



BÁO CÁO ĐỒ ÁN

LIGHT CONTROL



Môn học: Hệ thống nhúng và Mạng không dây

Lớp: NT131.N11

THÀNH VIÊN THỰC HIỆN

(Nhóm 10):

Họ và tên

MSSV

Nguyễn Đàm Nhật Anh 20520880

Giảng viên hướng dẫn:

TS. Lê Trung Quân

MỤC LỤC

A. TỔNG QUAN.....	3
1. Tổng quan về đề tài.....	3
a. Giới thiệu đề tài	3
b. Mục đích	3
c. Các thiết bị được sử dụng:	3
2. Cơ sở lý thuyết.....	3
a. Các thiết bị phân cứng:.....	3
b. Cơ chế hoạt động.....	5
3. Tổng kết và phát triển.....	10
a. Tổng kết:	10
b. Phát triển:	11
B. TÀI LIỆU THAM KHẢO	11

A. TỔNG QUAN

1. Tổng quan về đề tài

a. Giới thiệu đề tài

Trong thời đại ngày nay, các thiết bị và hệ thống tiết kiệm năng lượng là nhu cầu của thời đại. Con người đang lãng phí quá nhiều năng lượng điện khi không tắt đèn, quạt và nhiều thiết bị điện khác khi không sử dụng. Để khắc phục vấn đề này, một thiết bị IoT được phát triển có thể được lắp ở trong phòng với mục đích phát hiện sự hiện diện của con người để có thể tự động bật và tắt (hoặc có thể điều khiển trực tiếp thông qua các thiết bị laptop(web app/ mobile app)).

b. Mục đích

Một thiết bị tự động sẽ giúp người dùng sẽ đơn giản hoá việc sử dụng đèn, có thể điều khiển được từ xa, đồng thời có thể kiểm soát được tình trạng của đèn. Từ đó giảm thiểu được thời gian sử dụng đèn đồng thời tiết kiệm được điện năng.

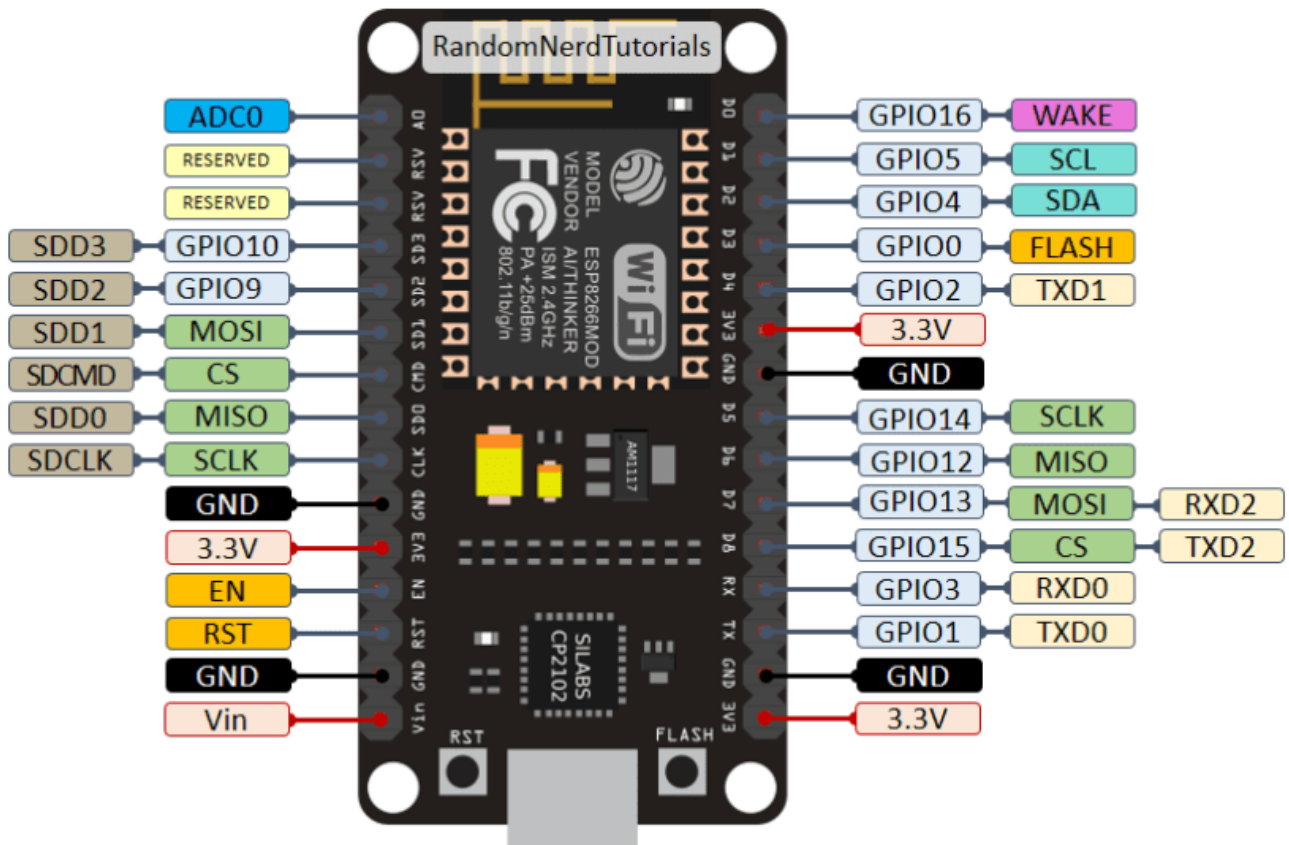
c. Các thiết bị được sử dụng:

1. Module ESP8266 NodeMCU
2. Cảm biến vật cản hồng ngoại
3. Switch
4. Led
5. Dây điện bus 2 đầu

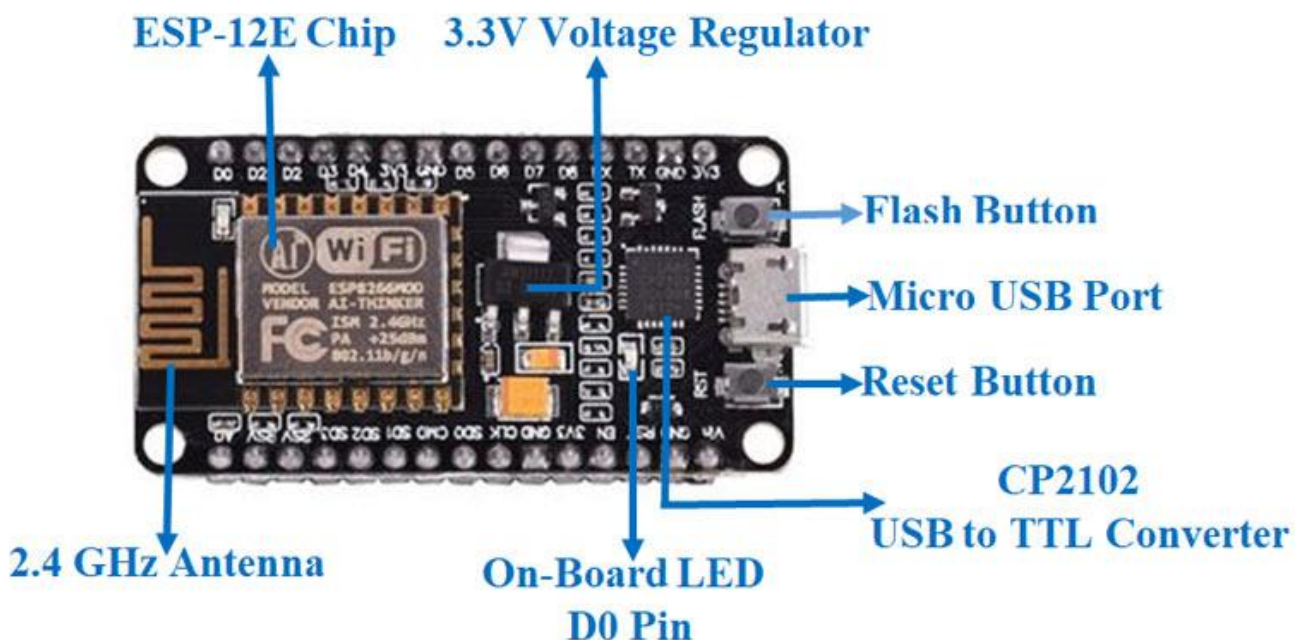
2. Cơ sở lý thuyết

a. Các thiết bị phần cứng:

1. Module ESP8266 NodeMCU:



Module ESP8266 NodeMCU là một module có ESP-12E chứa chip ESP8266 có bộ vi xử lý Tensilica Xtensa 32-bit LX106 RISC. NodeMCU có 128 KB RAM và 4MB bộ nhớ Flash để lưu trữ dữ liệu và chương trình. NodeMCU có thể được cấp nguồn bằng giắc cắm Micro USB và chân VIN (Chân nguồn cung cấp bên ngoài)



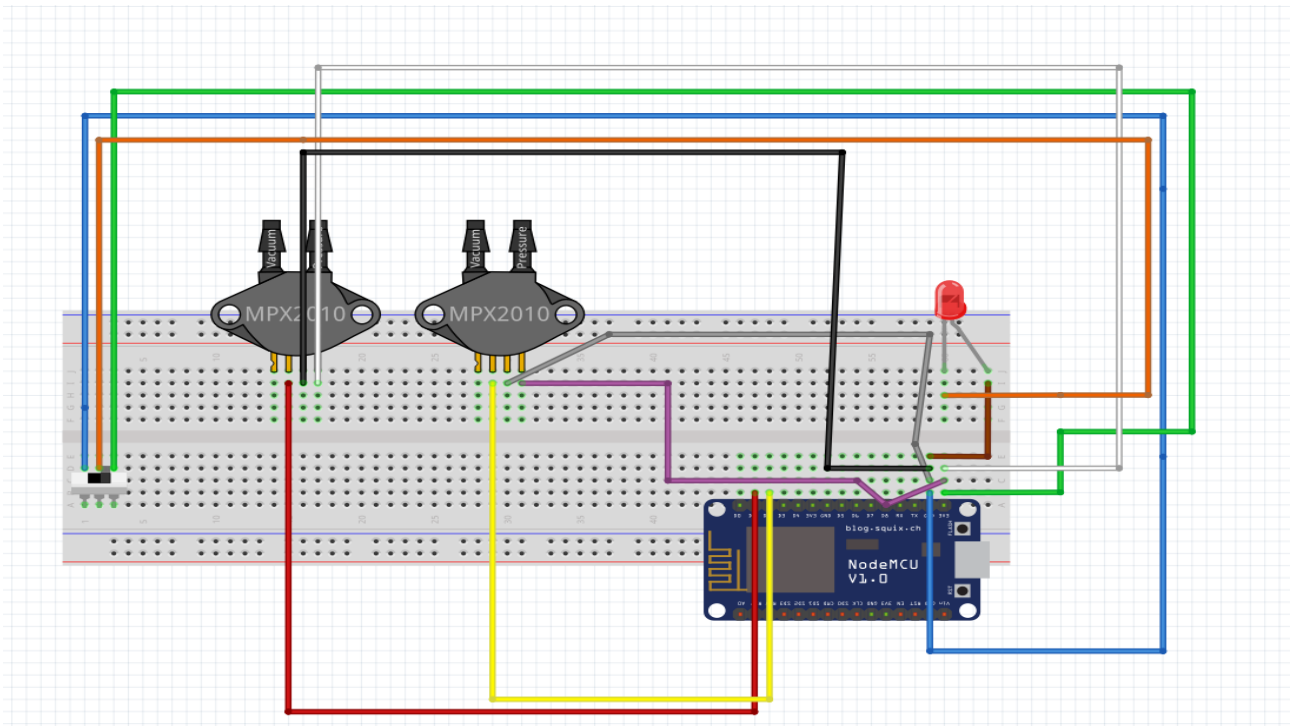
2. Cảm biến vật cản hồng ngoại

Cảm biến có khả năng nhận biết vật cản ở môi trường với một cặp LED thu phát hồng ngoại để truyền và nhận dữ liệu hồng ngoại. Tia hồng ngoại phát ra với

tần số nhất định, khi có vật cản trên đường truyền của LED phát nó sẽ phản xạ vào LED thu hồng ngoại, khi đó LED báo vật cản trên module sẽ sáng, khi không có vật cản, LED sẽ tắt. Với khả năng phát hiện vật cản trong khoảng 2 ~ 30cm và khoảng cách này có thể điều chỉnh thông qua chiết áp trên cảm biến cho thích hợp với từng ứng dụng cụ thể như: xe dò line, xe tránh vật cản.

3. Switch, Led, Dây bus

Switch để làm công tắc bật đèn, dây bus để kết nối các chân với nhau



b. Cơ chế hoạt động

b1. Cảm biến nhận diện

Sử dụng 2 cảm biến hồng ngoại để xác định người ra vào bằng cách xem thứ tự tín hiệu của cảm biến.

Luồng hoạt động:

- Đi vào:

Khi 2 cảm biến chưa nhận được tín hiệu hồng ngoại: lúc này sẽ ở trạng thái Free.

Khi cảm biến 1 nhận được vật cản trước, đọc được tín hiệu và lúc này sẽ ở trạng thái Entering1.

Khi cảm biến 2 nhận được vật cản, đọc được tín hiệu và lúc này sẽ ở trạng thái Entering2. Khi ở trạng thái Entering2, một biến đếm người (count) sẽ được tăng lên 1 đơn vị. Và khi biến đếm người > 0 thì đèn sẽ được bật.

Khi cảm biến 2 đang vật cản và cảm biến 2 không có vật cản, lúc này sẽ ở trạng thái Entering3.

Và khi cảm biến 2 không còn vật cản, có nghĩa lúc này người đã đi vào. Và trạng thái lúc này sẽ là Free.

- Đi ra:

Khi cảm biến 2 nhận được vật cản trước, đọc được tín hiệu và trạng thái lúc này là Exiting1.

Khi cảm biến 1 nhận được vật cản, đọc được tín hiệu và lúc này sẽ ở trạng thái Exiting2.

Khi cảm biến 2 không còn vật cản, đọc được tín hiệu và lúc này sẽ ở trạng thái Exiting3. Khi ở trạng thái Exiting3, biến đếm người (count) sẽ được giảm 1 đơn vị. Và khi biến đếm người = 0 thì đèn sẽ được tắt.

Khi cảm biến 1 không còn vật cản, lúc này sẽ ở trạng thái Free

b2. Điều khiển đèn bằng web

Sử dụng thư viện ESP8266Wifi.h để kết nối vào wifi. Sử dụng thư viện ESPAsyncTCP.h và ESPAsyncWebServer.h để ESP tạo một web server. Từ đó nhúng code của web vào server đó để có thể hiện thị 1 trang web.

Ngôn ngữ để thực hiện trang web bao gồm: Frontend gồm HTML, CSS, JS. Backend gồm: PHP (Laravel framework) và SQL. Giao tiếp giữa Frontend và Backend là API.

Đầu tiên: ESP sẽ kết nối vào wifi. Rồi tạo ra 1 server bằng thư viện AsyncWebServer.h. Nhúng trang web vào bằng cách gán toàn bộ code web vào 1 biến và sẽ push lên web server của esp bằng lệnh server.on

Để dùng nút bật tắt đèn trên web, đặt trong code web 1 biến %BUTTONPLACEHOLDER% và sẽ cho biến đó thay đổi khi tín hiệu đèn được thay đổi trên esp. Khi tín hiệu đèn được thay đổi, ở biến %BUTTONPLACEHOLDER% sẽ được in ra dòng lệnh html để thể hiện đèn đang bật hay tắt. Đồng thời nếu thay đổi biến trên web thì tín hiệu đèn ở esp cũng được thay đổi.

Khi đèn được bật hoặc tắt, ở esp sẽ tạo 1 đường dẫn cho webserver đồng thời gửi giá trị của đèn ở dạng plain text. Từ đó có thể đọc được giá trị của đèn để hiển thị đèn đang được bật hay tắt.

Khi bật đèn, web sẽ gửi API về cho backend với nội dung là giá trị của đèn. Lúc này backend sẽ nhận được giá trị đèn bật hay tắt và sẽ trả về



API với nội dung bao gồm trạng thái đèn, thời gian đèn bật, thời gian đèn tắt và thời gian đèn được sử dụng.

Khi đăng nhập, web sẽ gửi API về cho backend với nội dung là username và password. Và backend server sẽ gửi lại 1 API có nội dung là 1 thông báo login success, thời gian đèn được sử dụng, token, user gồm ID, roleID và username.

POST

https://esp.f4koin.cyou/api/log

ParamsAuthorizationHeaders (9)BodyPre-request ScriptTestsSettings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	KEY	VALUE
<input checked="" type="checkbox"/>	status	on
	Key	Value

BodyCookiesHeaders (14)Test Results

PrettyRawPreviewVisualizeJSON

```
1  {
2    "message": "success",
3    "total_active_time": "00:00:13",
4    "status": "on"
5  }
```

Hình 1. API bật đèn

POST

▼

https://esp.f4koin.cyou/api/log

Params

Authorization ●

Headers (9)

Body ●

Pre-request Script

Tests

Settings

● none

● form-data

● x-www-form-urlencoded

● raw

● binary

● GraphQL

	KEY	VALUE
<input checked="" type="checkbox"/>	status	off
	Key	Value

Body

Cookies

Headers (14)

Test Results

Pretty

Raw

Preview

Visualize

JSON ▼

1

2

3

4

5

"message": "success",

"total_active_time": "00:00:29",

"status": "off"

Hình 2. API tắt đèn

POST

https://esp.f4koin.cyou/api/login

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	KEY	VALUE
<input checked="" type="checkbox"/>	username	ndn4
<input checked="" type="checkbox"/>	password	123456789
	Key	Value

Body

Cookies

Headers (14)

Test Results

Pretty

Raw

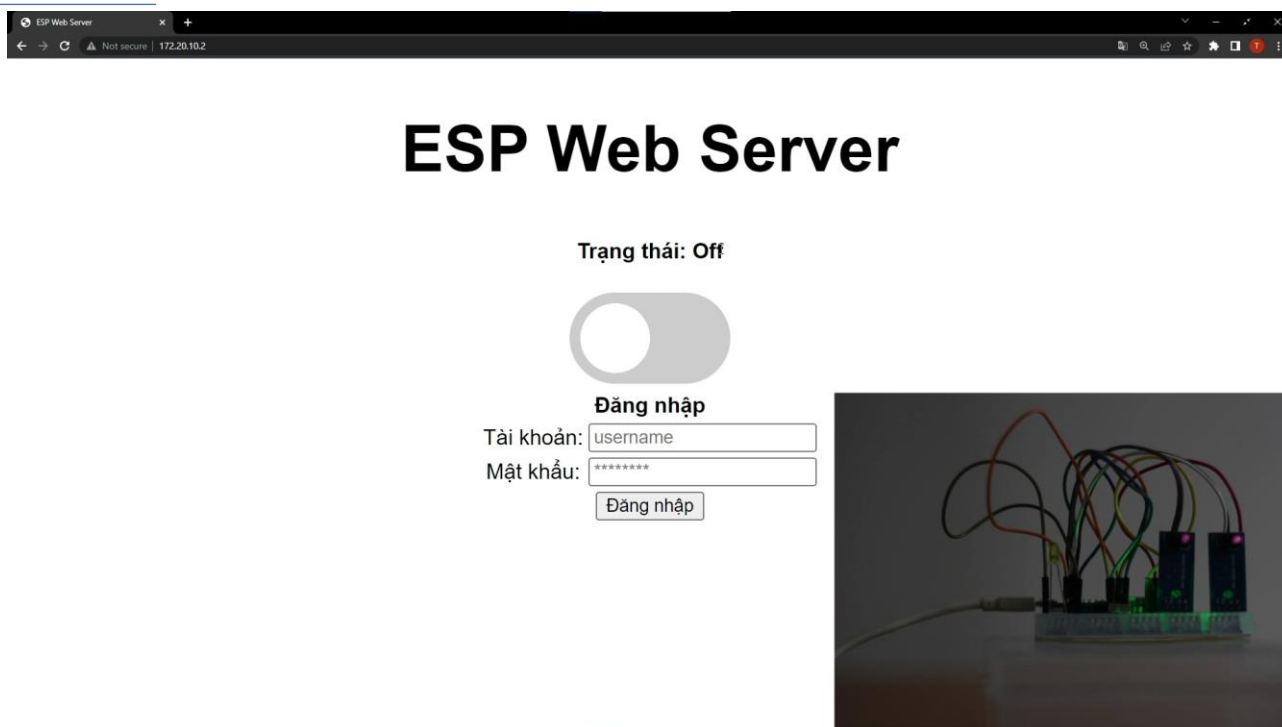
Preview

Visualize

JSON

```
1  {
2    "user": {
3      "userID": "fd2cd1b2-177d-49e8-a4f8-b5c30e9bedba",
4      "userRoleID": "2",
5      "username": "ndn4"
6    },
7    "token": "0|LWbduOB2YzGBEA0gfRgTuYq47Eye4ioYfYwgHpCn",
8    "total_active_time": "00:00:29",
9    "message": "Login success"
10 }
```

Hình 3. API đăng nhập



Hình 4. Trang web trước khi được đăng nhập



Hình 5. Trang web sau khi được đăng nhập

3. Tổng kết và phát triển

a. Tổng kết:

- Mô hình hiện tại đã có thể bật tắt đèn thông qua 3 hình thức: Website, công tắc và cảm biến

- Hiện tại chỉ điều khiển được 1 thiết bị
- Test case trong trường hợp lý tưởng, với các trường hợp hai người cùng lúc vào và ra khác nhau sẽ sai số, cảm biến không nhận được tín hiệu của người thứ 2.
- Vẫn còn delay và chưa đồng bộ được tính năng gửi tín hiệu đèn qua API với công tắc và cảm biến
- Hiện tại có thể sử dụng cho các phòng nhỏ có ít thiết bị để tăng sự tự động hoá.

b. Phát triển:

- Điều khiển đa thiết bị, đa người dùng:
 - o Với việc phát triển đa người dùng: API hiện đã có userRole để có thể phân quyền tính năng cho người dùng
 - o Với việc phát triển đa thiết bị: Sẽ cấp thêm cho các thiết bị các id để có thể định danh nhằm quản lý. Vì sử dụng API nên các thiết bị sẽ được độc lập
- Đồng bộ tính năng gửi tín hiệu đèn
- Nâng cấp thuật toán để nhận diện tín hiệu được nhiều trường hợp phức tạp hơn

B. TÀI LIỆU THAM KHẢO

[1].<https://arduinoakit.vn/dieu-khien-led-bang-webserver-su-dung-nodemcu-esp8266/>

[2].<https://techtutorialsx.com/2016/07/21/esp8266-post-requests/>