17520258 Vũ Hà Anh

17520860 Phạm Thùy Nhung

18521476 Nguyễn Phương Thùy

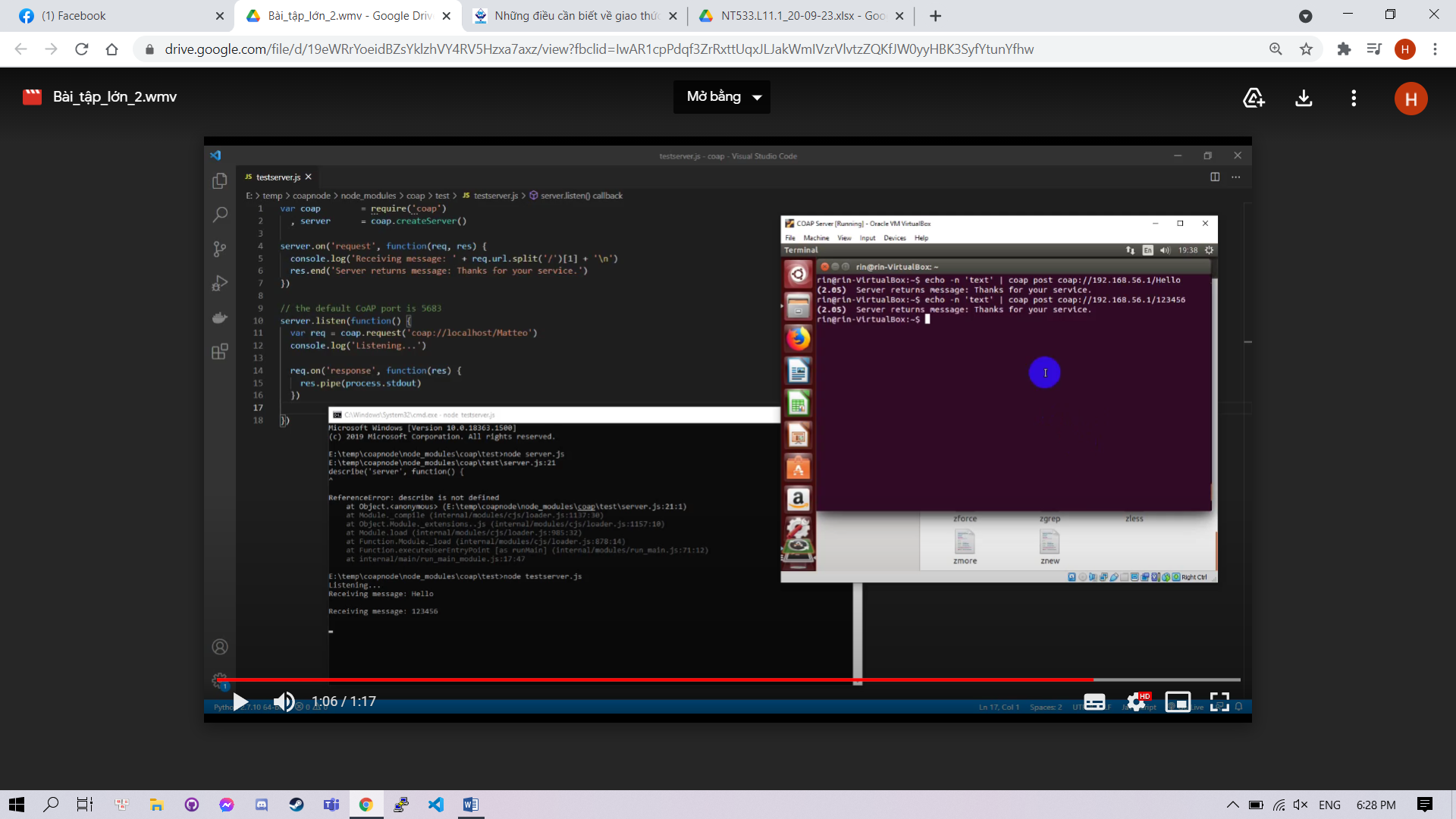
**BÁO CÁO BÀI TẬP LỚN SỐ 2 VÀ BÀI TẬP LỚN SỐ 3**

**BÀI TẬP LỚN SỐ 2**

Bài tập CoAP, gửi tin nhắn giữa client và server bằng giao thức CoAP được viết bằng nodejs. Link video : <https://drive.google.com/file/d/19eWRrYoeidBZsYklzhVY4RV5Hzxa7axz/view?fbclid=IwAR1cpPdqf3ZrRxttUqxJLJakWmIVzrVlvtzZQKfJW0yyHBK3SyfYtunYfhw>

Máy vật lý (Windows) đóng vai trò server.

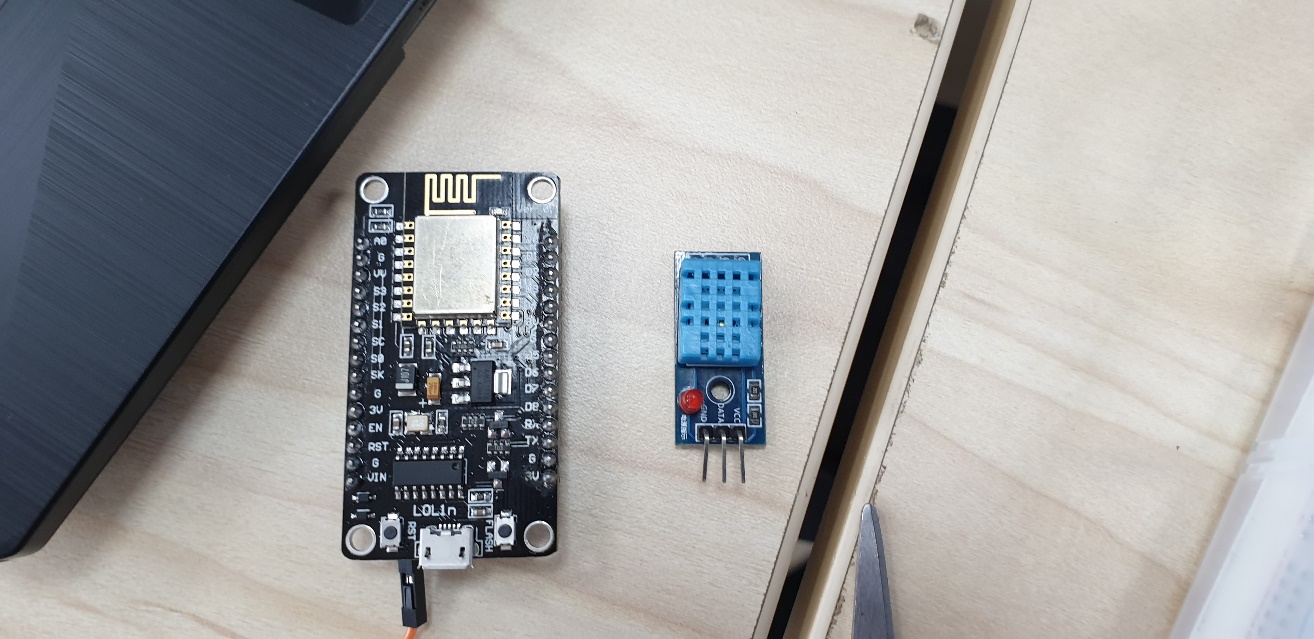
Máy ảo (Ubuntu) đóng vai trò Client.



**BÀI TẬP LỚN SỐ 3**

Vì không thể cấu hình được Bluetooth nên em thiết lập Raspberry thành gateway qua giao thức MQTT.

**Thiết bị sử dụng:** Raspberry pi 3 model A, ESP8266 NodeMCU, sensor DHT 11.





Video Demo: https://drive.google.com/drive/folders/1aCCyIdxMpg3FHXxjD5NG\_NA9RHEmmESA?usp=sharing

**Các bước thiết lập:**

**1. Thiết lập trên Raspberry (Gateway)**

Cài đặt Mosquitto server

sudo apt install -y mosquitto mosquitto-clients

Khởi động Server bằng lệnh sau

sudo systemctl enable mosquitto.service

Kiểm tra xem trạng thái hoạt động của Server

systemctl status mosquitto.service

Đầu tiên tạo một topic mới tên haanh

mosquitto\_sub -h localhost -t “haanh”

**2. Thiết lập trên ESP8266**

Nạp code sau vào thiết bị, sau mỗi 7 giây, Server sẽ nhận được thông tin độ ẩm từ sensor gửi đến.

#include <ESP8266WiFi.h>

#include <Wire.h>

#include <PubSubClient.h>

#include "DHT.h"

#define DHTPIN 12

//

#define DHTTYPE DHT11 // khai báo sensor

DHT dht(DHTPIN, DHTTYPE);

//Khai báo thông tin Wifi và địa chỉ của Raspberry

#define wifi\_ssid "asdfghjkl"

#define wifi\_password "yyyyyyyy"

#define mqtt\_server "192.168.137.162"

//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.print("Humidity:");

Serial.println(String(h).c\_str());

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

delay(7000);

}