17520258 Vũ Hà Anh

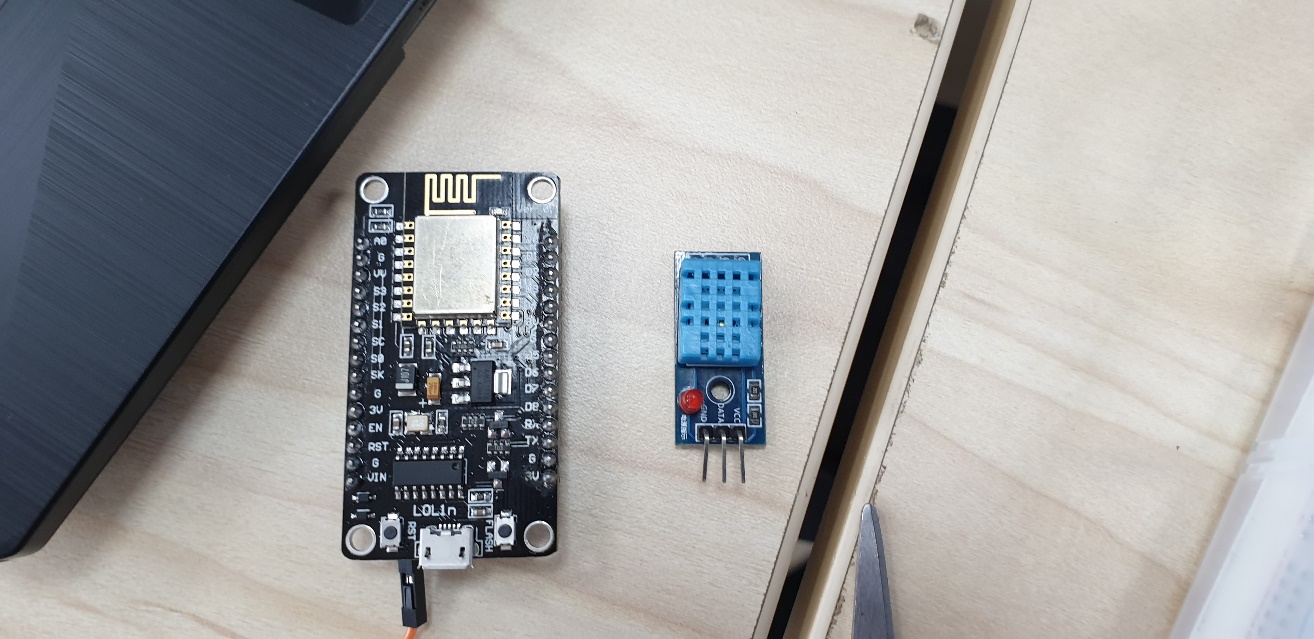
17520860 Phạm Thùy Nhung

18521476 Nguyễn Phương Thùy

**BÁO CÁO BÀI TẬP LỚN SỐ 4**

Sử dụng giao thức MQTT để giao tiếp giữa Client và Server. Host được sử dụng ở đây là: broker.hivemq.com , port 8000, tcp port: 1883.

**Thiết bị sử dụng:** ESP8266 NodeMCU, sensor DHT 11.



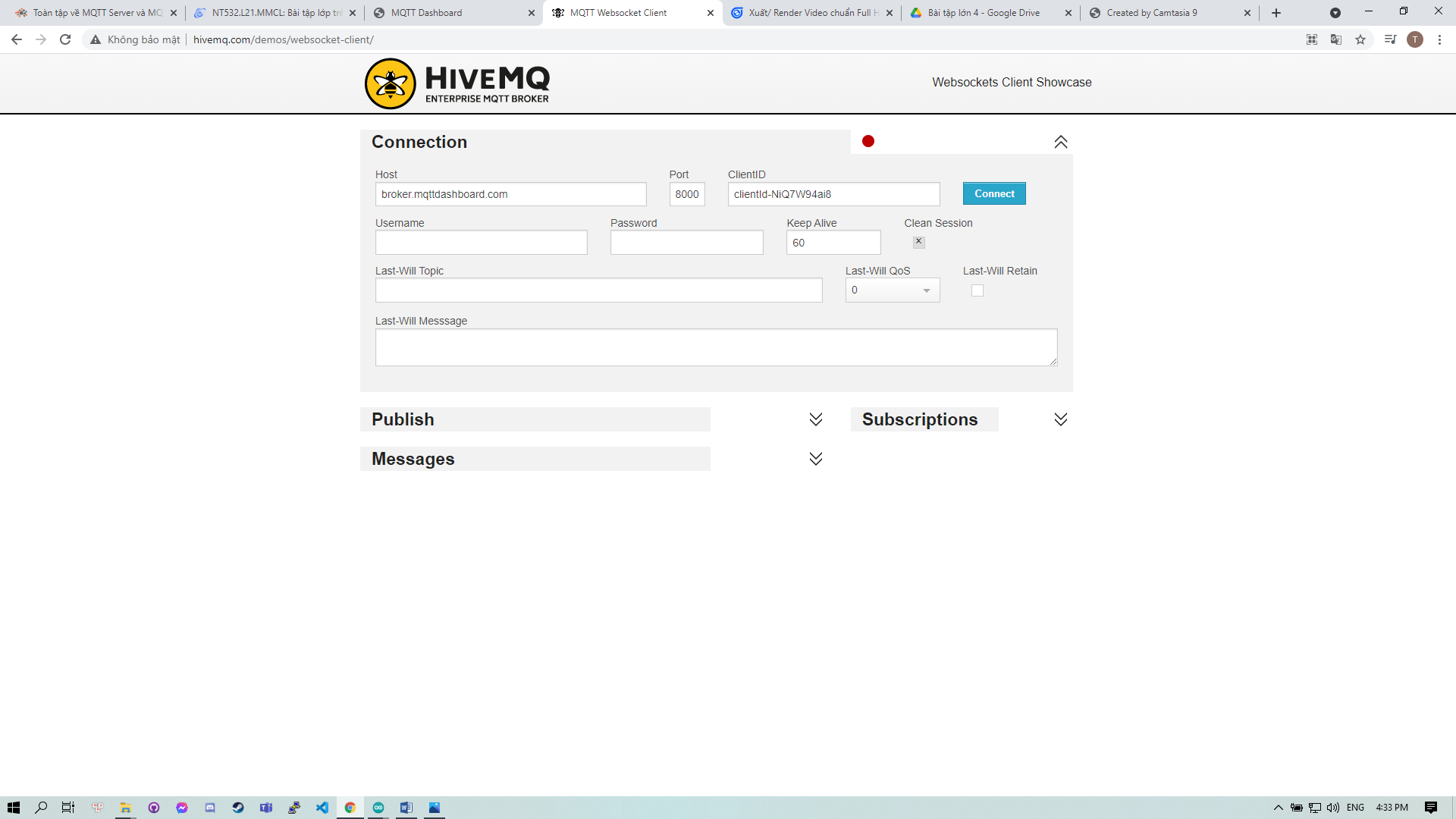
Video Demo: https://drive.google.com/file/d/1\_7XX9qmiCjh7KwUg\_h2YBmexyYKgNUDu/view?usp=sharing

**Các bước thiết lập:**

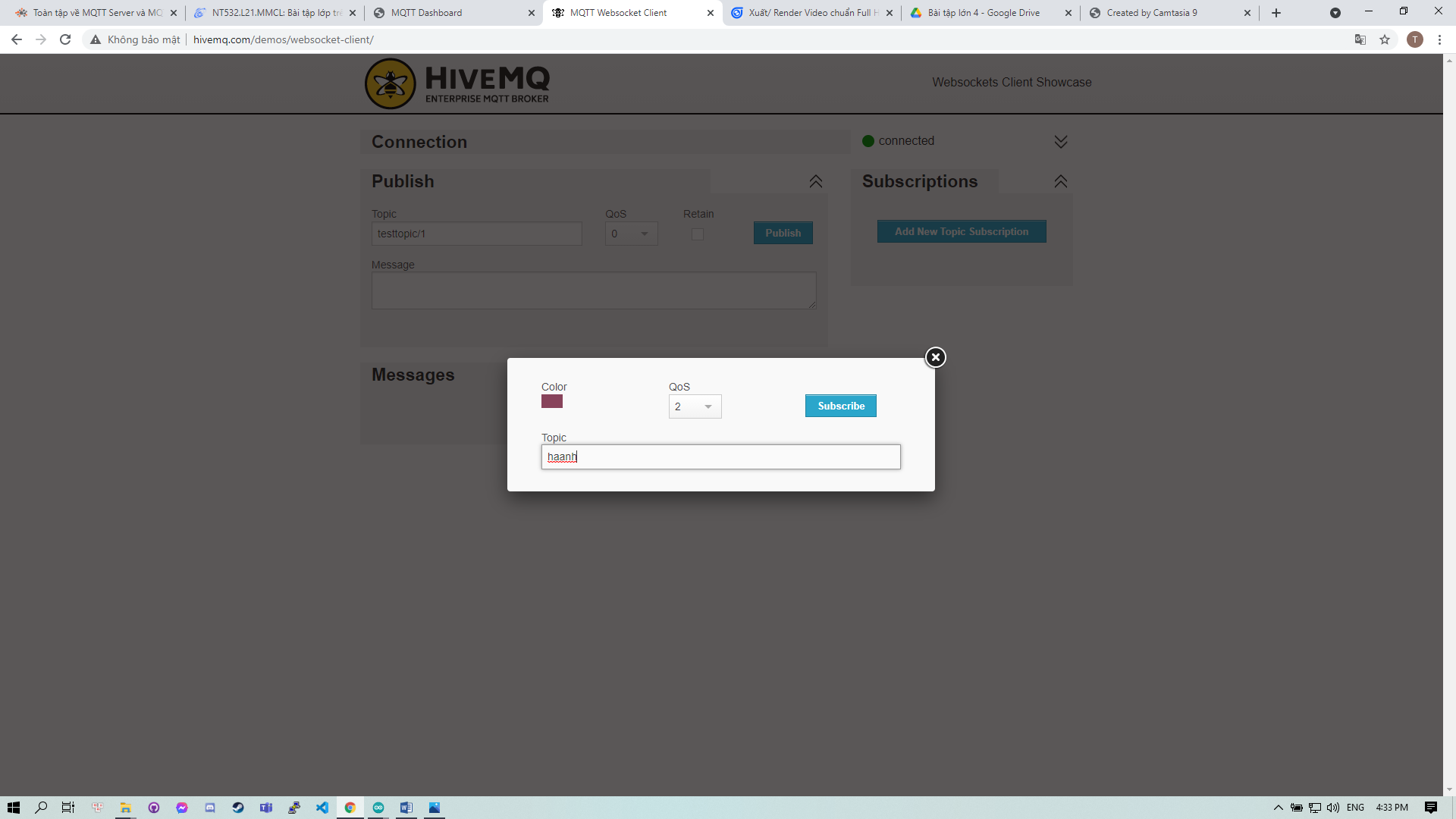
**1. Trên Cloud** [**http://www.hivemq.com/demos/websocket-client/**](http://www.hivemq.com/demos/websocket-client/)

**(Các bước thiết lập được thể hiện trong video)**

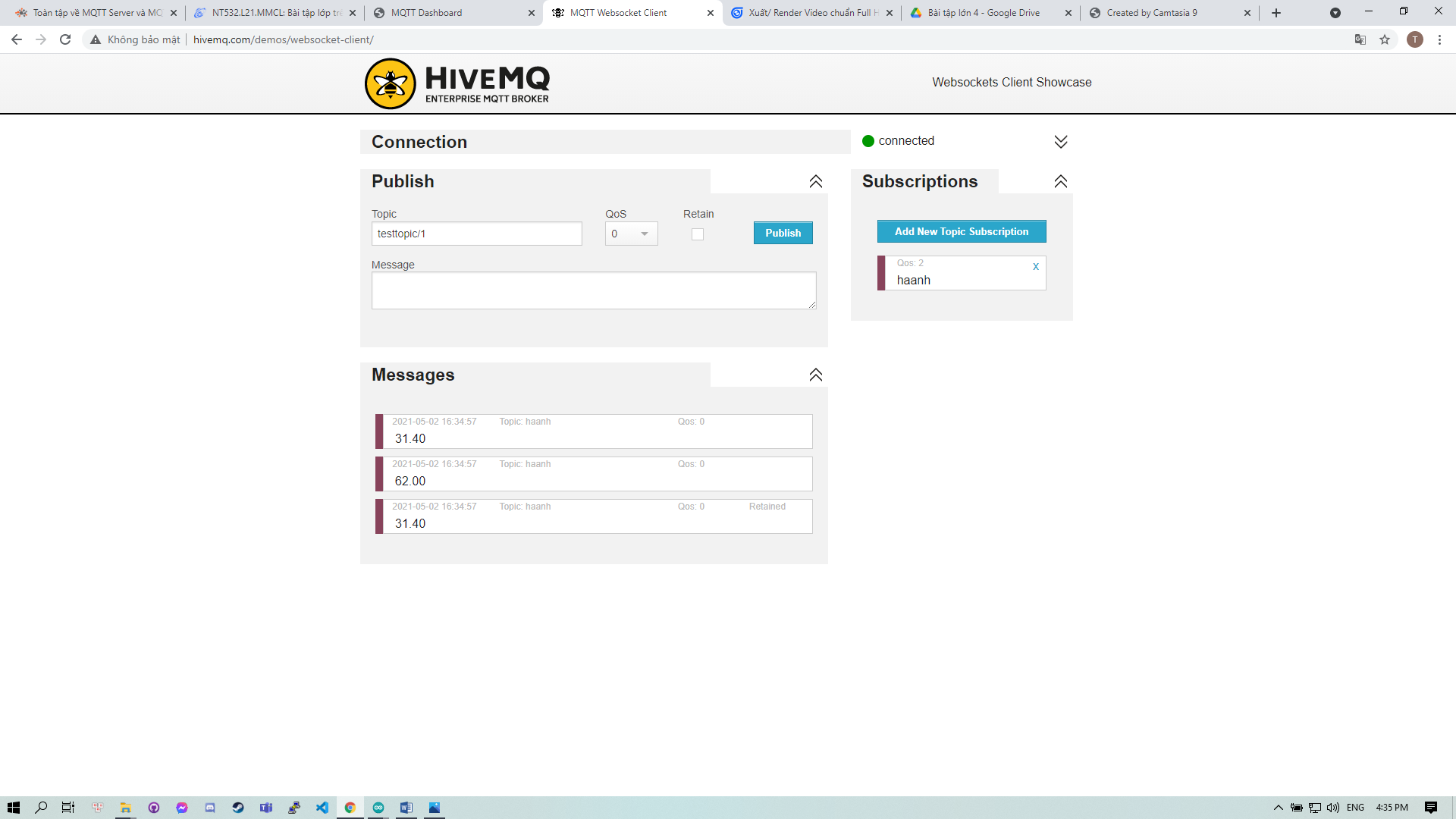
Thiết lập kênh Subcriber với thông tin như sau:



Truy cập vào Topic “haanh” để xem quá trình gửi dữ liệu



Sau khi truy cập vào kênh “haanh”, tại phần Messages sẽ là thông tin được gửi từ Sensor lên



**2. Thiết lập trên ESP8266**

Nạp code sau vào thiết bị, sau mỗi 5 giây, Server sẽ nhận được thông tin độ ẩm và nhiệt độ từ sensor gửi đến.

#include <ESP8266WiFi.h>

#include <Wire.h>

#include <PubSubClient.h>

#include "DHT.h"

#define DHTPIN 5

//

#define DHTTYPE DHT11 // khai báo sensor

DHT dht(DHTPIN, DHTTYPE);

#define wifi\_ssid "HaAnh"

#define wifi\_password "khongbiet"

#define mqtt\_server "broker.hivemq.com"

//Tên topic cần truyền thông tin

const char\* m\_topic = "haanh";

WiFiClient espClient;

PubSubClient client(espClient);

void setup() {

Serial.begin(115200);

dht.begin();

setup\_wifi();

client.setServer(mqtt\_server, 1883);

}

String macToStr(const uint8\_t\* mac)

{

String result;

for (int i = 0; i < 6; ++i) {

result += String(mac[i], 16);

if (i < 5)

result += ':';

}

return result;

}

// Khởi tạo kết nối wifi (dựa vào thư viện ESP8266 wifi)

void setup\_wifi() {

delay(10);

// We start by connecting to a WiFi network

Serial.println();

Serial.print("Connecting to ");

Serial.println(wifi\_ssid);

WiFi.begin(wifi\_ssid, wifi\_password);

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

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

}

void reconnect() {

// Loop until we're reconnected

while (!client.connected()) {

Serial.print("Attempting MQTT connection...");

String clientName;

clientName += "esp8266-";

uint8\_t mac[6];

WiFi.macAddress(mac);

clientName += macToStr(mac);

clientName += "-";

clientName += String(micros() & 0xff, 16);

Serial.print("Connecting to ");

Serial.print(mqtt\_server);

Serial.print(" as ");

Serial.println(clientName);

if (client.connect((char\*) clientName.c\_str())) {

Serial.println("connected");

} else {

Serial.print("failed, rc=");

Serial.print(client.state());

Serial.println(" try again in 5 seconds");

// Wait 5 seconds before retrying

delay(5000);

}

}

}

void loop() {

if (!client.connected()) {

reconnect();

}

client.loop();

// Wait a few seconds between measurements.

delay(2000);

// Reading temperature or humidity takes about 250 milliseconds!

// Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)

float h = dht.readHumidity();

// Read temperature as Celsius (the default)

float t = dht.readTemperature();

// Read temperature as Fahrenheit (isFahrenheit = true)

float f = dht.readTemperature(true);

// Check if any reads failed and exit early (to try again).

if (isnan(h) || isnan(t) || isnan(f)) {

Serial.println("Failed to read from DHT sensor!");

return;

}

// Compute heat index in Fahrenheit (the default)

float hif = dht.computeHeatIndex(f, h);

// Compute heat index in Celsius (isFahreheit = false)

float hic = dht.computeHeatIndex(t, h, false);

Serial.print("Humidity: ");

Serial.print(h);

Serial.print(" %\t");

Serial.println("Temperature: ");

Serial.print(t);

Serial.print(" %\t");

client.publish(m\_topic, String(h).c\_str(), true);

client.publish(m\_topic, String(t).c\_str(), true);

delay(5000);

}

Thông tin gửi đi :

