

# 11.3 Praktikum Membuat Aplikasi Controlling dan Monitoring dengan API Blynk IoT



Indobot Academy 21 November 2022

## 1. Alat/Instrumen/Apparatus/Bahan

- Wemos D1.
- Sensor DHT 11.
- Flame Sensor.
- Project board.
- Kabel jumper.
- MIT Apps Inventor.

## 2. Keselamatan Kerja

### 2.1. Pemasangan Komponen

Perhatikan bagian pin yang digunakan. Terutama bagian komponen yang memiliki polaritas, jangan sampai terbalik antara kaki positif dan negatif.

### 2.2. Penggunaan Ukuran Resistor

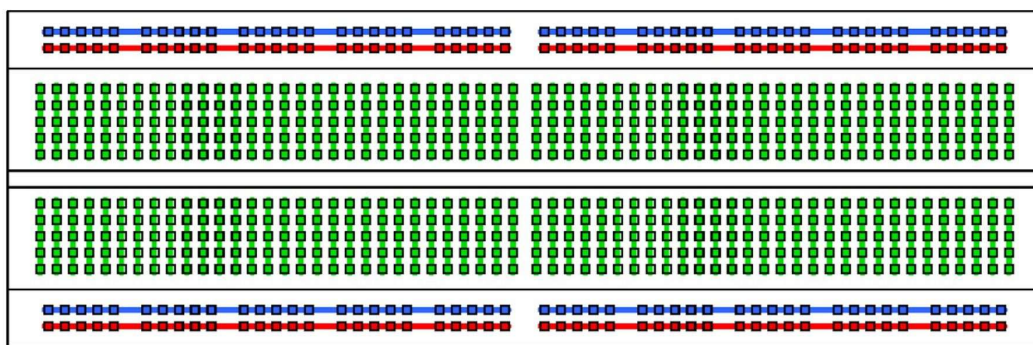
Hal yang perlu diperhatikan lainnya adalah mengenai ukuran resistor. Ukuran resistor bisa mengikuti sesuai gambar rangkaiannya.

### 2.3. Perhatikan pin

Selanjutnya kita juga perlu memperhatikan pin yang ada dalam wemos D1 maupun sensor.

### 2.4. Pemahaman Jalur Project Board

Agar kita mengetahui tentang jalur yang ada pada project board, kita bisa melihat gambar skema dalam project board seri MB-102 berikut ini.



Gambar 1. Jalur Project Board

- Bagian tengah project board akan saling terhubung secara vertikal setiap 5 pin. Kemudian akan ada celah, nah celah ini bisa anda gunakan untuk meletakkan push button atau komponen lainnya.
- Untuk bagian atas dan bawah ini terhubung secara horizontal, dengan celah juga di bagian tengah dari project board.

### 2.5. Peringatan Bahaya Api

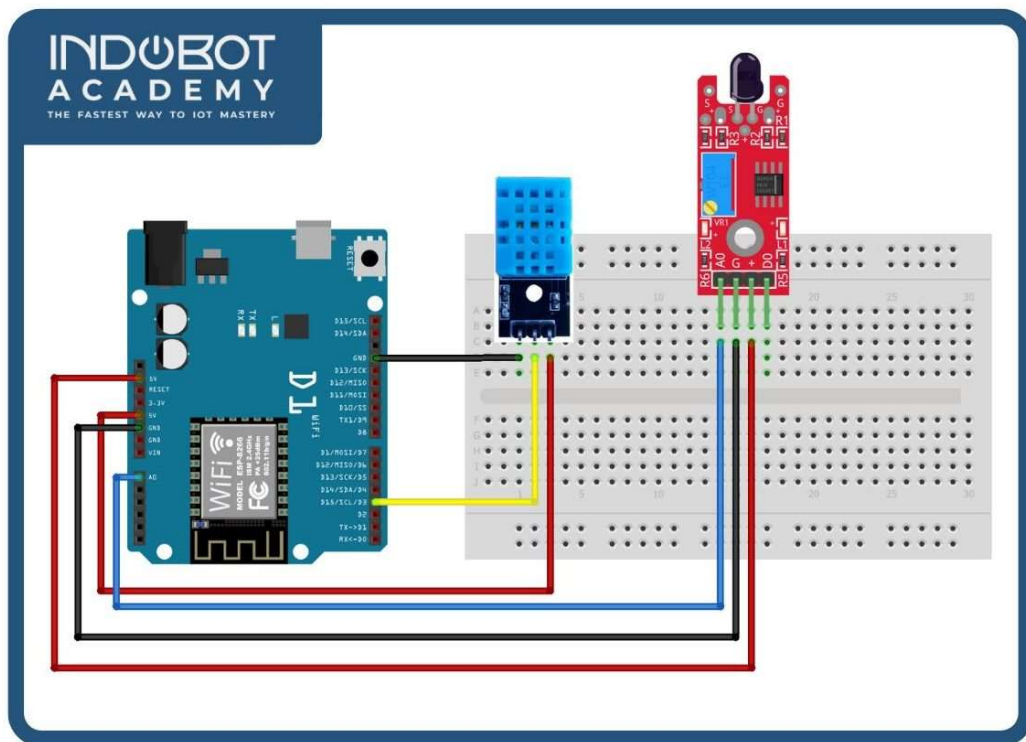
Hati-hati saat praktikum menggunakan sensor flame, karena praktikum ini akan melibatkan Api yang cukup berbahaya bila terkena benda-benda di sekitar kita.

### 3. Langkah Praktikum – Membuat Aplikasi Monitoring Suhu, Kelembapan Udara dan Deteksi Kebakaran

#### 3.1. Penjelasan Praktikum

Pada praktikum kali ini, kita akan mencoba membuat aplikasi untuk keperluan monitoring suhu dan kelembapan udara ketika terjadi kebakaran dengan menggunakan MIT App Inventor dan Blynk API.

#### 3.2. Skema Rangkaian



Gambar 2. Rangkaian Flame Sensor, DHT11, dan Wemos D1 R1

#### Keterangan :

- Hubungkan pin Ground kedua sensor ke pin GND Wemos D1 R1.
- Hubungkan pin VCC kedua sensor ke Pin 5V Wemos D1 R1.
- Hubungkan pin Data sensor DHT11 ke pin D3 Wemos D1 R1.
- Hubungkan pin Analog Flame sensor ke Pin A0 Wemos D1 R1.

### 3.3. Coding

```
// Blynk Template
#define BLYNK_TEMPLATE_ID "TMPLW4ujpnPf"
#define BLYNK_DEVICE_NAME "Blynk API"
#define BLYNK_AUTH_TOKEN
"3tRlv0RUItetWLdFG9ESFUTUXH6jxRi0"

// Pemanggilan Library
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>

// Inisialisasi pin, tipe, dan sebagainya
#define DHTPIN D3
#define DHTTYPE DHT11 //Tipe DHT11
DHT dht(DHTPIN, DHTTYPE);
BlynkTimer timer;

// Koneksi
char auth[] = BLYNK_AUTH_TOKEN ;
char ssid[] = "Tasya";
char pass[] = "Keling 07";

// Pembacaan batas minimum dan maksimum sensor
const int sensorMin = 0;
const int sensorMax = 1024;

// Method yang berfungsi untuk mengirim data sensor
void sendSensor()
{
    float t = dht.readTemperature();
    float h = dht.readHumidity();
    int sensorReading = analogRead(A0);
    int range = map(sensorReading, sensorMin, sensorMax,
0, 3);
    Serial.print("Suhu: ");
    Serial.print(t);
    Serial.print("°C --- ");
```

```

Serial.print("Kelembapan: ");
Serial.print(h);
Serial.println("%");

switch (range) {
    case 0:    // Api lebih dekat dari 1.5 kaki
jauhnya.
        Blynk.virtualWrite(V4,"3");
        break;
    case 1:    // Api antara 1-3 kaki jauhnya
        Blynk.virtualWrite(V4,"2");
        break;
    case 2:    // Api tidak terdeteksi
        Blynk.virtualWrite(V4,"1");
        break;
} delay(1);
// Mengirimkan data suhu udara ke Virtual pin V2 di
Blynk Cloud
Blynk.virtualWrite(V2, t);
// Mengirimkan data kelembapan udara ke Virtual pin
V3 di Blynk Cloud
Blynk.virtualWrite(V3, h);
}

// Method yang dikerjakan sekali saja setiap
menyalakan device
void setup() {
    // Serial monitor menggunakan baudrate 115200
    Serial.begin(115200);
    // Memulai Blynk
    Blynk.begin(auth, ssid, pass);
    // Mengaktifkan DHT11
    dht.begin();
    // Mengaktifkan timer untuk pengiriman data 1000ms
    timer.setInterval(1000L, sendSensor);
}

// Method yang dikerjakan secara berulang kali
void loop() {
    // Menjalankan blynk

```

```

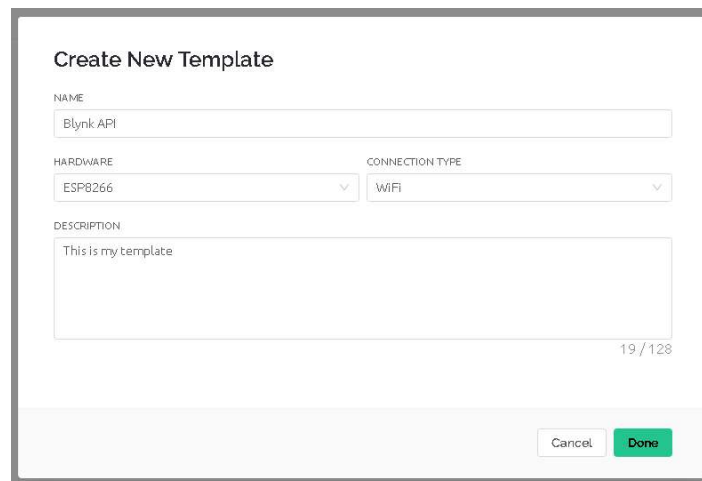
Blynk.run();
// Menjalankan timer
timer.run();
}

```

### 3.4. Konfigurasi MIT App Inventor

#### 3.4.1. Konfigurasi Blynk

- Buka Blynk Website, lalu buat template baru.



**Create New Template**

NAME  
Blynk API

HARDWARE  
ESP8266

CONNECTION TYPE  
WiFi

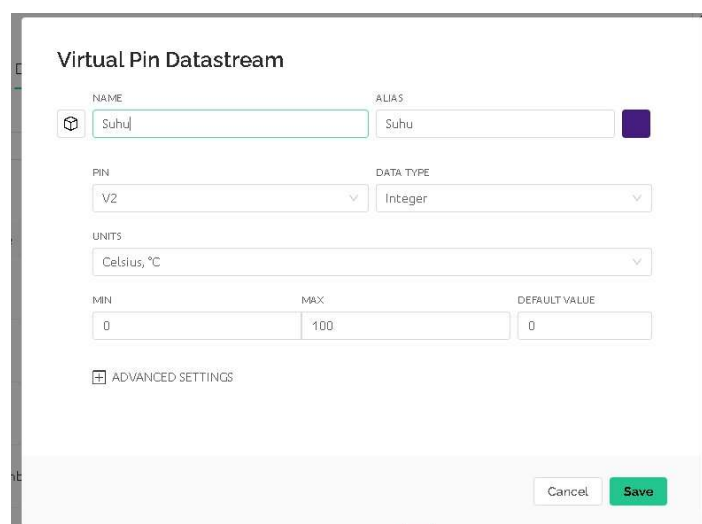
DESCRIPTION  
This is my template

19 / 128

Cancel Done

Gambar 3. Membuat template baru

- Lakukan Konfigurasi datastream untuk suhu kelembapan udara dan deteksi api seperti gambar dan keterangan di bawah ini.



**Virtual Pin Datastream**

NAME  
Suhu

ALIAS  
Suhu

PIN  
V2

DATA TYPE  
Integer

UNITS  
Celsius, °C

MIN  
0

MAX  
100



DEFAULT VALUE  
0

ADVANCED SETTINGS

Cancel Save



Gambar 4. Konfigurasi virtual pin data suhu

### Virtual Pin Datastream

NAME		ALIAS		
	<input type="text" value="Kelembaban"/>	<input type="text" value="Kelembaban"/>		
PIN		DATA TYPE		
<input type="text" value="V3"/>		<input type="text" value="Integer"/>		
UNITS				
<input type="text" value="Percentage, %"/>				
MIN		MAX		DEFAULT VALUE
<input type="text" value="0"/>		<input type="text" value="100"/>		<input type="text" value="0"/>
<input type="checkbox"/> ADVANCED SETTINGS				
				<input type="button" value="Cancel"/> <input type="button" value="Save"/>

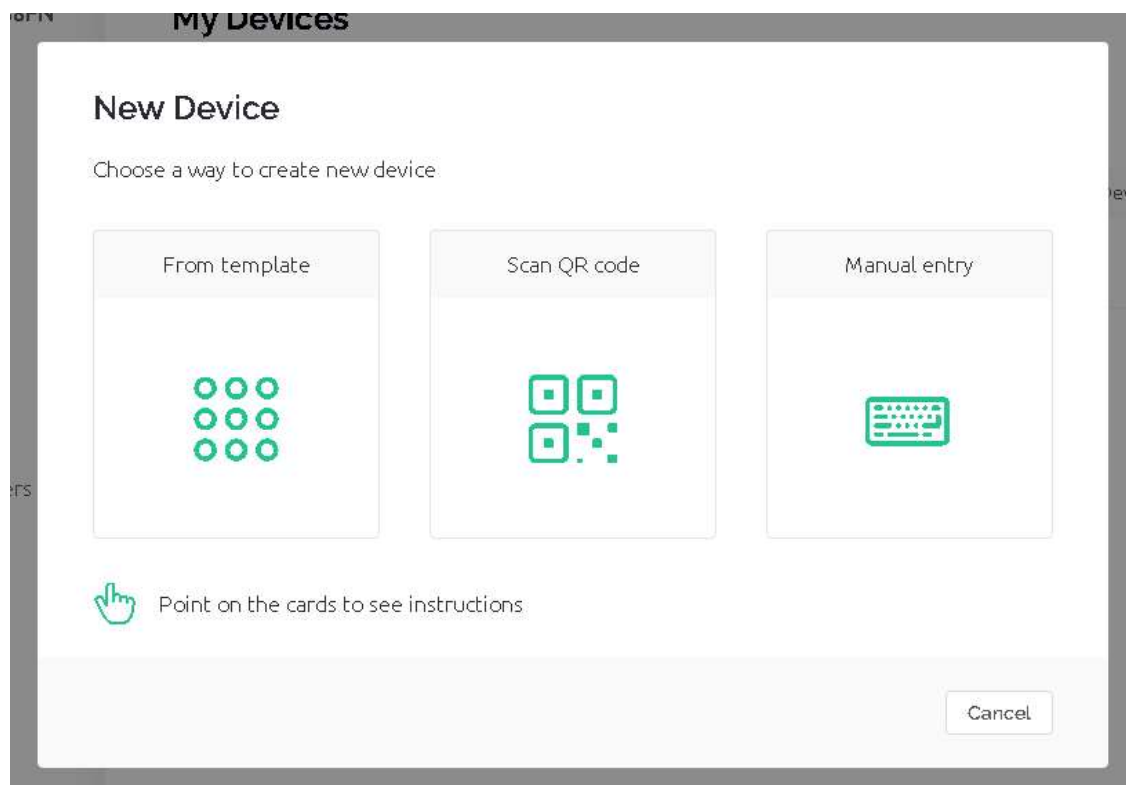
Gambar 5. Konfigurasi virtual pin data kelembapan

### Virtual Pin Datastream

NAME		ALIAS		
	<input type="text" value="Data Api"/>	<input type="text" value="Data Api"/>		
PIN		DATA TYPE		
<input type="text" value="V4"/>		<input type="text" value="String"/>		
DEFAULT VALUE				
<input type="text" value="Default Value"/>				
<input type="checkbox"/> ADVANCED SETTINGS				
				<input type="button" value="Cancel"/> <input type="button" value="Save"/>

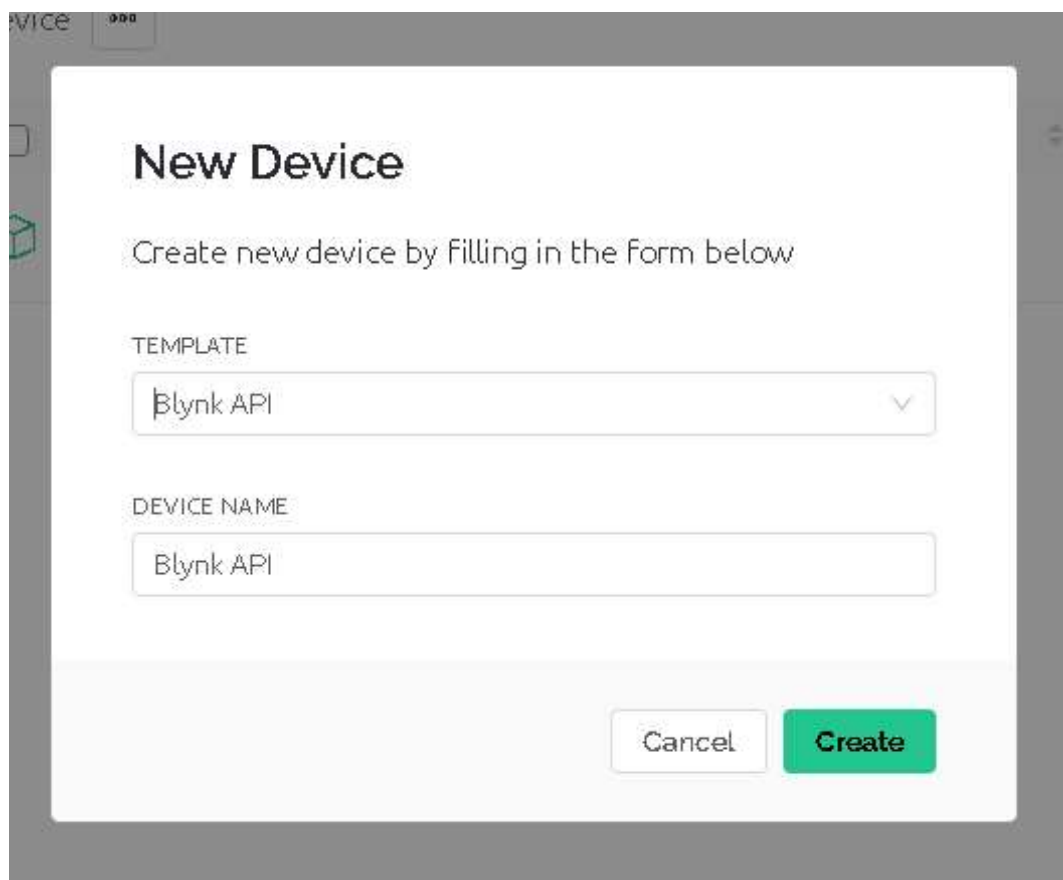
Gambar 6. Konfigurasi virtual pin data deteksi api

- Selanjutnya membuat device -> Klik Icon Menu Device -> New Device -> Pilih From Template.



Gambar 7. Membuat device baru

- Pilih Template yang baru kita buat.



Gambar 8. Memilih template

- Pada device tersebut, copylah tiga baris kode firmware configuration dan pastekan pada kode program yang digunakan.
- Device siap digunakan.



- Selanjutnya kita akan membuat tampilan aplikasi.

### 3.4.2. Desain Tampilan Aplikasi

- Buka MIT App Inventor -> New Project -> Beri nama.
- Pada layar smartphone tambahkan palette, lakukan konfigurasi seperti gambar dan keterangan di bawah ini.



Gambar 9. Menambahkan Palette

- Selanjutnya kita akan konfigurasi kode blok untuk aplikasinya.

### 3.4.3. Konfigurasi Kode Blok

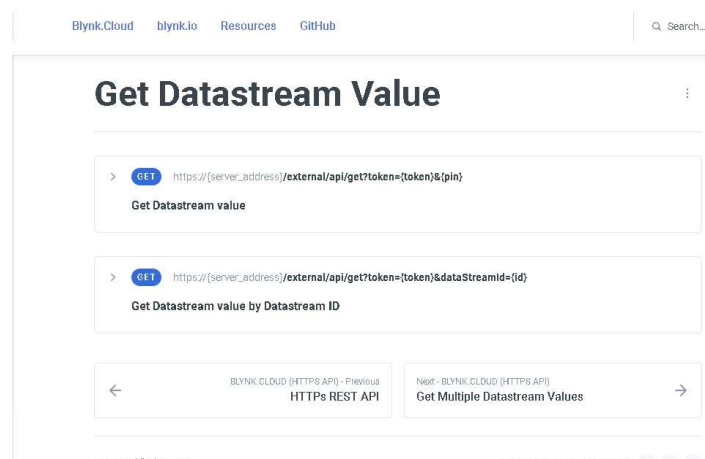
- Pada kode Block, tambahkan block variabel Initialize Global dan beri nama seperti gambar di bawah.



- lalu tambahkan block input string pada block variabel initialize global.



- Pada block input string dengan initialize global GetV2, GetV3, dan GetV4. Selanjutnya, masukkanlah baris kode API Get Datastream dari website dokumentasi Blynk ke aplikasi, seperti yang dapat anda lihat pada gambar berikut.



Gambar 10. API Get Datastream Value Blynk

Kode baris API yang digunakan adalah Get Datastream value seperti baris kode di bawah ini.

```
https://{server_address}/external/api/get?token=
{token}&{pin}
```

Konfigurasikan dengan mengganti isi baris kode seperti keterangan di bawah :

- {server\_addres} diganti menjadi region.blynk.cloud (region terdapat pada ujung kanan bawah blynk website).
- {token} diganti dengan token device yang digunakan.
- {pin} diganti dengan Virtual pin yang digunakan pada Datastream.

Jika menggunakan token dan virtual pin pada contoh program praktikum ini, maka baris kode ditulis seperti ini :

- untuk initialize global GetV2 :

```
https://sgp1.blynk.cloud/external/api/get?
token=3tRlv0RUItetWLdFG9ESFUTUXH6jxRi0&v2
```

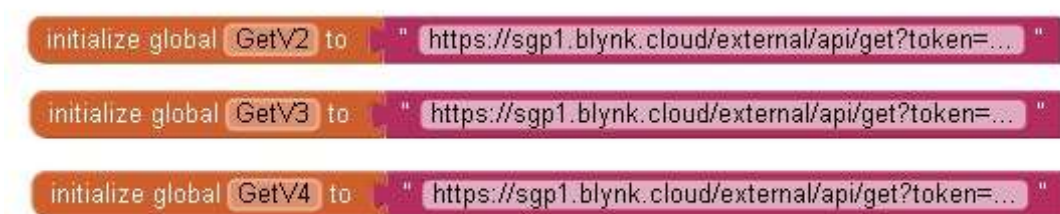
- untuk initialize global GetV3 :

```
https://sgp1.blynk.cloud/external/api/get?
token=3tRlv0RUItetWLdFG9ESFUTUXH6jxRi0&v3
```

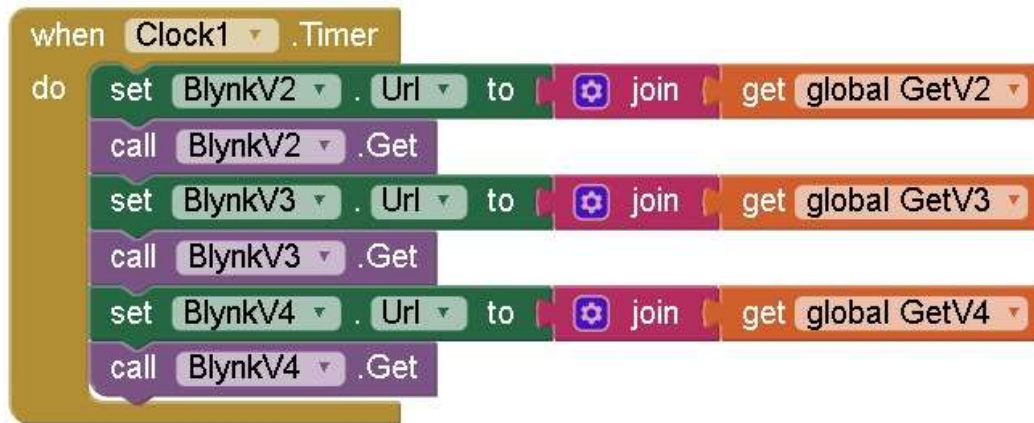
- untuk initialize global GetV4 :

```
https://sgp1.blynk.cloud/external/api/get?
token=3tRlv0RUItetWLdFG9ESFUTUXH6jxRi0&v4
```

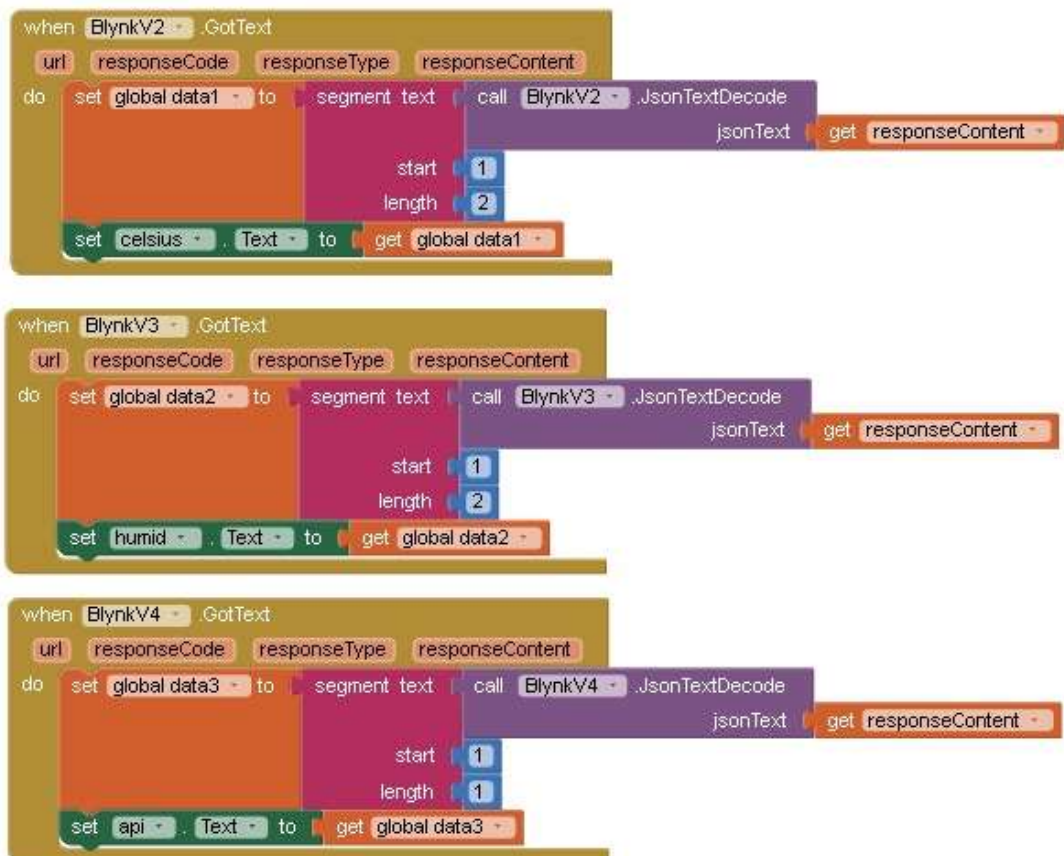
Tambahkan baris kode API pada input string initialize global GetV2, GetV3, dan GetV4.



- Selanjutnya, buatlah kode block untuk menghitung waktu dan jam. Buatlah seperti gambar di bawah ini.



- Setelah itu kita buat kode blok untuk menerima dan menampilkan data sensor pada Aplikasi, buatlah seperti gambar di bawah.



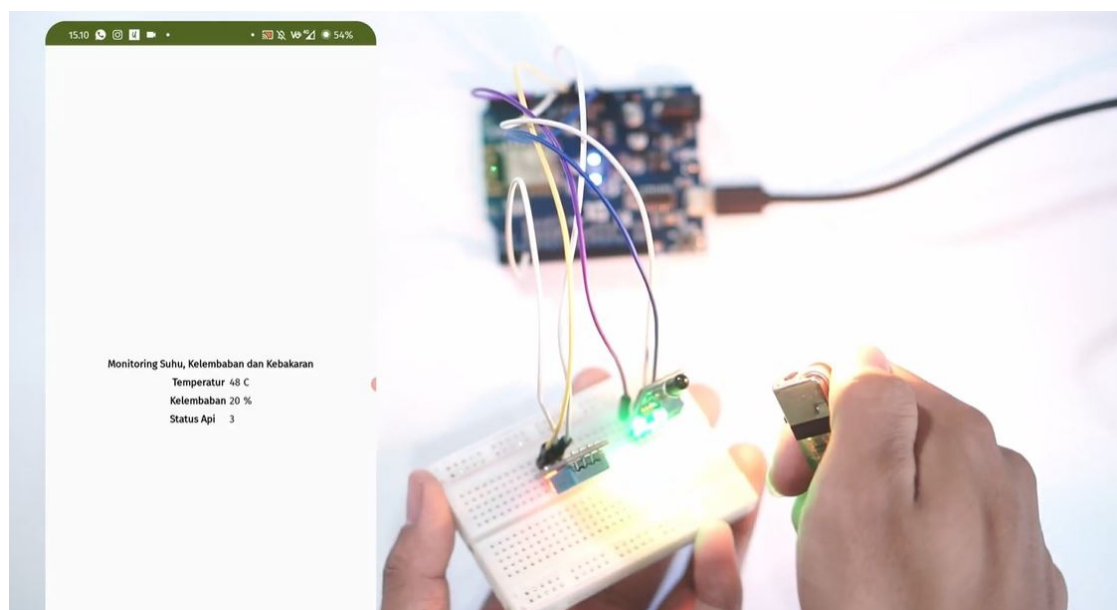
- Kode blok telah selesai kita buat, selanjutnya kita akan mencoba aplikasi yang kita buat.

#### 3.4.4. Menguji Aplikasi

- Untuk menguji aplikasi ini berhasil atau tidaknya, maka cobalah untuk mendownload aplikasi yang sudah kita buat tadi.
- Untuk mengetahui bagaimana cara mendownload aplikasi, silahkan membaca materi sebelumnya.

#### 3.5. Hasil Program

Bukalah MIT App inventor silahkan klik connect -> Ai Companion -> scan barcode atau masukkan kode ke aplikasi MIT App Inventor di Smartphone. Maka akan muncul tampilan layar yang sudah dirancang sebelumnya. Perhatikan angka suhu dan kelembapan udara, nilai pengukuran akan berubah secara realtime. Nilai pengukuran sensor Api pada aplikasi ini dinyatakan sebagai Status Api. Adapun ketetapan status api terbagi menjadi 3 opsi, yaitu jika status Api = 1 maka sensor tidak mendeteksi adanya Api. Jika status api = 2 maka sensor telah mendeteksi adanya Api dalam radius yang dekat. Jika status api = 3 maka sensor telah mendeteksi adanya Api dalam radius yang sangat dekat.



## **4. Tugas dan Tantangan**

Tambahkan palette lainnya untuk menghias tampilan aplikasi monitoring diatas, misalnya dengan menambahkan gambar, warna atau mengubah letak isi. Buatlah se-kreatif mungkin. Dan tambahkan loading screen yang sebelumnya telah anda buat di materi sebelumnya ke aplikasi anda saat ini.