

Lampiran 1. Surat Kesiediaan Membimbing 1

SURAT KESEDIAAN MEMBIMBING TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Miftakhul Huda, M.Kom

NIDN : 0620127801

NIPY : 04.007.033

Jabatan Struktural :

Jabatan Fungsional : Lektor

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

Nama : Rafli Ramadhan

NIM : 21040098

Program Studi : D-III Teknik Komputer

Judul TA : RANCANG BANGUN PEMANTAUAN TINGGI AIR BERBASIS IoT DAN SOLUSI OTOMATISASI DENGAN POMPA DI DESA SIDAKATON

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 18 Maret 2024

Mengetahui,
Ka. Prodi D-III Teknik Komputer



Ida Airliana, ST, M.Kom
NIPY. 12.013.168

Dosen Pembimbing I

Miftakhul Huda, M.Kom
NIPY.04.007.033

Lampiran 2. Surat Kesiediaan Membimbing 2

SURAT KESEDIAAN MEMBIMBING TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Arfan Haqiqi Sulasmoro, M.Kom
NIDN : 0623037704
NIPY : 02.009.054
Jabatan Struktural :
Jabatan Fungsional : Asisten Ahli

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

Nama : Rafli Ramadhan
NIM : 21040098
Program Studi : D-III Teknik Komputer

Judul TA : RANCANG BANGUN PEMANTAUAN TINGGI
AIR BERBASIS IoT DAN SOLUSI
OTOMATISASI DENGAN POMPA DI DESA
SIDAKATON

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 18 Maret 2024

Mengetahui,
Ka. Prodi D-III Teknik Komputer

Dosen Pembimbing II



Ida Afrilliana, ST, M.Kom
NIPY. 12.013.168

Arfan Haqiqi Sulasmoro, M.Kom
NIPY.02.009.054

Lampiran 3 *Script Program ESP8266 ke Arduino IDE*

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define trig D5 // Definisikan pin trig sebagai D5 (GPIO 14)
#define echo D6 // Definisikan pin echo sebagai D6 (GPIO 12)
#define relayPin D7 // Definisikan pin relay sebagai D7 (GPIO 13)
#define buzzerPin D8 // Definisikan pin buzzer sebagai D8 (GPIO 15)
#define greenLedPin D3 // Definisikan pin LED hijau sebagai D3 (GPIO 0)
#define redLedPin D4 // Definisikan pin LED merah sebagai D4 (GPIO 2)

const char* ssid = "TA";
const char* password = "testtal23";
const char* serverName = "https://monitor.arketipe.my.id/monitoring/simpanMonitoring";

long durasi;
int jarak;

LiquidCrystal_I2C lcd(0x27, 16, 2); // Alamat I2C mungkin berbeda,
sesuaikan jika diperlukan

void setup() {
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(relayPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
  pinMode(greenLedPin, OUTPUT);
  pinMode(redLedPin, OUTPUT);
  Serial.begin(9600); // Inisialisasi komunikasi serial pada baud
  rate 9600

  lcd.init(); // Inisialisasi LCD
  lcd.backlight(); // Mengaktifkan backlight LCD
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Mengukur Jarak");

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Menghubungkan ke WiFi...");
  }

  Serial.println("Terhubung ke WiFi");
}

void loop() {
  // Mengirim sinyal ultrasonik
```

```

digitalWrite(trig, LOW);
delayMicroseconds(5);
digitalWrite(trig, HIGH);
delayMicroseconds(10);
digitalWrite(trig, LOW);

// Mengukur durasi pantulan sinyal ultrasonik
durasi = pulseIn(echo, HIGH);

// Menghitung jarak berdasarkan durasi
jarak = durasi * 0.034 / 2;

// Menampilkan jarak di Serial Monitor
Serial.print("Jarak = ");
Serial.print(jarak);
Serial.println(" cm");

// Menampilkan jarak di LCD
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Jarak: ");
lcd.print(jarak);
lcd.print(" cm");

// Mengendalikan relay, buzzer, dan LED berdasarkan jarak
if (jarak < 25) { // Jika jarak kurang dari 25 cm
digitalWrite(relayPin, LOW); // Matikan relay
digitalWrite(buzzerPin, HIGH); // Nyalakan buzzer
digitalWrite(redLedPin, HIGH); // Nyalakan LED merah
digitalWrite(greenLedPin, LOW); // Matikan LED hijau
lcd.setCursor(0, 1);
lcd.print("Relay ON");
} else { // Jika jarak 25 cm atau lebih
digitalWrite(relayPin, HIGH); // Nyalakan relay
digitalWrite(buzzerPin, LOW); // Matikan buzzer
digitalWrite(redLedPin, LOW); // Matikan LED merah
digitalWrite(greenLedPin, HIGH); // Nyalakan LED hijau
lcd.setCursor(0, 1);
lcd.print("Relay OFF ");
}

// Mengirim data ke server
if (WiFi.status() == WL_CONNECTED) {
WiFiClientSecure client;
HTTPClient http;

client.setInsecure(); // Hanya digunakan untuk testing, untuk
produksi harus menggunakan sertifikat yang valid

http.begin(client, serverName);

http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

// Persiapkan data untuk dikirim
String httpRequestData = "tinggi_air=" + String(jarak);

```

```

int httpResponseCode = http.POST(httpRequestData);

if (httpResponseCode > 0) {
String response = http.getString();
Serial.println(httpResponseCode);
Serial.println(response);
digitalWrite(greenLedPin, HIGH); // Nyalakan LED hijau
digitalWrite(redLedPin, LOW); // Matikan LED merah
} else {
Serial.print("Error saat mengirim POST: ");
Serial.println(httpResponseCode);
digitalWrite(greenLedPin, LOW); // Matikan LED hijau
digitalWrite(redLedPin, HIGH); // Nyalakan LED merah
}

http.end();
} else {
Serial.println("WiFi Terputus");
digitalWrite(greenLedPin, LOW); // Matikan LED hijau
digitalWrite(redLedPin, HIGH); // Nyalakan LED merah
}

delay(1000); // Tunggu 1 detik sebelum pengukuran berikutnya
}

```

Lampiran 4. Foto Alat Ketinggian Air dan Pompa Otomatis

