

LAMPIRAN

Lampiran 1. Surat Kesiediaan Membimbing TA Dosen Pembimbing I

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

Nama : Ida Afriliana, S.T.M.Kom
NIPY : 12.013.168
Jabatan Struktural : KA. Prodi DIII Teknik Komputer
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut:

Nama : Andika Ristianto
NIM : 22041084
Program Studi : Diploma III Teknik Komputer

Judul TA : RANCANG BANGUN PROTOTYPE PENERING BAHAN MENTAH
JAMU TRADISIONAL DENGAN THERMOELECTRIC BERBASIS
INTERNET OF THINGS (IOT)

Dengan ini menyatakan ini dibuat agar dilaksanakan sebagaimana mestinya

Tegal, 6 Februari 2025

Mengetahui

Ka Prodi DIII Teknik Komputer,

Dosen Pembimbing I,


Ida Afriliana, S.T.M.Kom
NIPY. 12.013.168


Ida Afriliana, S.T.M.Kom
NIPY. 12.013.168

Lampiran 2. Surat Kesiediaan Membimbing TA Dosen Pembimbing II

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

Nama : Abdul Basit, S.Kom, M.T
NIPY : 01.015.198
Jabatan Struktural : Sekretaris Prodi DIII Teknik Komputer
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut:

Nama : Andika Ristianto
NIM : 22041084
Program Studi : Diploma III Teknik Komputer

Judul TA : RANCANG BANGUN PROTOTYPE PENGERING BAHAN MENTAH JAMU TRADISIONAL DENGAN THERMOELECTRIC BERBASIS INTERNET OF THINGS (IOT)

Dengan ini menyatakan ini dibuat agar dilaksanakan sebagaimana mestinya

Tegal, 5 Februari 2025

Mengetahui

Ka Prodi DIII Teknik Komputer,



Abdul Basit, S.T. M.Kom
NIPY. 01.015.198

Dosen Pembimbing II,

Abdul Basit, S.Kom, M.T
NIPY. 01.015.198

Lampiran 3. *Source Code* Program

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include "DHT.h"
#include "HX711.h"
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <ArduinoJson.h>

// ----- DHT Configuration -----
-----
#define DHTPIN D4
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// ----- LCD I2C Configuration -----
-----
LiquidCrystal_I2C lcd(0x27, 20, 4);

// ----- Relay & Buzzer Configuration -----
-----
#define RELAY_PIN D3 // GPIO0
#define BUZZER_PIN D8 // GPIO14 (ubah sesuai sambungan)

// ----- HX711 Load Cell Configuration -----
-----
#define DT D7 // GPIO13
#define SCK D6 // GPIO12
HX711 scale;

// ----- WiFi Configuration -----
-----
const char* ssid = "Redmi 9C";
const char* password = "888888888";
const char* serverUrl = "http://192.168.43.222/monitoring/simpan_data.php";
const int user_id = 1;

// ----- Variabel Berat -----
---
float berat_awal = 0;

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

    // Inisialisasi sensor
    dht.begin();
    scale.begin(DT, SCK);
    scale.set_scale(-920.f);

    // Inisialisasi Output
    pinMode(RELAY_PIN, OUTPUT);
    digitalWrite(RELAY_PIN, LOW);

    pinMode(BUZZER_PIN, OUTPUT);
```

```

digitalWrite(BUZZER_PIN, LOW);

// Inisialisasi LCD
lcd.begin();
lcd.backlight();

// Koneksi WiFi
WiFi.begin(ssid, password);
lcd.setCursor(0, 0);
lcd.print("Menghubungkan");
lcd.setCursor(0, 1);
lcd.print("ke WiFi...");

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    lcd.print(".");
}

Serial.println("\nWiFi Terhubung");
Serial.print("IP Address: ");
Serial.println(WiFi.localIP());

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("WiFi Terhubung");
lcd.setCursor(0, 1);
lcd.print("IP: ");
lcd.print(WiFi.localIP());
delay(3000);
lcd.clear();

// Tahapan kalibrasi berat awal
lcd.setCursor(0, 0);
lcd.print("Kosongkan timbangan");
Serial.println("Kosongkan timbangan...");
delay(30000); //Delay selama 30 detik

scale.tare();
Serial.println("Tare selesai.");

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Letakkan benda");
Serial.println("Letakkan benda...");
delay(60000); // Delay selama 1 menit

berat_awal = scale.get_units(10);
Serial.print("Berat awal: ");
Serial.println(berat_awal);

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Berat Awal:");
lcd.setCursor(0, 1);

```

```

    lcd.print((int)berat_awal);
    lcd.print(" gram");
    delay(2000);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Sistem Siap");
    Serial.println("Sistem siap.");
    delay(1000);
    lcd.clear();
}

void loop() {
    float humidity = dht.readHumidity();
    float temperature = dht.readTemperature();
    float berat = scale.get_units(10);

    if (abs(berat) < 10) berat = 0;

    // Validasi DHT
    if (isnan(humidity) || isnan(temperature)) {
        Serial.println("Sensor DHT Error");
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Sensor DHT Error");
        digitalWrite(RELAY_PIN, LOW);
        digitalWrite(BUZZER_PIN, LOW);
        delay(2000);
        return;
    }

    // --- Tampilkan Serial Monitor ---
    Serial.print("Suhu: ");
    Serial.print(temperature);
    Serial.print(" C | Hum: ");
    Serial.print(humidity);
    Serial.print(" % | Berat: ");
    Serial.print((int)berat);
    Serial.println(" gram");

    // --- Tampilkan LCD ---
    lcd.clear();

    // Baris 1: Judul
    lcd.setCursor(0, 0);
    lcd.print("PENGERING REMPAH");

    // Baris 2: Suhu
    lcd.setCursor(0, 1);
    lcd.print("SUHU: ");
    lcd.print(temperature, 1);
    lcd.print((char)223);
    lcd.print("C");

    // Baris 3: Kelembapan
    lcd.setCursor(0, 2);

```

```

lcd.print("KELEMBAPAN: ");
lcd.print(humidity, 0);
lcd.print("%");

// Baris 4: Berat
lcd.setCursor(0, 3);
lcd.print("BERAT: ");
lcd.print((int)berat);
lcd.print("gram");

    // --- Kontrol Relay ---
    if (berat < berat_awal * 0.9) {
        digitalWrite(RELAY_PIN, LOW); // Matikan peltier jika berat
turun >10%
    } else {
        if (temperature < 45.0) {
            digitalWrite(RELAY_PIN, HIGH);
        } else {
            digitalWrite(RELAY_PIN, LOW);
        }
    }

    // --- Logika Buzzer Aktif: Bip-Bip tiap 500ms saat berat turun
>10% ---
    static unsigned long buzzerLastToggle = 0;
    const unsigned long buzzerOnDuration = 1000;
    const unsigned long buzzerOffDuration = 1000;
    static bool buzzerState = false;

    if (berat < berat_awal * 0.9) {
        unsigned long currentMillis = millis();
        if (buzzerState && currentMillis - buzzerLastToggle >=
buzzerOnDuration) {
            digitalWrite(BUZZER_PIN, LOW);
            buzzerState = false;
            buzzerLastToggle = currentMillis;
        } else if (!buzzerState && currentMillis - buzzerLastToggle
>= buzzerOffDuration) {
            digitalWrite(BUZZER_PIN, HIGH);
            buzzerState = true;
            buzzerLastToggle = currentMillis;
        }
    } else {
        digitalWrite(BUZZER_PIN, LOW);
        buzzerState = false;
    }

    // --- Kirim Data ke Server ---
    if (WiFi.status() == WL_CONNECTED) {
        WiFiClient client;
        HTTPClient http;

        // Format data JSON
        DynamicJsonDocument doc(200);

```

```

doc["suhu"] = temperature;
doc["kelembaban"] = humidity;
doc["berat_awal"] = berat_awal;
doc["berat"] = berat;
doc["peltier"] = digitalRead(RELAY_PIN);
doc["user_id"] = user_id;

String jsonData;
serializeJson(doc, jsonData);

http.begin(client, serverUrl);
http.addHeader("Content-Type", "application/json");

int httpCode = http.POST(jsonData);

if (httpCode > 0) {
    String payload = http.getString();
    Serial.println("Response: " + payload);

    // Parsing response JSON
    DynamicJsonDocument resDoc(200);
    deserializeJson(resDoc, payload);

    String status = resDoc["status"];
    String message = resDoc["message"];
    String peltierStatus = resDoc["peltier_status"];

    Serial.println("Status: " + status);
    Serial.println("Message: " + message);
    Serial.println("Peltier: " + peltierStatus);
} else {
    Serial.println("Error on HTTP request: " +
String(httpCode));
}
http.end();
} else {
    Serial.println("WiFi Disconnected");
}

delay(60000); // Update setiap 1 menit
}

```