

**Học viện Công nghệ Bưu chính Viễn thông**  
**Khoa Kỹ thuật điện tử 1**

---



**Bài tiểu luận cuối kỳ:**  
**HỌC PHẦN: IOT VÀ ỨNG DỤNG**  
**Chủ đề: HỆ THỐNG NHÀ THÔNG MINH**

***Họ và tên:*** Lê Trung Thành

***Mã sinh viên:*** B21DCAT176

***Giảng viên hướng dẫn:*** TS. Nguyễn Quốc Uy

*Hà Nội, 09/2024*

## Mục lục

<b>CHƯƠNG I. TỔNG QUAN.....</b>	<b>4</b>
1. Lý do chọn chủ đề.....	4
2. Mục tiêu nghiên cứu .....	4
3. Phạm vi nghiên cứu .....	4
4. Đối tượng nghiên cứu .....	4
5. Phương pháp thực hiện .....	4
6. Kết quả mong đợi của đề tài .....	5
<b>CHƯƠNG II. CÁC THÀNH PHẦN CHI TIẾT CỦA HỆ THỐNG .....</b>	<b>6</b>
1. Phần cứng.....	6
<b>1.1. ESP32.....</b>	<b>6</b>
a) Tổng quan .....	6
b) Sơ đồ chân.....	6
c) Thông số kĩ thuật .....	6
<b>1.2. DHT11 .....</b>	<b>7</b>
a) Tổng quan .....	7
b) Sơ đồ chân DHT11 .....	7
c) Thông số kĩ thuật .....	7
<b>1.3. Dây breadboard .....</b>	<b>8</b>
<b>1.4. Đèn LED .....</b>	<b>8</b>
2. Phần mềm .....	8
<b>2.1. Arduino IDE 2.0.....</b>	<b>8</b>
a) Tổng quan .....	9
b) Hình ảnh thực tế.....	9
<b>CHƯƠNG III. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG .....</b>	<b>10</b>
1. Mô hình chung của hệ thống.....	10
2. Thiết kế hệ thống .....	10
ii) Đèn LED .....	10

3.	Cách thức hoạt động .....	11
<b>3.1.</b>	<b>Chế độ điều khiển tự động (phụ thuộc vào dữ liệu của cảm biến).....</b>	<b>11</b>
a)	Quạt.....	11
b)	Đèn LED .....	11
<b>3.2.</b>	<b>Chế độ điều khiển thủ công .....</b>	<b>12</b>
a)	Quạt.....	12
b)	Đèn LED .....	12
4.	Thiết kế giao diện và điều khiển.....	12
<b>CHƯƠNG IV. MÃ NGUỒN DỰ ÁN VÀ KẾT QUẢ.....</b>		<b>13</b>
1.	Kết quả đạt được .....	13
<b>1.1.</b>	<b>Phần cứng.....</b>	<b>13</b>
<b>1.2.</b>	<b>Phần mềm.....</b>	<b>14</b>
a)	Trên ESP32 .....	14
b)	Trên web .....	14
3.	Thuận lợi.....	14
4.	Khó khăn.....	14

## **CHƯƠNG I. TỔNG QUAN**

### **1. Lý do chọn chủ đề**

- Cuộc cách mạng công nghệ ngày càng có những bước tiến mạnh mẽ, góp phần nâng cao chất lượng cuộc sống của con người và lợi ích xã hội, tiêu biểu đó chính là ứng dụng thiết bị điều khiển từ xa cho hệ thống thiết bị điện trong gia đình.
- IoT – Internet of thing, cũng được coi là một thành quả lớn trong cuộc cách mạng công nghệ, bởi nó là sự kết hợp của các thiết bị điện tử và mạng không dây, khiến các đồ vật có thể dễ dàng kết nối với mạng Internet không dây và được điều khiển tự động hoặc thủ công bởi con người.
- Từ đó, em đã ứng dụng IoT để thiết kế mô hình nhà thông minh – smart home, sử dụng module ESP32 để đưa ra những ứng dụng thiết thực, có thể vận dụng trong thực tế.

### **2. Mục tiêu nghiên cứu**

- Mục tiêu em đề ra là thiết kế được một hệ thống nhà thông minh đơn giản, sử dụng module ESP32 có giao thức kết nối không dây WiFi để thu thập dữ liệu và hiển thị chúng trên thiết bị di động. Cùng với đó, hệ thống có khả năng điều khiển các thiết bị tự động thông qua dữ liệu, và thủ công bằng thiết bị di động hoặc trình duyệt web. Cuối cùng, các dữ liệu thu về được biểu diễn và phân tích dưới dạng đồ thị, biểu đồ.

### **3. Phạm vi nghiên cứu**

- Hệ thống sử dụng module ESP và các cảm biến, thiết bị ngoại vi khác có khả năng nhận dữ liệu và gửi tín hiệu điều khiển từ xa đến các thiết bị điện được kết nối với mạng không dây Wifi trong nhà.

### **4. Đối tượng nghiên cứu**

- Nghiên cứu cách ESP giao tiếp không dây và truyền nhận dữ liệu. Ứng dụng web server và điện thoại để kết nối hiển thị thông tin; điều khiển bật/tắt các thiết bị điện tử thủ công và tự động.

### **5. Phương pháp thực hiện**

- Tìm hiểu về ESP32
- Nghiên cứu các sensor DHT11

## ***6. Kết quả mong đợi của đề tài***

- Hệ thống module, các sensor và các thiết bị ngoại vi kết nối được với nhau và truyền nhận thông tin dữ liệu để hiển thị hoặc điều khiển các thiết bị

## CHƯƠNG II. CÁC THÀNH PHẦN CHI TIẾT CỦA HỆ THỐNG

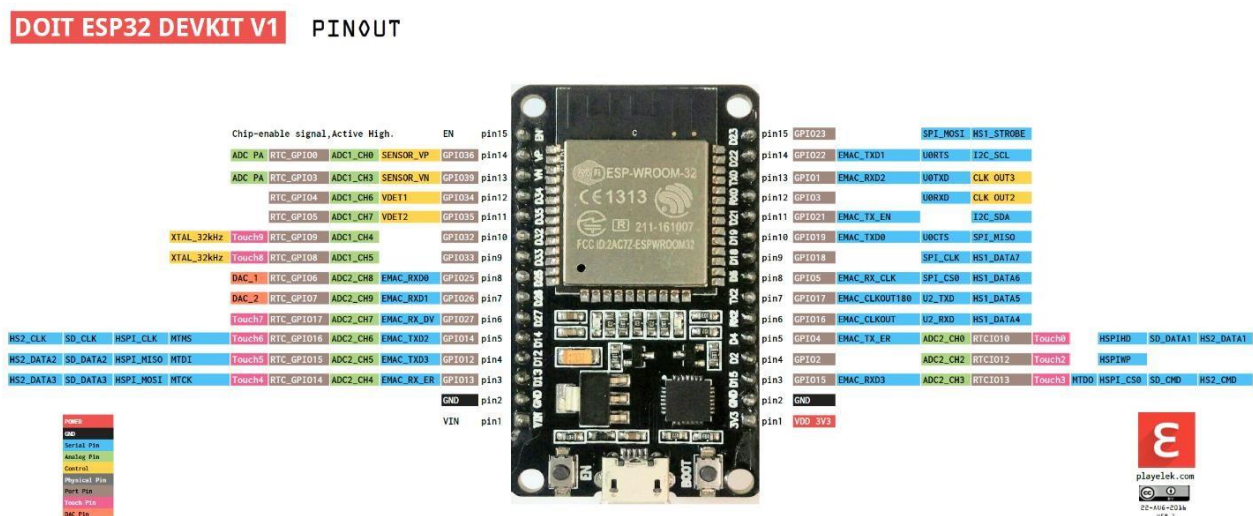
### 1. Phần cứng

#### 1.1. ESP32

##### a) Tổng quan

- ESP32 là một series các vi điều khiển trên một vi mạch giá rẻ, năng lượng thấp có hỗ trợ WiFi và dual-mode Bluetooth. Dòng ESP32 sử dụng bộ vi xử lý Tensilica Xtensa LX6 ở cả hai biến thể lõi kép và lõi đơn, và bao gồm các công tắc antenna tích hợp, RF balun, bộ khuếch đại công suất, bộ khuếch đại thu nhiễu thấp, bộ lọc và module quản lý năng lượng. [1]

##### b) Sơ đồ chân



Hình 1. Sơ đồ chân ESP32

##### c) Thông số kỹ thuật

- Tên Module: Wifi BLE ESP32 NodeMCU LuaNode32
- Nguồn sử dụng: 5VDC từ cổng Micro USB
- Tích hợp mạch nạp và giao tiếp UART CP2102
- Ra chân đầy đủ module ESP32, chuẩn chân cắm 2.54mm
- Tích hợp Led Status, nút BOOT và ENABLE
- Kích thước: 28.33x51.45mm [2]

## 1.2. DHT11

### a) Tổng quan

- DHT11 là một cảm biến kỹ thuật số giá rẻ để cảm nhận nhiệt độ và độ ẩm. Cảm biến này có thể dễ dàng giao tiếp với bất kỳ bộ vi điều khiển nào như Arduino, Raspberry Pi,... để đo độ ẩm và nhiệt độ ngay lập tức. DHT11 là một cảm biến độ ẩm nhiệt độ tương đối. Để đo không khí xung quanh, cảm biến này sử dụng một điện trở nhiệt và một cảm biến độ ẩm điện dung.
- Cảm biến Nhiệt độ và độ ẩm DHT11 là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1 wire (giao tiếp 1 dây truyền dữ liệu duy nhất). Bộ tiền xử lý tín hiệu tích hợp trong cảm biến giúp có được dữ liệu chính xác mà không phải qua bất kỳ tính toán nào. So với cảm biến đời mới hơn là DHT22 thì DHT11 cho khoảng đo và độ chính xác kém hơn rất nhiều.

### b) Sơ đồ chân DHT11

DHT11 gồm có 3 chân:



- (1) Vcc – Nguồn 3.5V đến 5.5V
- (2) Data – Đầu ra cả nhiệt độ và độ ẩm thông qua dữ liệu nối tiếp.
- (3) Ground – Chân nối đất.

Hình 2. Sơ đồ chân DHT11

### c) Thông số kỹ thuật

- Điện áp hoạt động: 5VDC.
- Chuẩn giao tiếp: TTL, 1 wire.
- Khoảng đo độ ẩm: 20%-80% RH sai số  $\pm 5\%$  RH.
- Khoảng đo nhiệt độ: 0-50°C sai số  $\pm 2^\circ\text{C}$ .
- Tần số lấy mẫu tối đa 1Hz (1 giây / lần).
- Kích thước: 28mm x 12mm x 10mm.

### 1.3. Dây breadboard

- Dây breadboard là dây dùng để nối giữa breadboard với các mạch khác. Nó gồm 3 loại: male to male (dương - dương); male to female (dương - âm) và female to female (âm - âm). Trong bài này em sử dụng dây cắm dương – âm.



*Hình 6. Dây nối dương – dương*

### 1.4. Đèn LED



*Hình 7. Đèn LED*

## 2. Phần mềm

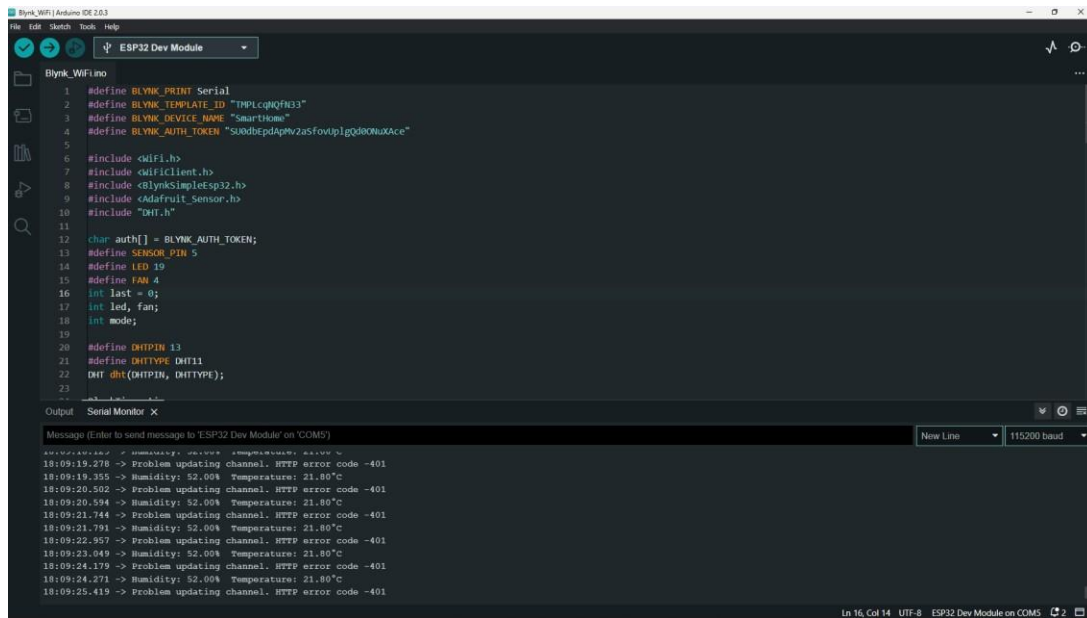
### 2.1. Arduino IDE 2.0



### **a) Tổng quan**

- Arduino IDE 2.0 là mã nguồn mở, một bước tiến lớn từ người tiền nhiệm mạnh mẽ của nó, Arduino IDE 1.x và đi kèm với UI được tân trang, Trình quản lý bảng & Thư viện cải tiến, trình gỡ lỗi, tính năng tự động hoàn chỉnh và nhiều hơn nữa.
- Chức năng chính của Arduino IDE, là giúp người dùng viết code để nạp vào bo mạch Arduino một cách nhanh chóng và dễ dàng.

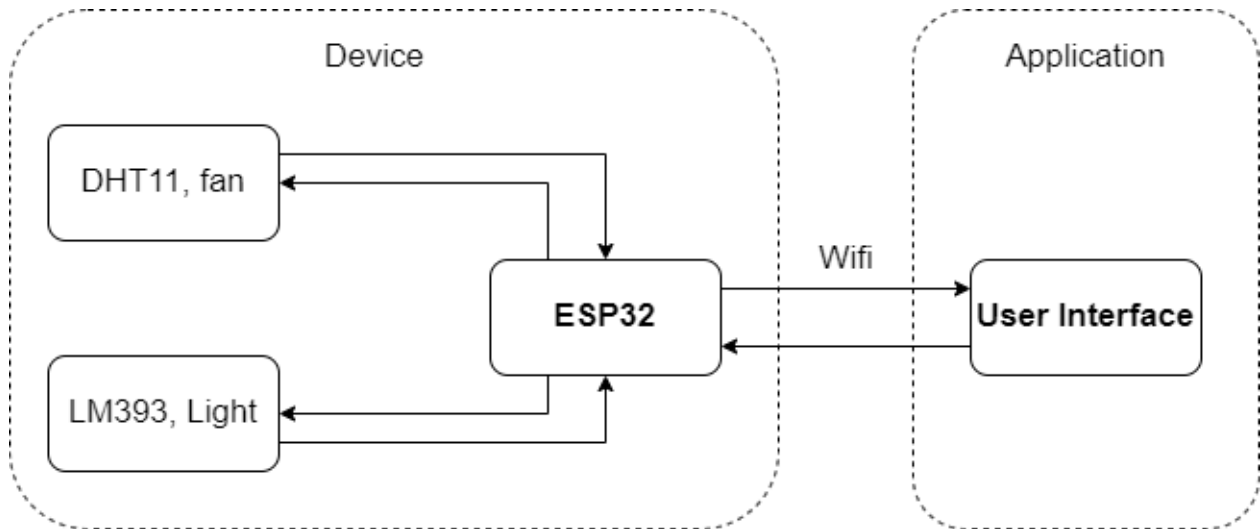
### **b) Hình ảnh thực tế**



Hình 9. Arduino IDE 2.0

### CHƯƠNG III. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

#### 1. Mô hình chung của hệ thống



Hình 10. Sơ đồ tổng quát của hệ thống

- Với mô hình tổng quát trên, em sử dụng 1 ESP32 để cảm biến nhiệt độ, độ ẩm, và điều khiển quạt; cảm biến hồng ngoại để bật tắt tự động đèn cầu thang.

#### 2. Thiết kế hệ thống

##### a) Kết nối ESP32 với cảm biến DHT11 (cảm biến nhiệt độ, độ ẩm) để bật tắt quạt

###### i) DHT11

- DHT11 có 3 chân, có chức năng và kết nối như sau:
  - Chân 1-VCC: kết nối tới chân VIN của ESP32
  - Chân 2-GND: kết nối tới chân GND của ESP32
  - Chân 3-Data: kết nối tới chân D13 (GPIO 13) của ESP32

###### ii) Đèn LED

- Đèn LED có 2 chân, kết nối như sau:
  - Chân 1-DC+: kết nối tới chân D19 (GPIO 19) của ESP32 và điện trở 120 Ohm
  - Chân 2-DC-: kết nối tới chân GND của ESP32

Hình 13. Hình ảnh kết nối thực tế ESP32 với LM393 và đèn LED

### 3. Cách thức hoạt động

#### 3.1. Chế độ điều khiển tự động (phụ thuộc vào dữ liệu của cảm biến)

##### a) Quạt

- Sau khi nhận được dữ liệu của cảm biến DHT11 từ chân D13, ESP32 bắt đầu xử lý nguồn dữ liệu.
- Nếu nhiệt độ vượt quá 30°C, ESP32 truyền tín hiệu mức cao (HIGH) đến chân D4 đang nối với relay, relay sẽ bật quạt.

##### b) Đèn LED

- Cảm biến hồng ngoại LM393 khi phát hiện ra vật cản, sẽ gửi tín hiệu mức thấp (LOW) đến chân D5 của ESP32 để ESP32 xử lý nguồn dữ liệu.
- ESP32 sẽ truyền tín hiệu mức cao (HIGH) đến chân D19 đang nối với đèn LED, đèn sẽ được bật.
- Sử dụng biến 'last' như một biến đếm timer để cài thời gian sáng cho đèn là xấp xỉ 100 giây. Sau 100 giây, đèn tự động tắt.

Hình 17. Lưu đồ thuật toán chế độ điều khiển tự động đèn

### 3.2. Chế độ điều khiển thủ công

#### a) Quạt

- Blynk có chân ảo (virtual pin) V1, có chức năng truyền tín hiệu mức cao/thấp đến chân D4 của ESP32 đang nối đến relay.
- Nếu muốn bật quạt, ấn nút 'ON' của quạt trên App Blynk, Blynk sẽ truyền tín hiệu mức cao (HIGH) đến chân D4 của ESP32, relay sẽ bật quạt.
- Quá trình trên thực hiện tương tự với thao tác tắt quạt.

#### b) Đèn LED

- Blynk có chân ảo (virtual pin) V0, có chức năng truyền tín hiệu mức cao/thấp đến chân D19 của ESP32 đang nối đến relay.
- Nếu muốn bật quạt, ấn nút 'ON' của đèn trên App Blynk, Blynk sẽ truyền tín hiệu mức cao (HIGH) đến chân D19 của ESP32, đèn sẽ được bật.
- Quá trình trên thực hiện tương tự với thao tác tắt đèn.

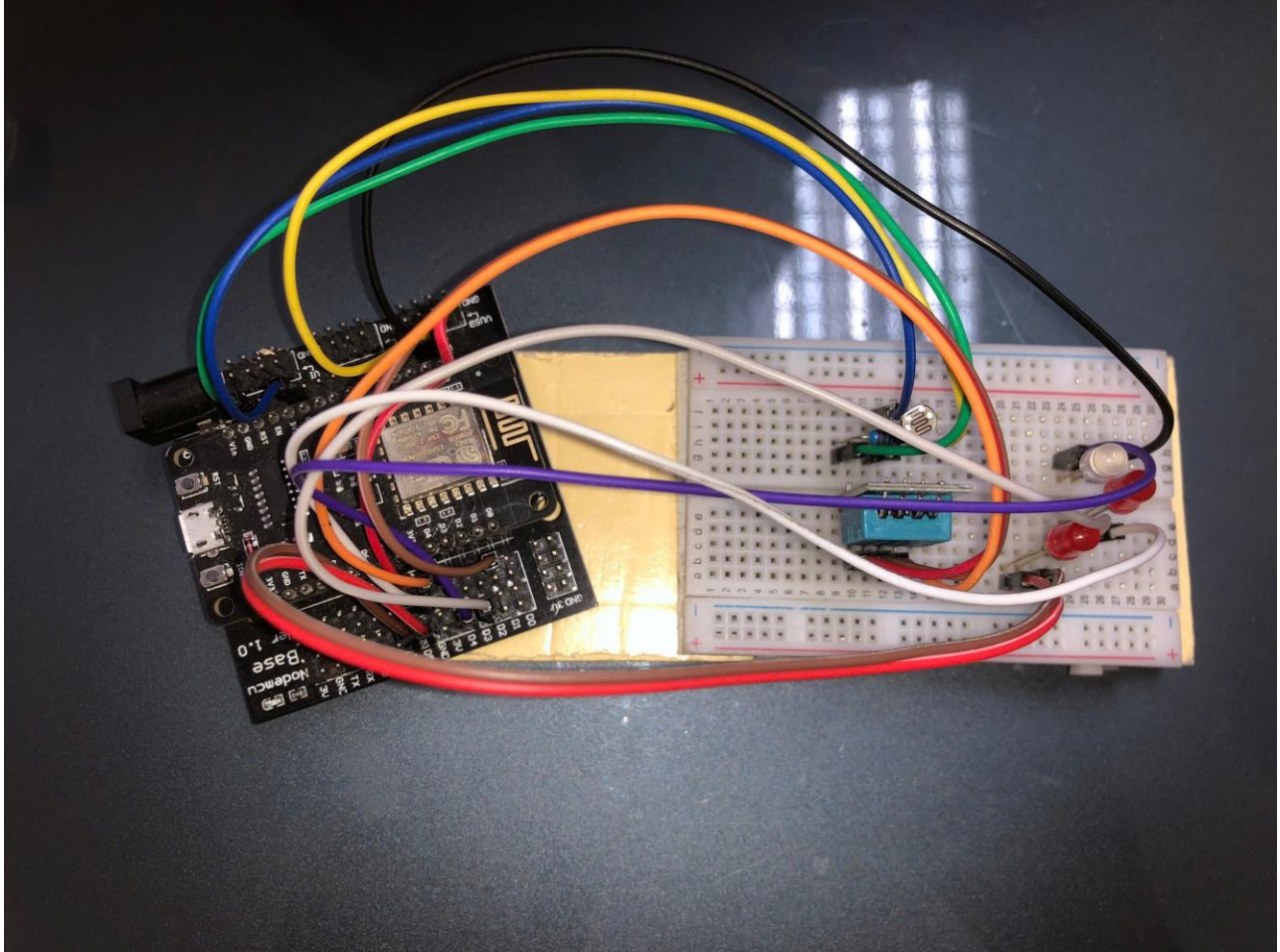
## 4. Thiết kế giao diện và điều khiển



## CHƯƠNG IV. MÃ NGUỒN DỰ ÁN VÀ KẾT QUẢ

### 1. Kết quả đạt được

#### 1.1. Phần cứng



- Sau khi phân tích và kết nối các phần của hệ thống lại với nhau, ta được một hệ thống nhà thông minh (smart home) hoàn chỉnh như trên (cá nhân em chưa có điều kiện hàn mạch cũng như trang trí lại hệ thống cho hoàn chỉnh hơn, chỉ dừng ở mức cắm các thiết bị trên bảng test board).
- Hệ thống gồm có 1 ESP32 cùng các thiết bị ngoại vi khác, bao gồm 2 cảm biến: cảm biến hồng ngoại LM393 và cảm biến nhiệt độ, độ ẩm DHT11, 2 thiết bị đầu ra gồm: quạt tản nhiệt và đèn LED.
- Để cấp nguồn cho hệ thống, em sử dụng nguồn có sẵn cấp từ cổng USB của máy tính, nối đến cổng microUSB của ESP32.

- DHT11, relay và quạt sẽ lấy nguồn 5V từ chân VIN của ESP32; LM393 và đèn LED sẽ lấy nguồn 3.3V từ chân 3V3 của ESP32.

## **1.2. Phần mềm**

### ***a) Trên ESP32***

- Áp dụng thành công sử dụng ESP32 vào việc điều khiển các thiết bị trong gia đình.
- Áp dụng Arduino IDE vào việc xây dựng chương trình và nạp chương trình cho ESP32.

### ***b) Trên web***

- Laptop cá nhân được sử dụng để làm server cho cả web với các chức năng:
  - o Điều khiển bật/tắt các thiết bị điện
  - o Hiển thị trạng thái bật/tắt của các thiết bị điện
  - o Nhận dữ liệu nhiệt độ, độ ẩm và hiển thị các dữ liệu lên màn hình web/app

## **2. Mã nguồn dự án: Link Github**

## **3. Thuận lợi**

- Vì là chủ đề đã được nghiên cứu từ rất lâu và rất nhiều dự án, đồ án, blog thảo luận về chủ đề này nên nguồn tài liệu rất dồi dào và phong phú, thuận lợi trong việc nghiên cứu ban đầu.
- Lượng kiến thức cần thiết trên lớp do giảng viên cung cấp, và do bản thân tự tiếp thu trong quá trình làm bài tập nhóm giữa kì, đã giúp ích rất nhiều cho bài tập cá nhân cuối kì.

## **4. Khó khăn**

- Chưa có nhiều kinh nghiệm tiếp xúc ESP và Arduino, thiếu nhiều kiến thức cơ bản về điện tử nên gặp khó khăn trong việc hình thành mạch điện hệ thống.
- Mất thời gian trong việc thiết lập các chế độ, chức năng đề ra, tối ưu tốc độ và cải thiện độ ổn định chương trình theo đúng mong muốn của bản thân.

