/ Escrito por Renan Romão Oliveira e Regimar Francisco dos Santos

// Uso permitido para quaisquer fins, modifique e divirta-se =)

#include <WiFi.h>

#include "DHT.h"

#define DHTPIN 19

#define DHTTYPE DHT11

// Pinos dos leds de conexão, erro e sucesso

#define CONNLED 21

#define ERRORLED 22

#define SUCCESSLED 23

const char\* ssid = "Casanet";

const char\* pwd = "b@uru735";

IPAddress server(192,168,0,100); // Substitua pelo IP do seu servidor

WiFiClient client;

DHT dht(DHTPIN, DHTTYPE);

float u, t;

void setup() {

 dht.begin();

 Serial.begin (115200);

 WiFi.begin(ssid, pwd);

 // Define o modo de operação dos pinos ligados aos LED como saída

 pinMode(CONNLED, OUTPUT);

 pinMode(ERRORLED, OUTPUT);

 pinMode(SUCCESSLED, OUTPUT);

 // Enquando estiver tentando se conectar com a rede, faz o LED amarelo piscar

 while (WiFi.status() != WL\_CONNECTED){

 digitalWrite(CONNLED, HIGH);

 delay(250);

 digitalWrite(CONNLED, LOW);

 delay(250);

 }

 digitalWrite(CONNLED, LOW);

}

void loop() {

 u = dht.readHumidity();

 Serial.print("hUMIDADE: ");

 Serial.print(u);

 Serial.print("%\t");

 t = dht.readTemperature();

 if(isnan(u) || isnan(t)){

 error();

 }else{

 String response = "";

 if(client.connect(server, 80)) {

 // Acende o LED amarelo indicando uma conexão com o servidor

 digitalWrite(CONNLED, HIGH);

 client.printf("GET /monit/dbotest.php?u=%f2.2&t=%f2.2&key=1234 HTTP/1.1\r\n", u, t);

 client.println("Host: 192.168.0.100:80");

 client.println("Connection: close");

 client.println();

 }

 delay(1000); // Espera um pouco pela resposta do servidor

 // Armazena a resposta do servidor na variável

 while (client.available()) {

 char c = client.read();

 response += c;

 }

 if (!client.connected()) {

 // Apaga o LED amarelo indicando o fim de uma conexão com o servidor

 digitalWrite(CONNLED, LOW);

 // Verifica se existe a palavra sucesso na resposta do servidor

 // em caso positivo acende o LED verde, negativo o vermelho

 if(response.indexOf("sucesso") >= 0)

 success();

 else

 error();

 client.stop();

 }

 // Espera 5 minutos (1800000 milissegundos)

 delay(300000);

 }

}

// Métodos para acender e apagar os LEDs verde e vermelho indicando sucesso ou erro

void success(){

 digitalWrite(SUCCESSLED, HIGH);

 digitalWrite(ERRORLED, LOW);

}

void error(){

 digitalWrite(ERRORLED, HIGH);

 digitalWrite(SUCCESSLED, LOW);

}