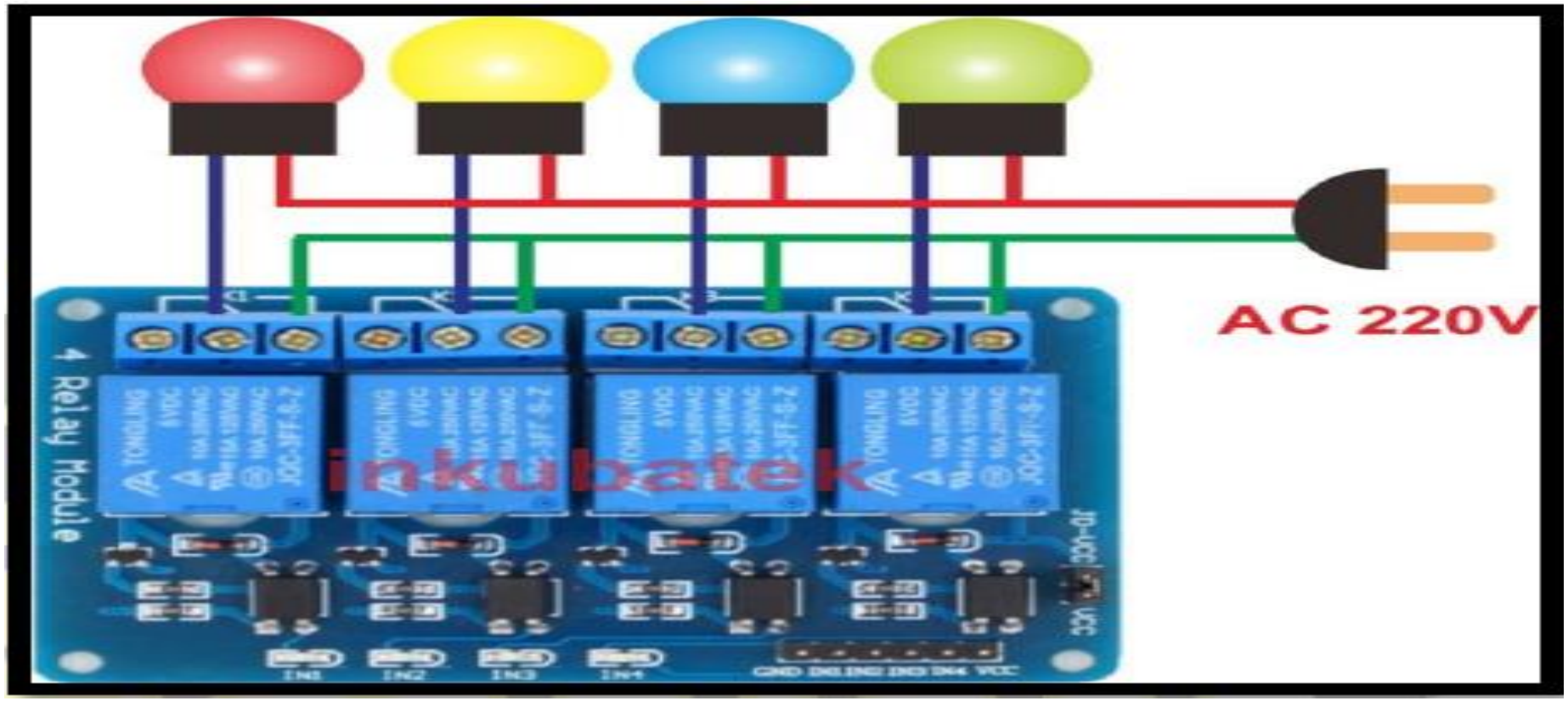
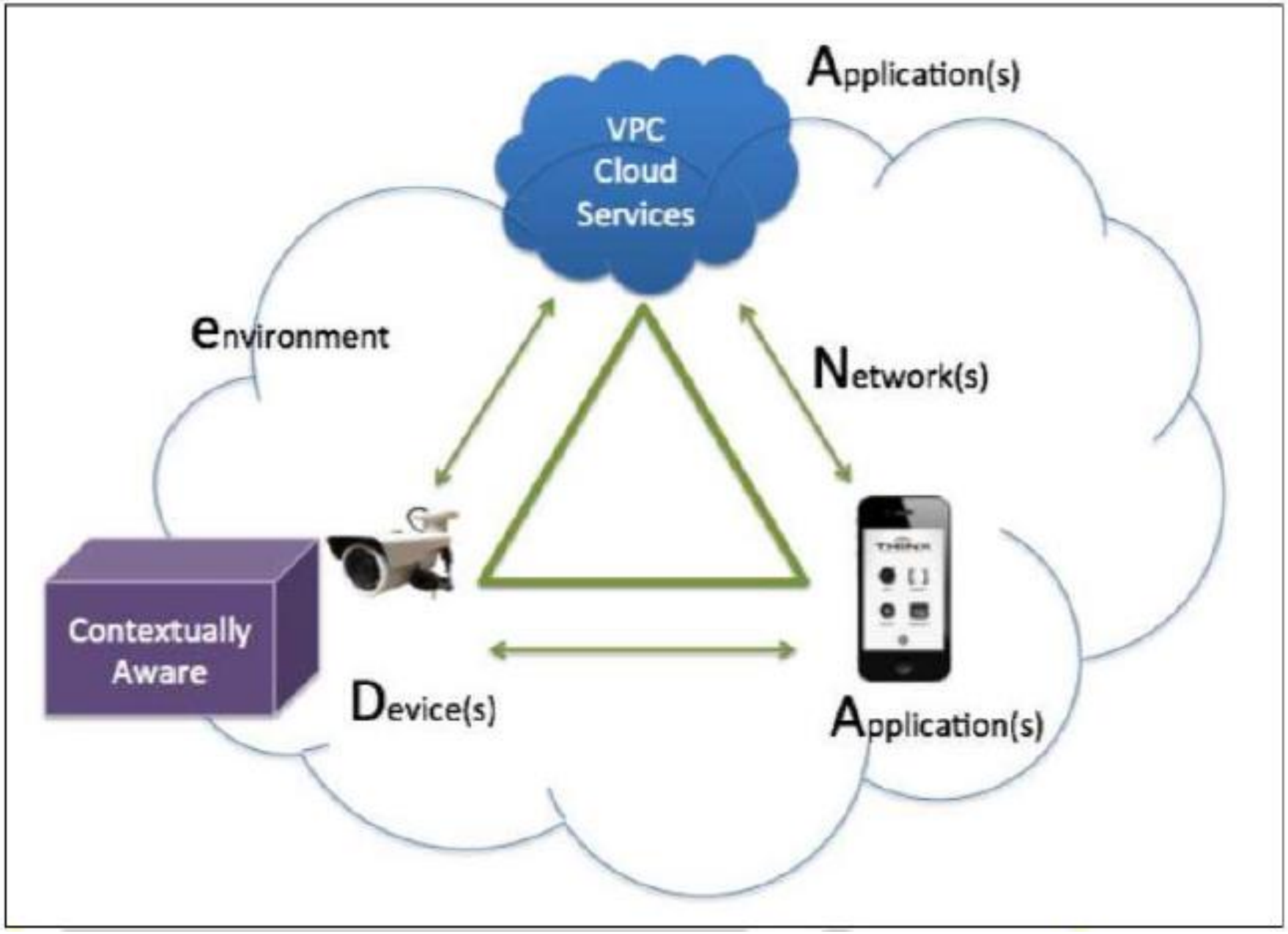
Smart Lamps



By: Tim i-ot.net

Sistem IoT


Application(s)
VPC
Cloud
Services
Environment
Network(s)
Contextually
Aware
Device(s)
Application(s)

Yang dibutuhkan

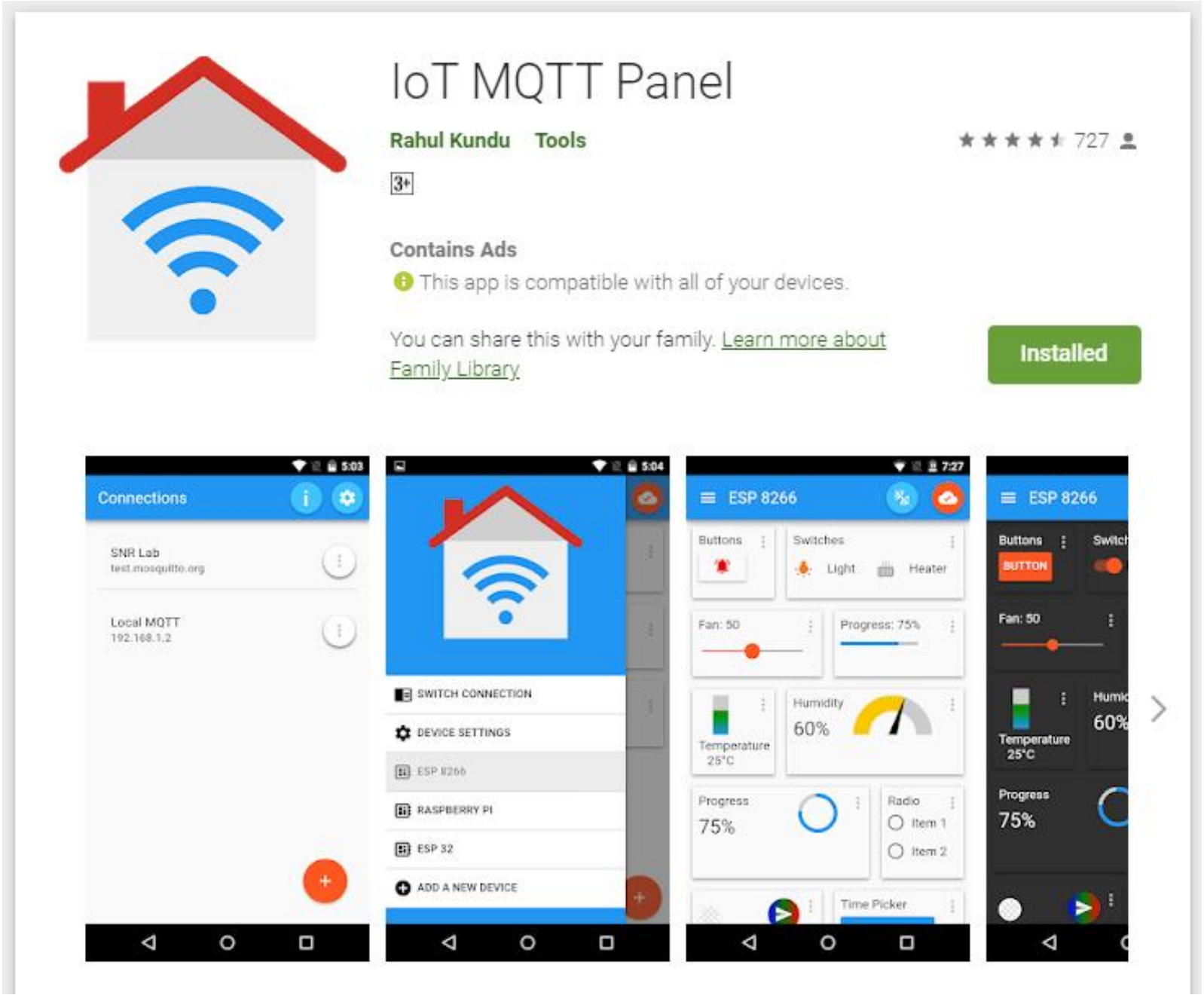
- **Application**

  Silahkan diinstall IoT MQTT Panel dari PlayStore di HP Android
- **Cloud IoT**

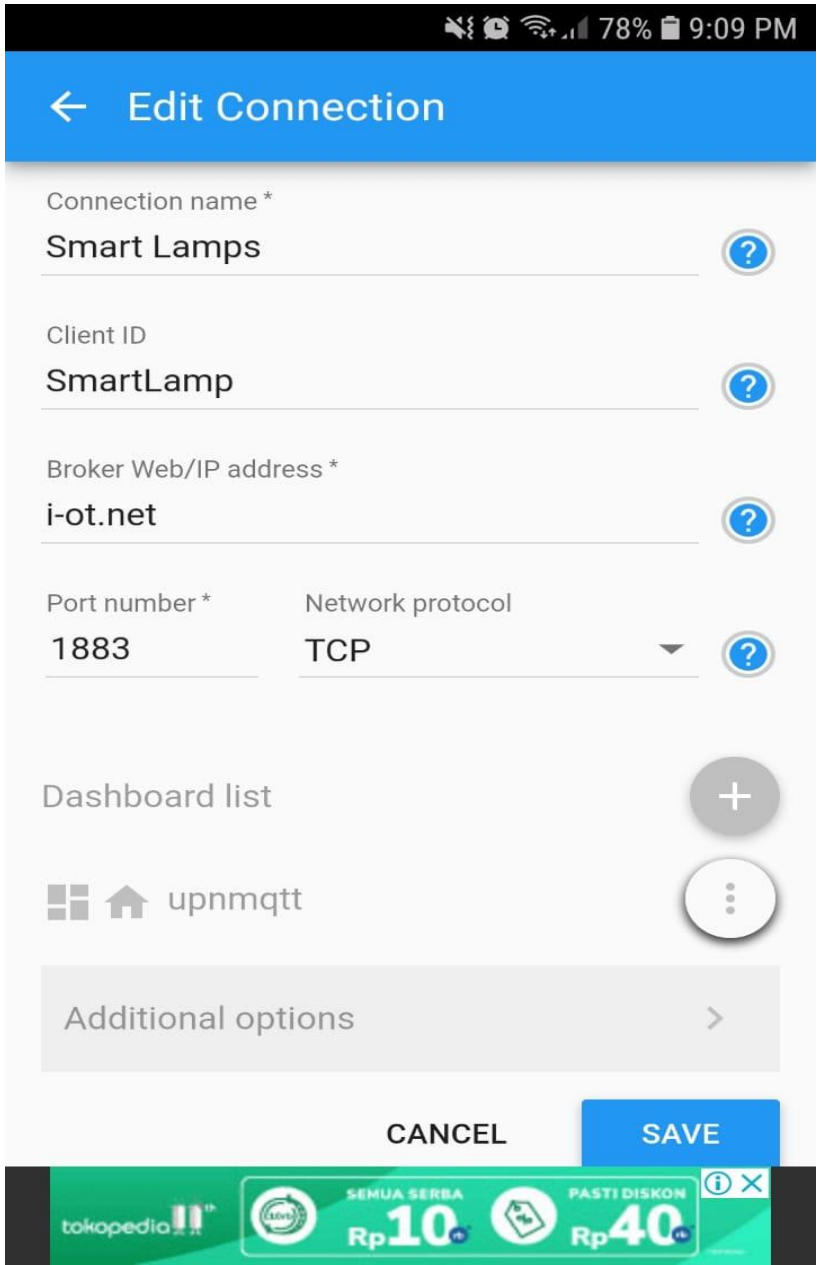
  Digunakan broker i-ot.net sebagai Pengganti cloudmqtt.com
- **Device**

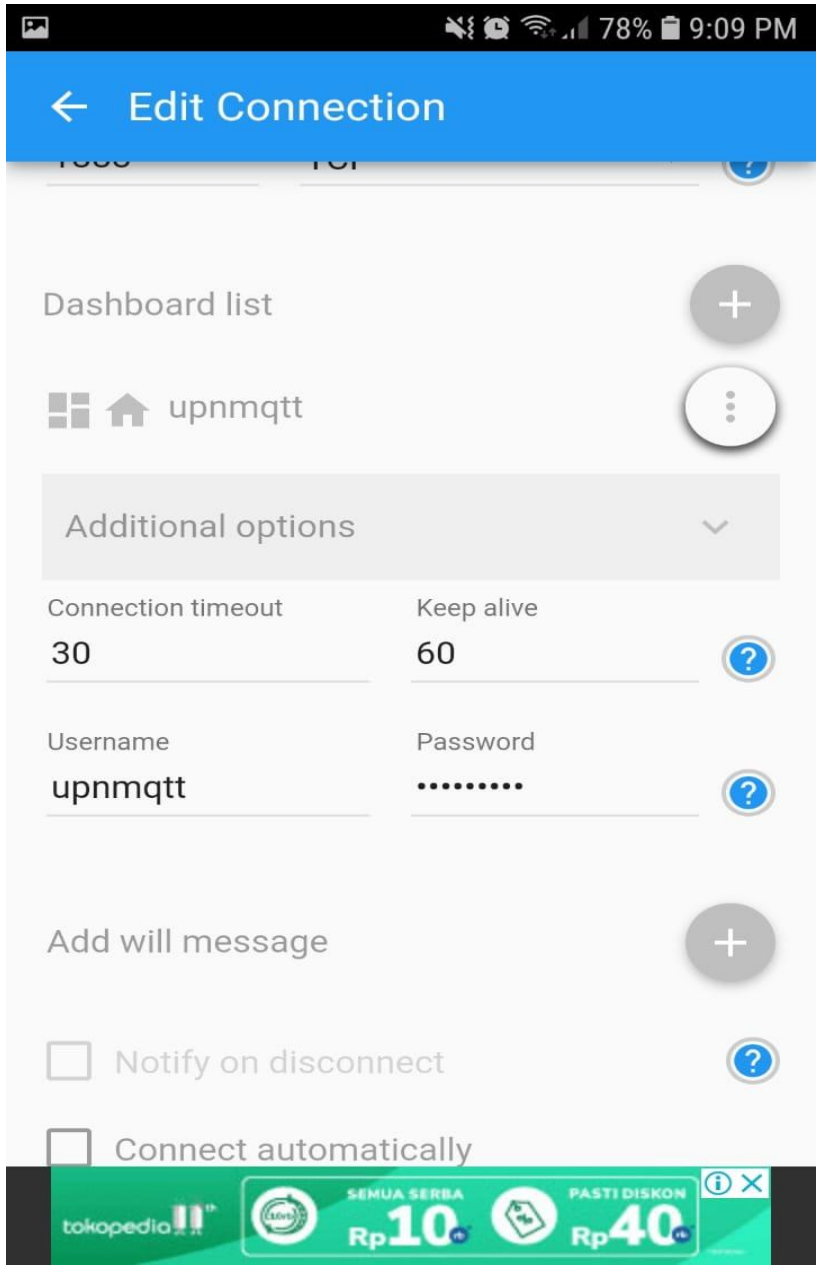
  Bisa digunakan IoT Starker Kit Produk Tokotronik atau Rakit Sendiri

IoT MQTT Panel



Seting Cloud IoT (i-ot.net) di IoT MQTT Panel





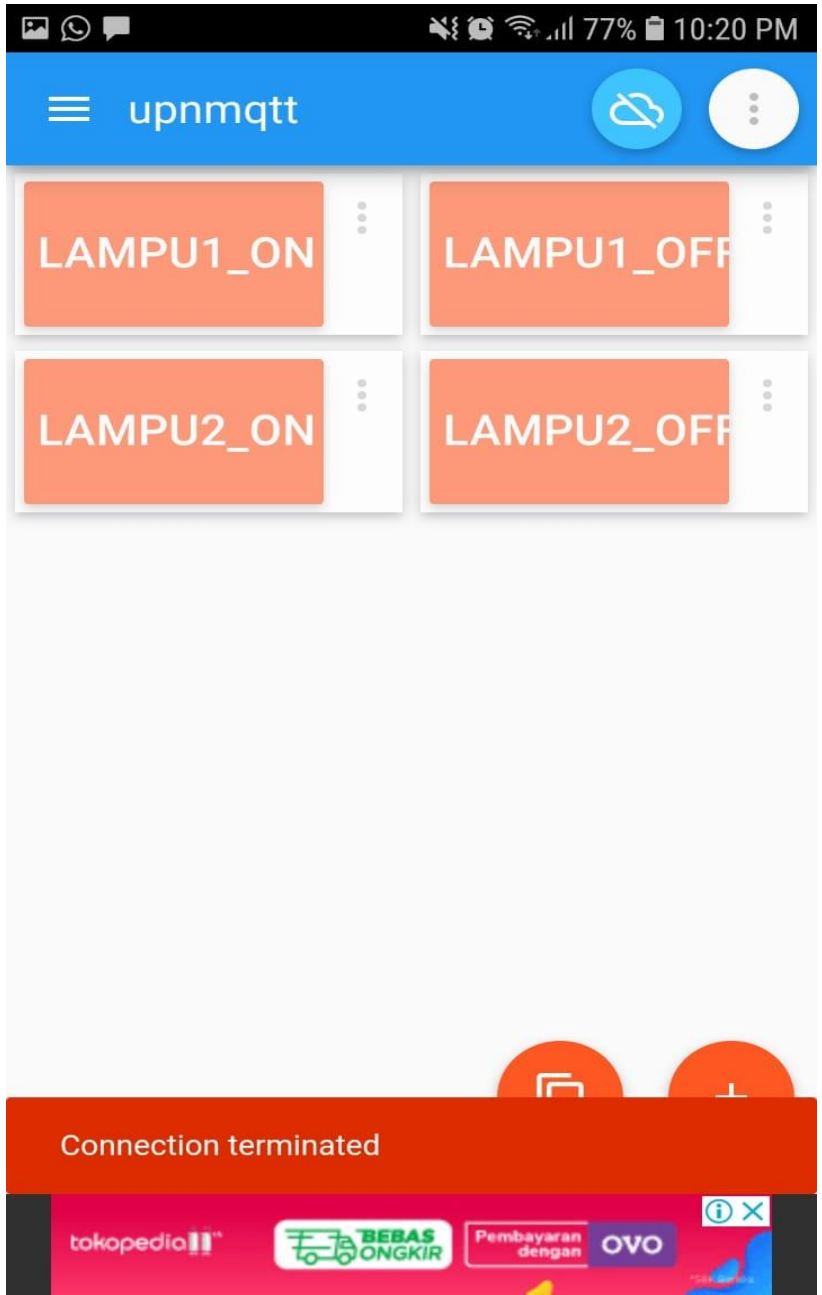
Isikan:

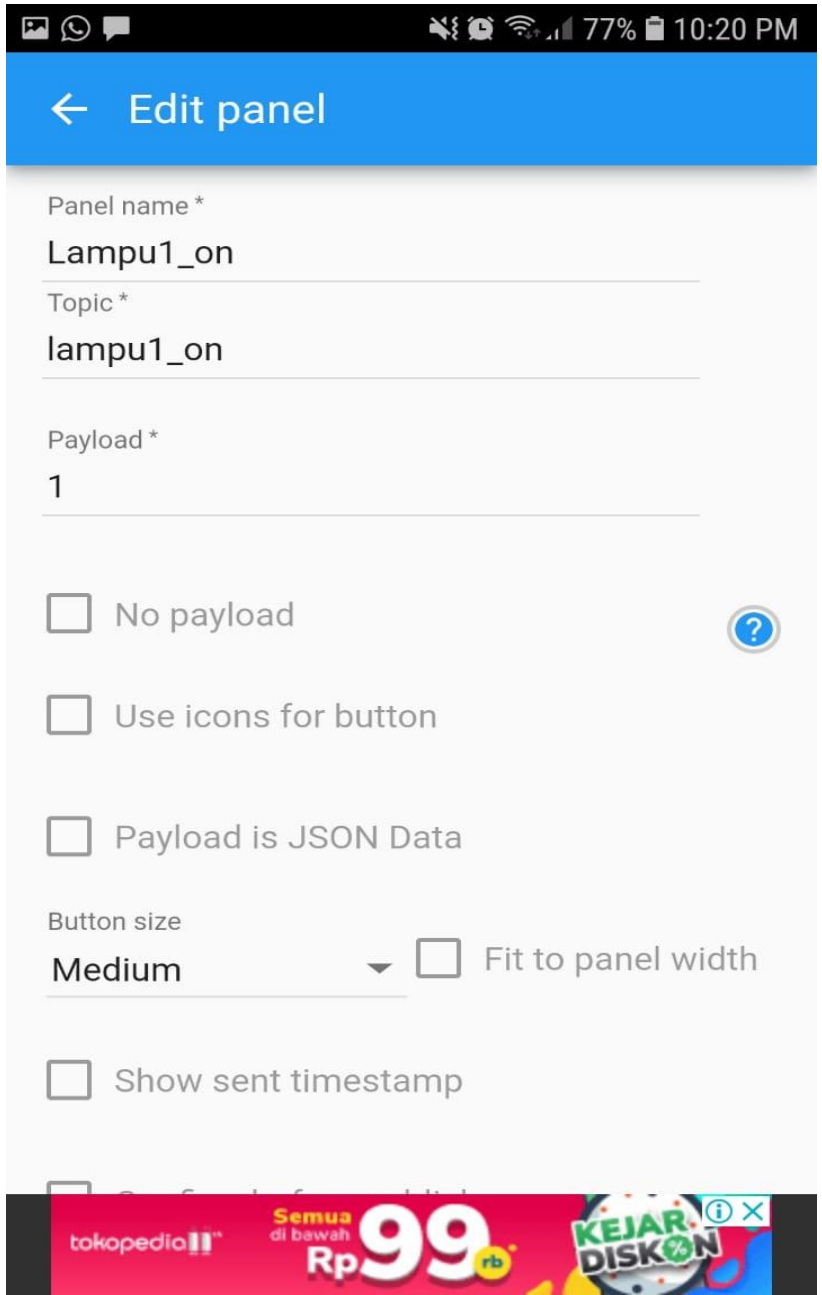
Connection name dll

User: upnmqtt

Password: 20upnmqtt

Buat Panel Kendali di IoT MQTT Panel





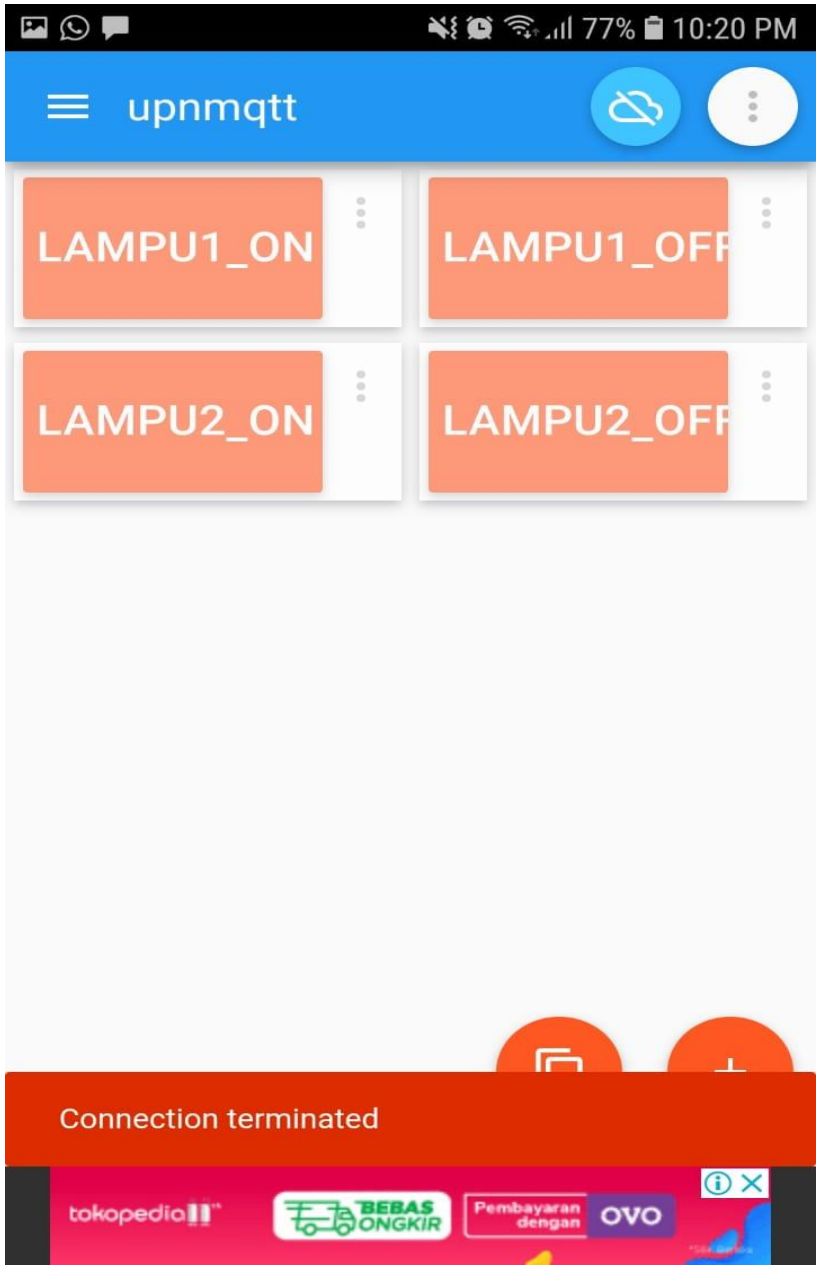
Yg penting:

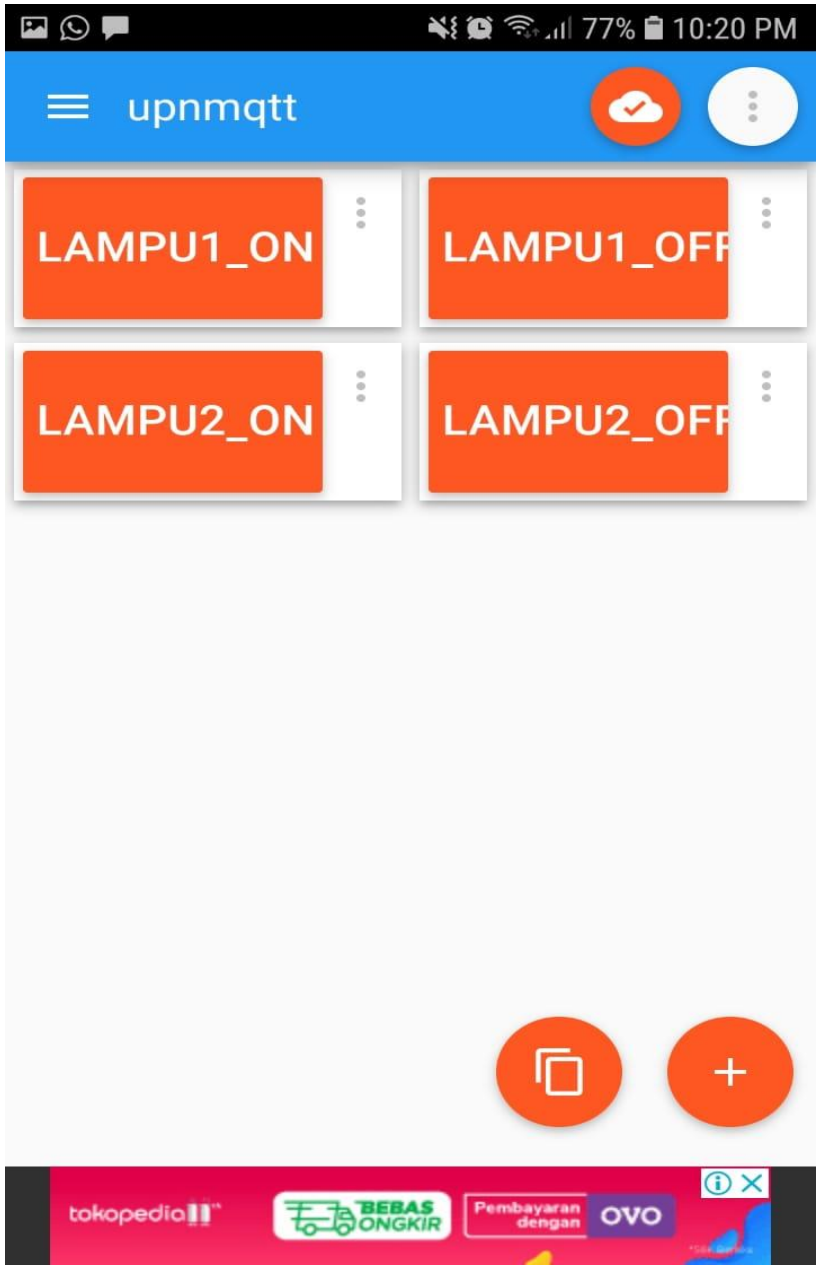
Nama Topik harus sesuai dengan program di Mikrokontroller

Misal Panel 1, Topik diisi lampu1_on, Lampu1_off dst

Payload diisi 1, 2, dst

Tes Koneksi ke Cloud IoT (i-ot.net)

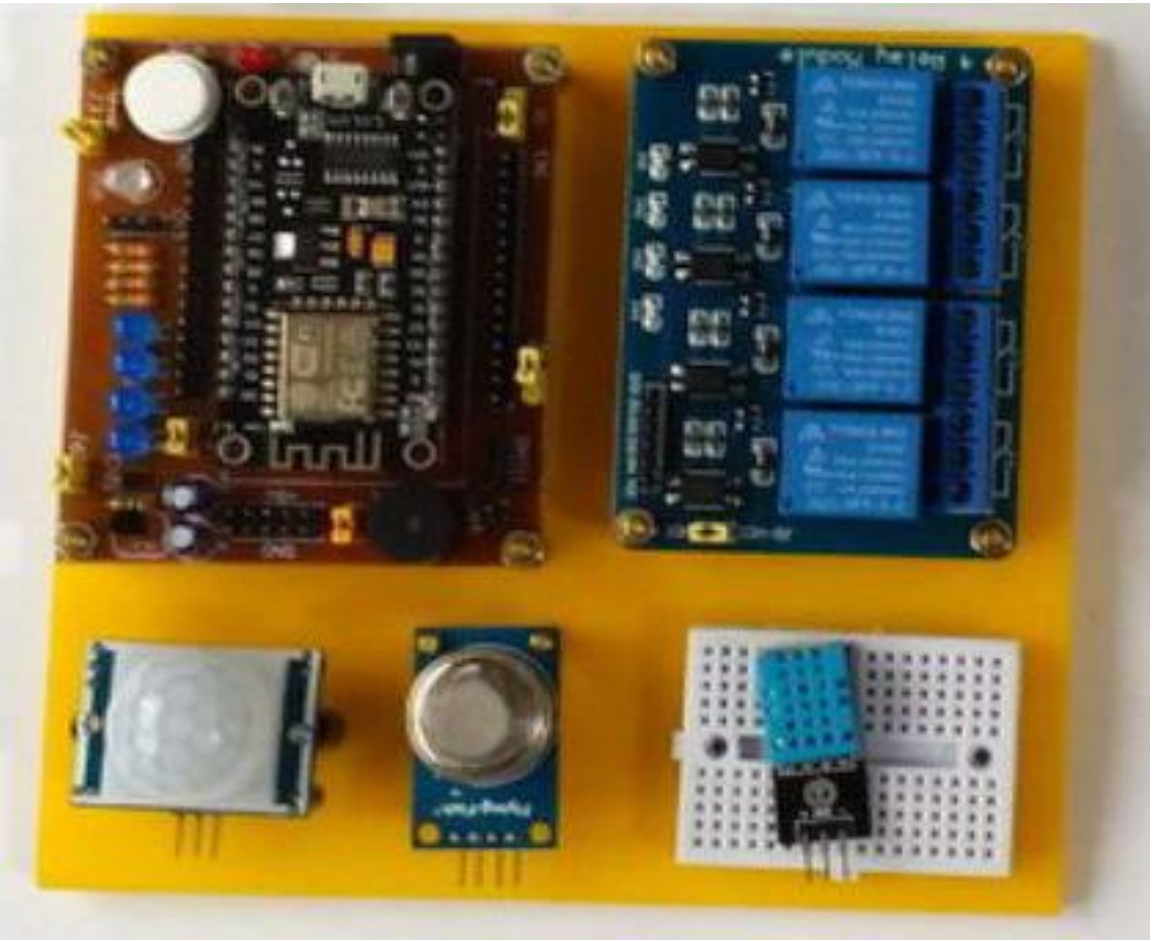




Buat dulu Panel seperti contoh
Coba tes koneksi
Warna akan berubah

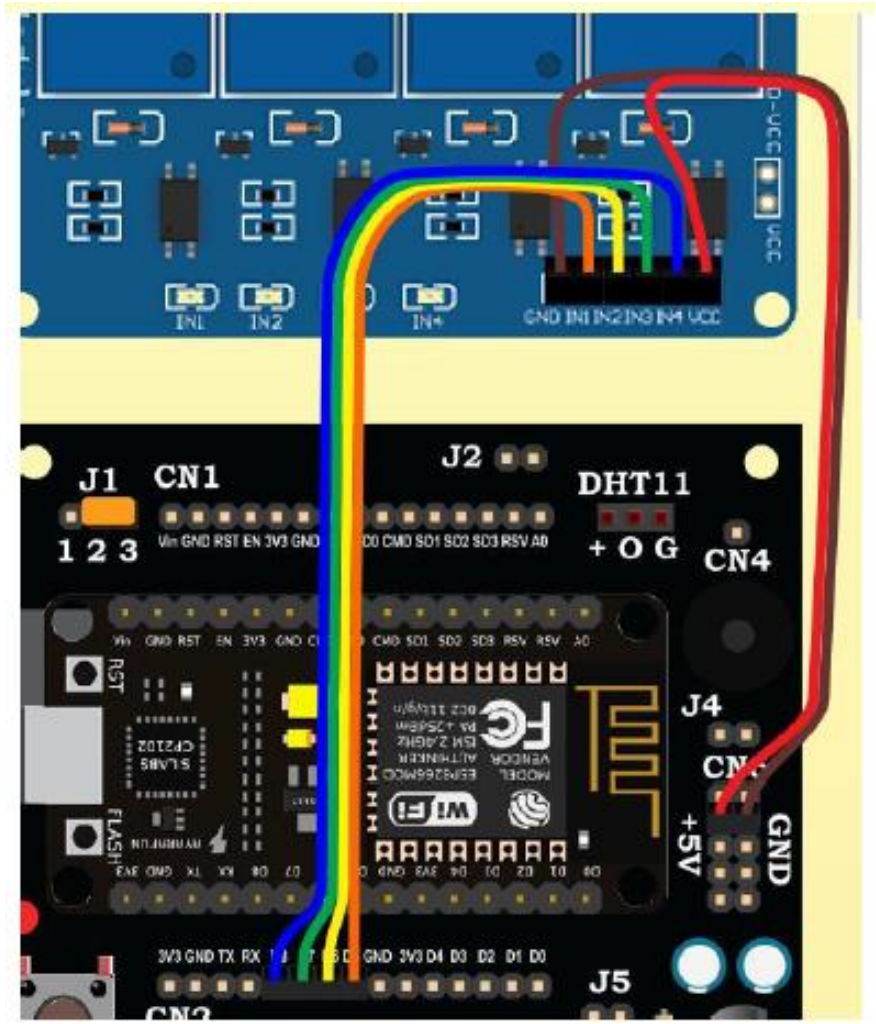
Device

Bisa digunakan IoT Starker Kit Produk Tokotronik atau Rakit Sendiri

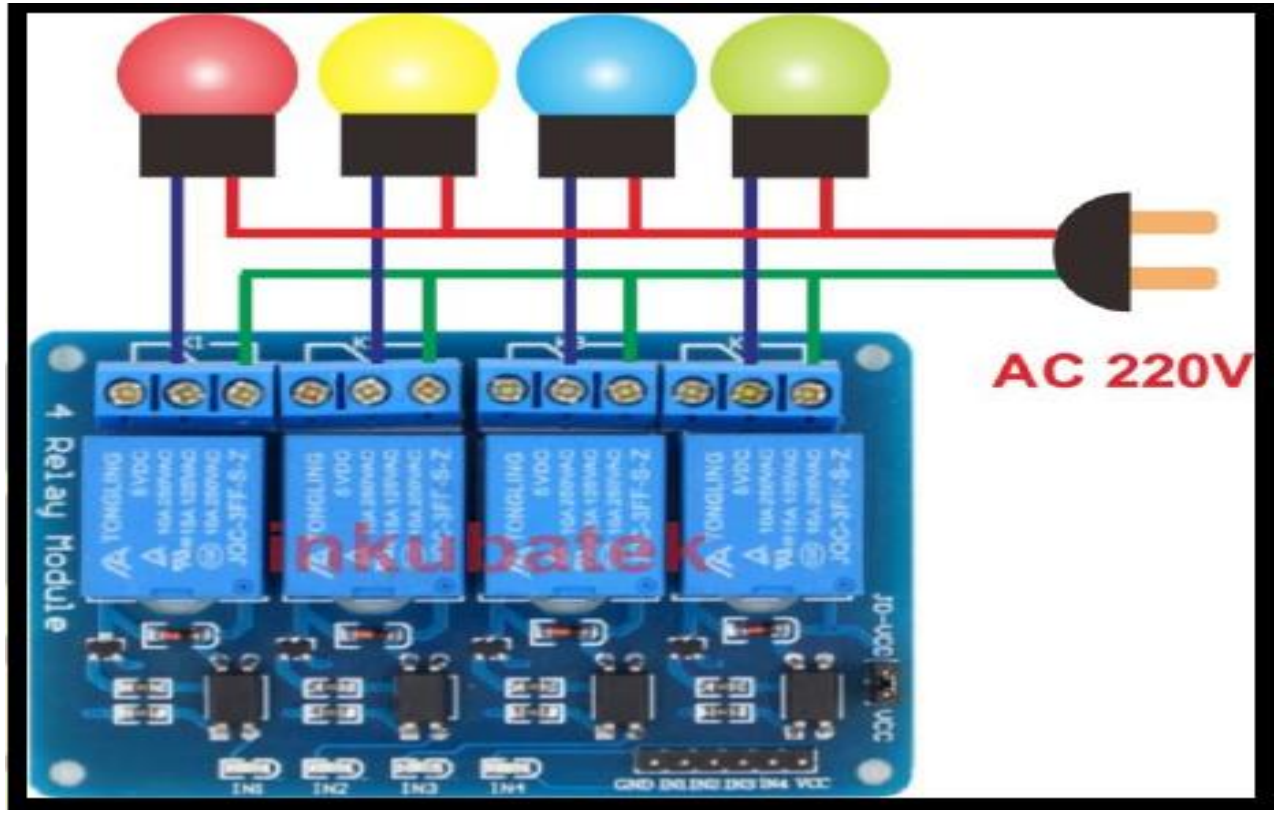


Device

Koneksi Port



Relay modul	Board IoT Starter Kit
IN1	D5
IN2	D6
IN3	D7
IN4	D8
VCC	+5V
GND	GND



Pemrograman Arduino

```
/**************************************
 * Program : SMART LAMPS
 * Tim i-ot.net
 * ************************************/

#include <ESP8266WiFi.h>
#include <PubSubClient.h>

String Topic;
String Payload;

const char* ssid = “nama_wifi";  // Tergantung wifi yang digunakan
const char* password = “pass_wifi"; // Password wifi
```

```
#define IN_1 D5 // Lampu1
#define IN_2 D6 // Lampu2
#define IN_3 D7 // Lampu3
#define IN_4 D8 // Lampu4

#define mqttServer "i-ot.net"
#define mqttPort 1883
#define mqttUser "upnmqtt"
#define mqttPassword "20upnmqtt"

WiFiServer server(80);
WiFiClient espClient;
PubSubClient client(espClient);
```

```
void receivedCallback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message received: ");
  Serial.println(topic);
  Serial.print("payload: ");

  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
  Serial.println();

  /* we got '1' -> Lampu1_on */
  if ((char)payload[0] == '1') {
    digitalWrite(IN_1, HIGH);
  }

/* we got '2' -> Lampu1_off */
  if ((char)payload[0] == '2') {
    digitalWrite(IN_1, LOW);
  }
```

```
 /* we got '3' -> Lampu2_on */
  if ((char)payload[0] == '3') {
    digitalWrite(IN_2, HIGH);
  }

  /* we got '4' -> Lampu2_off */
  if ((char)payload[0] == '4') {
    digitalWrite(IN_2, LOW);
  }
}

void setup() {
  Serial.begin(115200);
  delay(10);
  pinMode(IN_1, OUTPUT);
  pinMode(IN_2, OUTPUT);
  pinMode(IN_3, OUTPUT);
  pinMode(IN_4, OUTPUT);

  digitalWrite(IN_1, LOW);
  digitalWrite(IN_2, LOW);
  digitalWrite(IN_3, LOW);
  digitalWrite(IN_4, LOW);
```

```
// Connect to WiFi network
 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");

 server.begin();
 Serial.println("Server started");

 Serial.print("Use this URL to connect: ");
 Serial.print("http://");
 Serial.print(WiFi.localIP());
 Serial.println("/");
```

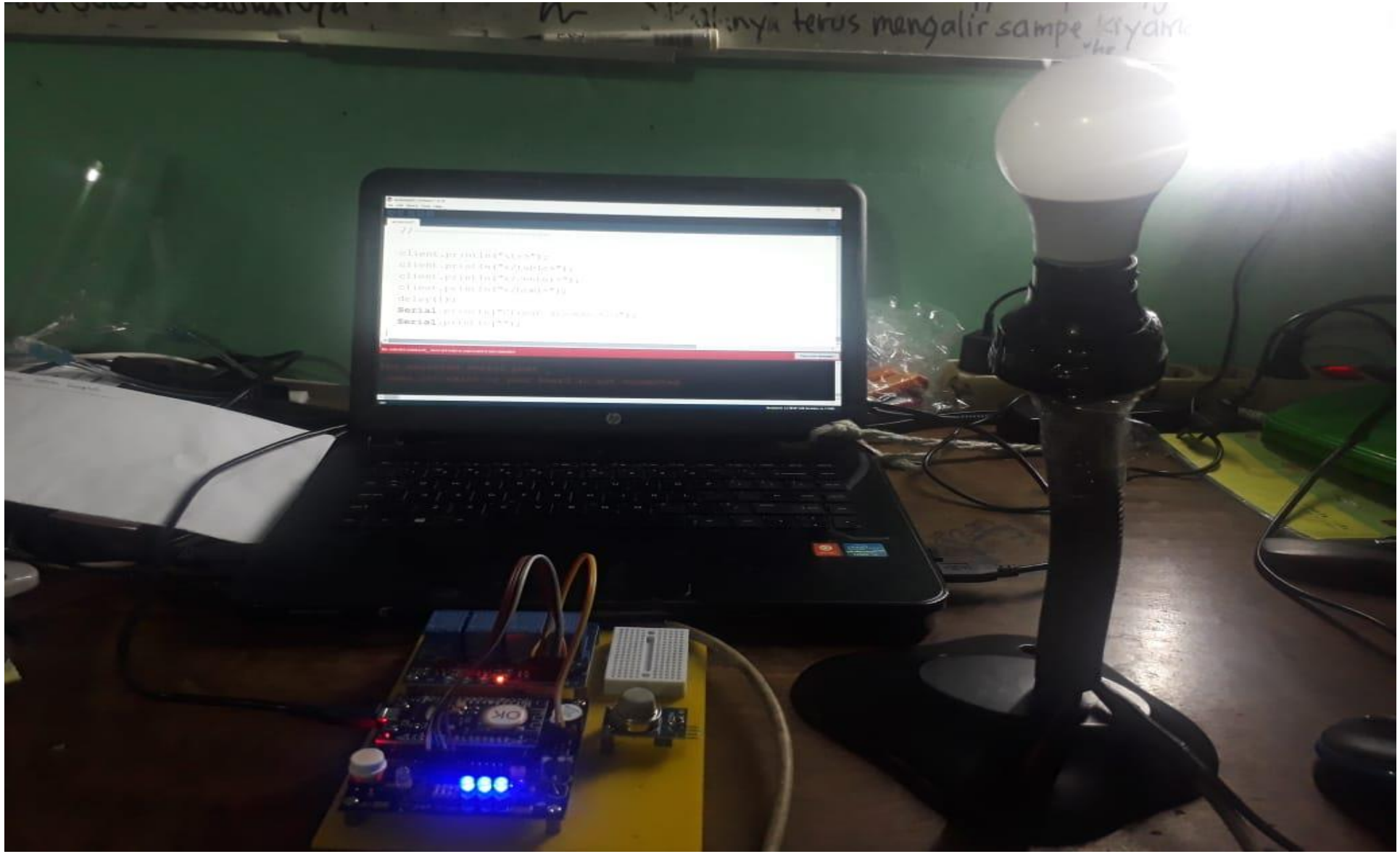
```
// Connect to Server IoT (CloudMQTT)

 client.setServer(mqttServer, mqttPort);
 client.setCallback(receivedCallback);

 while (!client.connected()) {
   Serial.println("Connecting to CLoudMQTT...");

   if (client.connect("ESP32Client", mqttUser, mqttPassword )) {

     Serial.println("connected");

   } else {
     Serial.print("failed with state ");
     Serial.print(client.state());
     delay(2000);
   }
 }
 client.subscribe("lampu1_on");
 client.subscribe("lampu1_off");
 client.subscribe("lampu2_on");
 client.subscribe("lampu2_off");
}
```

```
void loop() {

  client.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("/IN_1on") > 0)  {
    digitalWrite(IN_1, HIGH);
  }
  if (request.indexOf("/IN_1off") >0)  {
    digitalWrite(IN_1, LOW);
  }
   if (request.indexOf("/IN_2on") > 0)  {
    digitalWrite(IN_2, HIGH);
  }
  if (request.indexOf("/IN_2off") >0)  {
    digitalWrite(IN_2, LOW);
  }
```

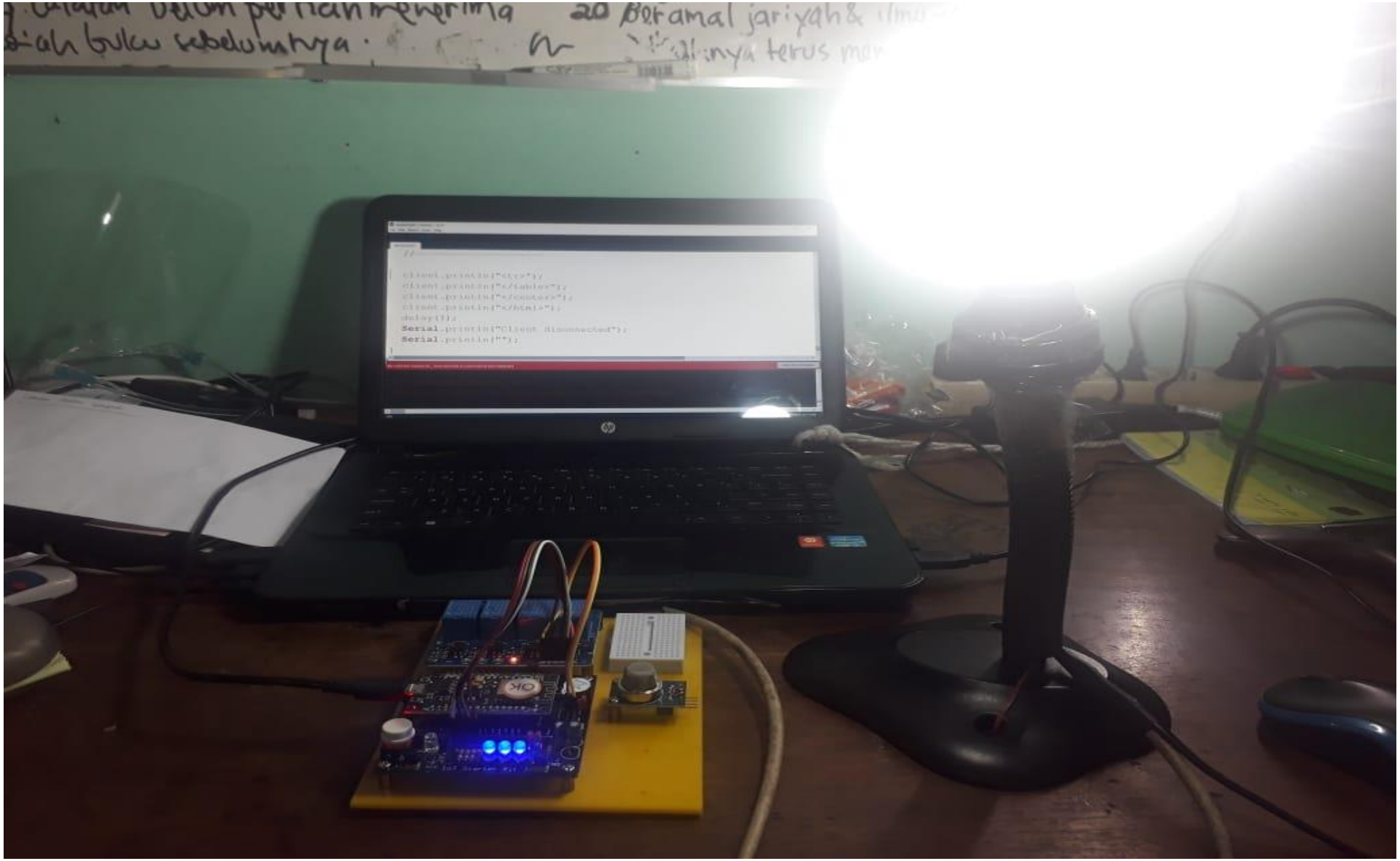
```
// Return the response
 client.println("HTTP/1.1 200 OK");
 client.println("Content-Type: text/html");
 client.println("");
 client.println("<!DOCTYPE HTML>");
 client.println("<html>");
 client.println("<head>");
 client.println("<meta name='apple-mobile-web-app-capable' content='yes' />");
 client.println("<meta name='apple-mobile-web-app-status-bar-style' content='black-translucent' />");
 client.println("</head>");
 client.println("<body bgcolor = \"#f7e6ec\">");
 client.println("<hr/><hr>");
 client.println("<h4><center> Smart Lamps  </center></h4>");
 client.println("<hr/><hr>");
 client.println("<br><br>");
 client.println("<br><br>");
 client.println("<center>");
 client.println("ROBOT");
 client.println("<a href=\"/IN_1on\"\"><button>Lampu1_On </button></a>");
 client.println("<a href=\"/IN_1off\"\"><button>Lampu1_Off </button></a><br />");
 client.println("</center>");
 client.println("<br><br>");
 client.println("<center>");
 client.println("ROBOT");
 client.println("<a href=\"/IN_2on\"\"><button>Lampu2_On </button></a>");
 client.println("<a href=\"/IN_2off\"\"><button>Lampu2_Off </button></a><br />");
 client.println("</center>");
 client.println("<br><br>");
 client.println("<center>");
 client.println("<table border=\"5\">");
 client.println("<tr>");
 //====================
```

```
 client.println("<tr>");
  client.println("</table>");
  client.println("</center>");
  client.println("</html>");
  delay(1);
  Serial.println("Client disonnected");
  Serial.println("");
}
```

Gambaran Riset



Gambaran Riset



Terimakasih…