

Dokumentasi Alat Smoke Bending

1. Source Code

a. Raw Source Code

- Source ESP8266

(Github) SmokeBending_ESP8266

(bit.ly/smokebendingcode)

[illegible]

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0xfc, 0x78, 0xf0, 0xfc, 0x38, 0x38,
0xff, 0x80, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x06, 0x03, 0xfc, 0x78, 0xf1, 0xfe, 0x38, 0x70,
0xff, 0x80, 0x00, 0x00, 0x00,
0x00, 0x00, 0x01, 0xff, 0x07, 0x8c, 0x78, 0xf1, 0xfe, 0x38, 0xf0,
0xff, 0x80, 0x00, 0x00, 0x00,
0x00, 0x00, 0x0f, 0xff, 0x07, 0x00, 0x7d, 0xf3, 0x87, 0x39, 0xe0,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x3f, 0xff, 0x07, 0x00, 0x7d, 0xf3, 0x87, 0x39, 0xc0,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x01, 0xff, 0xff, 0x07, 0x80, 0x7d, 0xf3, 0x87, 0x3f, 0x80,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x7f, 0xff, 0xff, 0x03, 0xf0, 0x77, 0x73, 0x87, 0x3f, 0x80,
0xff, 0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xff, 0xff, 0x01, 0xf8, 0x77, 0x73, 0x87, 0x3f, 0xc0, 0xff,
0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xff, 0xff, 0x00, 0x7c, 0x77, 0x73, 0x87, 0x3f, 0xc0, 0xff,
0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xff, 0xff, 0x00, 0x1e, 0x77, 0x73, 0x87, 0x3d, 0xe0,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xff, 0xff, 0x00, 0x0e, 0x70, 0x73, 0x87, 0x38, 0xe0,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xff, 0xfe, 0x00, 0x0e, 0x70, 0x73, 0x87, 0x38, 0xf0,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xff, 0x00, 0x06, 0x1e, 0x70, 0x71, 0xcf, 0x38, 0x70,
0xe0, 0x00, 0x00, 0x00, 0x00,
0x03, 0xff, 0xf8, 0x00, 0x07, 0xfc, 0x70, 0x71, 0xfe, 0x38, 0x38,
0xff, 0x80, 0x00, 0x00, 0x00,

0x03, 0xff, 0xc0, 0x00, 0x07, 0xf8, 0x70, 0x70, 0xfc, 0x38, 0x38,
0xff, 0x80, 0x00, 0x00, 0x00,
0x03, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x03, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x01, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x0f, 0xff, 0x07, 0xf8, 0x3f, 0xe3, 0x87, 0x1f, 0xc0,
0xff, 0x8e, 0x1c, 0x1f, 0xc0,
0x00, 0x00, 0x3f, 0xff, 0x07, 0xfc, 0x3f, 0xe3, 0xc7, 0x1f, 0xe0,
0xff, 0x8e, 0x1c, 0x3f, 0xc0,
0x00, 0x01, 0xff, 0xff, 0x07, 0x1e, 0x38, 0x03, 0xc7, 0x1f, 0xf0,
0x3c, 0x0f, 0x1c, 0x78, 0xc0,
0x00, 0x7f, 0xff, 0xff, 0x07, 0x0e, 0x38, 0x03, 0xe7, 0x1c, 0x70,
0x1c, 0x0f, 0x1c, 0x70, 0x00,
0x03, 0xff, 0xff, 0xff, 0x07, 0x0e, 0x38, 0x03, 0xe7, 0x1c, 0x78,
0x1c, 0x0f, 0x9c, 0xf0, 0x00,
0x03, 0xff, 0xff, 0xff, 0x07, 0x1c, 0x38, 0x03, 0xe7, 0x1c, 0x38,
0x1c, 0x0d, 0x9c, 0xe0, 0x00,
0x03, 0xff, 0xff, 0xff, 0x07, 0xf8, 0x3f, 0xe3, 0xb7, 0x1c, 0x38,
0x1c, 0x0d, 0x9c, 0xe0, 0x00,
0x03, 0xff, 0xff, 0xff, 0x07, 0xfc, 0x3f, 0xe3, 0xb7, 0x1c, 0x38,
0x1c, 0x0c, 0xdc, 0xe3, 0xc0,
0x03, 0xff, 0xff, 0xfe, 0x07, 0x0e, 0x38, 0x03, 0xbf, 0x1c, 0x38,
0x1c, 0x0c, 0xdc, 0xe3, 0xc0,
0x03, 0xff, 0xff, 0x00, 0x07, 0x0e, 0x38, 0x03, 0x9f, 0x1c, 0x38,
0x1c, 0x0c, 0xfc, 0xe0, 0xc0,
0x03, 0xff, 0xf8, 0x00, 0x07, 0x0e, 0x38, 0x03, 0x9f, 0x1c, 0x78,
0x1c, 0x0c, 0x7c, 0xf0, 0xc0,
0x03, 0xff, 0xc0, 0x00, 0x07, 0x0e, 0x38, 0x03, 0x8f, 0x1c, 0x70,
0x1c, 0x0c, 0x7c, 0x70, 0xc0,
0x03, 0xff, 0x00, 0x00, 0x07, 0x1e, 0x3f, 0xe3, 0x8f, 0x1f, 0xf0,
0xff, 0x0c, 0x3c, 0x7d, 0xc0,
0x03, 0xf0, 0x00, 0x00, 0x07, 0xfe, 0x3f, 0xe3, 0x8f, 0x1f, 0xe0,
0xff, 0x8c, 0x3c, 0x3f, 0xc0,
0x00, 0x00, 0x00, 0x00, 0x07, 0xf8, 0x3f, 0xe3, 0x87, 0x1f, 0x80,
0xff, 0x8c, 0x3c, 0x1f, 0x80,

```

0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00
};

```

```

#define FIREBASE_HOST
"smokebending-86277-default-rtdb.asia-southeast1.firebaseio.com" //Without http:// or https:// schemes //sesuaikan dengan
FIREBASE_HOST masing-masing
#define FIREBASE_AUTH "RWi*****cWg" //sesuaikan dengan
FIREBASE_HOST masing-masing
#define WIFI_SSID "SmokeBending" //sesuaikan dengan hotspot yang
digunakan
#define WIFI_PASSWORD "1234567890"

//Define FirebaseESP8266 data object
FirebaseData MQ7;
FirebaseData FAN;

```

```
FirebaseData SPEED;

void setup()
{

  Serial.begin(9600);

  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(RELAY1, OUTPUT);
  pinMode(RELAY2, OUTPUT);
  pinMode(RESET, INPUT);

  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  display.drawBitmap(0, 0, smokebendingLogo, 128, 64, 1);
  display.display();
  delay(3000);

  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println("Wifi connecting to ");
  display.println(WIFI_SSID);
  display.println("\nConnecting");
  display.display();

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
  {
    WiFiManager wm;
    wm.autoConnect("Smoke-Bending");
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  display.clearDisplay();
  display.setCursor(0, 24);
  display.setTextWrap(0);
  display.setCursor(19, 24);
  display.println("Wifi connected!");
```

```

display.setCursor(0, 38);
display.drawRoundRect(13, 22, 100, 12, 3, 1);
display.setCursor(13, 38);
display.println(WiFi.localIP());

display.display();
delay(2000);

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true);
}

void loop() {
  int val;
  int statusfan = 0;
  float sensorValue;
  float sensorVoltage;

  sensorValue = analogRead(A0);
  sensorVoltage = sensorValue/1024*5.0;

  val = digitalRead(RESET); // check the state of the button
  if(val==LOW) // if button is pressed, turn LED on
  {
    Serial.println("RESTART");
    ESP.restart();
  }

  if (Firebase.setFloat(MQ7, "/smokebending/sensor1/mq7",
sensorVoltage))
  {
    Serial.println("PASSED");
  } else
  {
    displayolederror();
    Serial.println("REASON: " + MQ7.errorReason());
    return;
  }

  int kondisi = 1;

  if(sensorVoltage > 1 && sensorVoltage < 3 && online==0){
    digitalWrite(RELAY1, HIGH);
  }
}

```

```

digitalWrite(RELAY2, LOW);
fanstatus = "ON";
tone(13, 450);
speedstatus = "SLOW";
}
else if (sensorVoltage >= 3 && online==0){
digitalWrite(RELAY1, HIGH);
digitalWrite(RELAY2, HIGH);
tone(13, 900);
fanstatus = "ON";
speedstatus = "FAST";
}
else if (online==0) {
digitalWrite(RELAY1, LOW);
digitalWrite(RELAY2, LOW);
noTone(13);
fanstatus = "OFF";
speedstatus = "OFF";
}

if (Firebase.getInt(FAN, "/smokebending/sensor2/fan")){
if (FAN.stringData() == "1") {
online = 1;
digitalWrite(RELAY1, HIGH);
fanstatus = "ON";
statusfan = 1;
}
else if (FAN.stringData() == "0" && online == 1){
online = 0;
digitalWrite(RELAY1, LOW);
fanstatus = "OFF";
statusfan = 0;
speedstatus = "OFF";
}
}

if (Firebase.getInt(SPEED, "/smokebending/sensor3/fanspeed") &&
statusfan == 1 && online == 1){
if (SPEED.stringData() == "1") {
digitalWrite(RELAY2, HIGH);
speedstatus = "FAST";
}
else if (SPEED.stringData() == "0"){
digitalWrite(RELAY2, LOW);

```

```

    speedstatus = "SLOW";
    }
}

displayoled(sensorVoltage, fanstatus);
delay(1000);
}

void displayoled(float mq7value, String fanstatus){
    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.print("MQ-7: ");
    display.setTextSize(2);
    display.setCursor(0, 10);
    display.print(mq7value);
    display.print(" %");
    display.setCursor(0, 30);
    display.setTextSize(1);
    display.println("Fan:");
    display.setCursor(0, 40);
    display.setTextSize(2);
    display.println(fanstatus+" (" +speedstatus+"");
    display.display();
}

void displayolederror() {
    digitalWrite(RELAY1, LOW);
    digitalWrite(RELAY2, LOW);
    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0, 10);
    display.setTextWrap(0);
    display.setCursor(13, 10);
    display.println("Terjadi Kesalahan");
    display.setTextWrap(0);
    display.setCursor(19, 18);
    display.println("dalam melakukan");
    display.setTextWrap(0);
    display.setCursor(40, 26);
    display.println("koneksi!");
    display.setTextWrap(0);
    display.setCursor(28, 34);
    display.println("(Cek kembali)");
    display.setTextWrap(0);

```



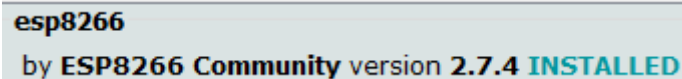
```
display.setCursor(13, 42);  
display.println("koneksi internet");  
display.display();  
}
```

- Source Website
[\(\[Github\]\(https://github.com/SmokeBending/SmokeBending-Website\)\) SmokeBending-Website](https://github.com/SmokeBending/SmokeBending-Website)
(bit.ly/smokebendingweb)

b. Penjelasan Code

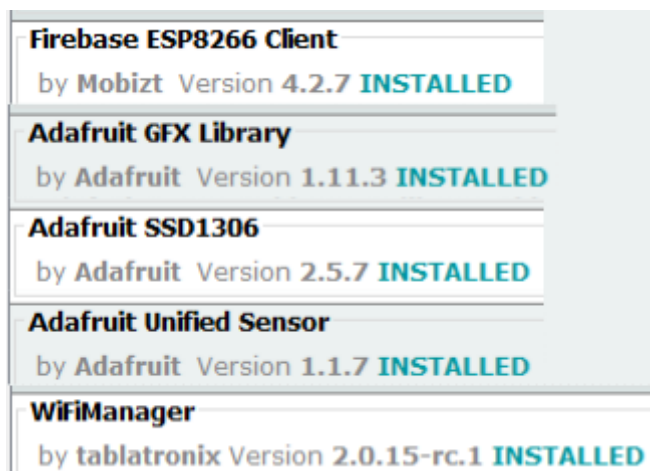
Pada alat ini kita menggunakan ESP8266 dan Arduino IDE sebagai teks editor kita, untuk menyambungkan ESP8266 ke Arduino IDE sebelumnya kita harus menginstall board yang sesuai. Pertama kita ke menu File>Preference dan pada bagian Additional Board Manager URL's kita isi dengan "https://arduino.esp8266.com/stable/package_esp8266com_index.json" kemudian klik OK.

Kemudian ke menu Tools>Board>Boards Manager, lalu cari "ESP8266" lalu pilih versi 2.7.4 dikarenakan jika kita memilih diatas versi tersebut, terdapat sebuah bug yang tidak bisa mengcompile program yang sudah kita buat.



esp8266
by ESP8266 Community version 2.7.4 INSTALLED

Kemudian install library yang dibutuhkan pada alat kita dengan list di bawah ini:



Firebase ESP8266 Client by Mobizt Version 4.2.7 INSTALLED
Adafruit GFX Library by Adafruit Version 1.11.3 INSTALLED
Adafruit SSD1306 by Adafruit Version 2.5.7 INSTALLED
Adafruit Unified Sensor by Adafruit Version 1.1.7 INSTALLED
WiFiManager by tablatronix Version 2.0.15-rc.1 INSTALLED

untuk mengupload value dari sensor gas kita, yaitu sensor MQ7, kita gunakan kondisi if else sebagai berikut:

```
if (Firebase.setFloat(MQ7, "/smokebending/sensor1/mq7", sensorVoltage))
{
    Serial.println("PASSED");
} else
{
    displayolederror();
    Serial.println("REASON: " + MQ7.errorReason());
    return;
}
```

Dimana pada kode tersebut kita menggunakan *setFloat* untuk mengirim nilai sensor MQ7 ke database Firebase dengan path *"/smokebending/sensor1/mq7"*. Fungsi ini akan mengembalikan boolean yang menandakan apakah operasi penyimpanan berhasil atau tidak.

untuk mengambil data dari firebase ke ESP8266 kita, kita gunakan kode sebagai berikut:

```
if (Firebase.getInt(FAN, "/smokebending/sensor2/fan")){
    if (FAN.stringData() == "1") {
        online = 1;
        digitalWrite(RELAY1, HIGH);
        fanstatus = "ON";
        statusfan = 1;
    }
    else if (FAN.stringData() == "0" && online == 1){
        online = 0;
        digitalWrite(RELAY1, LOW);
        fanstatus = "OFF";
        statusfan = 0;
        speedstatus = "OFF";
    }
}
```

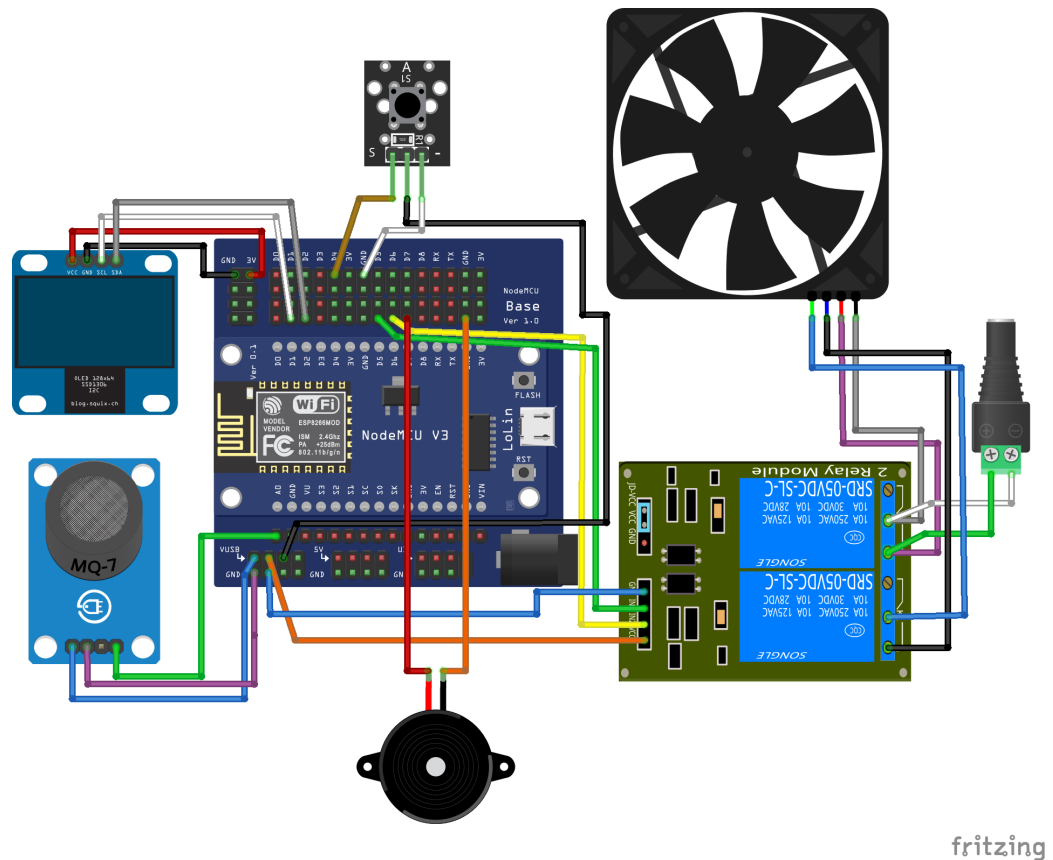
Sama seperti kode sebelumnya, yang membedakan adalah kita menggunakan perintah *getInt* untuk mengambil nilai yang ada di Firebase ke alat yang kita miliki.

Sebagai referensi, silahkan lihat dokumentasi library repository berikut:

(<https://github.com/mobizt/Firebase-ESP8266>)

2. Embedded System

a. Rancangan Model



b. Penjelasan Rancangan

Alat ini terdiri dari beberapa komponen utama, yaitu:

1. Baseboard Nodemcu V3: Merupakan papan sirkuit yang menjadi dasar alat IoT sensor gas. Papan ini terhubung ke internet melalui modul ESP8266.
2. ESP8266: Merupakan modul wifi yang memungkinkan alat ini terhubung ke internet.
3. Sensor Gas MQ-7: Merupakan sensor yang digunakan untuk mendeteksi tingkat gas di sekitar lingkungan.
4. Oled ssd1306: Merupakan layar yang digunakan untuk menampilkan informasi tingkat gas yang dideteksi oleh sensor.
5. Buzzer: Merupakan alat yang digunakan untuk memberikan notifikasi suara saat tingkat gas di sekitar lingkungan melebihi batas normal.
6. Relay 2 Channel: Merupakan alat yang digunakan untuk mengontrol kipas, saat tingkat gas melebihi batas normal.
7. Kipas 12 V: Merupakan alat yang digunakan untuk mengalirkan udara segar ke dalam ruangan saat tingkat gas melebihi batas normal.
8. KY-004 Key Switch: Merupakan tombol yang digunakan untuk mengaktifkan atau menonaktifkan alat ini.

9. DC Power Jack: Merupakan port yang digunakan untuk menghubungkan alat ini ke sumber daya listrik.
10. Kabel Jumper: Merupakan kabel yang digunakan untuk menghubungkan komponen-komponen alat ini.

Untuk list komponen yang dibutuhkan adalah sebagai berikut:

No	Nama alat	Jumlah	Pin
1	Baseboard Node MCU V3	1	-
2	ESP8266	1	-
3	Sensor Gas MQ-7	1	A0, 5V, GND
4	Oled ssd1306	1	D1, D2, 3V, GND
5	Buzzer	1	D7, GND
6	Relay 2 Channel	1	D5, D6, 3V, GND
7	Kipas 12 V	1	Relay, Power Jack
8	KY-004 Key Switch	1	D4, 5V, GND
9	DC Power Jack	1	Adaptor
10	Kabel Jumper	18	-

3. Protocol

a. Protokol yang digunakan

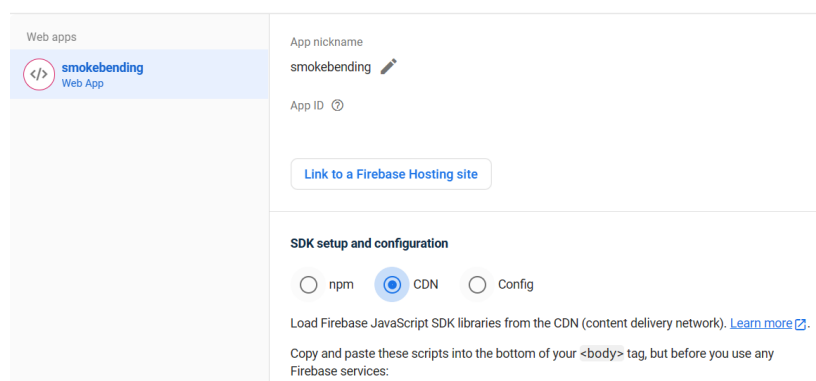
Protokol yang digunakan dalam alat ini adalah menggunakan protokol HTTP (Hypertext Transfer Protocol) merupakan protokol yang digunakan untuk mentransfer data di internet, sedangkan API (Application Programming Interface) merupakan antarmuka yang memungkinkan aplikasi untuk berkomunikasi dengan aplikasi lain atau sistem. HTTP dan API seringkali digunakan bersama-sama untuk mengirim dan menerima data antara aplikasi yang berjalan di internet. API menggunakan HTTP sebagai protokol untuk mentransfer data antara aplikasi. Jadi, API menggunakan HTTP untuk mengirim permintaan ke server dan menerima respon dari server. HTTP digunakan oleh API untuk menentukan tindakan yang dilakukan pada permintaan, seperti mengirim atau menerima data, serta menyertakan informasi tambahan dalam header permintaan atau respon. Sebagai contoh, ketika aplikasi mobile mengirim permintaan ke server untuk mendapatkan data dari database, permintaan tersebut akan dikirim melalui HTTP menggunakan API yang telah ditentukan.

b. Penjelasan Protokol

Dari program diatas, kita menggunakan Google Firebase dengan menggunakan layanan “Realtime Database” sebagai database untuk menyimpan data-data yang dihasilkan oleh sensor alat kita. Pada “Firebase Console” kita buat terlebih dahulu table database sebagai gambar berikut:



Buatlah sebuah web apps, dimana pilih bagian CDN dan masukan link tersebut ke bagian javascript module, bisa dilihat pada source code website, dan buatlah rules read & write nya menjadi allow agar bisa diubah melalui website kita.



Maka kita akan mendapatkan link api yang berikan oleh firebase, seperti pada link berikut:

<https://smokebending-86277-default-rtdb.asia-southeast1.firebaseio.com/p/smokebending.json>

Jika kita mengunjungi link tersebut, maka akan mendapatkan respon mengenai data json yang sudah kita deploy di firebase.

```
{"sensor1":{"mq7":0.29297},"sensor2":{"fan":0},"sensor3":{"fanspeed":0}}
```

Untuk menampilkan data yang ada di json ke website kita, kita harus mem-fetch api yang sudah kita buat, dimana pada contoh ini kita ingin merubah nilai element h1 (dengan id value-mq7) pada html berdasarkan data yang ada di database. Disini kita gunakan javascript sebagai berikut:

```
const valueMq7 = document.getElementById("value-mq7");

const url = "https://smokebending-86277-default-rtdb.asia-southeast1.firebaseio.com/smokebending.json";

var api = {};

setInterval(() => {
  fetch(url)
    .then((hasil) => hasil.json())
    .then((res) => {
      api = res;
      // console.log(api);
      valueMq7.innerHTML = api.sensor1.mq7.toFixed(2).toString() + "V";
    });
}, 1000);
```

Maka nanti api database akan ditampilkan pada elemen h1 html kita, sebagai berikut:



Untuk melatih dalam penggunaan JavaScript, silahkan cari referensi di Internet tentang DOM (Document Object Model) untuk mengakses dan mengubah elemen HTML di halaman web dan Fetch API.

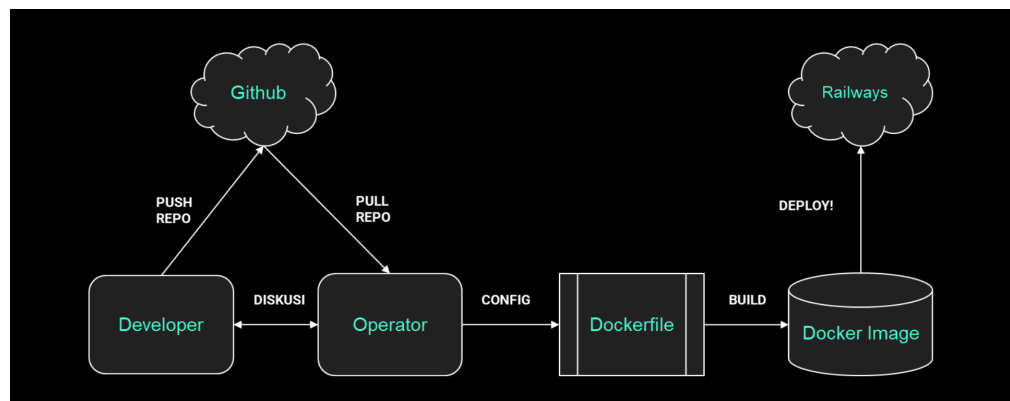
4. Deployment

a. Jenis Deployment yang digunakan

Jenis Deployment yang digunakan adalah menggunakan Railway dan Github. Dimana Railway sendiri adalah Iaas (Infrastructure as a Service). Iaas adalah salah satu jenis layanan cloud computing yang menyediakan infrastruktur IT seperti server, storage, dan jaringan kepada pelanggannya. Pelanggan dapat menggunakan layanan ini untuk menjalankan aplikasi dan menyimpan data tanpa harus membeli dan mengatur infrastruktur fisik sendiri.

Kemudian kita menggunakan Github. GitHub sendiri adalah platform yang menyediakan layanan version control menggunakan sistem Git. Dalam pengembangan website, version control dapat digunakan untuk mengelola perubahan-perubahan yang terjadi pada file-file yang terkait dengan website tersebut, seperti file HTML, CSS, dan JavaScript. Silahkan pelajari command git untuk melatih dalam penggunaan github.

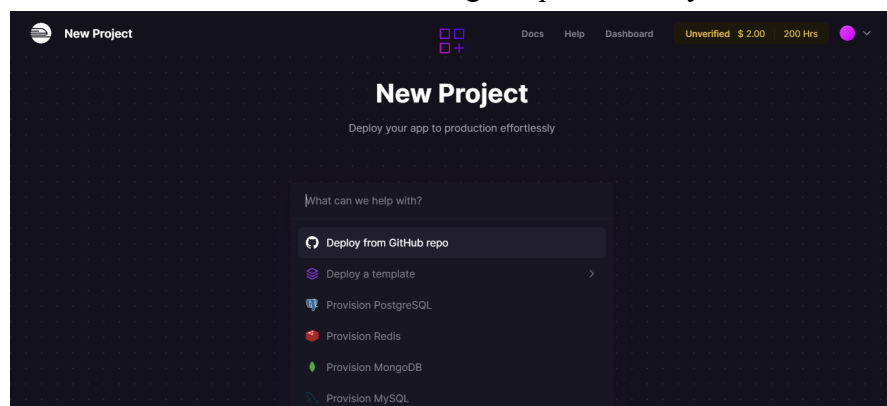
Berikut adalah Working Flow Deployment dari website yang akan kita deploy:



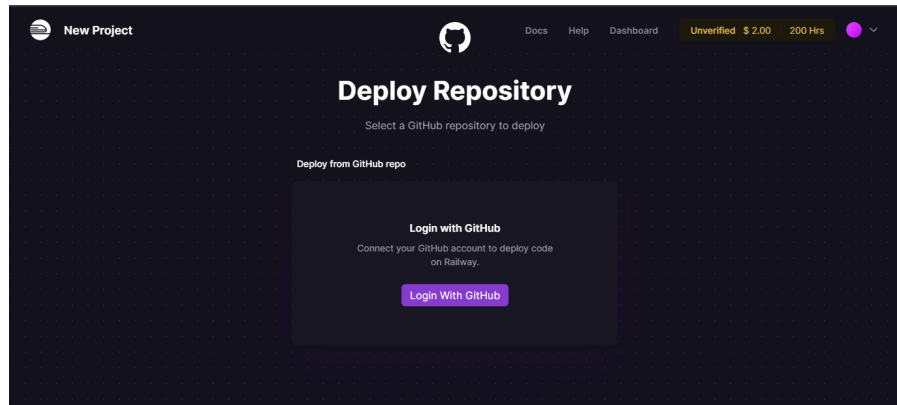
b. Prosedur Deployment

Buat sebuah repository baru pada website github, kemudian upload folder website yang ingin di deploy, disarankan untuk menguploadnya menggunakan command git pada terminal.

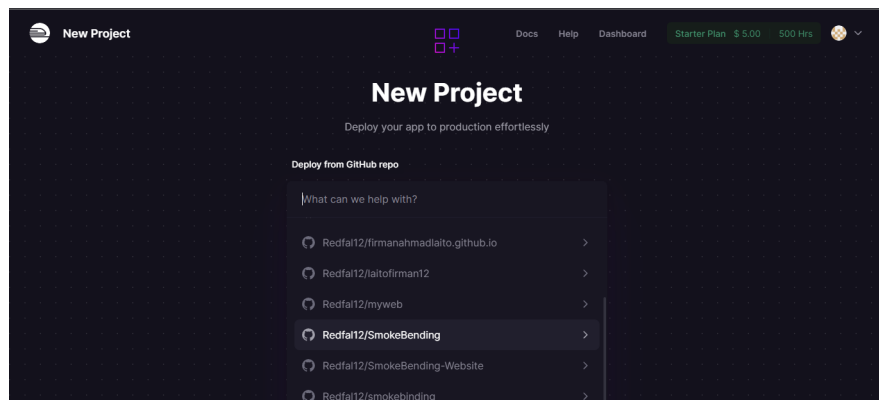
1. Pastikan sudah membuat akun dan sign in pada railway



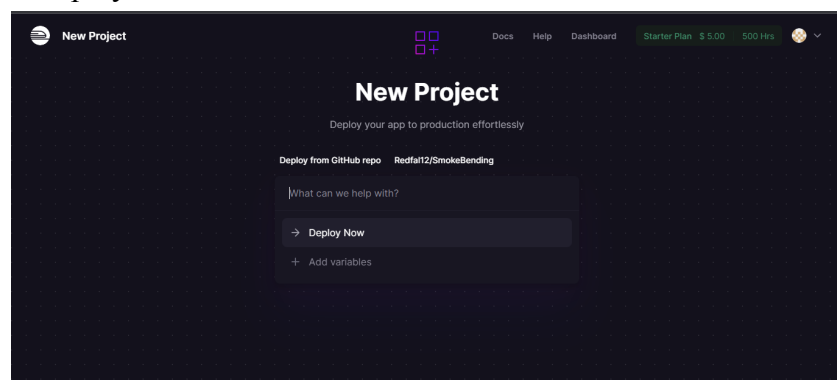
2. Setelah itu klik New Project
3. Klik Deploy from GitHub repo



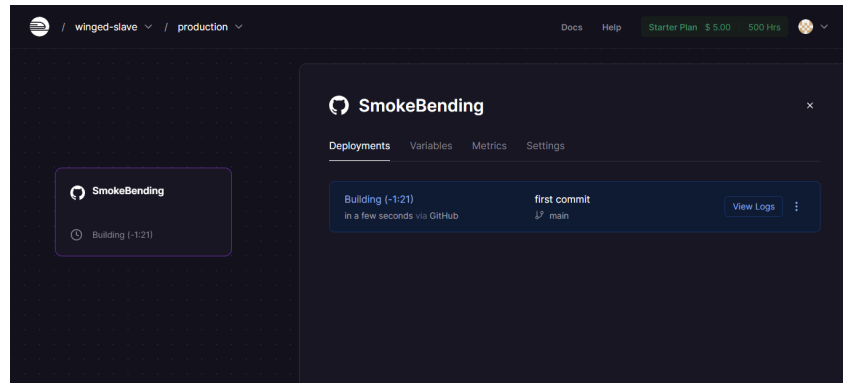
4. Setelah itu kita diharuskan untuk mengkoneksikan github kita dengan railway
5. Setelah Connect dengan Github kita milih repository mana yang akan kita Deploy, Disini kita ingin mendeploy Smoke Bending



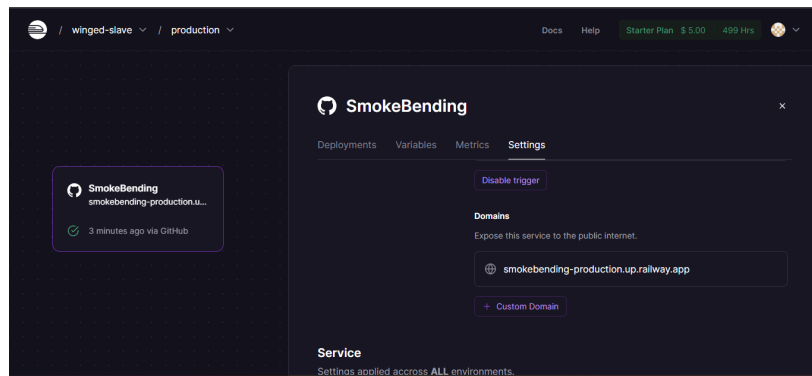
6. Klik Deploy Now



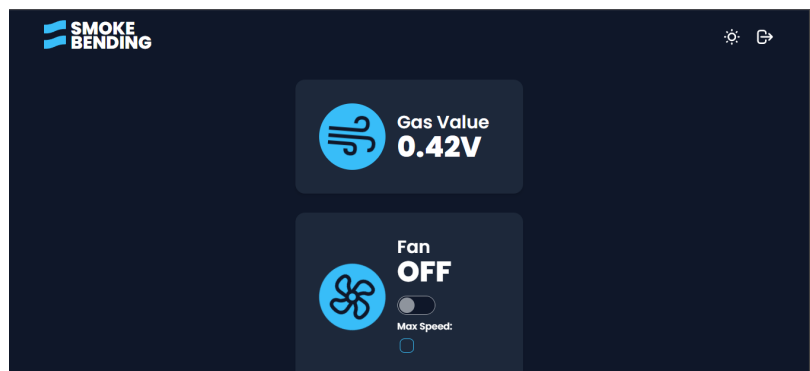
7. Untuk proses deploy memakan waktu 2-3 Menit. Setelah itu akan muncul Success pada catatannya.



8. Setelah itu kita menamai website kita, klik settings, selanjutnya kita klik generate domain kemudian klik domain kita jadi deh website kita.



9. Tampilan Website ketika berhasil di deploy



c. Penjelasan Deployment

Untuk proses deployment, disini kita menggunakan Express JS. Sebagai referensi, berikut backend yang digunakan dalam website Smoke Bending:



```
const express = require("express");
const app = express();
const port = process.env.PORT || 80;

//Middleware
app.use(express.static("public"));

app.get("/", (req, res) => {
  res.sendFile(__dirname + "/public/signinsignup.html");
});

app.get("/dashboard", (req, res) => {
  res.sendFile(__dirname + "/public/dashboard.html");
});

//Listen
app.listen(port, () => {
  console.log(`Aplikasi berjalan`);
});
```

Di sini, kita pertama-tama mengimport modul express dengan *require('express')* dan menyimpannya dalam variabel *express*. Kemudian, kita memanggil fungsi *express()* dan menyimpannya dalam variabel *app*, yang merupakan instance dari aplikasi Express.

Variabel port diinisialisasi dengan nilai dari environment variable PORT, atau nilai 80 jika environment variable tersebut tidak tersedia. Environment variable adalah variabel yang dapat diterapkan di lingkungan runtime aplikasi dan digunakan untuk mengkonfigurasi aplikasi.

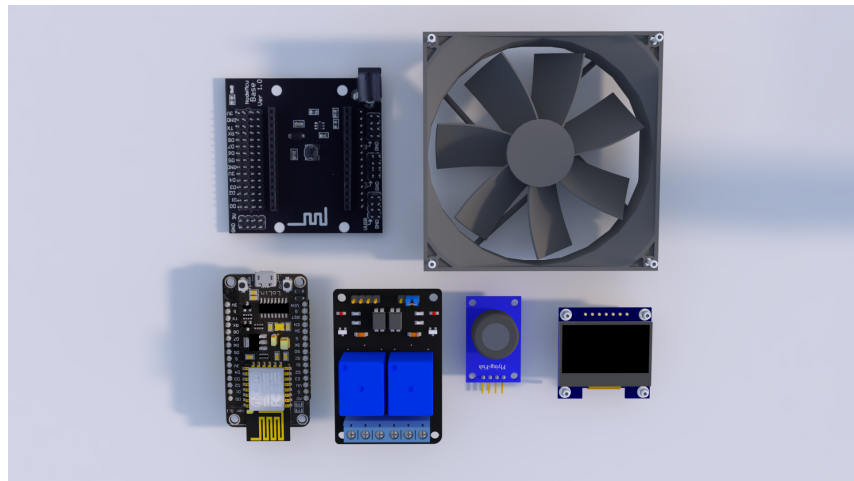
Kemudian, kita menggunakan method *app.use(express.static("public"))* sebagai middleware untuk menangani request ke file static seperti gambar, file CSS, dan file JavaScript dari direktori "public".

Setelah itu, kita menggunakan method *app.get()* untuk menangani request HTTP GET ke alamat '/' dan '/dashboard'. Dalam contoh ini, kita menggunakan method *res.sendFile()* untuk mengirim file HTML sebagai response. File HTML tersebut dipecah menjadi 2 yaitu *signinsignup.html* dan *dashboard.html*.

Terakhir, kita menggunakan method *app.listen()* untuk menjalankan server pada port yang telah ditentukan (port). Fungsi callback akan dipanggil setelah server siap untuk menerima request.

5. Lampiran

a. Foto Komersil



b. Foto Cara kerja

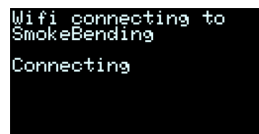


Cara Kerja Alat

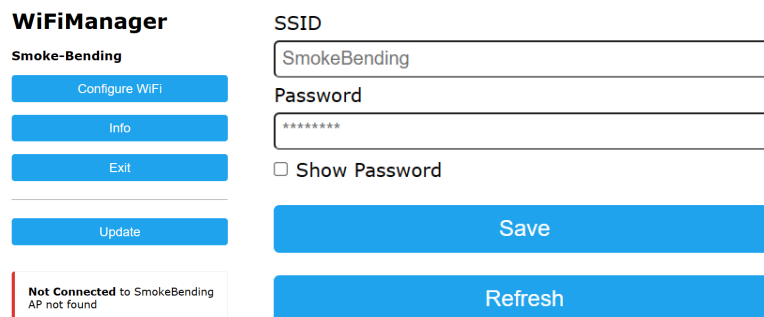
1. Saat pertama kali perangkat dihubungkan dengan daya, layar oled akan menampilkan logo dari smoke bending yang dimana pada bagian source code kita sudah mendeklarasikan kode bitmap untuk logo yang nantinya akan ditampilkan pada layar oled.



2. Kemudian setelah jeda beberapa saat, oled akan berganti ke bagian koneksi WiFi, pada bagian ini terdapat dua kondisi dimana jika jaringan yang sudah kita tentukan ditemukan, maka ESP8266 akan melakukan penyambungan.



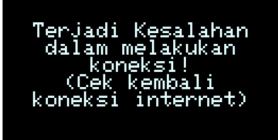
Namun apabila jaringan tidak dapat diakses, maka ESP8266 akan membuat sebuah Access Point baru yang bernama “Smoke-Bending”, yang dimana jika kita sambungkan ke wifi tersebut, kita akan diarahkan ke sebuah website gateway untuk memilih jaringan yang tersedia.



3. Setelah berhasil terhubung dengan wifi, maka tampilan akan masuk ke dashboard, dimana pada tampilan ini terdapat nilai sensor MQ 7 dan kondisi dari kipas. Pada alat smoke bending ini, value atau rentang nilai yang ditangkap oleh sensor gas adalah 0V - 5V, dimana jika sensor gas bernilai $0 < 1$ maka kipas akan “OFF”, jika sensor gas bernilai $1 > 3$ maka kipas akan “ON” dengan kecepatan pelan dan jika sensor gas bernilai $3 > 5$ maka kipas akan “ON” dengan kecepatan maksimal.



4. Jika pada saat alat yang dijalankan mengalami kendala dalam mengirim data ke database, maka akan muncul tampilan bahwa koneksi tidak tersedia. Jika kita ingin mengganti jaringan saat ini ke jaringan lain, maka kita tekan tombol “Reset” yang berada pada belakang alat.



```
Terjadi Kesalahan  
dalam melakukan  
koneksi!  
(Cek kembali  
koneksi internet)
```