HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

IOT VÀ ỨNG DỤNG

**⎯⎯⎯⎯⎯⎯⎯⎯⎯**



**BÁO CÁO CUỐI KỲ**

**Thành viên: Lê Đức Thắng – B21DCNN664**

**Nhóm lớp học: nhóm 05**

**Giảng viên giảng dạy: Thầy Nguyễn Quốc Uy**

**HÀ NỘI, Tháng 12 - 2024**

**Mục lục**

Contents

[**I. Giới Thiệu** 1](#_Toc186161508)

[1.1. IoT (Internet of Things) 1](#_Toc186161509)

[1.1.1. Khái niệm 1](#_Toc186161510)

[1.1.2. Các thành phần của IoT 1](#_Toc186161511)

[1.1.3. Ứng dụng 1](#_Toc186161512)

[1.2. Mô tả và mục đích của đề tài 1](#_Toc186161513)

[1.2.1. Mô tả đề tài 1](#_Toc186161514)

[1.2.2. Mục đích của đề tài 2](#_Toc186161515)

[1.3. Thiết bị sử dụng 2](#_Toc186161516)

[1.3.1. ESP8266 2](#_Toc186161517)

[*1.3.2. Cảm biến nhiệt độ - độ ẩm DHT11* 3](#_Toc186161518)

[*1.3.3. Cảm biến cường độ ánh sáng quang trở* 4](#_Toc186161519)

[1.3.4. Board Test MB-102 16.5x5.5 5](#_Toc186161520)

[1.3.5. Điện trở vạch 1/4W sai số 5% 250V 1R-10M 5](#_Toc186161521)

[1.3.6. Dây nối chân các thiết bị 5](#_Toc186161522)

[1.3.7. Led 5mm, 1 màu 2 chân (1,8-3V) 6](#_Toc186161523)

[1.4. Công nghệ sử dụng 6](#_Toc186161524)

[1.4.1. Trình biên dịch Arduino IDE 6](#_Toc186161525)

[*1.4.2. Frontend: HTML* 7](#_Toc186161526)

[1.4.3 Backend: Spring Boot 7](#_Toc186161527)

[*1.4.4 Database: SQL* 7](#_Toc186161528)

[1.4.5. Mosquitto 7](#_Toc186161529)

[**II. Giao diện** 8](#_Toc186161530)

[2.1. Giao diện website 8](#_Toc186161531)

[2.2. Giao diện API, API Docs 10](#_Toc186161532)

[2.3. Giao diện thiết bị 12](#_Toc186161533)

[**III. Thiết kế chi tiết** 12](#_Toc186161534)

[Thiết kế hệ thống 12](#_Toc186161535)

[**IV. Code** 13](#_Toc186161536)

[Arduino code 13](#_Toc186161537)

[**V. Kết quả thu được** 16](#_Toc186161538)

[5.1. Tổng quan 16](#_Toc186161539)

[5.2. Từ DHT11 và cảm biến ánh sáng 16](#_Toc186161540)

[5.3. Khi người dùng điều khiển thiết bị 17](#_Toc186161541)

[**VI. Tài liệu tham khảo** 17](#_Toc186161542)

# **I. Giới Thiệu**

## 1.1. IoT (Internet of Things)

### 1.1.1. Khái niệm

Mạng lưới vạn vật kết nối (IoT) là một hệ thống mạng lưới trong đó các thiết bị và đối tượng thông minh có khả năng giao tiếp và trao đổi dữ liệu với nhau thông qua internet. IoT cho phép các thiết bị không chỉ kết nối với mạng internet, mà còn có khả năng truyền nhận thông tin và tương tác với nhau mà không cần sự can thiệp của con người.

### 1.1.2. Các thành phần của IoT

* Thiết bị IoT: Đây là các thiết bị thông minh được trang bị cảm biến, các công nghệ kết nối và khả năng thu thập, gửi và nhận dữ liệu.
* Mạng: Đảm bảo việc truyền dữ liệu giữa các thiết bị IoT và hệ thống xử lý dữ liệu.
* Hệ thống xử lý dữ liệu: Bao gồm các máy chủ và hệ thống phần mềm để lưu trữ, xử lý và phân tích dữ liệu từ các thiết bị IoT.
* Ứng dụng và dịch vụ: Cung cấp các ứng dụng và dịch vụ dựa trên dữ liệu từ các thiết bị IoT, nhằm giúp cải thiện cuộc sống và công việc của con người.

### 1.1.3. Ứng dụng

IoT có thể được áp dụng trong nhiều lĩnh vực khác nhau, bao gồm:

* Nhà thông minh: Điều khiển ánh sáng, nhiệt độ, an ninh, thiết bị gia đình thông qua điện thoại di động.
* Công nghiệp: Giám sát và quản lý các quy trình sản xuất, dự báo bảo trì thiết bị, tăng cường an toàn lao động.
* Giao thông: Điều khiển giao thông thông minh, quản lý đỗ xe, theo dõi vận chuyển hàng hóa.
* Y tế: Theo dõi sức khỏe, quản lý thuốc, giám sát bệnh nhân từ xa.
* Năng lượng: Theo dõi và quản lý tiêu thụ năng lượng, tối ưu hóa sử dụng tài nguyên.

## 1.2. Mô tả và mục đích của đề tài

### 1.2.1. Mô tả đề tài

Đề tài này tập trung vào việc xây dựng một hệ thống IoT để giám sát và điều khiển các yếu tố môi trường như nhiệt độ, độ ẩm, ánh sáng và các thiết bị điện (đèn, quạt, system, tivi) thông qua một website. Hệ thống được phát triển với backend sử dụng Java Spring Boot và cơ sở dữ liệu SQL để lưu trữ và quản lý dữ liệu. Các cảm biến sẽ thu thập dữ liệu nhiệt độ, độ ẩm và ánh sáng trong thời gian thực, và người dùng có thể điều khiển các thiết bị điện từ xa thông qua giao diện website.

### 1.2.2. Mục đích của đề tài

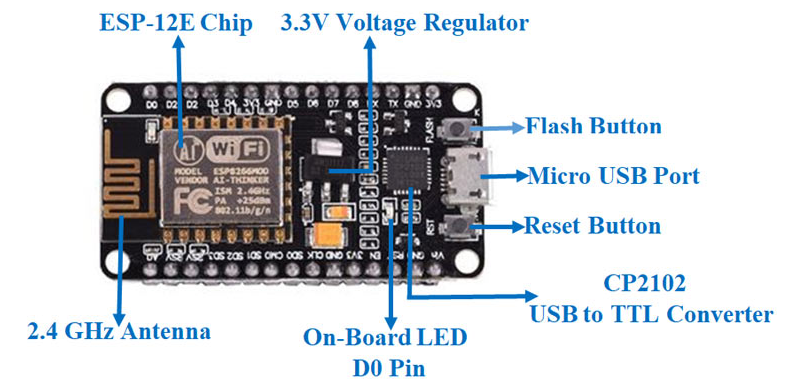
Mục đích của đề tài là xây dựng một hệ thống quản lý và điều khiển thông minh với các mục tiêu chính như sau:

* Giám sát và cảnh báo:
  + Thu thập và hiển thị dữ liệu nhiệt độ, độ ẩm, ánh sáng từ các cảm biến một cách trực quan trên website.
  + Cảnh báo người dùng khi các chỉ số môi trường vượt quá ngưỡng an toàn (ví dụ: nhiệt độ quá cao hoặc độ ẩm quá thấp).
* Điều khiển thiết bị từ xa:
  + Cho phép người dùng bật/tắt và điều chỉnh các thiết bị như đèn, quạt, tivi, hệ thống nào đó thông qua giao diện website.
* Quản lý và lưu trữ dữ liệu:
  + Lưu trữ dữ liệu giám sát (nhiệt độ, độ ẩm) và lịch sử hoạt động của các thiết bị vào cơ sở dữ liệu SQL.
  + Cung cấp các báo cáo thống kê, biểu đồ phân tích dữ liệu giúp người dùng nắm bắt tình trạng hoạt động của hệ thống.

## 1.3. Thiết bị sử dụng

### 1.3.1. ESP8266

* Thông số kỹ thuật:
* Bộ vi điều khiển: CPU RISC 32-bit Tensilica Xtensa LX106
* Điện áp hoạt động: 3.3V
* Điện áp đầu vào: 7-12V
* Chân I / O kỹ thuật số (DIO): 16
* Chân đầu vào tương tự (ADC): 1
* UARTs: 1
* SPI: 1
* I2Cs: 1
* Bộ nhớ Flash: 4 MB
* SRAM: 64 KB
* Tốc độ đồng hồ: 80 MHz
* USB-TTL dựa trên CP2102 được bao gồm trên bo mạch, cho phép Plug n Play
* Ăng-ten PCB
* Mô-đun có kích thước nhỏ để phù hợp thông minh bên trong các dự án IoT của bạn



*Hình 1: ESP8266*

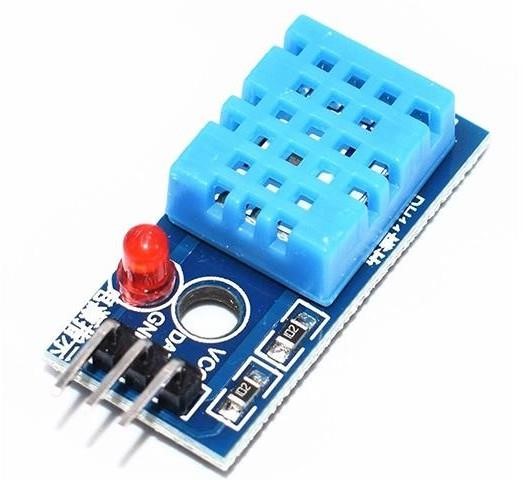
- Các chân ESP8266:

* + - * GND: Chân mẫu điện âm.
      * 3V3: Chân nguồn cung cấp 3.3V.
      * USB: Cổng USB được sử dụng để kết nối board với máy tính để lập trình và cung cấp nguồn.

### *1.3.2. Cảm biến nhiệt độ - độ ẩm DHT11*

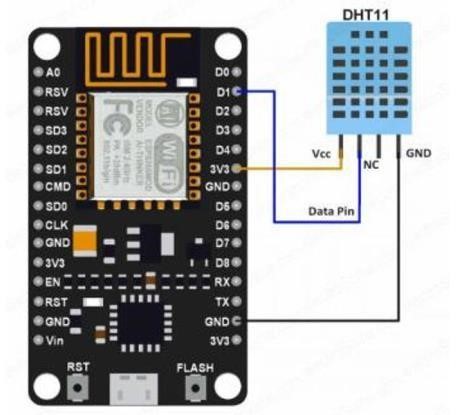
- Thông số kỹ thuật:

* + Điện áp hoạt động: 3V - 5V DC
  + Dòng điện tiêu thụ: 2.5mA
  + Phạm vi cảm biến độ ẩm: 20% - 90% RH, sai số ±5%RH
  + Phạm vi cảm biến nhiệt độ: 0°C ~ 50°C, sai số ±2°C
  + Tần số lấy mẫu tối đa: 1Hz (1 giây 1 lần)
  + Kích thước: 23 \* 12 \* 5 mm



*Hình 2: Cảm biến DHT11* - Sơ đồ chân DHT11:

|  |  |  |
| --- | --- | --- |
| Số chân | Tên chân | Mô tả |
| 1 | VCC | Nguồn 3.3V đến 5V |
| 2 | Data | Đầu ra nhiệt độ độ ẩm thông qua dữ liệu nối tiếp |
| 3 | NC | Không có kết nối và do đó không sử dụng |
| 4 | Ground | Nối đất |



*Hình 3: Sơ đồ chân nối DHT11*

### *1.3.3. Cảm biến cường độ ánh sáng quang trở*

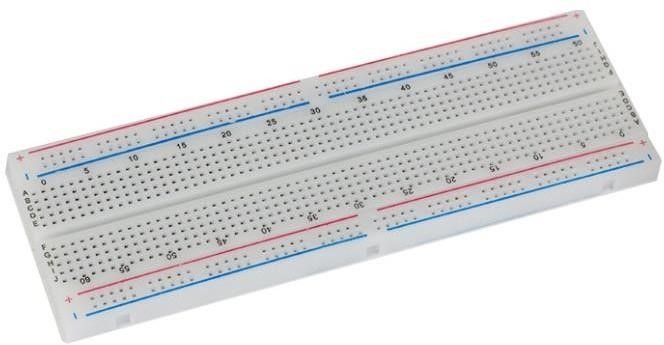
* Thông số kỹ thuật
  + Điện áp hoạt động: 3.3V – 5V
  + Kết nối 4 chân với 2 chân cấp nguồn (VCC và GND) và 2 chân tín hiệu ngõ ra (AO và DO).
  + Hỗ trợ cả 2 dạng tín hiệu ra Analog và TTL. Ngõ ra Analog 0– 5V tỷ lệ thuận với cường độ ánh sáng, ngõ TTL tích cực mức thấp.
  + Độ nhạy cao với ánh sáng được tùy chỉnh bằng biến trở .
  + Kích thước: 32 x 14mm



*Hình 4: Cảm biến ánh sáng*

### 1.3.4. Board Test MB-102 16.5x5.5

Dùng để test mạch trước khi làm mạch in hoàn chỉnh. Giúp bạn dễ dàng thực hiện các mạch điện thực tế trước khi hàn trực tiếp linh kiện lên board mạch đồng



*Hình 5: Board test*

### 1.3.5. Điện trở vạch 1/4W sai số 5% 250V 1R-10M



*Hình 6: Điện trở*

### 1.3.6. Dây nối chân các thiết bị



*Hình 7: Dây nối*

### 1.3.7. Led 5mm, 1 màu 2 chân (1,8-3V)



*Hình 8: Led*

## 1.4. Công nghệ sử dụng

### 1.4.1. Trình biên dịch Arduino IDE

- Arduino IDE là một môi trường phát triển tích hợp được sử dụng để viết và nạp mã cho các vi điều khiển Arduino. Đây là công cụ chính để lập trình các thiết bị phần cứng, cho phép giao tiếp với các cảm biến (như nhiệt độ, độ ẩm) và thiết bị điều khiển (như quạt, đèn, điều hòa). - Tính năng chính:

* Hỗ trợ các thư viện và hàm đặc biệt dành riêng cho lập trình IoT.
* Dễ dàng nạp mã cho các bo mạch Arduino thông qua cổng USB.
* Giao diện đơn giản, dễ sử dụng cho việc lập trình và kiểm tra mã nguồn.

### *1.4.2. Frontend: HTML*

### 1.4.3 Backend: Spring Boot

* Spring Boot là một framework mạnh mẽ để phát triển các ứng dụng Java, đặc biệt là các ứng dụng web và RESTful API. Spring Boot cung cấp môi trường phát triển nhanh chóng và đơn giản với các tính năng bảo mật và quản lý dữ liệu hiệu quả. - Tính năng chính:
  + RESTful API: Xây dựng các dịch vụ API để giao tiếp với frontend, cung cấp dữ liệu từ cơ sở dữ liệu và thực hiện các chức năng điều khiển thiết bị.
  + Spring Security: Cung cấp bảo mật cho ứng dụng với các chức năng xác thực và phân quyền người dùng.
  + Spring Data JPA: Tương tác với cơ sở dữ liệu một cách dễ dàng, hỗ trợ các thao tác CRUD (Create, Read, Update, Delete).
  + Dependency Injection: Quản lý các thành phần và phụ thuộc trong ứng dụng một cách hiệu quả.

### *1.4.4 Database: SQL*

- SQL là một hệ quản trị cơ sở dữ liệu quan hệ phổ biến, được sử dụng để lưu trữ và quản lý thông tin về nhiệt độ, độ ẩm và trạng thái của các thiết bị. SQL cung cấp khả năng truy vấn và quản lý dữ liệu mạnh mẽ, phù hợp cho các ứng dụng có quy mô vừa và lớn.

* Tính năng chính:
  + Khả năng lưu trữ: Lưu trữ thông tin lịch sử hoạt động của các thiết bị và dữ liệu cảm biến.
  + Quan hệ dữ liệu: Tổ chức dữ liệu theo mô hình quan hệ, đảm bảo tính nhất quán và toàn vẹn dữ liệu.
  + Truy vấn nhanh chóng: Sử dụng các chỉ mục và khóa ngoại để tối ưu hóa hiệu suất truy vấn dữ liệu.
  + Bảo mật dữ liệu: Cung cấp các tính năng quản lý người dùng và phân quyền truy cập để đảm bảo an toàn dữ liệu.

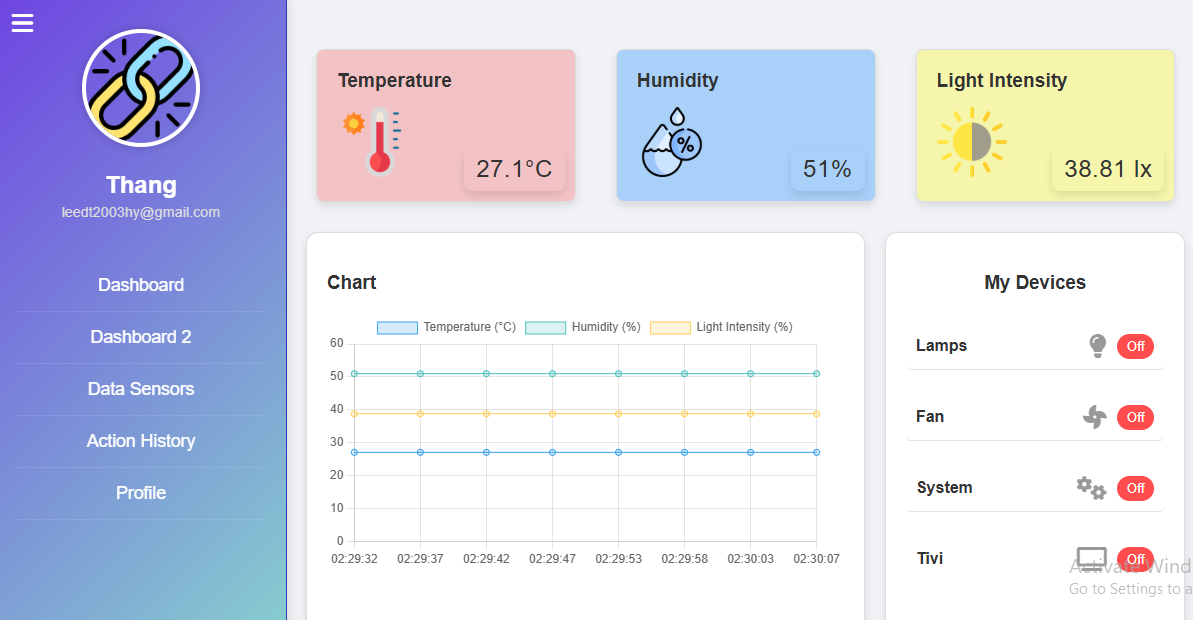
### 1.4.5. Mosquitto

* Mosquitto là một MQTT broker mã nguồn mở, cho phép giao tiếp giữa các thiết bị IoT với nhau và với server. Nó đảm bảo việc truyền tải thông tin nhanh chóng và ổn định, giúp hệ thống giám sát và điều khiển hoạt động trơn tru.
* Tính năng chính:
  + Giao tiếp MQTT: Hỗ trợ giao thức MQTT để truyền tải thông điệp giữa các thiết bị IoT và ứng dụng server.
  + Chất lượng dịch vụ (QoS): Cung cấp nhiều mức độ QoS để đảm bảo thông điệp được truyền tải một cách tin cậy.
  + Nhẹ và hiệu quả: Tiêu thụ ít tài nguyên hệ thống, phù hợp cho các thiết bị IoT có tài nguyên hạn chế.
  + Bảo mật: Hỗ trợ các cơ chế bảo mật như TLS và xác thực người dùng để đảm bảo an toàn trong việc truyền thông điệp.

# **II. Giao diện**

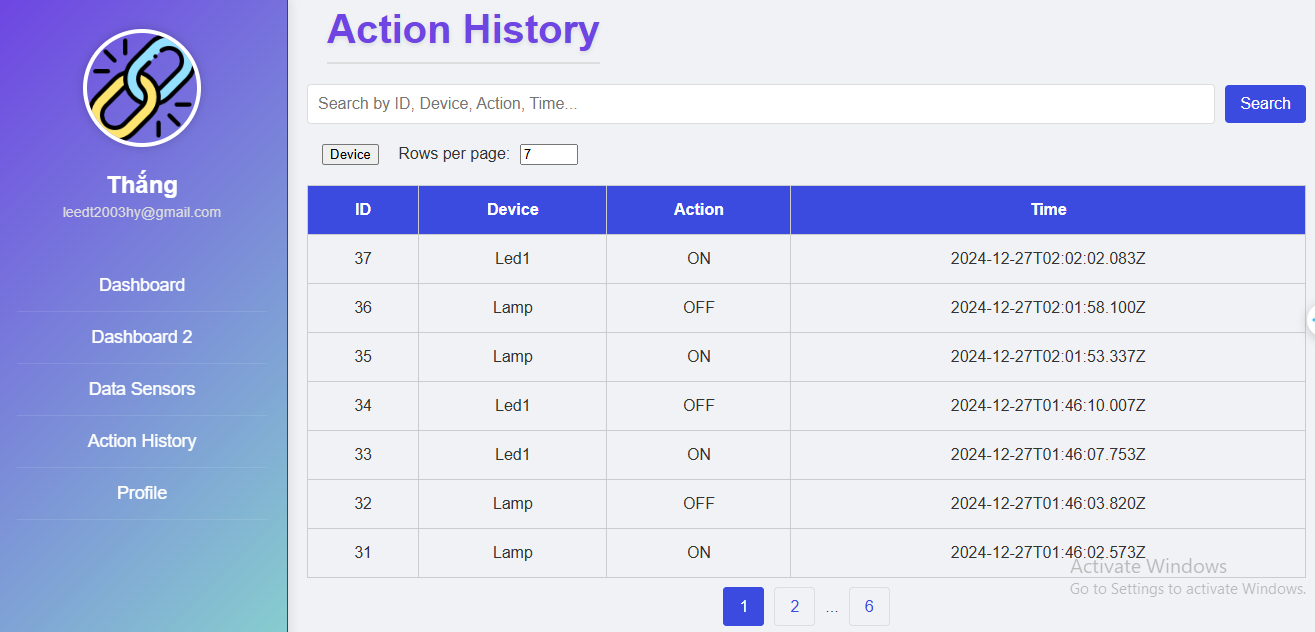
## 2.1. Giao diện website

* Trang Dashboard hiển thị nhiệt độ, độ ẩm, ánh sáng mà cảm biến đo được, cùng với biểu đồ thống kê 3 giá trị đó theo thời gian. Để điều khiển bật tắt các thiết bị có thể nhấn On/Off



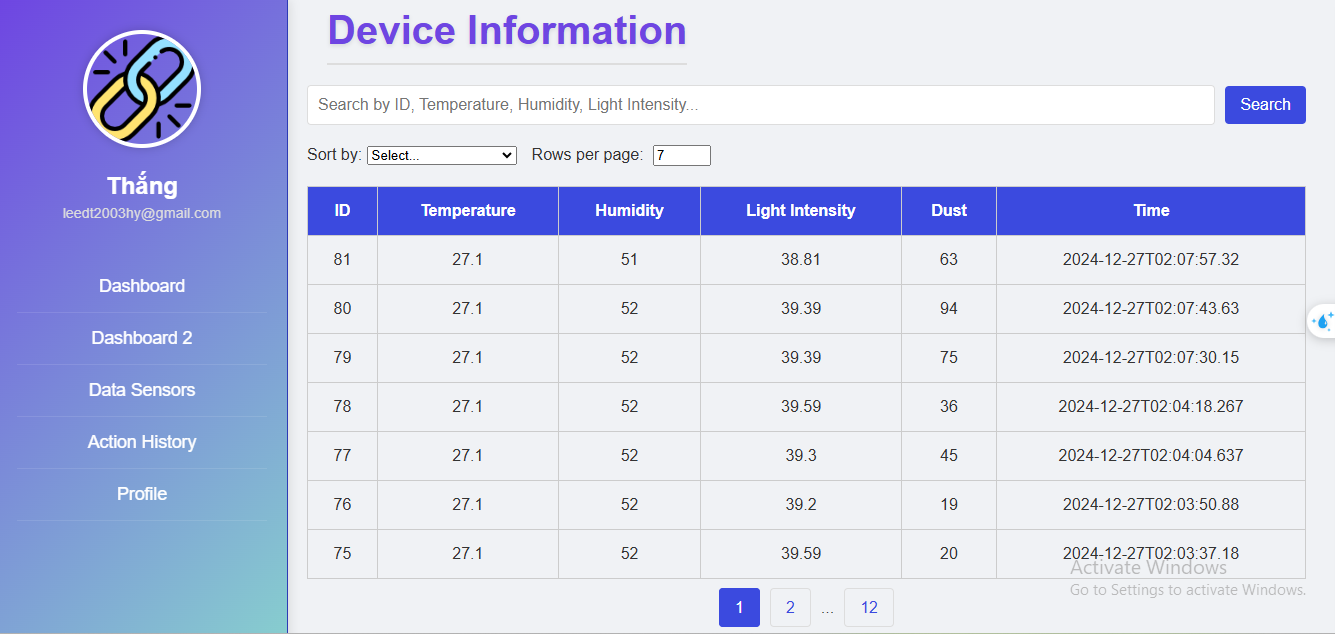
*Hình 9: Giao diện Dasboard*

* Trang Action History hiển thị hoạt động bật tắt của các thiết bị theo thời gian.



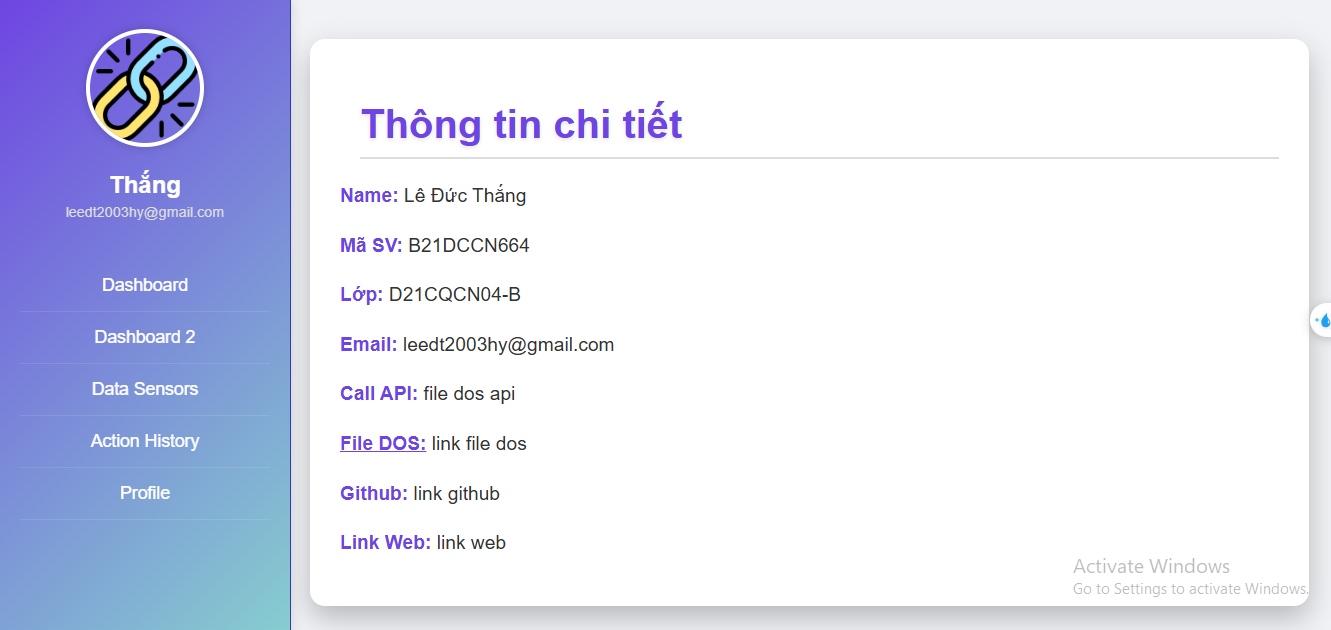
*Hình 10: Giao điện Action History*

* Trang Data Sensor hiển thị các giá trị nhiệt độ, độ ẩm, ánh sang theo thời gian



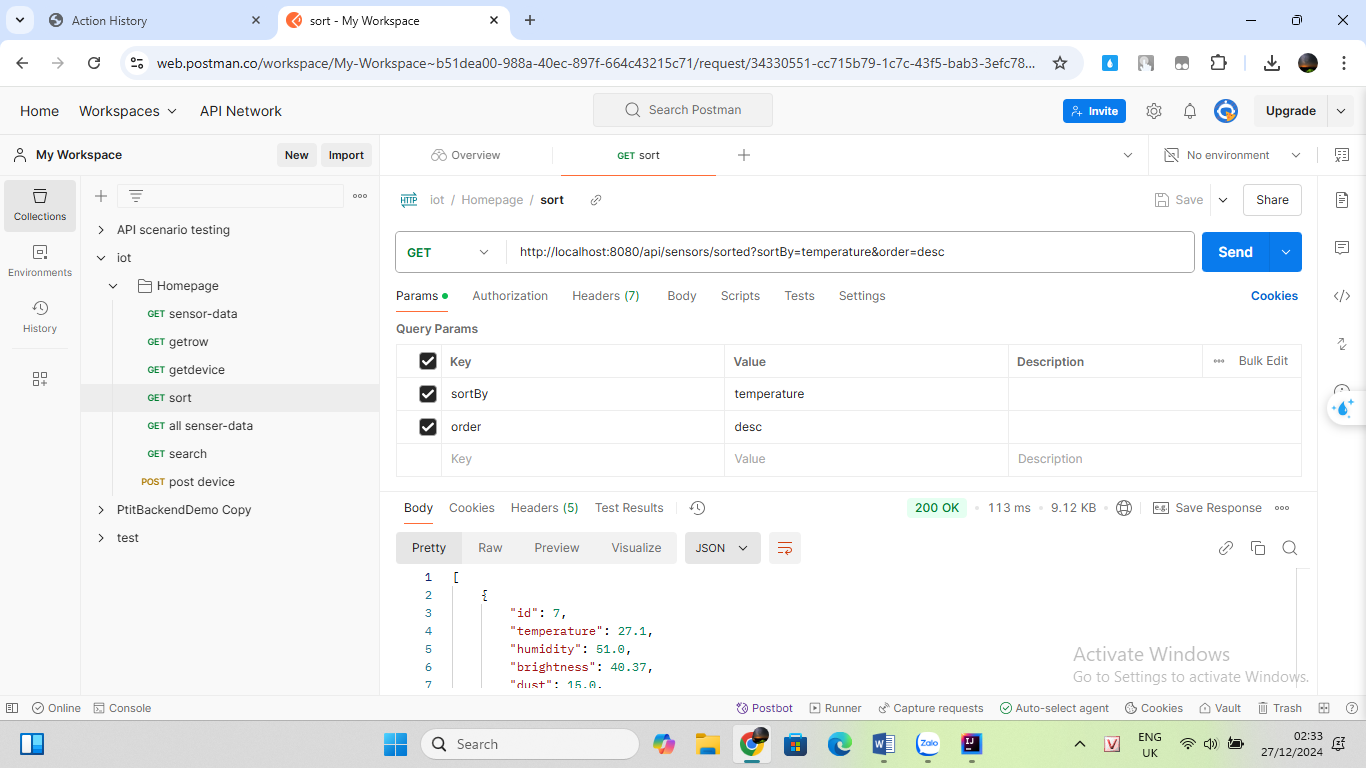
*Hình 11: Giao diện Data Sensor*

* Trang Profile hiển thị thông tin sinh viên và các liên kết theo yêu cầu



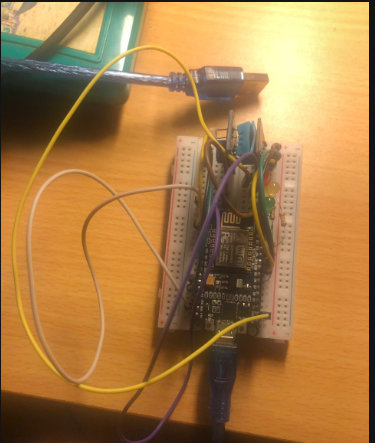
*Hình 12: Giao diện Profile*

## 2.2. Giao diện API, API Docs



*Hình 13: Giao diện API*

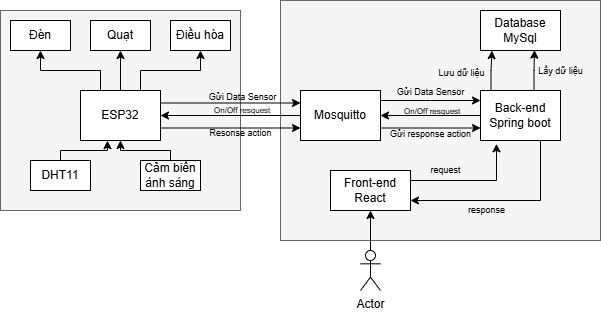
## 2.3. Giao diện thiết bị



*Hình 15: Giao diện thiết bị*

# **III. Thiết kế chi tiết**

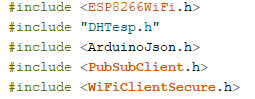
## Thiết kế hệ thống



*Hình 16: Thiết kế hệ thống*

# **IV. Code**

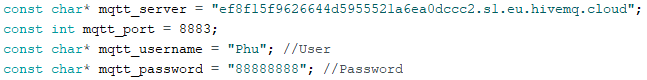
## Arduino code



* Khai báo các thư viện cần thiết gồm:
  + DHespT.h: Đọc dữ liệu nhiệt độ và độ ẩm từ cảm biến DHT.
  + WiFiClient.h: Thư viện kết nối với wifi
  + PubSubClient.h: Thư viện sử dụng mqtt
  + ESP8266WiFi.h: Thư viện esp8266



* Định nghĩa các thông tin như tên wifi, mật khẩu wifi mà ESP8266 sẽ kết nối



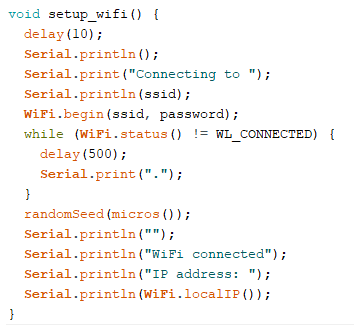
* Định nghĩa các thông tin về server của MQTT Broker (ở đây là Mosquitto), tên các topic, port



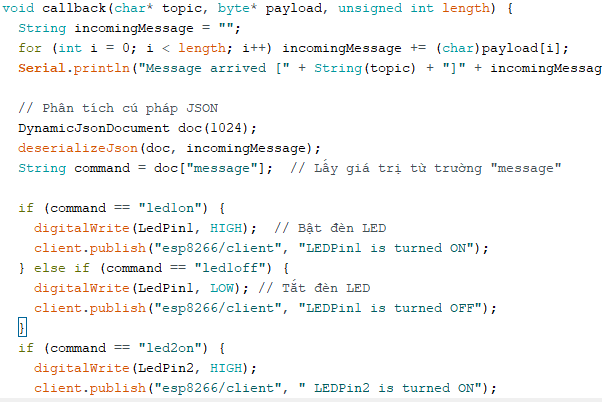
* Định nghĩa các chân (pin) mà cảm biến ánh sáng và các thiết bị (quạt và đèn LED) được kết nối trên ESP8266. Cụ thể:
  + Cảm biến ánh sáng được kết nối với chân analog A0 của ESP8266.
  + Chân D1, D2, D6 được dùng để điều khiển LED.



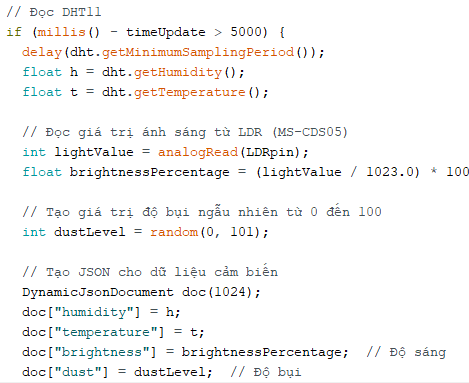
* Định nghĩa chân kết nối với DHT11 là chân D4
* Setup kết nối wifi



* Nhận yêu cầu bật đèn và trả về thông báo đèn đã bật hay chưa



* Gửi data sensor

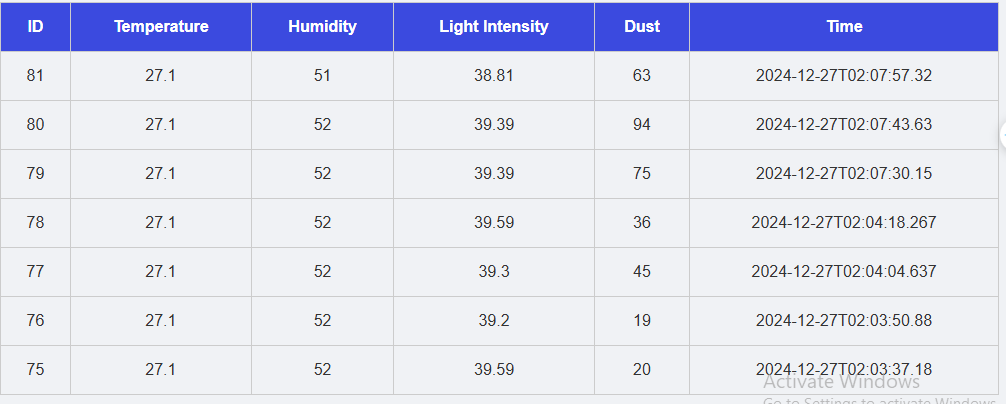


# **V. Kết quả thu được**

## 5.1. Tổng quan

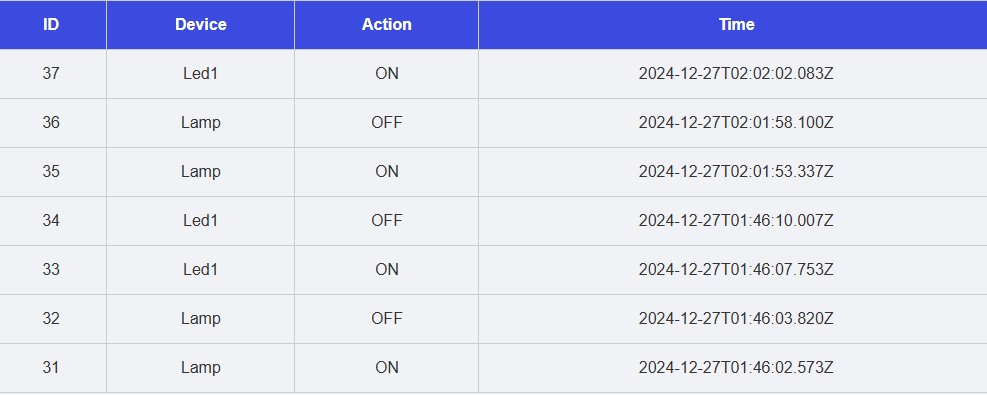
Hệ thống IoT đã thực hiện thành công các chức năng đo thông số môi trường theo thời gian thực, điều khiển thiết bị điện từ xa, và lưu trữ dữ liệu cho người dùng. Hệ thống đã được triển khai và thử nghiệm, đáp ứng tốt các yêu cầu đề ra.

## 5.2. Từ DHT11 và cảm biến ánh sáng



- Hệ thống đã đo được các thông số của nhiệt độ, độ ẩm, ánh sáng và truyền lên website theo thời gian thực để hiện thị cho người dùng một cách tương đối trên trang Data Sensor.

## 5.3. Khi người dùng điều khiển thiết bị



* Hệ thống đã lưu thành công các dữ liệu khi người dùng nhấn On/Off trên website và hiển thị trên trang Action History

# **VI. Tài liệu tham khảo**

* Tham khảo về Esp8266, mosquito, DHT11, Arduino IDE:
* [**https://dienthongminhesmart.com/lap-trinh-esp8266/giao-thuc-mqtt-va-esp/#Ket\_noi\_phan\_cung\_va\_nap\_code\_chuong\_trinh**](https://dienthongminhesmart.com/lap-trinh-esp8266/giao-thuc-mqtt-va-esp/#Ket_noi_phan_cung_va_nap_code_chuong_trinh).
* **https://www.slideshare.net/slideshow/123doc-dieukhienthietbiquamodulewifiesp8266/249546180**.