

LAMPIRAN A
KEGIATAN PENGAMBILAN DATA







LAMPIRAN B

Program Rancang Bangun Tingkat Multifungsi Sebagai Alat Bantu Petunjuk Jalan Tunanetra

```
// library
#include <DFPlayer_Mini_Mp3.h>
#include <SoftwareSerial.h>

// inialisasi playlist suara
#define depan_ada_halangan 1
#define kanan_ada_halangan 2
#define kiri_ada_halangan 3
#define bawah_ada_halangan 4
#define depan_ada_lubang 5

// pin sensor
int echoPin1 = 6;
int initPin1 = 7;
int distance1 = 0;

int echoPin2 = 8;
int initPin2 = 9;
int distance2 = 0;

int echoPin3 = 10;
int initPin3 = 11;
int distance3 = 0;

int echoPin4 = 12;
int initPin4 = 13;
int distance4 = 0;

//
unsigned int uS_atas;
unsigned int uS_bawah;
unsigned int uS_kanan;
```

```

unsigned int uS_kiri;

int sp_atas = 60;
int sp_kanan = 30;
int sp_kiri = 30;
int sp_pendek = 35;
int sp_panjang = 100;

int flag;

long proveus;
String bawah, kondisi;

SoftwareSerial dfplayer(3, 2);

void setup() {
  Serial.begin(57600);
  dfplayer.begin(9600);
  mp3_set_serial(dfplayer);
  delay(5);
  mp3_set_volume(30);
  pinMode(initPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(initPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
  pinMode(initPin3, OUTPUT);
  pinMode(echoPin3, INPUT);
  pinMode(initPin4, OUTPUT);
  pinMode(echoPin4, INPUT);
}

void loop() {
  readultra();
  mainn();
}

void mainn() {
  if (uS_bawah > sp_pendek && uS_bawah < sp_panjang) {
    bawah = "aman";
  }
}

```

```

}
else if (uS_bawah < sp_pendek && uS_bawah < sp_panjang) {
    bawah = "halangan";
}
// else if (uS_bawah > sp_pendek && uS_bawah > sp_panjang) {
//     bawah = "lubang";
// }
// else if (uS_bawah < sp_pendek && uS_bawah > sp_panjang) {
//     bawah = "lubang dan halngan"; // tapi ndak mungkin
// }

if (uS_atas < sp_atas) {
    flag = 1;
    kondisi = "depan ada halangan";
    mp3_play (1);
    delay (2500);
}
if (uS_kanan < sp_kanan) {
    flag = 2;
    kondisi = "kanan ada halangan";
    mp3_play (2);
    delay (2500);
}
if (uS_kiri < sp_kiri) {
    flag = 3;
    kondisi = "kiri ada halangan";
    mp3_play (3);
    delay (2500);
}
if (bawah == "halangan") {
    flag = 4;
    kondisi = "bawah ada halangan";
    mp3_play (4);
    delay (2500);
}
if (bawah == "lubang") {
    flag = 5;
    kondisi = "bawah ada lubang";
    mp3_play (5);
}

```

```

    delay (2500);
}

if ((millis() - proveus) > 500) {
    Serial.print("depan: ");
    Serial.print(uS_atas);
    Serial.println(" cm");
    Serial.print("bawah: ");
    Serial.print(uS_bawah);
    Serial.println(" cm");
    Serial.print("kanan: ");
    Serial.print(uS_kanan);
    Serial.println(" cm");
    Serial.print("kiri: ");
    Serial.print(uS_kiri);
    Serial.println(" cm");
    Serial.print("bawah: ");
    Serial.println(bawah);
    Serial.println("kondisi: " + kondisi);
    Serial.println("flag: " + String(flag));
    Serial.println();
    proveus = millis();
}
}

void readultra() {
    distance1 = getDistance(initPin1, echoPin1);
    delay(150);
    distance2 = getDistance(initPin2, echoPin2);
    delay(150);
    distance3 = getDistance(initPin3, echoPin3);
    delay(150);
    distance4 = getDistance(initPin4, echoPin4);
    delay(150);

    uS_kanan = distance1 + 11;
    uS_bawah = distance2 + 12;
    uS_atas = distance3 + 10;
    uS_kiri = distance4 + 14;
}

```



```

printDistance("kanan", distance1); // kanan
printDistance("bawah", distance2); // bawah
printDistance("depan", distance3); // atas
printDistance("kiri", distance4); // kiri
Serial.println();
}

int getDistance (int initPin, int echoPin) {
digitalWrite(initPin, LOW);
delayMicroseconds(2);
digitalWrite(initPin, HIGH);
delayMicroseconds(10);
digitalWrite(initPin, LOW);
unsigned long pulseTime = pulseIn(echoPin, HIGH);
int distance = pulseTime / 58;
return distance;
}

void printDistance(String id, int dist) {
Serial.print(id);
Serial.print("----->");
Serial.print(dist, DEC);
Serial.println(" cm");
}

```

Pemrograman di Mikrokontroler ESP 32

```

#include <TinyGPS++.h>

#include <SoftwareSerial.h>

#include <WiFi.h>

#include <FirebaseESP32.h>

#include <addons/TokenHelper.h>

#include <addons/RTDBHelper.h>

#include <DFPlayer_Mini_Mp3.h>

```

```
#define WIFI_SSID "p"

#define WIFI_PASSWORD "qwertyuiop"

#define API_KEY "AIzaSyCP9pczmXF4L6ElUYZiK-
15NJAEPn-y3w"

#define DATABASE_URL "gps-tracking-37542-default-
rtdb.firebaseio.com" //<databaseName>.firebaseio.com or
<databaseName>.<region>.firebasedatabase.app

#define USER_EMAIL "gpstrackingpnc2022@gmail.com"

#define USER_PASSWORD "gpstracking2022"

#define pintombol 25

float latitude , longitude;

String lat_str = "0", lng_str = "0";

long proveus;

long interval = 5000;

int echoPin1 = 4;

int initPin1 = 5;

int us_atas = 0;

int echoPin2 = 18;

int initPin2 = 13;

int us_bawah = 0;

int distance1 = 0;
```

```
int distance2 = 0;
```

```
int sp_bawah_panjang = 50;
```

```
int sp_bawah_pendek = 50;
```

```
int sp_atas = 50;
```

```
byte didepanadahalangan = 1;
```

```
byte dibawahadahalangan = 1;
```

```
byte dibawahadalubang = 1;
```

```
FirebaseData fbdo;
```

```
FirebaseAuth auth;
```

```
FirebaseConfig config;
```

```
TinyGPSPlus gps;
```

```
SoftwareSerial SerialGPS;
```

```
SoftwareSerial dfplayer;
```

```
void setup()
```

```
{
```

```
  Serial.begin(115200);
```

```
  SerialGPS.begin(9600, SWSERIAL_8N1, 17, 16, false);
```

```
  dfplayer.begin(9600, SWSERIAL_8N1, 32, 33, false);
```

```

mp3_set_serial(dfplayer);

delay(5);

mp3_set_volume(15);

delay(100);

pinMode(pintombol, INPUT_PULLUP);

pinMode(initPin1, OUTPUT);

pinMode(echoPin1, INPUT);

pinMode(initPin2, OUTPUT);

pinMode(echoPin2, INPUT);

WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

Serial.print("Connecting to Wi-Fi");

while (WiFi.status() != WL_CONNECTED)

{

  Serial.print(".");

  delay(300);

}

Serial.println();

Serial.print("Connected with IP: ");

Serial.println(WiFi.localIP());

Serial.println();

Serial.printf("Firebase Client v%s\n\n",
FIREBASE_CLIENT_VERSION);

```

```

config.api_key = API_KEY;

auth.user.email = USER_EMAIL;

auth.user.password = USER_PASSWORD;

config.database_url = DATABASE_URL;

config.token_status_callback = tokenStatusCallback; //see
addons/TokenHelper.h

Firebase.begin(&config, &auth);

Firebase.reconnectWiFi(true);

}

void loop()

{

  read_gps();

  if (digitalRead(pintombol) == 0) {

    Serial.printf("Set string... %s\n", Firebase.setString(fbdo,
F("/notif"), "1") ? "ok" : fbdo.errorReason().c_str());

    delay(1000);

    Serial.printf("Set string... %s\n", Firebase.setString(fbdo,
F("/notif"), "0") ? "ok" : fbdo.errorReason().c_str());

  }

}

```

```

void read_gps() {
    if (SerialGPS.available() > 0) {
        if (gps.encode(SerialGPS.read()))
        {
            if (gps.location.isValid())
            {
                latitude = gps.location.lat();
                lat_str = String(latitude , 6);
                longitude = gps.location.lng();
                lng_str = String(longitude , 6);

                Serial.printf("Set string... %s\n", Firebase.setString(fbdo,
F("/Latitude"), String(lat_str)) ? "ok" : fbdo.errorReason().c_str());

                Serial.printf("Set string... %s\n", Firebase.setString(fbdo,
F("/Longitude"), String(lng_str)) ? "ok" : fbdo.errorReason().c_str());

                Serial.print("Latitude = ");
                Serial.println(lat_str);

                Serial.print("Longitude = ");
                Serial.println(lng_str);
            }
            Serial.println();
        }
    }
}

```

```
}
```

```
void readultra() {
```

```
    us_atas = getDistance(initPin1, echoPin1);
```

```
    delay(150);
```

```
    us_bawah = getDistance(initPin2, echoPin2);
```

```
    delay(150);
```

```
    printDistance(1, distance1);
```

```
    printDistance(2, distance2);
```

```
    Serial.println("");
```

```
}
```

```
int getDistance (int initPin, int echoPin) {
```

```
    digitalWrite(initPin, HIGH);
```

```
    delayMicroseconds(10);
```

```
    digitalWrite(initPin, LOW);
```

```
    unsigned long pulseTime = pulseIn(echoPin, HIGH);
```

```
    int distance = pulseTime / 58;
```

```
    return distance;
```

```
}
```

```
void printDistance(int id, int dist) {  
    Serial.print(id);  
    Serial.print("----->");  
    Serial.print(dist, DEC);  
    Serial.println(" cm");  
}
```


LAMPIRAN C

DOKUMENTASI MENGGUNAKAN ALAT



2. Tombol untuk menyalakan dan mematikan alat



3. Sebagai tombol darurat pada saat tunanetra dalam keadaan bahaya



4. Sebagai tombol untuk mengatur hendak menggunakan headset atau speaker.



5. Sensor ultrasonic untuk mendeteksi adanya halangan depan .



6. Sensor ultrasonik untuk mendeteksi adanya halangan kanan.



7. Sensor ultrasonik untuk mendeteksi adanya halangan kiri



8. Sensor ultrasonik untuk mendeteksi adanya halangan bawah

LAMPIRAN D
DOKUMENTASI ALAT



1. Tongkat multifungsi tampak depan



2. Tongkat multifungsi tampak samping

BIODATA PENULIS



Nama : Yunita Eka Pratiwi
Tempat, Tanggal Lahir : Cilacap, 8 Juni 2001
Agama : Islam
Alamat : Jalan Diponegoro Utara no 27 rt 02
rw 09 Kuripan, Kesugihan, Cilacap
Email : yunitaekapратиwi58@gmail.com
Nomor Telepon : 0858-4272-8913
Hobi : Berbisnis dan menyanyi
Moto Hidup : Ridho Allah tergantung dari Ridho
orang tua dan murka Allah
tergantung dari murka orangtua.

Riwayat Pendidikan

Sekolah / Institusi/ Universitas	Jurusan	Periode
SD Negeri 2 Kuripan	-	2007-2013
SMP Negeri 3 Kesugihan	-	2013-2016
SMA Negeri 1 Jeruklegi	IPA	2017-2019
Politeknik Negeri Cilacap	D3 Teknik Elektronika	2019-2022

Penulis telah mengikuti seminar Tugas Akhir pada 19 Agustus 2022 sebagai salah satu persyaratan untuk memperoleh gelar Ahli Madya (A.Md).