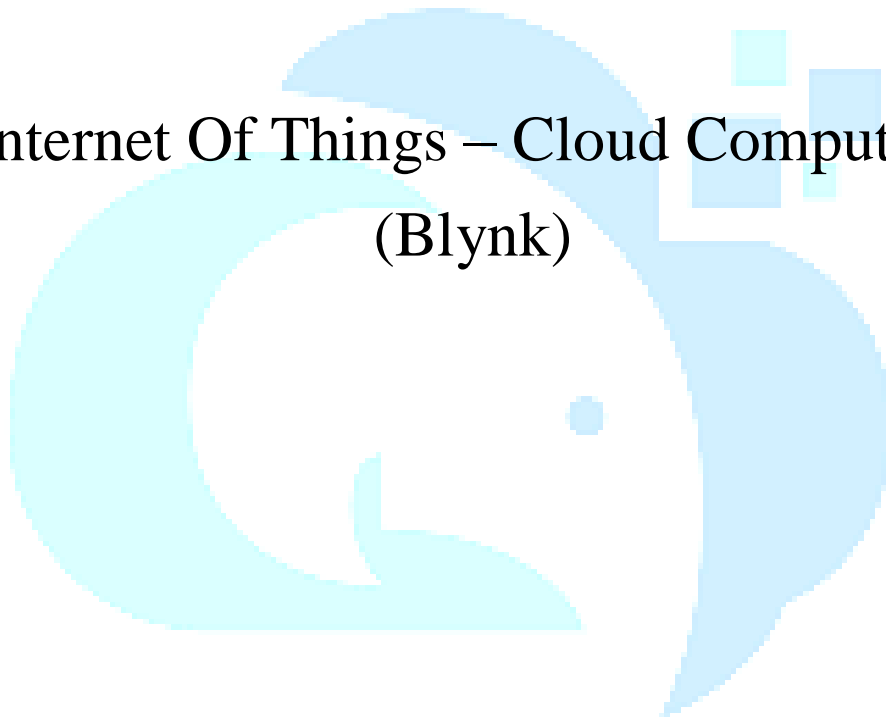


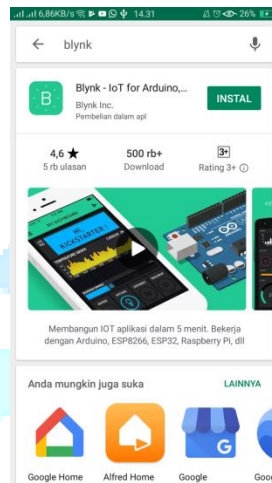
Internet Of Things – Cloud Computing (Blynk)



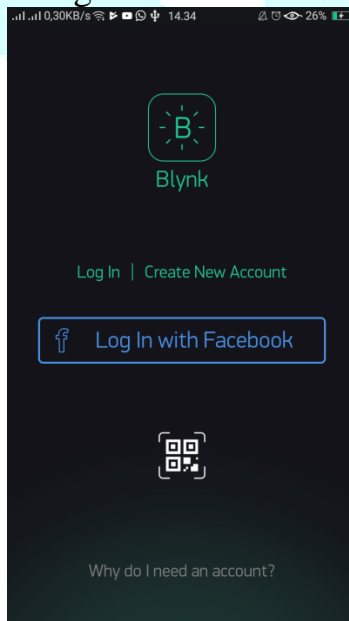
Kontrol Lampu dengan Blynk

Pada project Kontrol Lampu dengan Blynk kali ini, kita harus mendownload terlebih dahulu aplikasi blynk di playstore .Yuk... kita mulai saja.

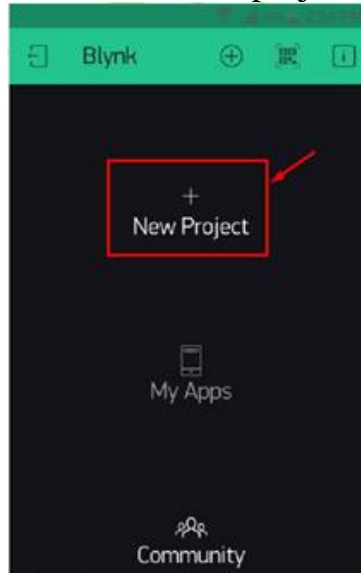
- Pertama kita download terlebih dahulu aplikasinya di playstore.



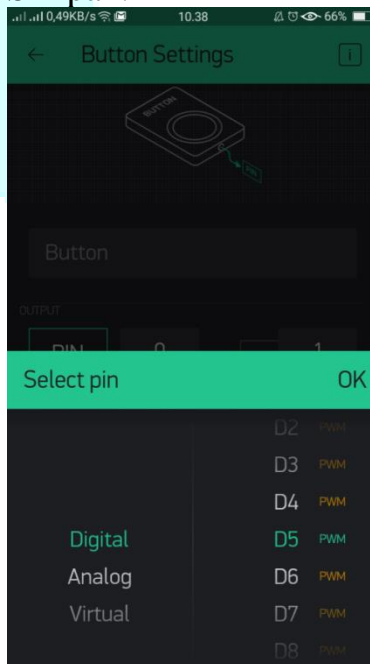
- Langkah kedua adalah kita membuat Aplikasi terlebih dahulu di Blynknya.
 - Buka apk Blynk
 - Lalu anda akan di minta untuk mendaftar dengan email anda sebagai verifikasi atau nantinya sebagai pengirim kode token auth.



- Kemudian New project



- lalu akan muncul Create New project dan anda bisa membuat nama sesuka hati anda. Lalu pilih choose device (NodeMCU) dan connection typenya (WiFi).
- Setelah itu akan ada email sebagai auth token nya dan pilih tanda + di kanan atas pilih widget button dan atur Pin Di D5 kemudian Simpan.



- Langkah ketiga kita buat program, atau anda bisa membuka saja program yang sudah kita sediakan dan hal yang paling penting jangan lupa mengganti SSID, Passwordnya dan auth token yang sudah di kirim melalui email.

```
/******  
* Program : Kontrol Lampu via Blynk  
* Input :  
* Output : 1 buah Relay  
* Iot Starter Kit Basic  
* *****/  
#define BLYNK_PRINT Serial  
#include <ESP8266WiFi.h>  
#include <BlynkSimpleEsp8266.h>  
  
// Ganti dengan token anda yang dikirim via email.  
char auth[] = "ee2ba9a5b0aa438cb697c4f9846e4f5f";  
  
// Ganti dengan WiFi dan Password anda  
char ssid[] = "SEKOLAHAN.ID";  
char pass[] = "foritastasolusindo";  
  
void setup()  
{  
  Serial.begin(9600);  
  Blynk.begin(auth, ssid, pass);  
}  
  
void loop()  
{  
  Blynk.run();  
}
```

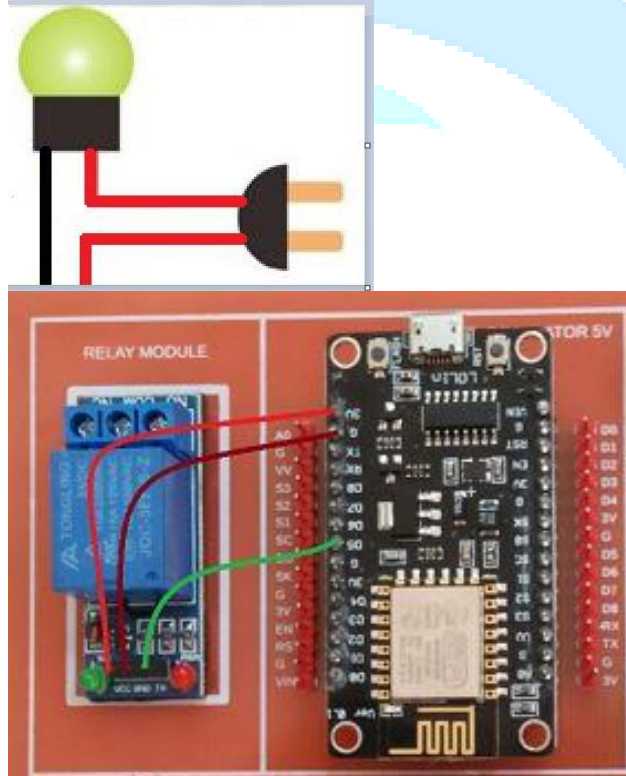
- Langkah ke empat atau yang terakhir adalah simpan (save) kemudian Upload dan tunggu sebentar. Kemudian coba anda klik dan apa yang terjadi, apabila WiFi tidak connect coba lihat program dan reset NodeMCU

1. Project 4 : Kontrol Lampu via Internet

Project ke 6 ini yaitu Kontrol Lampu via Internet merupakan sistem IoT pengontrolan berbasis web karena kita akan mengaksesnya melalui web hampir sama dengan project 1 dan 2 hanya saja perbedaannya dari penggunaan relay dan pemrogramannya.

Dimana pemrograman di NodeMCUnya pembacaan relay-nya yaitu aktif LOW jadi relay akan berfungsi ketika relay diberi perintah “LOW” namun sebaliknya jika relay di berikan perintahnya “HIGH” maka relay tidak akan berfungsi. Mari kita langsung aja ke TKP

- Pertama kita wiring terlebih dahulu antara modul relay dan NodeMCU seperti di bawah ini :



Hubungkan kabel hitam ke com relay dan kabel merah ke **NO** dari relay.

- Kedua kita buat programnya dan ada beberapa yang harus di perhatikan yaitu penggunaan perintah di void setup. Dan void loop dimana relay harus diberikan perintah LOW untuk aktif dan HIGH untuk tidak aktif. Pada saat di void setup relay harus di beri perintah (`digitalWrite(relay, HIGH)`) agar pada saat pembacaan pertama relay akan tetap tidak berfungsi sehingga lampu padam, dan di beri program

```
( if (request.indexOf("/relay1on") > 0) {  
    digitalWrite(Relay1, LOW);  
}  
if (request.indexOf("/relay1off") > 0) {  
    digitalWrite(Relay1, HIGH);  
})
```

pada void loop agar relay dapat berfungsi pada saat di beri LOW dan tidak berfungsi pada saat di beri HIGH seperti pada program di bawah ini

```
/*  
*****  
* Program : Kontrol Lampu via Internet  
* Input : -  
* Output : 1 buah Relay  
* *****/  
#include <ESP8266WiFi.h>  
const char* ssid = "Sekolahan.id";  
const char* password = "foritastasolusindo";  
#define Relay1 D5  
; //  
WiFiServer server(80);  
  
void setup() {  
    Serial.begin(115200);  
    delay(10);  
    pinMode(Relay1, OUTPUT);  
    digitalWrite(Relay1, HIGH);  
    Serial.println();  
    Serial.println();  
    Serial.print("Connecting to ");  
    Serial.println(ssid);  
  
    WiFi.begin(ssid, password);  
    while (WiFi.status() != WL_CONNECTED) {  
        delay(500);  
        Serial.print(".");  
    }  
    Serial.println("");  
    Serial.println("WiFi connected");  
}
```

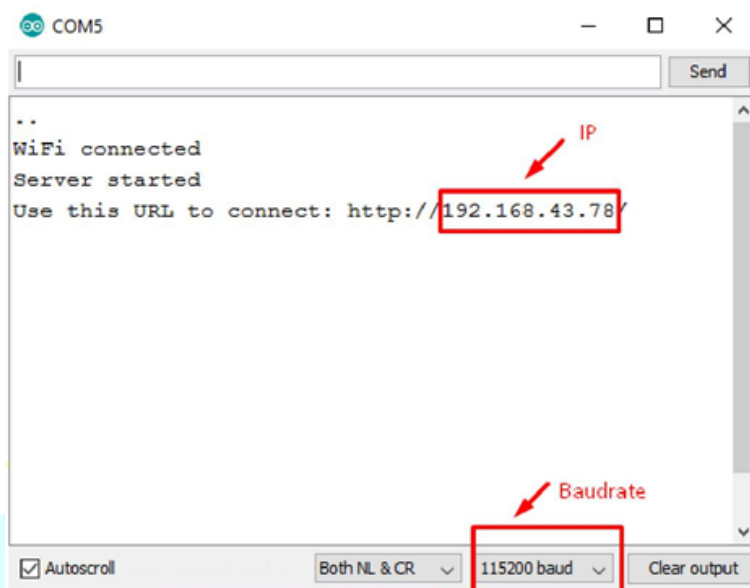
```
// Start the server
server.begin();
Serial.println("Server started");
Serial.print("Use this URL to connect: ");
Serial.print("http://");
Serial.print(WiFi.localIP());
Serial.println("/");
}

void loop() {
  WiFiClient client = server.available();
  if (!client) {
    return;
  }
  Serial.println("new client");
  while(!client.available()){
    delay(1);
  }
  String request = client.readStringUntil('\r');
  Serial.println(request);
  client.flush();
  if (request.indexOf("/relay1on") > 0) {
    digitalWrite(Relay1, LOW);
  }
  if (request.indexOf("/relay1off") > 0) {
    digitalWrite(Relay1, HIGH);
  }
  client.println("HTTP/1.1 200 OK");
  client.println("Content-Type: text/html");
  client.println(""); // do not forget this one
  client.println("<!DOCTYPE HTML>");
  client.println("<html>");
  client.println("<head>");
  client.println("<meta name='apple-mobile-web-app-capable'");
  client.println("content='yes' />");
  client.println("<meta name='apple-mobile-web-app-status-bar-style'");
  client.println("content='black-translucent' />");
  client.println("</head>");
  client.println("<body bgcolor = \"#f7e6ec\">");
```

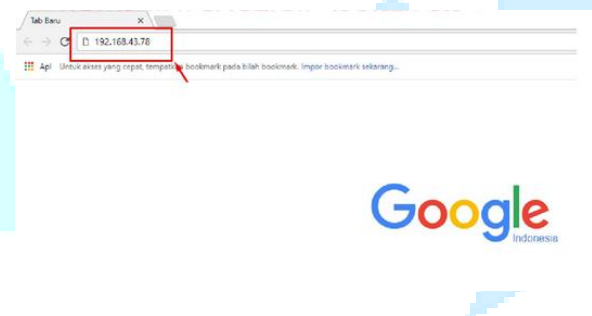
```
client.println("<hr/><hr>");
client.println("<h1><center> IoT Starter Kit Basic : Kontrol 1
Lampu </center></h1>");
client.println("<h1><center> SEKOLAHAN.ID </center></h1>");
client.println("<hr/><hr>");
client.println("<br><br>");
client.println("<br><br>");
client.println("<center>");
client.println("Lampu 1");
client.println("<a href='/relay1on/'><button>Turn On
</button></a>");
client.println("<a href='/relay1off/'><button>Turn Off
</button></a><br />");
client.println("</center>");
client.println("<br><br>");
client.println("<center>");
client.println("<table border='5'>");
client.println("<tr>");
//=====
if (digitalRead(Relay1))
{
    client.print("<td>Lampu 1 = ON</td>");
}
else
{
    client.print("<td>Lampu 1 = OFF</td>");
}
client.println("<br />");
client.println("</tr>");
client.println("</table>");
client.println("</center>");
client.println("</html>");
delay(1);
Serial.println("Client disconnected");
Serial.println("");
}
```

- Lalu penggunaan dari perintah `client.println` yaitu agar bisa berkomunikasi dengan web dan dapat menampilkan kata atau bentuk di web

- Setelah itu anda bisa upload program dan setelah selesai bisa buka Serial Monitor yang ada di pojok kanan atas aplikasi Arduino IDE, dan nanti akan muncul sebuah alamat IP

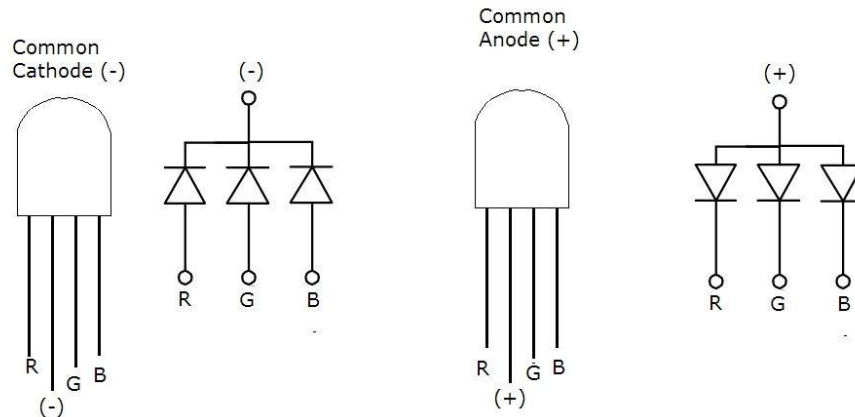


Perhatikan baudratanya juga. Setelah mendapat alamat IP maka anda bisa serch alamat IPnya di web anda masing – masing. Dan akan tampil seperti di bawah ini :



Kontrol LED RGB dengan Aplikasi Android

Led RGB merupakan LED yang dapat menampilkan 3 warna dalam 1 LED : Merah (R), Hijau (G) dan Biru (B). Kombinasi dari 3 warna tersebut akan menghasilkan warna LED yang berwarna – warni.

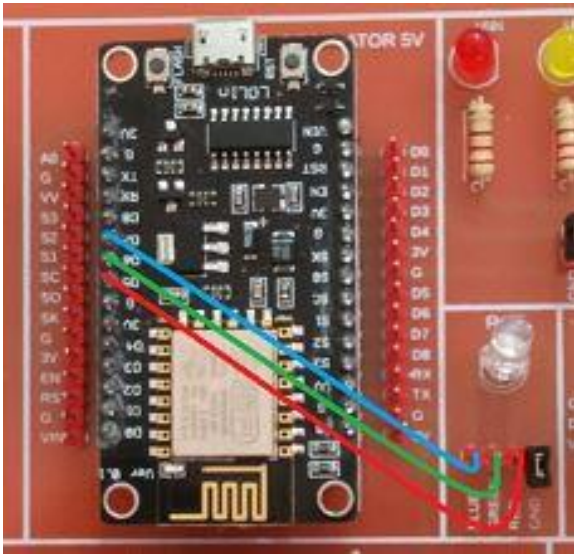


Nah nilai tegangan (PWM nantinya) di setiap warna tersebut dapat kita kontrol sehingga menampilkan perpaduan warna diantara ke-3 warna dasar tersebut.

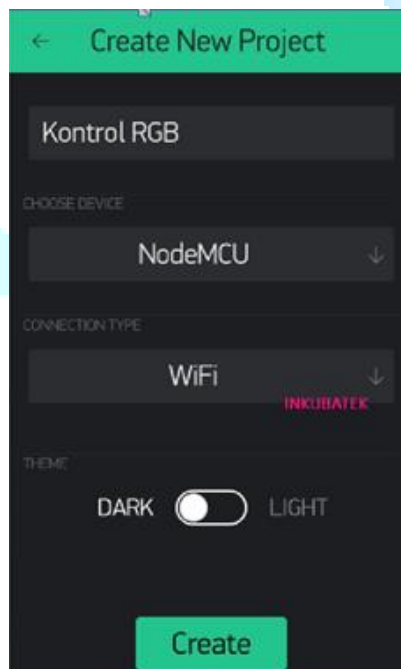
Sebelum ke membuat aplikasinya kita sambungak terlebih dahulu antara modul RGB ke NodeMCU yaitu :

- R masuk ke D5 NodeMCU
- G masuk ke D6 NodeMCU
- B masuk ke D7 NodeMCU
- (-) masuk ke GND NodeMCU

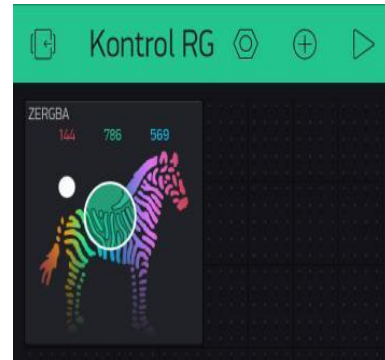
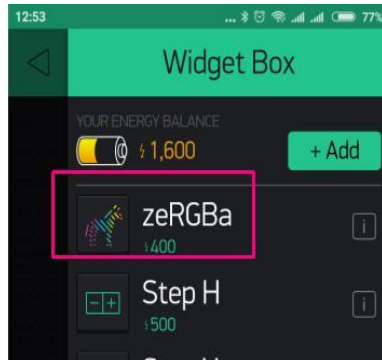
Seperti pada wiring dibawah ini :



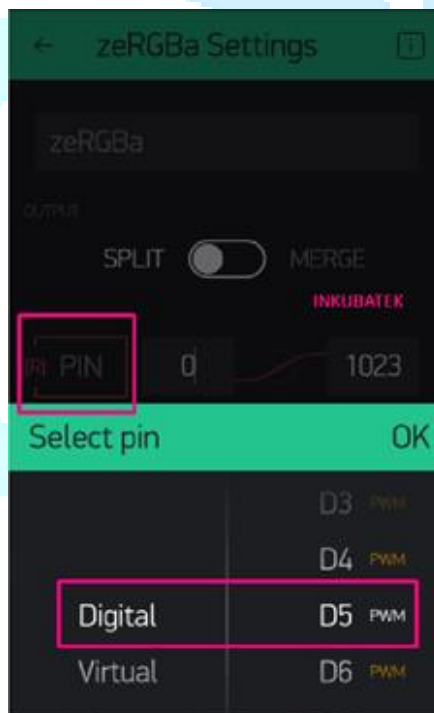
Kita masih memakai Blynk. Bikin proyek baru, *New Project* kemudian beri nama **Kontrol RGB**, DEVICE-nya NodeMCU dan koneksinya WiFi. Caranya masih sma seperti yang sebelumnya.



Klik “*Create*” sehingga kode Token dikirim ke email anda. Tambahkan sebuah widget zeRGBa dari widget box.



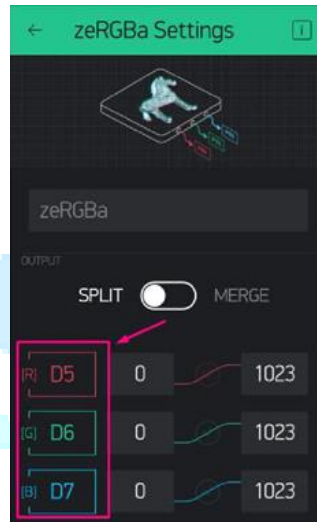
Seting widget zeRGBa. Klik di widget zeRGBa.



Klik “**PIN [R]**” kemudian pada “*Select pin*” pilih “**Digital**” dan “**D5**” karena pin Red (R) nantinya kita hubungkan dengan D5.

Klik “**PIN [G]**” kemudian pada “*Select pin*” pilih “**Digital**” dan “**D6**” karena pin Red (G) nantinya kita hubungkan dengan D6

Klik “**PIN [B]**” kemudian pada “*Select pin*” pilih “**Digital**” dan “**D7**” karena pin Red (B) nantinya kita hubungkan dengan D7



Kalau sudah sekarang kita siapkan program dengan Arduino IDE. Ganti token dengan kepunyaan anda, juga SSID dan password WiFi-nya.

```
/*
*****
* Program : Kontrol LED RGB dengan Aplikasi Android
* Input :
* Output : 1 buah LED RGB
* Iot Starter Kit Basic
* *****/

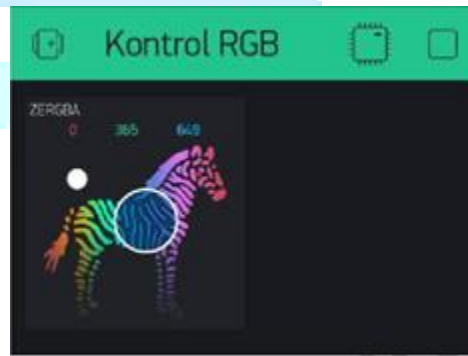
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = "b3db33a093944a979053ea58d497db99";
char ssid[] = "SEKOLAHAN.ID";
char pass[] = "foritastasolusindo";

void setup()
{
  // Debug console
  Serial.begin(9600);
}
```

```
Blynk.begin(auth, ssid, pass);  
}  
  
void loop()  
{  
  Blynk.run();  
}
```

Save programnya kemudian *Upload*. Tunggu sampai selesai. Jika sudah tersambung dengan WiFi, buka aplikasi Kontrol RGB tadi kemudian jalankan (Run) tombol play di atas kanan.

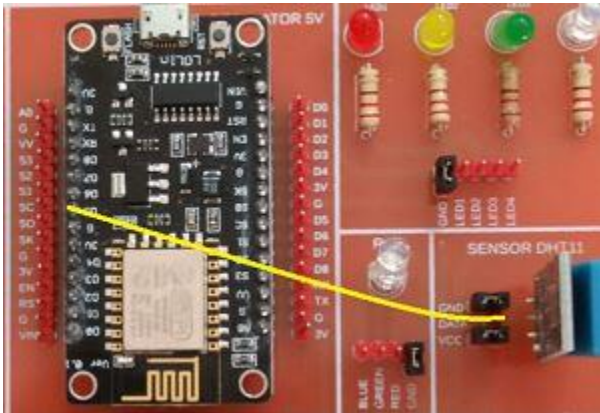


Geser dengan jari lingkaran putih untuk meenentukan warna LED RGB, perhatikan LED RGB di board **IoT Starter Kit Basic** juga berubah warna menyesuaikan warna yang kita pilih.

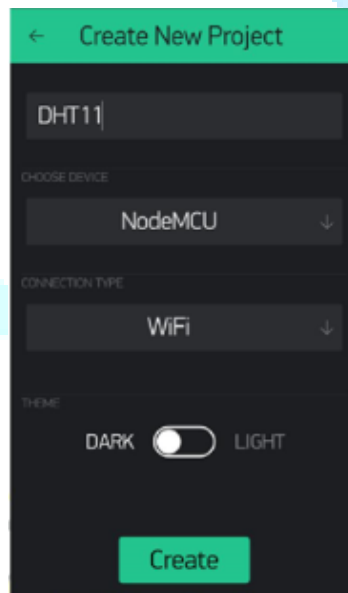
2. Project 6 : Monitoring Kelembaban dan Suhu Udara dengan Blynk

Masih setia dengan *Blynk* ya ... hehe... Kita lanjutkan membuat aplikasi monitoring suhu dan kelembaban dengan sensor DHT11. Rangkaian sama dengan Project 8.

Dibawah ini adalah gambar rangkaian yang harus di rangkai antara Modul DHT11 dengan NodeMCU.

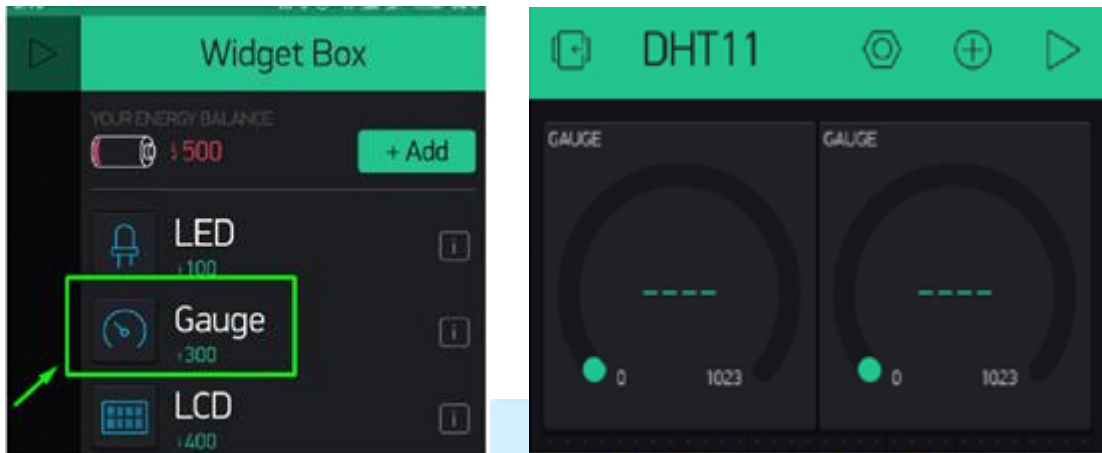


Sekarang kita buka *Blynk* kemudian buat proyek baru. Beri nama **DHT11**.

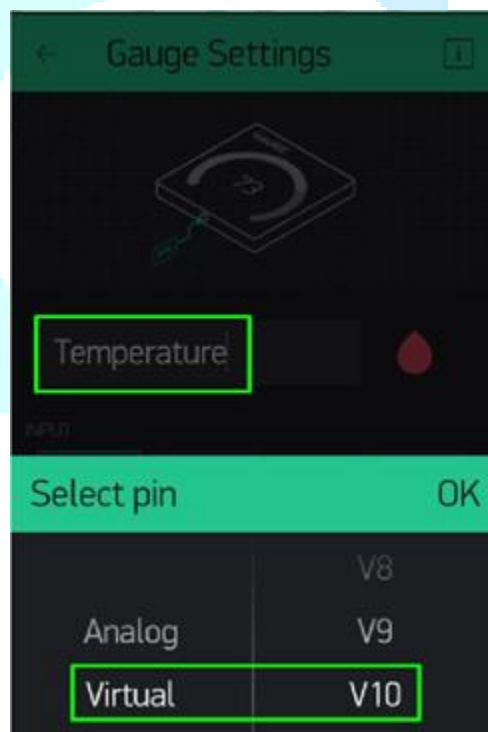


CHOOSE DEVICE : *NodeMCU*. lalu create dan blynk akan mengirim anda auth token.

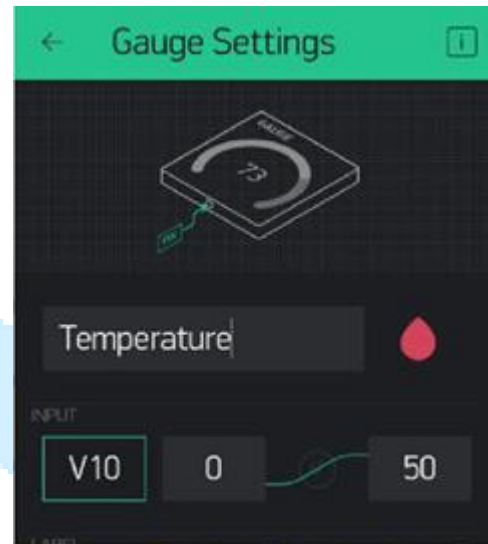
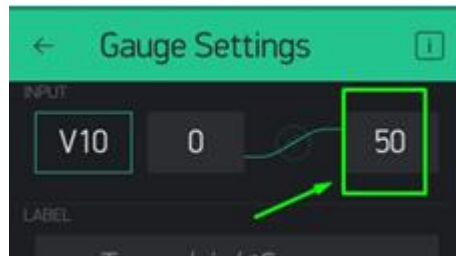
Selanjutnya tambahkan 2 buah “Gauge” di Widget Box



Seting Widget Gauge 1 untuk temperature (suhu). Klik pada widget “GAUGE” dan Beri label/nama “**Temperature**” terus klik INPUT, pilih **Virtual V10**.



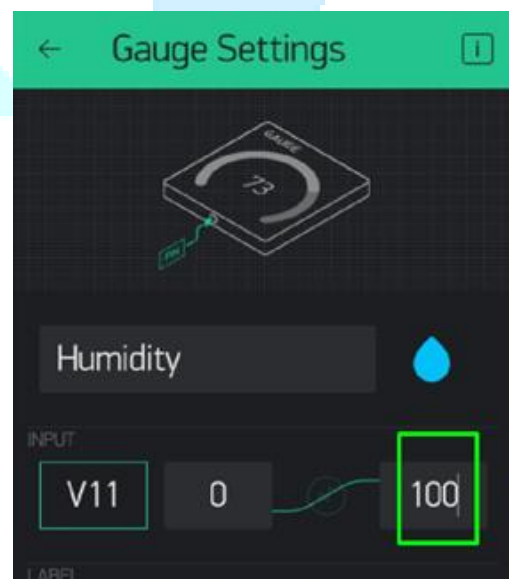
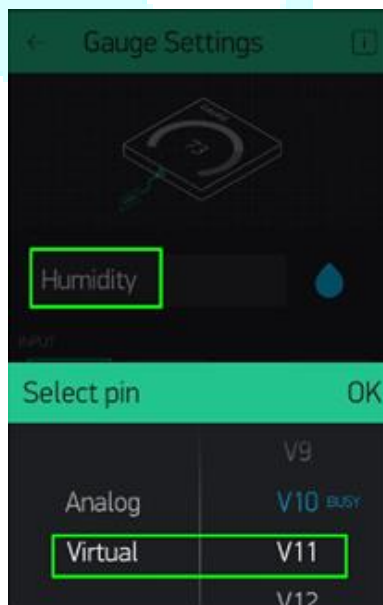
Klik di kotak sebelah kanan INPUT, isikan nilai maksimal 50, hal ini karena pembacaan suhu maksimal DHT11 adalah 50°C.



Kembali ke tampilan utama, sekarang kita seting Gauge2, klik pada widget Gauge 2.

Beri label/nama “**Humidity**” nantinya untuk tampilan nilai kelembaban.

Seting inputnya : Virtual V11 kemudian klik OK



Atur nilai maksimalnya menjadi 100.

Selanjutnya kita buat program di Arduino IDE. **Ganti Token, ssid dan password WiFi-nya.**

```

/*****
* Program :Monitoring Kelembaban dan Suhu dengan Blynk
* Input : DHT11 di pin D5
* Output :
* Iot Starter Kit Basic
* *****/

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>
#include <SimpleTimer.h>
#define DHTPIN D5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

SimpleTimer timer;

float humi, temp;
// Ganti dengan kode token anda
char auth[] = "489a863dcc2e4302ba0a5f2a3a4914ac";
// Ganti dengan WiFi dan Password anda
char ssid[] = "SEKOLAHAN.ID";
char pass[] = "foritastasolusindo";

void setup()
{
  Serial.begin(115200);
  delay(10);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(1000, sendData);
}

void sendData()
{
  humi = dht.readHumidity();
  temp = dht.readTemperature();

```

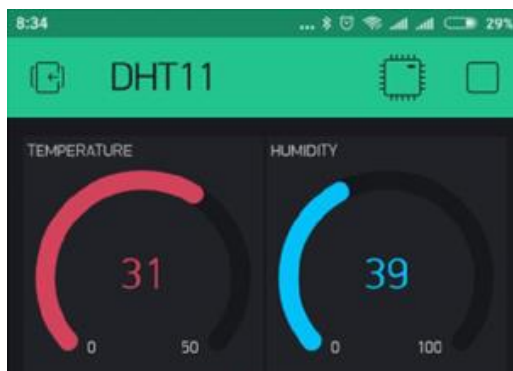
```
Blynk.virtualWrite(10, temp);//virtual input V10 di Blynk (suhu)
Blynk.virtualWrite(11, humi);//virtual input V11 di Blynk (kelembaban)
}

void loop()
{
  Blynk.run();
  timer.run();
}
```

Save programnya kemudian *Upload*. Tunggu sampai selesai. **Jangan lupa untuk mengganti Token, ssid dan password WiFi**. Buka Serial Monitor, jika sudah terhubung dengan WiFi tampilannya :



Kembali ke Project DHT11 di Blynk Android anda. Klik tombol “Run” untuk menjalankannya.

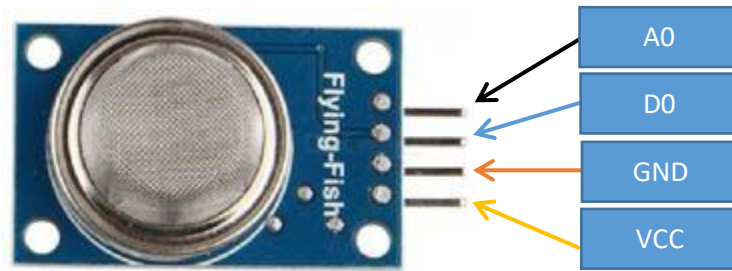


Hasilnya tampak nilai temperature dan kelembaban ditambah dengan tampilan grafis-nya.



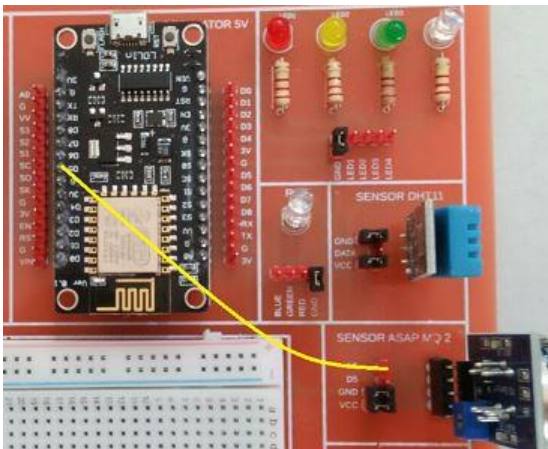
Deteksi Kebocoran Gas

Project yang ke 8 ini kita akan mendeteksi kebocoran gas dengan menggunakan sebuah sensor yaitu sensor MQ2 atau bisa di sebut juga sebagai sensor gas.



Di atas merupakan data sheet untuk sensor MQ2 atau sensor gas.

Kemudian setelah kita tahu data sheet dari sensor ini, lalu kita buat wiringnya pada modul IoT starter kit Basic.



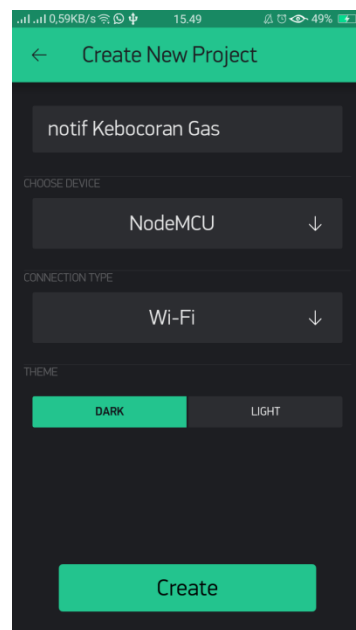
Wiringnya yaitu :

Hubungkan GND NodeMCU -> GND sensor

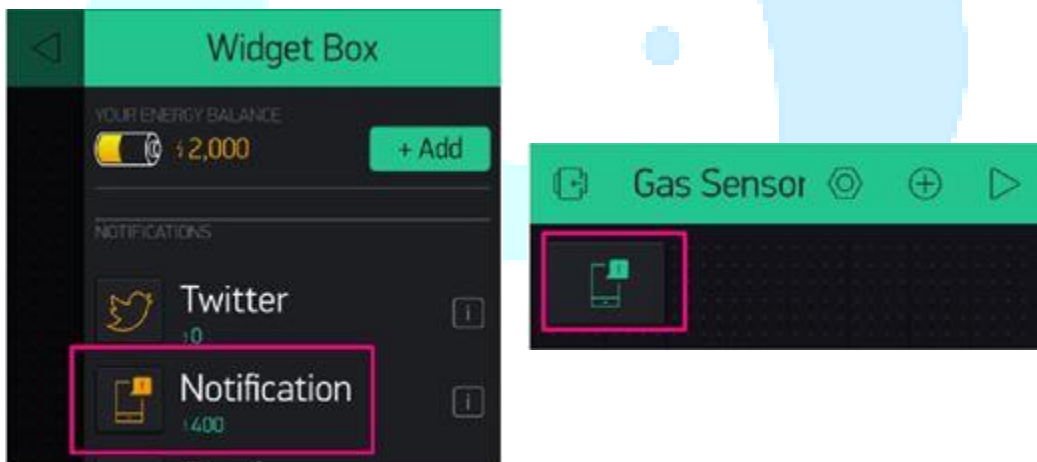
Hubungkan VCC NodeMCU -> VCC sensor

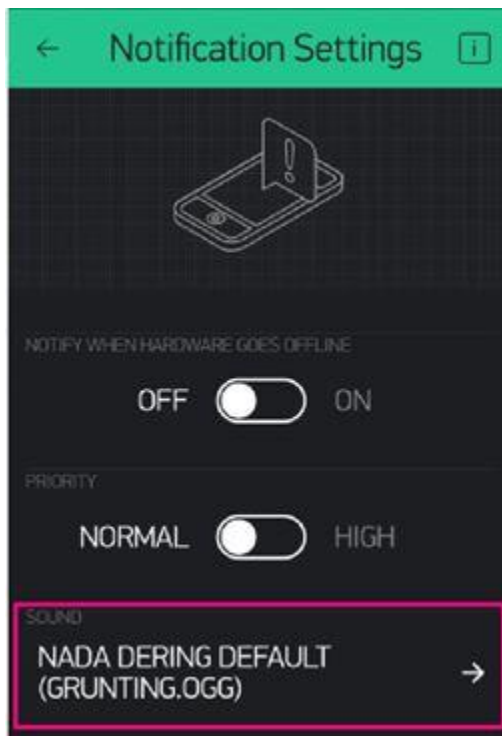
Hubungkan D5 NodeMCU -> D0 sensor

Nah sekarang kita buat aplikasinya di blynk android. seperti biasa buka project baru +New Project , lalu beri nama **"Notif Kebocoran Gas"**



Klik tombol “*Create*” kemudian cek di email anda kode Token-nya. Catat untuk pembuatan program nanti. Tambahkan sebuah Notification di Widget Box.





Untuk memilih nada dering (suara alarm) ketika notifikasi aktif, anda dapat mengganti suara/nada dering dengan memilih „sound“, klik di menu „ *SOUND* “ kemudian pilih nada yang dikehendaki.

Sekarang kita siapkan program di Arduino IDE nya. **Ganti auth token, ssid dan passwordnya.**

```

/*****
* Program :Project Deteksi Kebocoran Gas
* Input : MQ2 di pin D5
* Output : Email notification
* Iot Starter Kit Basic
* *****/

#include <ESP8266WiFi.h>
#define BLYNK_PRINT Serial // Comment this out to disable prints and save
space
#include <BlynkSimpleEsp8266.h>
char auth[] = "1928e00202647eb833bc78cb26b9d84";

/* Ganti dengan ssid dan password WiFi anda */
char ssid[] = "SEKOLAHAN.ID";

```

```
char pass[] = "foritastasolusindo";

#define MQ2Pin D5
int MQ2Value;

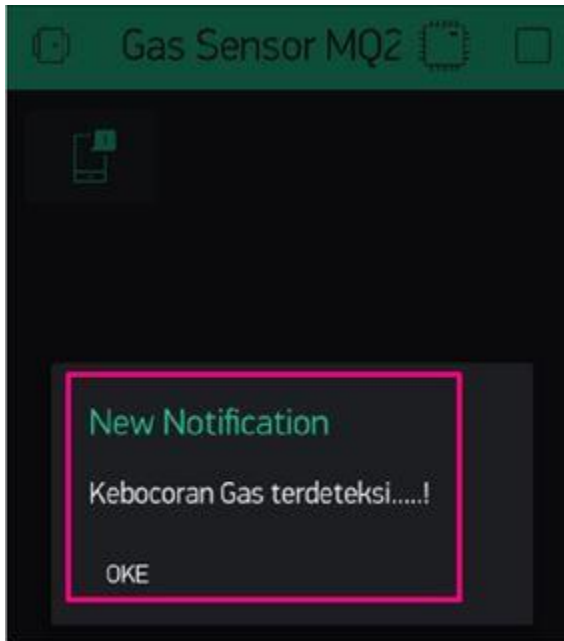
void setup()
{
  Serial.begin(115200);
  delay(10);
  Blynk.begin(auth, ssid, pass);
  pinMode(MQ2Pin, INPUT);
}

void loop()
{
  getMQ2Value();
  Blynk.run();
}

// *****/
void getMQ2Value(void)
{
  MQ2Value = digitalRead(MQ2Pin);
  if (!MQ2Value)
  {
    Serial.println("==>Gas terdeteksi");
    Blynk.notify("Kebocoran Gas terdeteksi.....!");
    delay(1000);
  }
}
```

Setelah ini anda bisa save dan upload program lalu lihat di serial monitor setelah ada bacaan ready anda bisa lihat ke aplikasi blynk dan run aplikasinya. Kemudian coba anda beri sensor tersebut gas dari korek api gas dan apa yang terjadi.

Maka akan ada notifikasi di blynk dengan bacaan “Kebocoran Gas Terdeteksi”



```
File Edit Sketch Tools Help
[Icons]
9_Deteksi_Kebocoran_Gas
{
  void loop()
  {
    getMQ2Value();
    Blynk.run();
  }

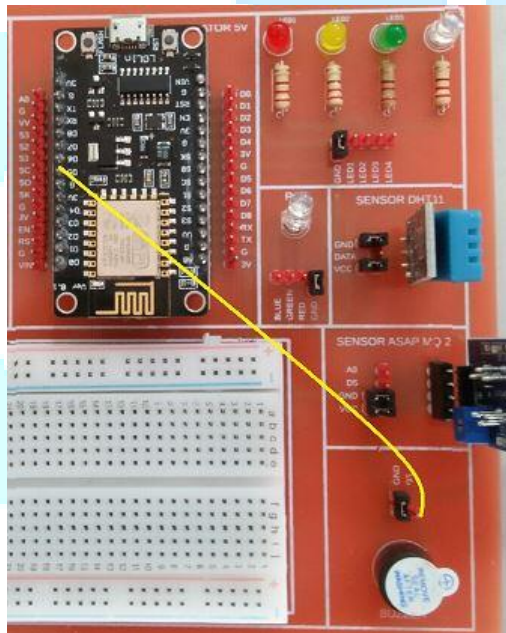
  // *****
  void getMQ2Value(void)
  {
    MQ2Value = digitalRead(MQ2Pin);
    if (!MQ2Value)
    {
      Serial.println("=>Gas terdeteksi");
      Blynk.notify("Kebocoran Gas terdeteksi.....!");
      delay(1000);
    }
  }
}
```

3. Project 9 : Kontrol Buzzer Via Android

Buzzer adalah sebuah alat yang biasa di pakai untuk kita sebagai alarm dan pada kali ini kita akan coba buzzer ini kita kontrol dengan aplikasi android yaitu blynk, yuk kita mulai saja .



Rangkai lah dengan kabel jumper antara NodeMCU dengan Modul Buzzer seperti gambar di bawah ini.

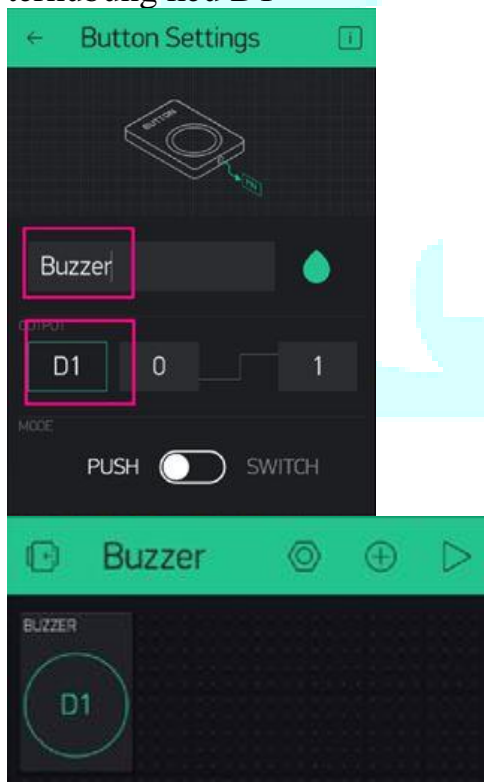


Setelah itu kita langsung menuju aplikasi androidnya.

Pertama anda harus membuka blynknya terlebih dahulu lalu anda bisa menekan create new project dan berinama “buzzer”, atur juga devicenya jangan lupa di “NodeMCU” dan connection typenya di atur sebagai “WiFi” lalu tekan tombol “Create” agar anda dapat mendapat kode tokenya melalui email.



Setelah anda menekan Create lalu anda bisa mendapat email dari aplikasi blynk , setelah itu pilih di kanan atas untuk widget **button** di aplikasi blynk dan atur labelnya sebagai buzzer dan pilih digital i/nya ke D1 maka akan terhubung ke D1



Setelah semuanya selesai maka langkah selanjutnya adalah membuat program di ArduinoIDE dan jangan lupa atur “SSID” dan “Pasword” WiFinya ok...

```
/******  
* Program :Project 16 Kontrol Buzzer  
* Input : -  
* Output : Buzzer di D1  
* Iot Starter Kit Basic  
* *****/  
  
#define BLYNK_PRINT Serial  
#include <ESP8266WiFi.h>  
#include <BlynkSimpleEsp8266.h>  
//Ganti dengan Token anda  
char auth[] = "6d2c7f338e0b498a87c8ec4fbaf07406";  
  
// Ganti dengan WiFi anda  
char ssid[] = "SEKOLAHAN.ID";  
char pass[] = "foritastasolusindo";  
  
void setup()  
{  
  Serial.begin(115200);  
  Blynk.begin(auth, ssid, pass);  
}  
  
void loop()  
{  
  Blynk.run();  
}
```

Lalu simpan dan upload programnya pastikan usb terhubung ke pc/laptop anda ok langsung saja upload.

Setelah selesai anda bisa membuka aplikasi blynk lalu run dan tekan tombol yang tadi kita buat maka apabila tombol di tekan maka buzzer akan berbunyi dan kita bisa mengontrol buzzer tersebut.