

```
//projeto 2 semestre, projeto ashton
```

```
#include <ESP8266WiFi.h>
```

```
//-----colocar aqui rede-----
```

```
const char* ssid    = "sua-rede";
```

```
const char* password = "sua-senha";
```

```
//-----
```

```
WiFiServer server(80);
```

```
String header;
```

```
//ESTADO DOS LEDS
```

```
String garagem = "Desligado";
```

```
String firstFloor = "Desligado";
```

```
String secondFloor= "Desligado";
```

```
unsigned long currentTime = millis();
```

```
unsigned long previousTime = 0;
```

```
const long timeoutTime = 2000
```

```
//FUNCOES RESPONSAVEIS PARA INICIAR O SERVIDOR DO ESP
```

```
void setup() {
```

```
    Serial.begin(115200);
```

```
    delay(5000);
```

```
    Serial.print("Conectando a rede ");
```

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

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

```
    while (WiFi.status() != WL_CONNECTED) {
```

```
        delay(500);
```

```
        Serial.print(".");
```

```
    }
```

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

```
    Serial.println("WiFi conectado.");
```

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

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

```
    server.begin();
```

```
}
```

```
//-----
```

```
void loop() {
```

```
// Aguardando o cliente se conectar no servidor
```

```
    WiFiClient client = server.available();
```

```
    if (client) {
```

```
        String currentLine = "";
```

```
        currentTime = millis();
```

```
        previousTime = currentTime;
```

```
        while (client.connected() && currentTime - previousTime <= timeoutTime) {
```

```

currentTime = millis();
if (client.available()) {
    char c = client.read();
    header += c;
    if (c == '\n') {

//cliente conectado entao bora gerar uma pagina html-----
        if (currentLine.length() == 0) {
            client.println("HTTP/1.1 200 OK");
            client.println("Content-type:text/html");
            client.println("Connection: close");
            client.println();

//-----
//Vamos entao ver o texto recebido do servidor e armazenar em uma string, já que a
//comunicação serial aconteceu com str
            //LED DA GARAGEM
            if (header.indexOf("GET /13/on") >= 0) {
                Serial.println("GARAGEM_ON");           //Comando que vai ser enviado para o
arduinoMega via serial
                garagem = "Ligado";
            } else if (header.indexOf("GET /13/off") >= 0) {
                Serial.println("GARAGEM_OFF"); //Comando que vai ser enviado para o
arduinoMega via serial
                garagem = "Desligado";
            }

            //LED FIRSTFLOOR
            if (header.indexOf("GET /12/on") >= 0) {
                Serial.println("FIRSTFLOOR_ON");           //Comando que vai ser enviado para o
arduinoMega via serial
                firstFloor = "Ligado";
            } else if (header.indexOf("GET /12/off") >= 0) {
                Serial.println("FIRSTFLOOR_OFF");           //Comando que vai ser enviado para o
arduinoMega via serial
                firstFloor = "Desligado";
            }

            //LED SECONDFLOOR
            if (header.indexOf("GET /11/on") >= 0) {
                Serial.println("SECONDFLOOR_ON");           //Comando que vai ser enviado
para o arduinoMega via serial
                secondFloor = "Ligado";
            } else if (header.indexOf("GET /11/off") >= 0) {
                Serial.println("SECONDFLOOR_OFF");           //Comando que vai ser enviado para o
arduinoMega via serial
            }

```

```
        secondFloor = "Desligado";
    }
}
```

//FIM DOS GET'S Agora vamos mandar o feedback de volta para o cliente :D

```
    client.println("<!DOCTYPE html><html>");
    client.println("<head><meta name=\"viewport\" content=\"width=device-width,
initial-scale=1\">");
    client.println("<link rel=\"icon\" href=\"data:;\">");
```

```
    // CSS para estilizar a pagina
    client.println("<style>html { font-family: Helvetica; display: inline-block; margin: 0px
auto; text-align: center;};");
    client.println("h1,p {font-weight: bold;color: #126e54; font-size: 32px;});");
    client.println("p {font-size: 16px;});");
    client.println(".button { background-color: #1BAE85; border: none; color: white;
padding: 16px 40px;}); //Estiliza o botao
    client.println("text-decoration: none; font-size: 30px; margin: 2px; cursor: pointer;});");
    client.println("</style></head>");
```

```
    client.println("<body><h1>Projeto Ashton</h1>"); //TITULO DA PAGINAAA
```

```
//-----
-----
```

//Alterar os botoes da garagem

```
    client.println("<p>Pino 13 - Estado " + garagem + "</p>"); //aqui vamos mostrar
como esta o botao do pino 13 (tirar isso depois dos testes.)
    if (garagem == "Desligado") {
        client.println("<p><a href=\"/13/on\"><button
class=\"button\">Ligar</button></a></p>");
    } else {
        client.println("<p><a href=\"/13/off\"><button class=\"button
button2\">Desligar</button></a></p>");
    }
}
```

//Alterar os botoes do primeiro andar

```
    client.println("<p>Pino 12 - Estado " + firstFloor + "</p>"); //aqui vamos mostrar
como esta o botao do pino 13 (tirar isso depois dos testes.)
    if (firstFloor == "Desligado") {
        client.println("<p><a href=\"/12/on\"><button
class=\"button\">Ligar</button></a></p>");
    } else {
        client.println("<p><a href=\"/12/off\"><button class=\"button
button2\">Desligar</button></a></p>");
    }
}
```

```

//Alterar os botoes do segundo andar
    client.println("<p>Pino 11 - Estado " + secondFloor + "</p>"); //aqui vamos mostrar
como esta o botao do pino 13 (tirar isso depois dos testes.)
    if (secondFloor == "Desligado") {
        client.println("<p><a href=\\\"/11/on\\\"><button
class=\\\"button\\\">Ligar</button></a></p>");
    } else {
        client.println("<p><a href=\\\"/11/off\\\"><button class=\\\"button
button2\\\">Desligar</button></a></p>");
    }

//-----
-----

    client.println("</body></html>");
    // A resposta HTTP termina com outra linha em branco.
    client.println();
    break;
} else { // Se você recebeu uma nova linha, limpe currentLine
    currentLine = "";
}
} else if (c != '\\r') {
    currentLine += c;
}
}
}
header = "";
client.stop();
}
}

```