

Nama : Solani

NIM : H1D022033

Shift : H

PRAKTIKU PEMROGRAMAN SISTEM TERTANAM

LED IOT

Program ini adalah aplikasi IoT sederhana yang menggunakan ESP8266 untuk mengendalikan tiga buah LED melalui halaman web yang diakses lewat jaringan WiFi lokal. ESP8266 bertindak sebagai Access Point (AP) dan Web Server, sehingga perangkat seperti HP atau laptop bisa terhubung langsung ke ESP8266 dan mengakses halaman kontrol LED.

1. Source Code

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>

const char* ssid = "Laniisa";
const char* password = "lanii2207";
int LED1 = D0;
int LED2 = D1;
int LED3 = D4;

ESP8266WebServer Server(80);

String webpage;

void setup() {
  Serial.begin(9600);
  delay(10);

  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);

  Serial.println();
  Serial.print("Configuring access point,,,");

  WiFi.mode(WIFI_AP);
```

```

WiFi.begin(ssid, password);

Serial.print("Access Point SSID: ");
Serial.println(WiFi.softAPSSID());

Serial.println("IP address: ");
Serial.println(WiFi.softAPIP());

webpage+= "<h1> Web Control ESP8266</h1>";
webpage+= "<h2> SOLANI </h2>";
webpage+= "<p>LED 1 : ";
webpage+= "<a href=\"LED1ON\"><button>ON</button></a><a href=\"LED1OFF\"><button>OFF</button></a></p><br>";
webpage+= "<p>LED 2 : ";
webpage+= "<a href=\"LED2ON\"><button>ON</button></a><a href=\"LED2OFF\"><button>OFF</button></a></p><br>";
webpage+= "<p>LED 3 : ";
webpage+= "<a href=\"LED3ON\"><button>ON</button></a><a href=\"LED3OFF\"><button>OFF</button></a></p>";

Server.on("/", []() {
    Server.send(200, "text/html", webpage);
});

Server.on("/LED1ON", []() {
    Server.send(200, "text/html", webpage);
    digitalWrite(LED1, HIGH);
    delay(1000);
});

Server.on("/LED2ON", []() {
    Server.send(200, "text/html", webpage);
    digitalWrite(LED2, HIGH);
    delay(1000);
});

Server.on("/LED3ON", []() {
    Server.send(200, "text/html", webpage);
    digitalWrite(LED3, HIGH);
    delay(1000);
});

Server.on("/LED1OFF", []() {
    Server.send(200, "text/html", webpage);
    digitalWrite(LED1, LOW);

```

```

    delay(1000);
  });

  Server.on("/LED2OFF", []() {
    Server.send(200, "text/html", webpage);
    digitalWrite(LED2, LOW);
    delay(1000);
  });

  Server.on("/LED3OFF", []() {
    Server.send(200, "text/html", webpage);
    digitalWrite(LED3, LOW);
    delay(1000);
  });

  Server.begin();
  Serial.println("Webserver dijalankan");
}

void loop() {
  Server.handleClient();
}

```

2. Penjelasan

a. Library dan Deklarasi

```

#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>

```

Fungsi: Mengimpor pustaka yang dibutuhkan agar ESP8266 bisa menjalankan fungsi WiFi dan Web Server.

- ESP8266WiFi.h: Mengatur koneksi WiFi.
- ESP8266WebServer.h: Mengatur server web di ESP8266.

b. Variabel dan Pin

```

const char* ssid = "Laniisa";
const char* password = "lanii2207";
int LED1 = D0;
int LED2 = D1;
int LED3 = D4;

```

ESP8266WebServer Server(80);

String webpage;

- ssid dan password: Nama dan sandi WiFi yang akan dibuat oleh ESP8266 (Access Point).
- LED1, LED2, LED3: Pin digital yang digunakan untuk mengontrol LED.
- Server(80): Membuat web server pada port HTTP standar (port 80).
- webpage: Variabel untuk menyimpan isi halaman web (HTML).

c. Fungsi setup()

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  delay(10);
```

- Serial.begin(9600): Mengaktifkan komunikasi serial untuk debugging (melalui Serial Monitor).

```
  pinMode(LED1, OUTPUT);
```

```
  pinMode(LED2, OUTPUT);
```

```
  pinMode(LED3, OUTPUT);
```

- Mengatur ketiga pin LED sebagai output.

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

```
  Serial.print("Configuring access point,,");
```

- Menampilkan pesan pada Serial Monitor sebagai penanda bahwa ESP sedang membuat access point.

```
  WiFi.mode(WIFI_AP);
```

```
  WiFi.begin(ssid, password);
```

- Mengatur ESP ke mode Access Point dan memulai WiFi dengan SSID dan password yang sudah ditentukan.

```
  Serial.print("Access Point SSID: ");
```

```
  Serial.println(WiFi.softAPSSID());
```

```
  Serial.println("IP address: ");
```

```
  Serial.println(WiFi.softAPIP());
```

- Menampilkan nama SSID dan alamat IP lokal dari access point ke Serial Monitor.

d. Membuat Halaman Web

```
  webpage+=" <h1> Web Control ESP8266</h1>";
```

```
  webpage+=" <h2> SOLANI </h2>";
```

- Menambahkan judul dan nama ke halaman web.

```

webpage+= "<p>LED 1 : ";
webpage+= "      <a      href=\"LED1ON\"><button>ON</button></a><a
href=\"LED1OFF\"><button>OFF</button></a></p><br>";

```

- o Menambahkan tombol ON dan OFF untuk LED1. Tombol ini mengarahkan ke link seperti /LED1ON dan /LED1OFF.

e. Routing URL Server

```

Server.on("/", []() {
  Server.send(200, "text/html", webpage);
});

```

- o Ketika user mengakses root URL (/), server akan mengirim halaman web (webpage) sebagai response HTML.

```

Server.on("/LED1ON", []() {
  Server.send(200, "text/html", webpage);
  digitalWrite(LED1, HIGH);
  delay(1000);
});

```

- o Ketika user membuka URL /LED1ON, LED1 akan dinyalakan, dan halaman web dikirim kembali ke browser.

```

Server.on("/LED1OFF", []() {
  Server.send(200, "text/html", webpage);
  digitalWrite(LED1, LOW);
  delay(1000);
});

```

- o Saat URL /LED1OFF dibuka, LED1 dimatikan.

f. Menjalankan Server

```

Server.begin();
Serial.println("Webserver dijalankan");

```

- o Memulai server web dan menampilkan pesan di Serial Monitor bahwa web server sudah aktif.

g. Fungsi loop()

```

void loop() {
  Server.handleClient();
}

```

- o Fungsi ini akan terus berjalan untuk melayani permintaan dari klien (browser). Jika ada yang membuka halaman atau klik tombol, akan diproses di sini.

3. Dokumentasi

