di wadah 1
#define metalSensorPin1 6 // Pin digital untuk sensor metal NJK-5001C (Sensor 1)
#define metalSensorPin2 3 // Pin digital untuk sensor metal NJK-5001C (Sensor 2)
#define servoPin1 9 // Pin untuk servo di wadah 1

#define infraredPin2 4 // Pin digital untuk sensor infrared di wadah 2
#define dhtPin 5 // Pin digital untuk sensor DHT22
#define servoPin2 10 // Pin untuk servo di wadah 2

#define DHTTYPE DHT22 // Jenis sensor DHT

DHT dht(dhtPin, DHTTYPE);
Servo servol, servo2;

const int servoPosisiAwal = 90;
const int servoPosisiLogam = 0;
const int servoPosisiKering = 0;
const int servoPosisiNonLogam = 180;
const int servoPosisiBasah = 180;

bool deteksiSampahWadah1 = false;
bool deteksiSampahWadah2 = false;
float kelembapanAwal = 0.0;

float bacaKelembapan() {
    float humidity = dht.readHumidity();
    return (isnan(humidity)) ? 0.0 : humidity;
}

void setup() {
    Serial.begin(9600);
    pinMode(infraredPin1, INPUT);
    pinMode(metalSensorPin1, INPUT);
    pinMode(metalSensorPin2, INPUT);
    pinMode(servoPin1, OUTPUT);
    pinMode(infraredPin2, INPUT);
    pinMode(dhtPin, INPUT);
    pinMode(servoPin2, OUTPUT);

    servol.attach(servoPin1);
    servo2.attach(servoPin2);

    servol.write(servoPosisiAwal);
    servo2.write(servoPosisiAwal);

    dht.begin();

    // Membaca kelembapan awal saat alat menyala
    kelembapanAwal = bacaKelembapan();
}

void loop() {
    // Wadah 1: Deteksi Sampah dan Pemilahan Logam/Non-Logam
    if (!deteksiSampahWadah1 && digitalRead(infraredPin1) == LOW) {
        Serial.println("Sampah terdeteksi di Wadah 1.");
        deteksiSampahWadah1 = true;
        delay(2000); // Jeda 1 detik sebelum sensor metal memulai pembacaan
        if (digitalRead(metalSensorPin1) == LOW || digitalRead(metalSensorPin2) == LOW) {
            Serial.println("Deteksi Logam. Sampah diarahkan ke Tempat Sampah Logam.");
            servol.write(servoPosisiLogam);
            delay(2000); // Jeda 2 detik untuk perubahan posisi servo
            servol.write(servoPosisiAwal);
            delay(2000);
            deteksiSampahWadah1 = false;
        } else {

```



```

    } else {
//      Serial.println("Deteksi Non-Logam. Sampah diarahkan ke wadah2.");
servo1.write(servoPosisiNonLogam);
delay(2000); // Jeda 2 detik untuk perubahan posisi servo
servo1.write(servoPosisiAwal);
    }
}

// Wadah 2: Deteksi Sampah dan Pemilahan Basah/Kering
if (deteksiSampahWadah1 && digitalRead(infraredPin2) == LOW) {
//      Serial.println("Sampah terdeteksi di Wadah 2.");
float humidity = bacaKelembapan();
delay(20000); // Jeda 20 detik sebelum membaca kelembapan
if (humidity - kelembapanAwal > 2.5) {
//      Serial.println("Deteksi Sampah Basah. Sampah diarahkan ke Tempat Sampah Basah.");
servo2.write(servoPosisiBasah);
delay(2000); // Jeda 2 detik untuk perubahan posisi servo
servo2.write(servoPosisiAwal);
    } else {
//      Serial.println("Deteksi Sampah Kering. Sampah diarahkan ke Tempat Sampah Kering.");
servo2.write(servoPosisiKering);
delay(2000); // Jeda 2 detik untuk perubahan posisi servo
servo2.write(servoPosisiAwal);
    }
}
deteksiSampahWadah1 = false;
delay(5000); // Jeda 5 detik sebelum kembali ke state awal
}
}

```

Lampiran 2 Program monitoring sampah.

```

esp_gabung | Arduino 1.8.19
File Edit Sketch Tools Help
esp_gabung$
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
#include <NewPing.h>

// Wifi network station credentials
#define WIFI_SSID "folkstar"
#define WIFI_PASSWORD "107107107"
// Telegram BOT Token (Get from Botfather)
#define BOT_TOKEN "7032699396:AAGuJL3Dkb6NrbpG8zZqfXTcpMSIOFqnEOg"
#define CHAT_ID "1035078145"

const unsigned long BOT_MTBS = 1000; // mean time between scan messages

X509List cert(TELEGRAM_CERTIFICATE_ROOT);
WiFiClientSecure secured_client;
UniversalTelegramBot bot(BOT_TOKEN, secured_client);
unsigned long bot_lasttime; // last time messages' scan has been done

#define TRIG_PIN D6
#define ECHO_PIN D1 // Pin D2
#define TRIG2_PIN D7
#define ECHO2_PIN D2 // Pin D8
#define TRIG3_PIN D8
#define ECHO3_PIN D4 // Pin D7
#define MAX_DISTANCE 200

#define Buz D5

NewPing sonar1(TRIG_PIN, ECHO_PIN, MAX_DISTANCE);
NewPing sonar2(TRIG2_PIN, ECHO2_PIN, MAX_DISTANCE);
NewPing sonar3(TRIG3_PIN, ECHO3_PIN, MAX_DISTANCE);

// Nilai awal untuk sensor dan tinggi tempat sampah
int initialValue1 = 40; // Nilai awal sensor 1
int initialValue2 = 49; // Nilai awal sensor 2
int initialValue3 = 49; // Nilai awal sensor 3
int trashHeight = 28; // Tinggi tempat sampah

```



```
// Variabel untuk menyimpan persentase isi tempat sampah
float percentagel = 0;
float percentage2 = 0;
float percentage3 = 0;

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

  // attempt to connect to Wifi network:
  Serial.print("Connecting to Wifi SSID ");
  Serial.print(WIFI_SSID);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  secured_client.setTrustAnchors(&cert); // Add root certificate for api.t

  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(".");
    delay(500);
  }
  Serial.print("\nWiFi connected. IP address: ");
  Serial.println(WiFi.localIP());

  Serial.print("Retrieving time: ");
  configTime(0, 0, "pool.ntp.org"); // get UTC time via NTP
  time_t now = time(nullptr);
  while (now < 24 * 3600)
  {
    Serial.print(".");
    delay(100);
    now = time(nullptr);
  }
  Serial.println(now);

  pinMode(Buz, OUTPUT);
}
```



```
void loop()
{
  unsigned long currentMillis = millis();

  if (currentMillis - bot_lasttime > BOT_MTBS)
  {
    int numNewMessages = bot.getUpdates(bot.last_message_received + 1);

    while (numNewMessages)
    {
      Serial.println("got response");
      handleNewMessages(numNewMessages);
      numNewMessages = bot.getUpdates(bot.last_message_received + 1);
    }

    bot_lasttime = millis();
  }

  delay(50);

  // Mengukur jarak dari masing-masing sensor
  unsigned int distance1 = sonar1.ping_cm();
  unsigned int distance2 = sonar2.ping_cm();
  unsigned int distance3 = sonar3.ping_cm();

  // Menghitung ketinggian aktual berdasarkan perbedaan dengan nilai awal
  int currentHeight1 = initialValue1 - distance1;
  int currentHeight2 = initialValue2 - distance2;
  int currentHeight3 = initialValue3 - distance3;

  // Menghitung persentase isi tempat sampah
  percentage1 = (float)currentHeight1 / trashHeight * 100;
  percentage2 = (float)currentHeight2 / trashHeight * 100;
  percentage3 = (float)currentHeight3 / trashHeight * 100;
}
```



```

// Kirim pesan ke Telegram jika ketinggian melebihi 26 cm
if (currentHeight1 > 26)
{
    bot.sendMessage(CHAT_ID, "Sampah logam penuh!", "");
    digitalWrite(Buz, HIGH);
}
if (currentHeight2 > 26)
{
    bot.sendMessage(CHAT_ID, "Sampah non-organik penuh!", "");
    digitalWrite(Buz, HIGH);
}
if (currentHeight3 > 26)
{
    bot.sendMessage(CHAT_ID, "Sampah organik penuh!", "");
    digitalWrite(Buz, HIGH);
}
else
{
    digitalWrite(Buz, LOW);
}
delay(1000);
}

void handleNewMessages(int numNewMessages)
{
    for (int i = 0; i < numNewMessages; i++)
    {
        String command = bot.messages[i].text; // Mendapatkan teks pesan

        // Menangani perintah /start
        if (command == "/start")
        {
            bot.sendMessage(bot.messages[i].chat_id, "Selamat datang!\n\nBot Pemberitahuan Sampa
        }
        // Menangani perintah /status
        else if (command == "/status")
        {
            String message = "Tempat Sampah Logam: Terisi " + String(percentage1) + " %\n" +

```

```

void handleNewMessages(int numNewMessages)
{
    for (int i = 0; i < numNewMessages; i++)
    {
        String command = bot.messages[i].text; // Mendapatkan teks pesan

        // Menangani perintah /start
        if (command == "/start")
        {
            bot.sendMessage(bot.messages[i].chat_id, "Selamat datang!\n\nBot Pemberitahuan Sampa
        }
        // Menangani perintah /status
        else if (command == "/status")
        {
            String message = "Tempat Sampah Logam: Terisi " + String(percentage1) + " %\n" +
                "Tempat Sampah Non-organik: Terisi " + String(percentage2) + " %\n"
                "Tempat Sampah Organik: Terisi " + String(percentage3) + " %";
            bot.sendMessage(bot.messages[i].chat_id, message, ""); // Membalas dengan status ter
        }
        // Menangani perintah lainnya
        else
        {
            bot.sendMessage(bot.messages[i].chat_id, "Perintah tidak dikenali", ""); // Membalas
        }
    }
}

```





Optimized using
trial version
www.balesio.com