ĐẠI HỌC THÁI NGUYÊN

**TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP**



**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN**

**NGHIÊN CỨU VÀ XÂY DỰNG MÔ HÌNH ĐIỀU KHIỂN HỆ ĐA VI ĐIỀU KHIỂN PHÂN TÁN THÔNG QUA MẠNG NỘI BỘ LAN.**

**MÃ SỐ: SV2022 – 29**

|  |  |
| --- | --- |
| Sinh viên thực hiện | **: Nguyễn Thọ Duy** |
| Mã số sinh viên | **: K195480106007** |
| Giáo viên hướng dẫn | **: ThS Tăng Cẩm Nhung** |

**Thái Nguyên, tháng 3/ 2023**

ĐẠI HỌC THÁI NGUYÊN

**TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN**

**NGHIÊN CỨU VÀ XÂY DỰNG MÔ HÌNH ĐIỀU KHIỂN HỆ ĐA VI ĐIỀU KHIỂN PHÂN TÁN THÔNG QUA MẠNG NỘI BỘ LAN.**

**MÃ SỐ: SV2022 – 29**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn**  *(ký, họ tên)*  **Tăng Cẩm Nhung** | **Chủ nhiệm đề tài**  *(ký, họ tên)*  **Nguyễn Thọ Duy** |
| **Xác nhận của tổ chức chủ trì**  **KT. HIỆU TRƯỞNG**  **PHÓ HIỆU TRƯỞNG**  **PGS.TS. Vũ Ngọc Pi** | |

**Thái Nguyên, tháng 3/ 2023**

|  |
| --- |
| **TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP**  **Đơn vị: Khoa Điện Tử**  **------o0o--------** |

**THÔNG TIN KẾT QUẢ NGHIÊN CỨU**

**1. Thông tin chung**

**-** Tên đề tài: **Nghiên cứu và xây dựng mô hình điều khiển đa hệ vi điều khiển phân tán thông qua mạng nội bộ LAN.**

- Mã số: **SV2022-29**

- Chủ nhiệm đề tài: Nguyễn Thọ Duy

- Giáo viên hướng dẫn: ThS. Tăng Cẩm Nhung

**2. Mục tiêu đề tài**

- Nghiên cứu truyền thông, giao tiếp giữa vi điều khiển và rasberry pi thông qua mạng nội bộ Lan

- Kết nối truyền nhận dữ liệu và tín hiệu điều khiển

**3. Kết quả nghiên cứu:**

- Kết nối đa vi điều khiển với Raspberry pi

- Điều khiển và truyền nhận dữ liệu 2 chiều

**4. Sản phẩm**

- Phương pháp điều khiển các thiết bị dùng Vi điều khiển ESP8266

**-** Website điều khiển thiết bị thông qua Lan

Ngày tháng năm 2023

|  |  |  |
| --- | --- | --- |
| **Cơ quan chủ trì**  **KT. HIỆU TRƯỞNG**  **PHÓ HIỆU TRƯỞNG**  **PGS.TS. Vũ Ngọc Pi** | **Giáo viên hướng dẫn**  **Tăng Cẩm Nhung** | **Chủ nhiệm đề tài**  **Nguyễn Thọ Duy** |

# MỤC LỤC

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 2](#_Toc131006390)

[1.1. Giới thiệu chung về vi điều khiển phân tán và đa vi điều khiển phân tán 2](#_Toc131006391)

[1.2. Tính cấp thiết của đề tài 3](#_Toc131006392)

[1.3. Phạm vi và Mục tiêu nghiên cứu 3](#_Toc131006393)

[1.4. Tính mới và sáng tạo của sản phẩm 4](#_Toc131006394)

[1.5. Đánh giá lợi ích thu được 4](#_Toc131006395)

[1.6. Sự đóng góp và khả năng áp dụng của sản phẩm 4](#_Toc131006396)

[1.6.1. Đóng góp về giáo dục và đào tạo 5](#_Toc131006397)

[1.6.2. Khả năng áp dụng của sản phẩm 5](#_Toc131006398)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 6](#_Toc131006399)

[2.1. Giới thiệu chung về thiết bị phần cứng 6](#_Toc131006400)

[2.1.1. Vi điều khiển ESP8266 6](#_Toc131006401)

[2.1.2. Mạch điều khiển Raspberry pi 7](#_Toc131006402)

[2.2. Giới thiệu về thiết bị phần mềm 8](#_Toc131006403)

[2.2.1. Web Server 8](#_Toc131006404)

[2.2.2. Cách thức xây dựng một hệ thống Web điều khiển 9](#_Toc131006405)

[CHƯƠNG 3. THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN QUA MẠNG NỘI BỘ LAN 12](#_Toc131006406)

[3.1. Sơ đồ mạch thiết kế 12](#_Toc131006407)

[3.2. Thiết kế giao diện website 13](#_Toc131006408)

[3.2.1. Môi trường sử dụng và công nghệ sử dụng 13](#_Toc131006409)

[3.2.2. Thiết kế giao diện 21](#_Toc131006410)

[3.3. Thiết kế mạch phần cứng 24](#_Toc131006411)

[3.3.1. Sơ đồ mạch nguyên lý 24](#_Toc131006412)

[3.3.2. Chương trình kết nối giữa ESP và mạng Lan 27](#_Toc131006413)

[CHƯƠNG 4. KIỂM THỬ VÀ ĐÁNH GIÁ 33](#_Toc131006414)

[4.1. Kiểm thử trương trình và đánh giá kết quả 33](#_Toc131006415)

[4.2. Phương hướng phát triển 33](#_Toc131006416)

[TÀI LIỆU THAM KHẢO 36](#_Toc131006417)

# DANH MỤC HÌNH ẢNH

[Hình 1. Mô hình DSC được áp dụng trong công nghiệp 2](#_Toc130996800)

[Hình 2. Sơ đồ cấu trúc tổng quan 5](#_Toc130996801)

[Hình 3. ESP8266 NodeMCU 6](#_Toc130996802)

[Hình 4. Raspberry Pi 7](#_Toc130996803)

[Hình 5. Danh sách những Web Server được sử dụng nhiều nhất 8](#_Toc130996804)

[Hình 6. Quy trình hoạt động của Web Server 9](#_Toc130996805)

[Hình 7. Hình ảnh trượng trưng cho việc lên ý tưởng trang Web 10](#_Toc130996806)

[Hình 8. Hình ảnh tượng trưng cho việc lập kế hoạch 10](#_Toc130996807)

[Hình 9. Sơ đồ mạch thiết kế 12](#_Toc130996808)

[Hình 10. Phần mềm Visual Studio Code 14](#_Toc130996809)

[Hình 11. Cài đặt ngôn ngữ python 15](#_Toc130996810)

[Hình 12. Sơ đồ Model View Template (MVT) của Django 16](#_Toc130996811)

[Hình 13. Sơ đồ tổ chức, hoạt động của Ajax 20](#_Toc130996812)

[Hình 14. Thiết kế chương trình 21](#_Toc130996813)

[Hình 15. Giao diện tổng quan các thiết bị 22](#_Toc130996814)

[Hình 16. Giao diện tổng quan khi truy cập vào bằng điện thoại 22](#_Toc130996815)

[Hình 17. Sử dụng Ajax làm công cụ giúp tránh việc phải load lại trang 23](#_Toc130996816)

[Hình 18. Sử dụng postmant để bắt dữ liệu kiểm tra 24](#_Toc130996817)

[Hình 19. Sơ đồ mạch nguyên lý hoạt động 24](#_Toc130996818)

[Hình 20. Chức năng các Menu chính 25](#_Toc130996819)

[Hình 21. Thực hiện thao tác biên dịch chương trình 26](#_Toc130996820)

[Hình 22. Chọn board thích hợp 26](#_Toc130996821)

[Hình 23. Chọn Serial Port 26](#_Toc130996822)

[Hình 24. Sơ đồ thể hiện quá trình chạy chương trình trong Arduino IDE 27](#_Toc130996823)

[Hình 25. Giao diện ban đầu khi sử dụng Arduino IDE 27](#_Toc130996824)

[Hình 26. Thực hiện viết chương trình điều khiển Arduino ESP 28](#_Toc130996825)

[Hình 27. Khai báo mạng Lan sử dụng 29](#_Toc130996826)

[Hình 28. Hàm setup của chương trình 29](#_Toc130996827)

[Hình 29. Hàm loop của chương trình 30](#_Toc130996828)

[Hình 30. Kiểm tra trạng thái của thiết bị 31](#_Toc130996829)

**LỜI MỞ ĐẦU**

Trong thời đại ngày nay, các mạng LAN ngày có khả năng truyền tải dữ liệu nhanh chóng và ổn định hơn, cho phép kết nối đa dạng các thiết bị như máy tính, điện thoại, máy in, máy chủ, cảm biến và actuator, tăng tính linh hoạt và hiệu quả trong việc quản lý và vận hành hệ thống. Ngoài ra, các mạng LAN cũng có tính bảo mật cao hơn, giúp ngăn chặn các cuộc tấn công và truy cập trái phép vào hệ thống.

Không chỉ thế, các mạng LAN còn cho phép kết nối với internet, giúp các tổ chức và doanh nghiệp có thể truy cập vào các nguồn thông tin và dịch vụ trực tuyến một cách dễ dàng và nhanh chóng. Các mạng LAN cũng cho phép chia sẻ tài nguyên như máy in, ổ cứng và thiết bị lưu trữ khác giữa các thiết bị trong hệ thống, tăng tính tiện lợi và giảm chi phí cho việc quản lý tài nguyên.

Chính vì sự tiện ích mà mạng nội bộ Lan mang lại, trong bài nghiên cứu của em, em xin được đưa ra đề tài của mình: “**Nghiên cứu và xây dựng mô hình điều khiển đa hệ vi điều khiển phân tán thông qua mạng nội bộ LAN**” với mục tiêu hướng tới không phải xây dựng những trang website quản lý, mà là xây dựng theo hướng website điều khiển – Một xu hướng của những trang web ngày nay.

Để thực hiện được đề tài này em xin gửi lời cảm ơn chân thành nhất đến tất cả các thầy cô trong bộ môn em suốt thời gian qua. Đặc biệt em xin chân thành cảm ơn cô giáo hướng dẫn **Tăng Cẩm Nhung**, đã nhiệt tình hướng dẫn, chỉ bảo và cung cấp cho em nhiều kiến thức cũng như tài liệu trong suốt quá trình làm đề tài.

Em xin chân thành cảm ơn!

**CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI**

* 1. **Giới thiệu chung về vi điều khiển phân tán và đa vi điều khiển phân tán**

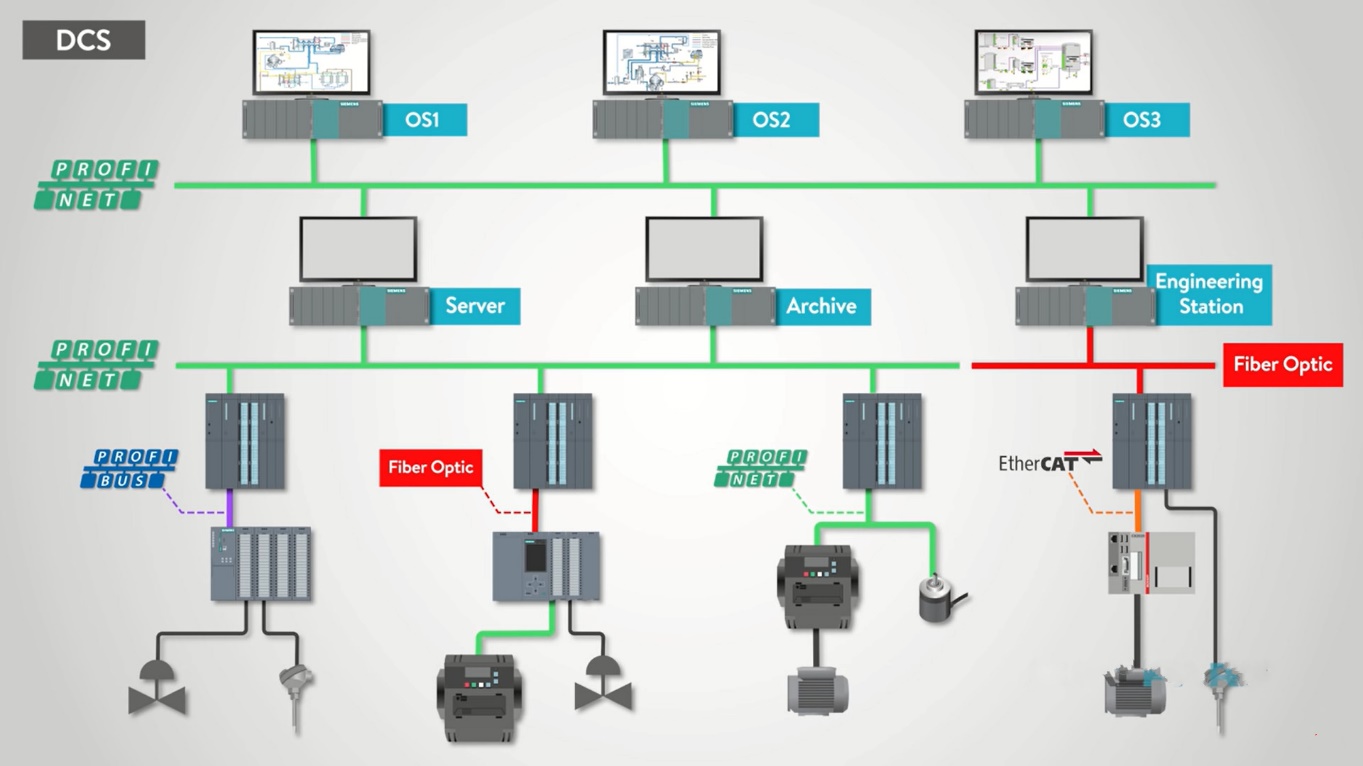
Vi điều khiển là một máy tính được thích hợp trên một chíp và thường được sử dụng để điều khiển các thiết bị điện tử. Vi điều khiển, thực chất là một hệ thống bao gồm một vi sử lý có hiệu suất đủ dùng và giá thành thấp (Khác với các bộ vi sử lý đa năng dùng trong máy tính) kết hợp với các khối ngoại vi như bộ nhớ, các module vào/ra, các module biến đổi số sang tương tự hoặc ngược lại, …

Vi điều khiển thường được sử dụng để xây dựng các hệ thống nhúng. Vi điều khiển thường xuất hiện trong các thiết bị điện, điện tử như máy giặt, lò vi sóng, điện thoại, …

Hầu hết các vi điều khiển ngày nay được xây dựng dựa trên kiến trúc Harvard, kiến trúc này định nghĩa 4 thành phần cần thiết của một hệ thống nhúng. Những thành phần này là lõi CPU, bộ nhớ chương trình (Thông thường là bộ nhớ ROM hoặc bộ nhớ FLASH), bộ nhớ dữ liệu (RAM), hoặc một vài bộ định thời và các cổng vào/ra để giao tiếp với các thiết bị ngoại vi và các môi trường bên ngoài – tất cả các khối này được thiết kế trong một vi mạch thích hợp.

Vi điều khiển phân tán - Distributed control system (DSC). Là một thuật ngữ dùng để mô tả một hệ thống sản xuất, quá trình hoặc bất kỳ loại hệ thống năng động. Mà điểm đặc biệt là các yếu tố điều khiển không phải là trung tâm trong vị trí. Chúng được phân tán trên toàn hệ thống với mỗi thành phần tiểu hệ thống được điều khiển bởi một hoặc nhiều bộ điều khiển.

DSC là một thuật ngữ được sử dụng rất rộng dãi trong nhiều ngành công nghiệp, để giám sát và kiểm soát thiết bị phân tán.



Hình 1. Mô hình DSC được áp dụng trong công nghiệp

**1.2. Tính cấp thiết của đề tài**

Trong thời đại ngày nay, hầu hết các thiết bị công nghiệp đều được kết nối với mạng Internet và mạng Lan, tạo nên một hệ thống kết nối phức tạp và đa dạng. Điều này đặt ra yêu cầu về việc quản lý và vận hành hệ thống, đặc biệt là trong các hệ thống đa vi điều khiển phân tán.. Việc tìm hiểu hay tra cứu thông tin đã không còn là vấn đề quá khó khăn với bất kỳ ai. Tuy nhiên, nếu chỉ dừng lại ở những công việc như tra cứu thông tin đơn giản như vậy, đã dần dần không đáp ứng được nhu cầu sử dụng ngày càng mở rộng của con người.

Từ đó, những Website bán hàng, những Website quản lý dần được mở rộng và phát triển, những kết quả mà web đem lại cho mọi người là rất lớn, dần dần những tính năng này cũng đã trở nên phổ biến và ai cũng để dàng thao tác đem lại những hiệu quả nhất định cho bản thân và cho cộng đồng. Nhưng liệu có nên dừng lại tính năng ở đó?

Câu trả lời là không. Nếu như mạng Lan đã đem lại hiệu quả tích cực với xã hội như vậy, tại sao lại chỉ sử dụng như một phương tiện tra cứu thông tin hay một phương tiện mua sắm, giải trí, … Với tính năng mở rộng nhanh chóng cùng với việc kết nối mọi nơi. Có thể áp dụng điều đó để giám sát và điều khiển thiết bị. Đúng vậy, điều nay hoàn toàn có thể.

Với một gia đình khá giả, cuộc sống điều khiển mọi thứ thiết bị trong gia đình, từ đèn điện, quạt, máy bơm, … đến những doanh nghiệp nhỏ cần quản lý, quan sát trạng thái thiết bị, điều khiển trạng thái thiết bị, … Liệu có phải mua một bộ điều khiển đắt tiền để thực hiện những công việc đó.

Với các doanh nghiệp lớn, có thể áp dụng thành quả của công nghệ hiện nay, là máy PLC để giám sát và điều khiển các thiết bị. Nhưng với các gia đình cá nhân hay các doanh nghiệm nhỏ, số lượng thiết bị lại không nhiều, việc phải tốn một khoản chi phí để mua một chiếc PLC về thật sự là tốn kém và không cần thiết.

Xuất phát từ các phân tích trên, đề tài “**Nghiên cứu và xây dựng mô hình điều khiển hệ đa vi điều khiển phân tán thông qua mạng nội bộ LAN**” hoàn toàn đảm bảo được tính cấp thiết mà thực tế đặt ra.

**1.3. Phạm vi và Mục tiêu nghiên cứu**

Mục tiêu của đề tài là nghiên cứu cơ sở lý thuyết về máy điều khiển phân tán, thiết kế một trang Web hoạt động trong mạng LAN để điều khiển và giám sát thiết bị thực thi trên máy tính, xây dựng phần kết cấu cơ bản giúp tăng cường tính linh hoạt, hiệu quả và tiết kiệm chi phí trong việc quản lý và vận hành các hệ thống đa vi điều khiển.. Kết quả của sản phẩm thu được cần có tính năng: giám sát được thiết bị cần điều khiển, điều khiển được các thiết bị thông qua mạng nội bộ Lan. Từ mục tiêu ban đầu em tiến hành thiết kế lắp đặt trang thiết bị phần cứng đến việc lập trình xây dựng trang Web.

Đề tài “**Nghiên cứu và xây dựng mô hình điều khiển hệ đa vi điều khiển phân tán thông qua mạng nội bộ LAN”** gồm những nội dung chính sau:

- Xây dựng ý tưởng cho đề tài.

- Thiết kế mô tả bản vẽ, bản lắp đặt các trang thiết bị phần cứng.

- Tìm hiểu về lập trình Web, ngôn ngữ lập trình C cho Arduino, trình biên dịch Arduino IDE, sử dụng ngôn ngữ python (Djago để thiết kế và lập trình web)

- Tìm hiểu về các linh kiện điện tử chính: ESP8266, Raspberry pi.

- Tìm hiểu về việc xây dựng một trang web server trên raspberry pi.

**1.4. Tính mới và sáng tạo của sản phẩm**

Tính mới của đề tài là đã áp dụng nhiều kỹ thuật, nhiều chuyên ngành như tin học, lập trình web, thiết kế xây dựng web server, …

Mạng LAN được sử dụng để liên kết và truyền thông giữa các thiết bị điều khiển trong hệ thống đa vi điều khiển phân tán. Điều này giúp tăng tính linh hoạt của hệ thống, cho phép các thiết bị điều khiển được kết nối và quản lý từ xa, giảm thiểu chi phí và thời gian vận hành, tăng tính bảo mật do hệ thống chỉ hoạt động trong mạng nội bộ.

Mô hình đề xuất có tính ứng dụng cao trong các ứng dụng thực tế, bao gồm các hệ thống tự động hóa, hệ thống điều khiển công nghiệp, hệ thống điều khiển truyền thông và các hệ thống thông tin khác.

Mô hình đề xuất giúp tăng hiệu quả và tiết kiệm chi phí trong việc quản lý và vận hành các hệ thống đa vi điều khiển, giúp tăng tính cạnh tranh và nâng cao hiệu suất sản xuất.

**1.5. Đánh giá lợi ích thu được**

Khi thực hiện được thành công đề tài cũng đồng nghĩa với việc làm chủ được việc thiết kế, xây dựng và thực hiện lập trình web điều khiển, từ đó:

- Đem lại lợi ích cho những gia đình, những doanh nghiệp có nhu cầu quan sát và điều khiển các thiết bị thay vì phải bỏ ra chi phí lắp đặt các thiết bị đắt tiền, tốn kém.

- Có kinh nghiệm và kỹ năng trong:

+ Xây dựng và thiết kế các trang web điều khiển, một lĩnh vực mới trong lĩnh vực thiết kế web.

+ Cho phép điều khiển và giám sát thiết bị từ xa thông qua mạng nội bộ Lan.

**1.6. Sự đóng góp và khả năng áp dụng của sản phẩm**

*1.6.1. Đóng góp về mặt kinh tế - xã hội*

Nâng cao chất lượng sản xuất: Thông qua việc giám sát và điều khiển thiết bị, giúp quá trình sản xuất diễn ra nhanh chóng, đễ dàng chỉ với một vài thao tác đơn giản.

Độ chính xác cao và nhanh chóng, an toàn: Việc giám sát và điều khiển từ xa giúp con người có thể tránh xa các khu vực làm việc nguy hiểm, độc hại. Mà lại vẫn có thể giám sát một cách dễ dàng, nhanh chóng.

*1.6.1. Đóng góp về giáo dục và đào tạo*

Trong sự nghiệp xây dựng đất nước công nghiệp hoá, hiện đại hoá đất nước ngày nay, xã hội ta ngày một phát triển, cùng với đó là sự hiểu biết về trình độ thì khả năng chuyên môn, việc áp dụng này sẽ giúp cho con người có nhiều thời gian để nghiên cứu và phát triển những lĩnh vực, những nội dung khác một cách có hiệu quả hơn.

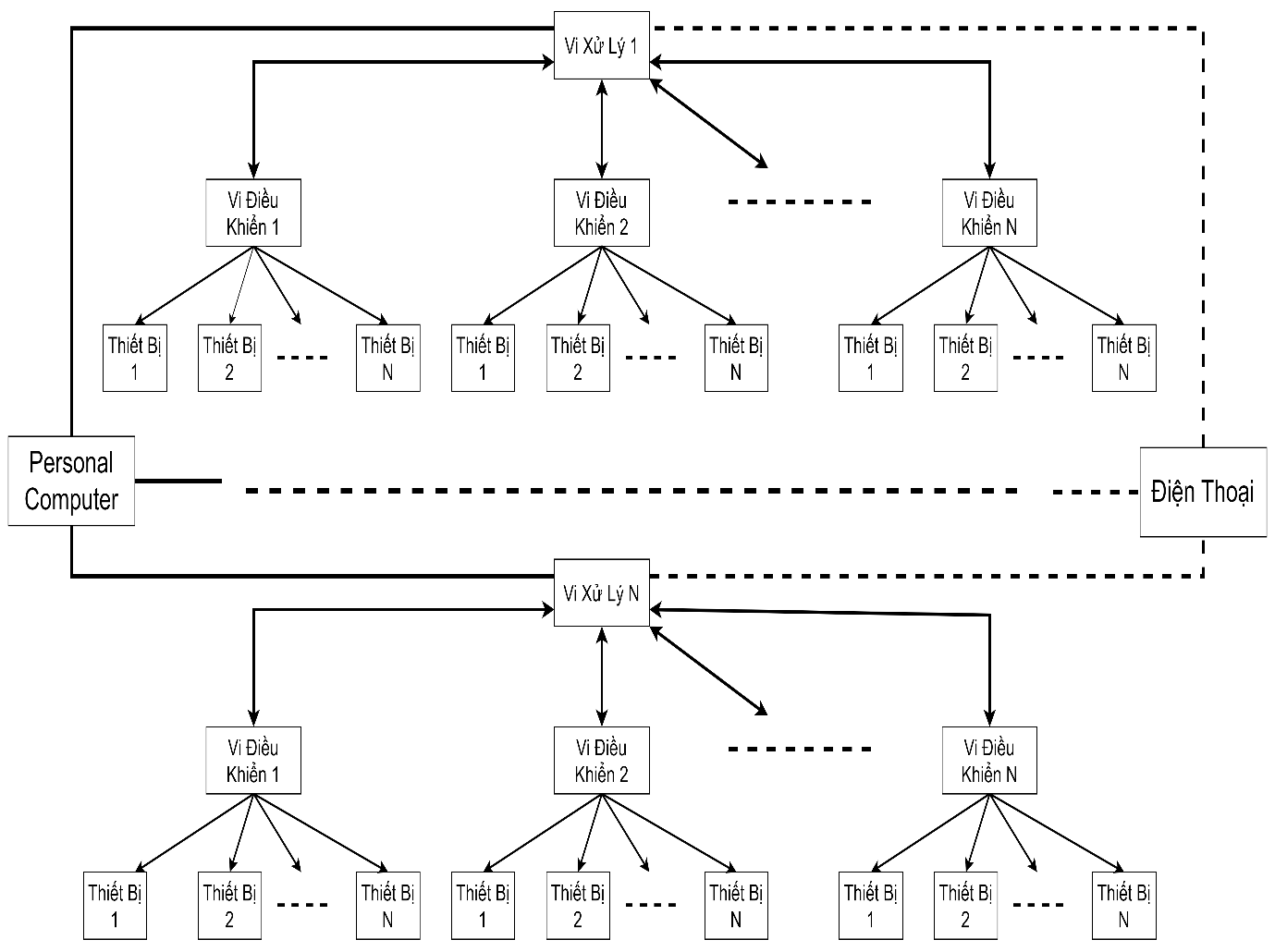
1.6.2. Khả năng áp dụng của sản phẩm

Phạm vi ứng dụng của đề tài là vô cùng rộng, có thể được dùng riêng cho mỗi cá nhân hay có thể vận hành rộng rãi với công suất lớn trong công nghiệp, có thể kể đến như:

- Áp dụng cho các doanh nghiệp, các công ty: Vừa quan sát thiết bị từ xa, vừa đảm bảo tính anh toàn, có thể điều kiển các thiết bị ở bất kỳ đâu.

- Áp dụng với các nhân: Không còn lo lắng về việc bật, tắt các thiết bị trước khi ra, vào nhà, …

- …



Hình 2. Sơ đồ cấu trúc tổng quan

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

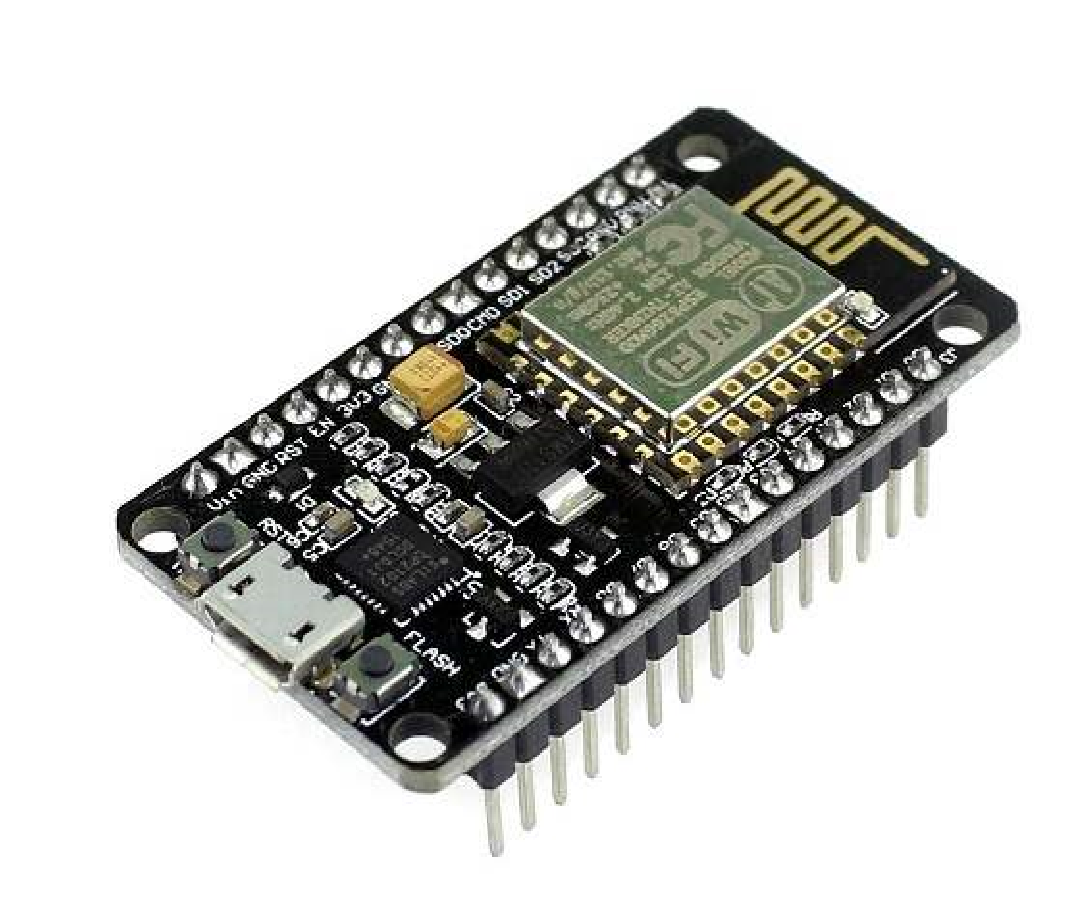
**2.1. Giới thiệu chung về thiết bị phần cứng**

*2.1.1. Vi điều khiển ESP8266*

ESP8266 là một vi mạch trong gói QFN có khả năng của cả bộ TCP/IP và bộ vi điều khiển. ESP8266 cung cấp giải pháp WiFi tích hợp cao đáp ứng nhu cầu của các ứng dụng IoT (LAN of Things) như chi phí thấp, sử dụng năng lượng hiệu quả, hiệu suất đáng tin cậy và thiết kế nhỏ gọn. ESP8266 được sản xuất bởi Espressif Systems ở Thượng Hải, Trung Quốc.

Có khả năng kết nối mạng WiFi hoàn chỉnh, ESP8266 có thể hoạt động như một thiết bị tớ cho một bộ vi điều khiển chủ hoặc như một ứng dụng độc lập. Khi chúng ta nói thiết bị tớ cho vi điều khiển chủ, điều đó có nghĩa là ESP8266 có thể sử dụng làm bộ điều hợp WiFi cho bất kỳ vi điều khiển nào sử dụng giao tiếp SPI hoặc UART. Trong khi sử dụng độc lập, ESP8266 có thể thực hiện được các chức năng của một bộ vi điều khiển và mạng WiFi.

ESP8266 dựa trên dòng L106 Diamond của Tensilica, là bộ xử lý 32-bit và có SRAM trên chip. Đồng thời tích hợp module nguồn, balun RF, bộ thu và phát RF, bộ thu và phát tương tự, băng tần số, bộ khuếch đại, bộ lọc và một số thành phần tối thiểu khác.



Hình 3. ESP8266 NodeMCU

**Thông số kỹ thuật:**

WiFi: 2.4 GHz hỗ trợ chuẩn 802.11 b/g/n

Điện áp hoạt động: 5VDC thông qua cổng microUSB

Số chân I/O: 11 (tất cả các chân I/O đều có Interrupt/PWM/I2C/One-wire, trừ D0)

Số chân Analog Input: 1 (điện áp vào tối đa 3.3V)

Bộ nhớ Flash: 4MB

Giao tiếp: Cable Micro USB (Tương đương cáp sạc điện thoại)

Hỗ trợ bảo mật: WPA/WPA2

Tích hợp giao thức TCP/IP

Lập trình trên các ngôn ngữ: C/C++, MicroPython, Lua

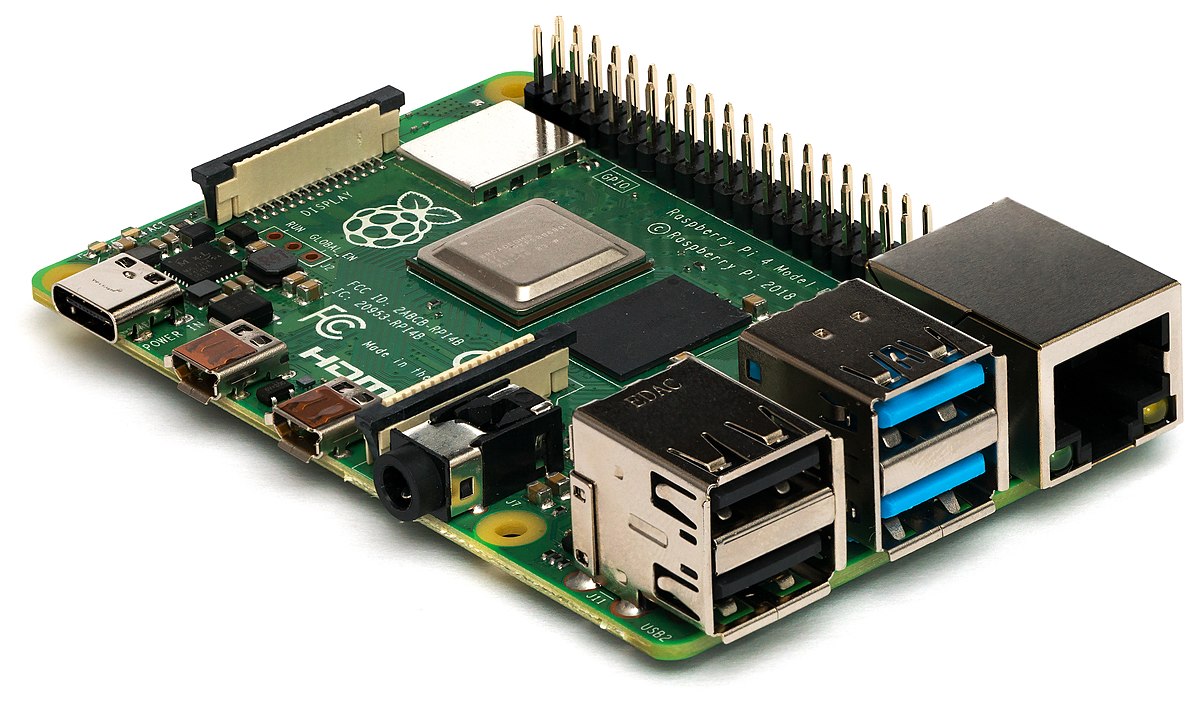
*2.1.2. Mạch điều khiển Raspberry pi*

Raspberry Pi là board máy tính nhúng được sử dụng nhiều nhất hiện nay, ngoài việc sử dụng như một máy tính bình thường chạy hệ điều hành Linux hoặc Windows 10 IoT, máy còn có khả năng xuất tín hiệu ra 40 chân GPIO giúp ta có thể giao tiếp và điểu khiển vô số các board mạch và ngoại vi bên ngoài để thực hiện vô số các ứng dụng khác nhau.

Raspberry Pi 3 Model B+ là sản phẩm mới nhất trong gia đình Raspberry Pi, nổi bật với chip 4 nhân 64-bit có tốc độ 1.4GHz – nhanh nhất từ trước đến nay! Phiên bản mới còn hỗ trợ Wifi Dual-band 2.4GHz và 5GHz, Bluetooth 4.2/Bluetooth Low Energy, cổng Ethernet tốc độ cao (300Mbps) và Power over Ethernet (PoE) thông qua PoE HAT.

Raspberry Pi 3 Model B+ (Made in UK/Japan) được sản xuất tại UK/Japan với quy trình gia công và linh kiện chất lượng cao, đảm bảo cho việc chạy bền bỉ và lâu dài, máy có kích thước nhỏ gọn, giá thành phải chăng, cách sử dụng dễ dàng, chỉ cần cài hệ điều hành vào thẻ nhớ và cấp nguồn là có thể sử dụng.

Raspberry Pi 3 Model B+ (Made in UK/Japan) có cộng đồng sử dụng rất lớn trên thế giới, đây chính là ưu điểm lớn nhất của Raspberry Pi, điều này giúp các bạn có thể tìm nguồn tài liệu cũng như hỗ trợ rất dễ dàng trên Google hoặc trang chủ Raspberry Pi.



Hình 4. Raspberry Pi

**Các cải tiến của Raspberry Pi 3 Model B+ so với các phiên bản cũ:**

- Nâng cấp CPU phiên bản mới BCM2837B0 từ Boardcom với vỏ bọc kim loại tản nhiệt cho khả năng tăng tốc độ lên đến 1.4Ghz 4 nhân với kiến trúc ARM Cortex-A53 64-bit.

- Tốc độ Ethernet nhanh hơn (Gigabit via USB) lên đến 300Mb/s bằng việc nâng cấp chip xử lý mới LAN7515 từ Microchip.

- Nâng cấp Wifi Dual-band 2.4GHz & 5GHz với vỏ bọc IC kim loại chống nhiễu và tản nhiệt cho tốc độ và độ ổn định cao hơn.

- Hỗ trợ Socket cắm Power over Ethernet (PoE) sử dụng với loại Raspberry Pi HAT tương thích.

- Nâng cấp IC quản lý nguồn thông minh MXL7704 giúp đơn giản hóa phần thiết kế nguồn trên board cho khả năng cấp nguồn ổn định và an toàn hơn.

**Thông số kỹ thuật chi tiết của Raspberry Pi 3 Model B+:**

- Vi xử lý: Broadcom BCM2837B0, quad-core A53 (ARMv8) 64-bit SoC @1.4GHz

- RAM: 1GB LPDDR2 SDRAM

- Kết nối: 2.4GHz and 5GHz IEEE 802.11 b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, Gigabit Ethernet over USB 2.0 (Tối đa 300Mbps).

- Cổng USB: 4 x 2.0

- Mở rộng: 40-pin GPIO

- Video và âm thanh: 1 cổng full-sized HDMI, Cổng MIPI DSI Display, cổng MIPI CSI Camera, cổng stereo output và composite video 4 chân.

- Multimedia: H.264, MPEG-4 decode (1080p30), H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics.

- Lưu trữ: MicroSD

- Nguồn điện sử dụng: 5V/2.5A DC cổng microUSB, 5V DC trên chân GPIO, Power over Ethernet (PoE).

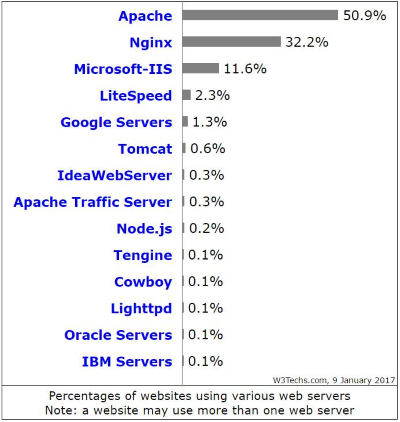
**2.2. Giới thiệu về thiết bị phần mềm**

*2.2.1. Web Server*

**a) Giới thiệu về Web Server**

Web server hay còn gọi là máy chủ web, trong đó được kết nối và liên kết mạng máy tính mở rộng. Máy chủ web được cài đặt các chương trình để phục vụ ứng dụng web, chứa toàn bộ dữ liệu và nắm quyền quản lý. Web server có thể lấy thông tin requess từ phía trình duyệt web và gửi phần hồi tới máy khách thông qua HTTP hoặc giao thức khác.

Những web server được sử dụng nhiều nhất hiện nay: Apache, Nginx, IIS...



Hình 5. Danh sách những Web Server được sử dụng nhiều nhất

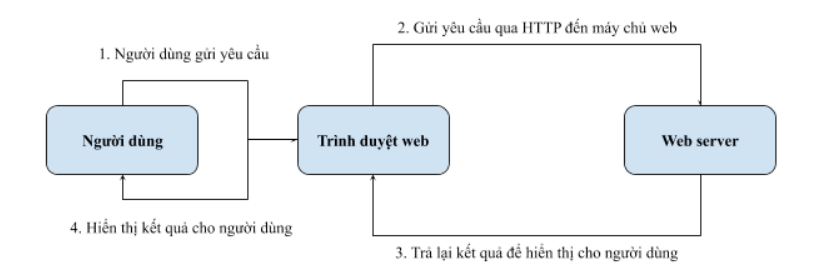
***b) Chức năng của Web Server***

- Xử lý dữ liệu qua giao thức HTTP: Xử lý và cung cấp thông tin cho khách hàng không qua các máy tính cá nhân trên LAN qua giao thức HTTP. Nội dung được chia sẻ từ máy chủ web server là những nội dung định dạng HTML, các thẻ style sheets, hình ảnh và những đoạn mã script hỗ trợ nội dung văn bản.

- Kết nối linh hoạt: Máy tính nào cũng có thể là một máy chủ nếu có được cài đặt một chương trình phần mềm server và có kết nối LAN.

- Kết nối linh hoạt: Máy tính nào cũng có thể là một máy chủ nếu có được cài đặt một chương trình phần mềm server và có kết nối LAN.

***c) Cách thức hoạt động của Web Server***



Hình 6. Quy trình hoạt động của Web Server

Mô hình hoạt động cơ bản nhất của 1 Web Server bao gồm có 3 thành phần chính: Web Server, trình duyệt web và giao thức HTTP.

Cách thức hoạt động của 1 Web Server bao gồm:

* Bước 1: Người dùng gửi yêu cầu.
* Bước 2: Trình duyệt gửi yêu cầu của người dùng tới Web Server để xử lý.
* Bước 3: Server thực hiện kiểm tra, trả về kết quả và trình duyệt hiển thị kết quả cho người dùng.

***d) Những điều cần lưu ý khi xây dựng Web Server***

Khi làm việc và hợp tác Web Server, cần phải lưu ý những điều sau đây:

* Cần chuẩn bị một máy tính có dung lượng lớn, cấu hình cao cũng như đáp ứng được một lượng lớn người dùng truy cập vào website cùng một lúc. Khi xây dựng một Web Server là đang xây dựng một ứng dụng giúp người truy cập vào trang, có thể tìm kiếm và tra cứu kết quả một cách nhanh chóng trên website.
* Để Web Server hoạt động tốt cần phải cho chạy 24/24, không được ngắt quãng nhằm giúp web hoạt động trực tuyến một cách tốt nhất cũng như phục vụ cho việc cung cấp thông tin đến người tiêu dùng.

*2.2.2. Cách thức xây dựng một hệ thống Web điều khiển*

**Bước 1**: Lên ý tưởng về trang web

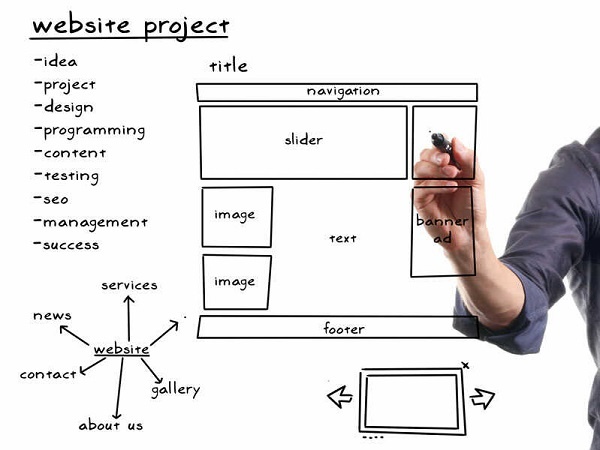


Hình 7. Hình ảnh trượng trưng cho việc lên ý tưởng trang Web

Việc đầu tiên trong quá trình thiết kế, xây dựng 1 hệ thống web (1 trang web) nói chung và web điều khiển nói riêng là việc thiết kế và xây dựng ý tưởng cho trang web muốn thực hiện. Cần phải xác định được một số vấn đề như:

* Website sẽ có những tính năng gì: Những website khác nhau sẽ có những tính năng khác nhau
* Yêu cầu về tính mỹ thuật: Yêu cầu tính mỹ thuật sẽ thể hiện qua giao diện màn hình như màu sắc phồng nền, phông chữ, cách bố trí các tính năng, … sao cho phù hợp với nội dung trang web.
* Xác định đối tượng phục vụ: Tùy thuộc vào các đối tượng khác nhau mà cần xây dựng các tính năng, giao diện khác nhau. Ví dụ: Đối tượng là một cá nhân cụ thể thì nên xây dựng trang web thiên về tính tiện ích, nhanh chóng, và đơn giản, tránh cầu kỳ, … Còn nếu đối tượng là một doanh nghiệp, cần thiết kế theo hướng phù hợp với doanh nghiệp, tính bao quát tổng quan, …

**Bước 2:** Lập kế hoạch



Hình 8. Hình ảnh tượng trưng cho việc lập kế hoạch

Sau khi đã xác định đầy đủ được những thông tin trong bước 1, tiếp theo sẽ tiến tới quá trình lập kế hoạch cụ thể:

* Phác thảo sơ đồ trang web.
* Phác thảo sơ đồ mạch, sơ đồ thiết kế phần cứng.
* Liệt kê toàn bộ các tính năng, chủ đề sẽ xuất hiện trong trang web.
* Xác định các thuộc tính, yêu cầu của từng tính năng, chủ đề.

**Bước 3:** Thi công, lắp đặt các thiết bị phần cứng

Việc quan trọng cần lưu ý rằng, một trang web điều khiển phải xây dựng dựa trên các thiết bị phần cứng.

**Bước 4:** Thiết kế giao diện cho trang Web

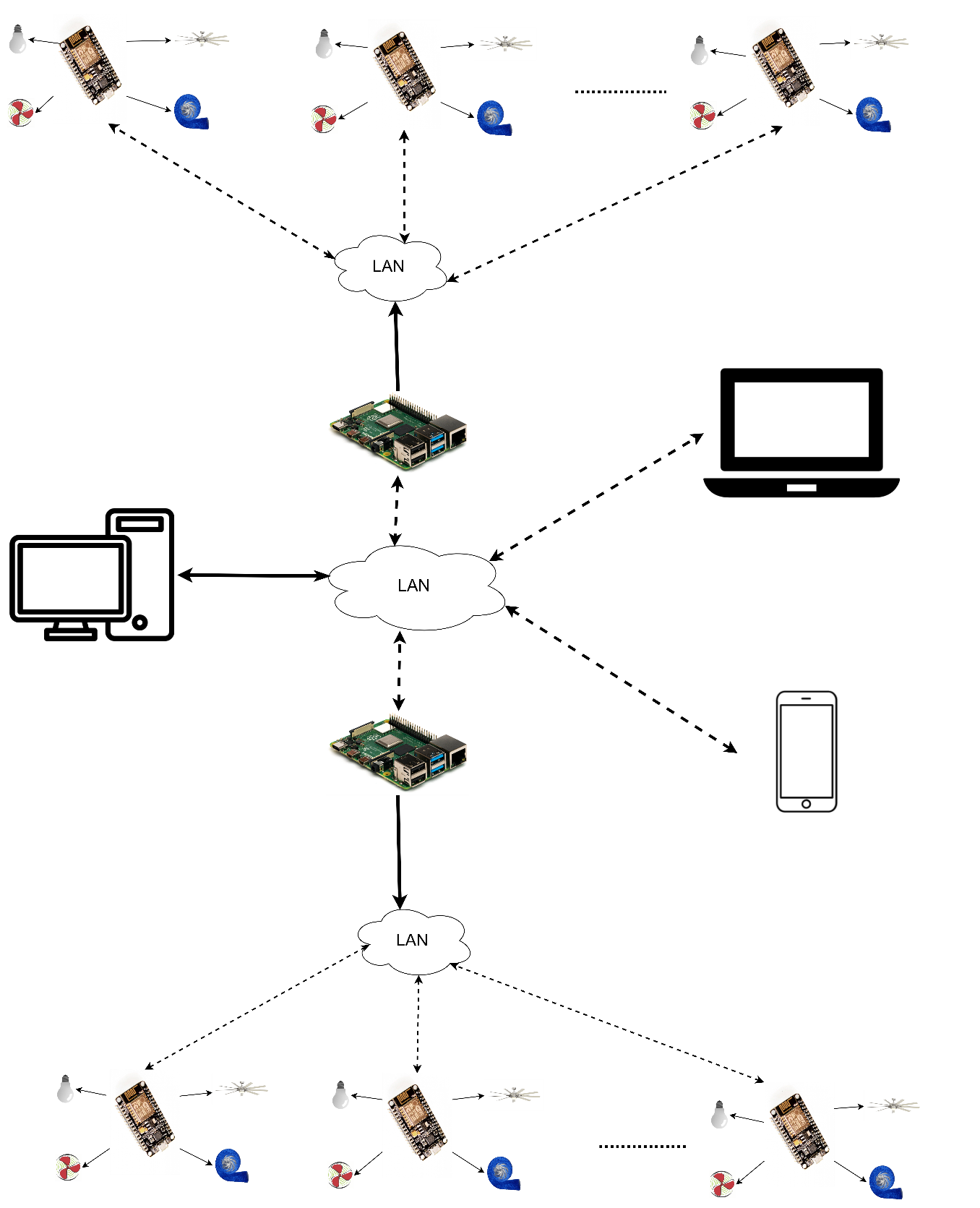
**Bước 5:** Xây dựng các tính năng cho trang Web

**Bước 6:** Thực hiện kiểm tra và chỉnh sửa

**Bước 7:** Hosting trang Web

**CHƯƠNG 3. THIẾT KẾ HỆ THỐNG ĐIỀU KHIỂN QUA MẠNG NỘI BỘ LAN**

**3.1. Sơ đồ mạch thiết kế**



Hình 9. Sơ đồ mạch thiết kế

Raspberry pi được cài đặt một web server để cung cấp giao diện web để điều khiển các thiết bị khác. Web server này sử dụng giao thức http để truyền tải dữ liệu giữa raspberry pi, laptop và điện thoại.

ESP8266 được lập trình để kết nối với mạng lan và gửi dữ liệu đến web server trên raspberry pi. ESP8266 sử dụng giao thức tcp/ip để truyền tải dữ liệu giữa ESP8266 và raspberry pi.

Khi esp8266 gửi dữ liệu đến web server trên raspberry pi, web server sẽ tiếp nhận và xử lý dữ liệu đó. Dữ liệu này chứa các lệnh điều khiển thiết bị được gửi từ laptop.

Sau khi xử lý dữ liệu, web server sẽ gửi lại thông tin điều khiển cho esp8266 thông qua giao thức tcp/ip. ESP8266 sẽ tiếp nhận và thực hiện các lệnh điều khiển được nhận từ raspberry pi và truyền lại kết quả cho web server trên raspberry pi.

Laptop truy cập vào web server trên raspberry pi để điều khiển các thiết bị khác thông qua giao diện web được cung cấp. Laptop sử dụng giao thức http để truyền tải dữ liệu giữa ESP8266 và web server trên raspberry pi.

**3.2. Thiết kế giao diện website**

*3.2.1. Môi trường sử dụng và công nghệ sử dụng*

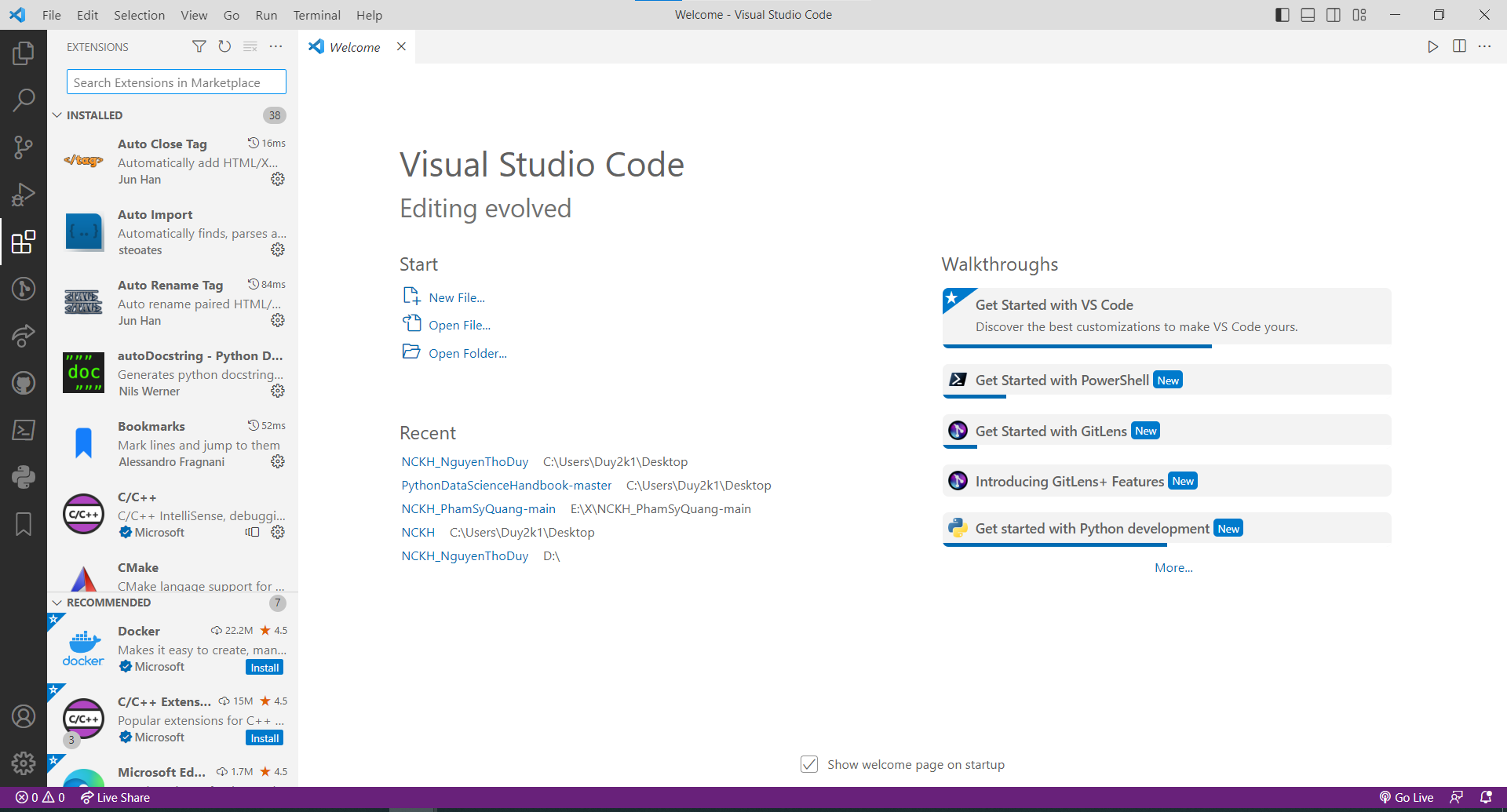
**a) Phầm mềm hỗ trợ lập trình Visual Studio Code**

Để có thể chạy được những chương trình việc đầu tiên chúng ta cần một phần mềm hỗ trợ việc chuyển đổi ngôn ngữ lập trình thành ngôn ngữ máy.

Ở đây chúng ta chọn phần mềm Visual Studio Code để làm những việc này. Ngoài Visual Studio Code thì còn rất nhiều những phầm mềm khác hỗ trợ như: Sublime Text, Visual Studio, PyCharm, Android Studio,…

Đầu tiên, Visual Studio Code là một trình soạn thảo mã nguồn được phát triển bởi Microsoft dành cho Windows, Linux và MacOS. Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có chức năng nổi bật cú pháp (syntax highlighting), tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Visual Studio Code cũng cho phép tùy chỉnh, do đó, người dùng có thể thay đổi theme, phím tắt, và các tùy chọn khác. Visual Studio Code miễn phí và là phần mềm mã nguồn mở theo giấy phép MIT, mặc dù bản phát hành của Microsoft là theo giấy phép phần mềm miễn phí.

Visual Studio Code được dựa trên Electron, một nền tảng được sử dụng để triển khai các ứng dụng Node.js máy tính cá nhân chạy trên động cơ bố trí Blink. Mặc dù Visual Studio Code sử dụng nền tảng Electron nhưng phần mềm này không phải là một bản khác của Atom, thực ra được dựa trên trình biên tập của Visual Studio Online (tên mã là "Monaco"). Điển hình chúng ta có thể thấy trong cuộc khảo sát vào năm 2022 trên Stack Overflow, Visual Studio Code được xếp hạng là trình biên tập mã phổ biến nhất, với 74.48% của 71,010 người trả lời tuyên bố sử dụng. Phần mềm có thể được download tại trang chủ theo link sau: <https://code.visualstudio.com/>



Hình 10. Phần mềm Visual Studio Code

Visual Studio Code mang rất nhiều ưu điểm vượt trội so với bất kỳ IDE nào khác:

+ Hỗ trợ đa nền tảng: Linux, Mac, Windows,...

+ Hỗ trợ đa ngôn ngữ: C/C++, C#, F#, JavaScript, JSON, Visual Basic, HTML, CSS,...

+ Ít dung lượng

+ Tính năng mạnh mẽ

+ Intellisense chuyên nghiệp

+ Giao diện thân thiện

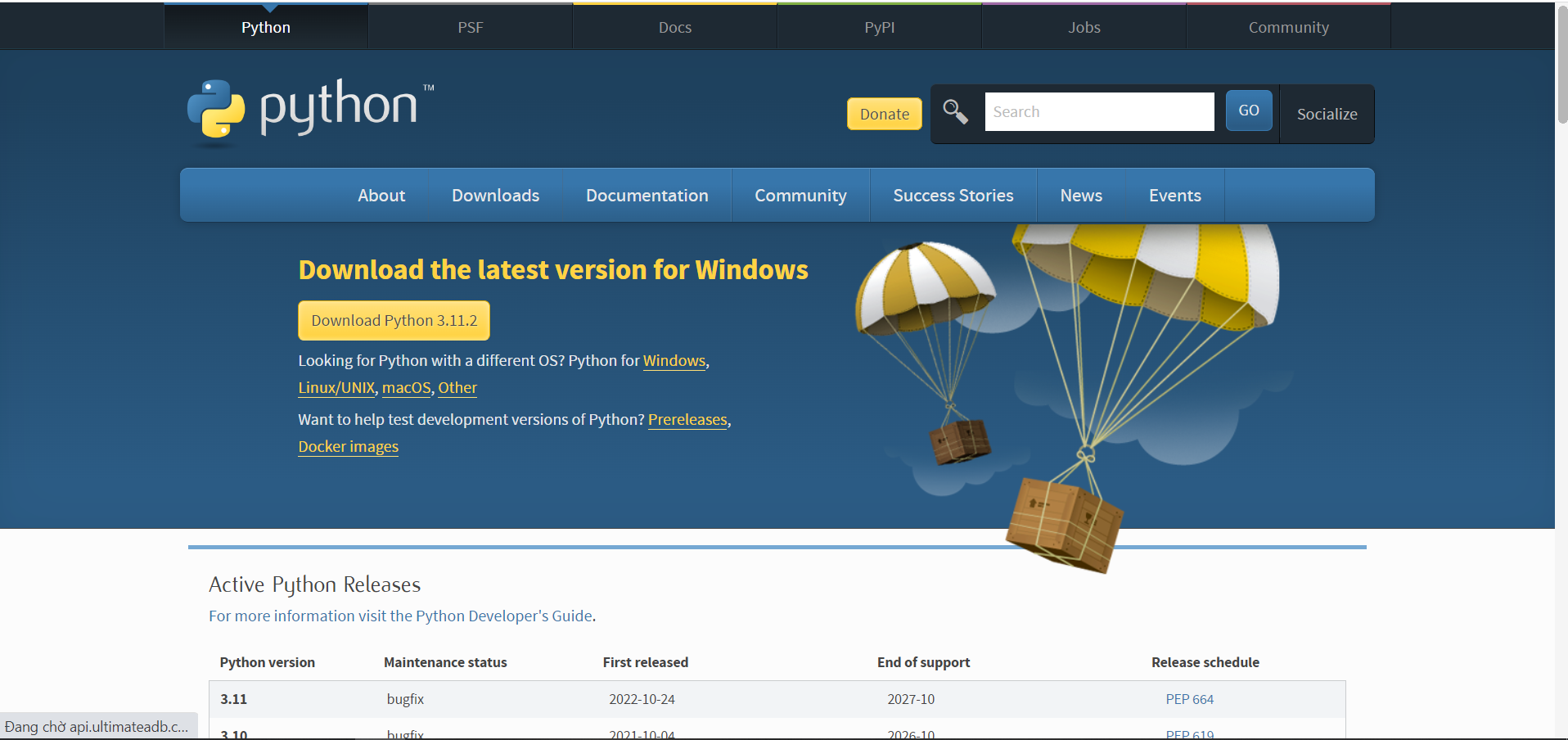
+ Kiến trúc mạnh mẽ và người dùng có thể khai thác mở rộng

+ Số lượng người sử dụng lớn tạo nên ộng đồng hỗ trợ rộng rãi

**b) Lập trình với Django**

Django là một Framework lập trình Web bậc cao, mã nguồn mở được viết bằng ngôn ngữ lập trình Python.

Việc cài đặt Django rất đơn giản. Như đã nói Framework Django được viết bằng ngôn ngữ lập trình Python nên việc đầu tiên là cài đặt môi trường trên máy (Trong trường hợp này chúng ta cài đặt trên Windows 10 và cài phiên bản python là 3.9). Có thể truy cập link sau để cài đặt: <https://www.python.org/downloads/>



Hình 11. Cài đặt ngôn ngữ python

Sau khi đã cài đặt xong môi trường để làm việc với python. Tiếp đến, người dùng có thể sử dụng những gói cài đặt và các thư viện mà sẽ được cài đặt sẵn trong quá trình cài đặt python cho thiết bị của mình. Một trong những gói cài đặt là “pip” – đây là một trình quản lý gói tiêu chuẩn cho python. Pip cho phép người dùng cài đặt và quản lý các gói bổ xung không phải là một phần của thư viện python.

Để có thể cài đặt Django. Người dùng chỉ việc mở termimal của máy tính lên và nhập câu lệnh: “python -m pip install Django” là đã có thể cài đặt Django cho thiết bị của mình.

Django giúp người lập trình làm việc dễ dàng hơn bằng cách nhóm các chức năng khác nhau thành một tập hợp lớn các mô-đun có thể tái sử dụng, được gọi là một khung ứng dụng web. Các nhà phát triển sử dụng khung web Django để sắp xếp và viết mã của họ hiệu quả hơn và giảm đáng kể thời gian phát triển web.

Mục tiêu chính của Django là đơn giản hóa việc tạo các website phức tạp có sử dụng cơ sở dữ liệu. Django tập trung vào tính năng “có thể tái sử dụng” và “có thể tự chạy” của các component, tính năng phát triển nhanh, không làm lại những gì đã làm.

Một số website phổ biến ít có người biết được xây dựng từ Django là Pinterest, Instagram, Mozilla, và Bitbucket, ....

Có một số khung web trên thị trường. Django được viết bằng ngôn ngữ Python và là một trong nhiều khung web Python. Tuy nhiên, các nhà phát triển thường ưu ái khung web Django hơn so với những khung web khác vì các lý do sau.

* Tốc độ phát triển:

Khung Django có cấu trúc hợp lý và dễ cài đặt cũng như học hỏi, vì vậy, mọi người đều có thể bắt đầu sử dụng trong vòng vài giờ. Những nhà thiết kế Django đã tạo ra khung để nhanh chóng triển khai bất kỳ kiến trúc web nào bằng mã. Nhờ có khung hỗ trợ, quá trình phát triển diễn ra nhanh chóng và thiết kế nhìn gọn ghẽ, thực tiễn. Mọi người đều có thể viết mã chỉ bằng vài dòng vì Django cung cấp cấu trúc sẵn sàng sử dụng cho một số tác vụ phát triển web phổ biến, chẳng hạn như:

+ Xác thực người dùng

+ Quản trị nội dung

+ Sơ đồ trang web

+ Nguồn cấp dữ liệu RSS

* Tiết kiệm chi phí:

Django là một dự án Python miễn phí và nguồn mở với một cộng đồng hoạt động tích cực trong việc đánh giá và bảo trì phần mềm. Django Software Foundation là tổ chức phi lợi nhuận thúc đẩy và hỗ trợ việc sử dụng cũng như duy trì Django. Tổ chức này tổ chức các buổi gặp mặt, hội họp và sự kiện cộng đồng thường xuyên nhằm khuyến khích những nhà phát triển khác đánh giá và đóng góp cho dự án Django. Qua đó, khung web thu được có chất lượng cao với tính năng phong phú, miễn phí.

* Phổ biến:

Có hàng nghìn dự án nguồn mở và các trang web tầm cỡ sử dụng Django như:

+ Instagram

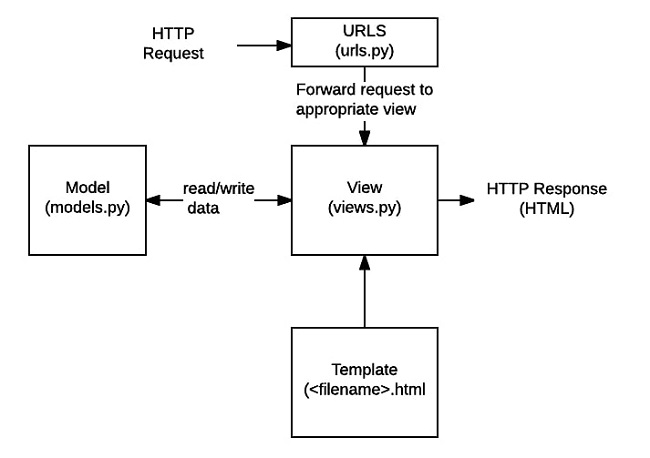
+ Mozilla Firefox

+ Pinterest

+ National Geographic

Vì được sử dụng phổ biến, khung không ngừng phát triển và có cơ sở hạ tầng hỗ trợ vững chắc. Có một lượng không hề nhỏ các cá nhân và công ty cung cấp gói hỗ trợ miễn phí và có trả phí cho bất kỳ thách thức nào trong quá trình phát triển mà mọi người có thể đối mặt khi đang sử dụng Django.

Bất kỳ ứng dụng web nào cũng đều được cấu tạo từ hai phần: mã máy chủ và mã máy khách. Khách thể hoặc đối tượng truy cập trang web có trình duyệt. Khi nhập URL vào trình duyệt của mình, họ sẽ gửi một yêu cầu tới máy chủ web chứa ứng dụng web đang chạy. Các máy chủ xử lý yêu cầu bằng cách sử dụng cơ sở dữ liệu và gửi thông tin ngược về máy khách dưới dạng phản hồi. Mã máy khách hiển thị thông tin cho đối tượng truy cập dưới dạng trang web.



Hình 12. Sơ đồ Model View Template (MVT) của Django

* Mô hình – Model:

Các mô hình Django đóng vai trò là giao diện giữa cơ sở dữ liệu và mã máy chủ. Chúng là nguồn thông tin chính xác duy nhất về dữ liệu của người lập trình. Những mô hình dữ liệu này chứa các trường và thao tác thiết yếu mà người lập trình cần có để tương tác với cơ sở dữ liệu của mình. Do đó, các mô hình Django chuyển đổi bảng cơ sở dữ liệu thành lớp hoặc đối tượng trong mã Python. Quá trình này được gọi là ánh xạ quan hệ đối tượng.

* Chế độ xem – Views:

Khung nhìn của Django sử dụng các mô hình để xử lý yêu cầu. Người lập trình có thể viết hàm khung nhìn cho từng loại yêu cầu mà đối tượng truy cập trang web có thể thực hiện đối với trang web. Hàm khung nhìn có thể lấy yêu cầu làm dữ liệu đầu vào và trả về phản hồi. Phản hồi có thể là mã lỗi, hình ảnh, tệp hoặc bất kỳ loại dữ liệu nào.

* Mẫu – Templates:

Các mẫu Django quản lý cách trình bày trang web trong trình duyệt. Vì hầu hết các trang web đều sử dụng Ngôn ngữ đánh dấu siêu văn bản (HTML), người lập trình có thể viết mã mẫu Django theo cách tương tự như HTML. Tệp mẫu chứa một số thành phần nhất định.

* Tính chất của Django:

+ Độ hoàn thành cao

Django cung cấp hầu hết mọi thứ mà các developer có thể muốn để phát triển web theo hướng mình muốn.

Tất cả các phần trong framework hoạt động liền mạch với nhau, tuân theo một nguyên tắc thiết kế nhất quán và có tài liệu để bạn tham khảo.

Nhờ đó, bất cứ lập trình viên có thể vừa phát triển web theo phong cách riêng, vừa tiết kiệm được thời gian.

+ Tính linh hoạt cao:

Django có thể sử dụng để xây dựng hầu hết mọi loại trang web-từ hệ thống quản lý nội dung (như wiki), cho đến các trang mạng xã hội, tin tức.

Django có thể hoạt động cùng với các framework bên ngoài và cũng có thể cung cấp nội dung ở hầu hết mọi định dạng (bao gồm HTML, RSS feeds, JSON, XML,..v..v..).

Framework này còn cung cấp các tùy chọn khác nhau cho hầu hết các chức năng (như công cụ tạo template, cơ sở dữ liệu phổ biến,..).

+ Khả năng bảo mật:

Django giúp developer tránh được nhiều lỗi bảo mật phổ biến bằng cách cung cấp framework có khả năng tự bảo vệ trang.

Ví dụ: Django cung cấp một giải pháp an toàn để quản lý tài khoản người dùng và mật khẩu (password), tránh những lỗi phổ biến như đưa thông tin phiên vào ở nơi dễ bị tấn công như cookie (thay vào đó cookie chỉ chứa key còn dữ liệu thực tế được lưu trữ trong database) hoặc trực tiếp lưu trữ password chứ không phải lưu trữ password hash.

Django còn bảo vệ website khỏi những lỗ hổng khỏi những loại tấn công mạng như: tấn công SQL injection, Cross-site Scripting, cross-site request forgery và clickjacking.

+ Khả năng mở rộng:

Django sử dụng kiến trúc thành phần riêng nên bạn có thể mở rộng quy mô bằng cách thêm phần cứng vào các cấp độ (máy chủ bộ nhớ đệm, máy chủ cơ sở dữ liệu hoặc máy chủ ứng dụng). Instagram là ví dụ rõ ràng nhất cho khả năng mở rộng này.

+ Khả năng duy trì:

Code của Django được viết bằng cách sử dụng các nguyên tắc và mẫu thiết kế khuyến khích việc tạo mã có thể bảo trì và tái sử dụng.

Django còn thúc đẩy việc nhóm các chức năng liên quan thành các “ứng dụng” có thể tái sử dụng, từ đó giúp website có khả năng duy trì cao hơn.

**c) Công cụ Postman**

Trước tiên thì postman là một công cụ phần mềm được sử dụng để kiểm thử các API (Application Programming Interface) khác nhau. Postman cho phép người dùng gửi các yêu cầu HTTP (GET, POST, PUT, DELETE,....) đến các API và nhận các phản hồi tương ứng từ chúng.

Postman có giao diện đồ họa thân thiện và dễ sử dụng, cho phép người dùng thực hiện các yêu cầu đến các API một cách dễ dàng, hiển thị kết quả trả về trong một cách dễ đọc. Postman cũng cung cấp cho người dùng các tính năng như lưu trữ lịch sử yêu cầu, chia sẻ yêu cầu với đồng nghiệp và thực hiện kiểm thử dữ liệu.

Postman là một công cụ phổ biến trong quá trình phát triển và kiểm thử ứng dụng web và mobile, cho phép người dùng tiết kiệm thời gian và nỗ lực trong việc kiểm thử các API khác nhau.

* Lý do cần kiểm thử API chương trình

Đảm bảo tính đúng đắn và chính xác của dữ liệu: Kiểm thử API giúp đảm bảo rằng dữ liệu được chuyển tới và từ các API là đúng đắn và chính xác.

Kiểm tra tính năng: Kiểm thử API giúp đảm bảo rằng các API hoạt động theo những cách được thiết kế và đáp ứng được các yêu cầu tính năng.

Đảm bảo tính ổn định và khả năng mở rộng: Kiểm thử API giúp đảm bảo rằng các API hoạt động ổn định và có khả năng mở rộng, có thể xử lý tốt các yêu cầu trong tương lai.

Tiết kiệm thời gian và chi phí: Kiểm thử API giúp tiết kiệm thời gian và chi phí trong quá trình phát triển phần mềm, bởi vì postman giúp phát hiện các lỗi và vấn đề sớm hơn, giảm thiểu thời gian và chi phí để sửa chữa sau này.

* Lý do sử dụng Postman:

+Khả năng tiếp cận (Accessibility) - Để sử dụng Postman người dùng chỉ cần đăng nhập vào tài khoản là đã có thể để dễ dàng truy cập các tệp mọi lúc, mọi nơi miễn là ứng dụng Postman được cài đặt trên máy tính.

+ Sử dụng bộ sưu tập (Use of Collections) - Postman cho phép người dùng tạo collections cho các lệnh API của họ. Mỗi một collections có thể tạo nhiều thư mục con và nhiều yêu cầu.

+ Sự hợp tác (Collaboration) - Collections và môi trường có thể nhập hoặc xuất giúp dễ dàng chia sẻ tệp. Một liên kết trực tiếp có thể sử dụng để chia sẻ collections.

+ Tạo môi trường (Creating Environments) - Có nhiều môi trường hỗ trợ ít lặp lại các thử nghiệm nên có thể sử dụng collection cho môi trường khác.

+ Tạo các bản Test (Creation of Tests) - Test checkpoints kiểm tra trạng thái phản hồi HTTP thành công có thể thêm vào mỗi lệnh gọi API giúp đảm bảo phạm vi kiểm tra.

+ Automation Testing - Thông qua việc sử dụng Collection Runner hoặc Newman, kiểm thử có thể chạy nhiều lần lặp lại tiết kiệm thời gian.

+ Debugging - Bảng điều khiển Postman giúp kiểm tra dữ liệu nào đã được truy xuất giúp dễ dàng gỡ lỗi.

+ Continuous Integration - Với khả năng tích hợp liên tục, các hoạt động phát triển được duy trì.

**d) Công nghệ Ajax**

AJAX là chữ viết tắt của Asynchronous JavaScript and XML, AJAX = Asynchronous JavaScript and XML. Đây là một công nghệ giúp lập trình viên tạo ra những Web động mà hoàn toàn không reload lại trang nên rất mượt và đẹp. Vậy Asynchronous, JavaScript, XML trong từ AJAX là gì:

+ Asynchronous, hay nói ngắn hơn là Async – bất đồng bộ. Bất đồng bộ có nghĩa là một chương trình có thể xử lý không theo tuần tự các hàm. Sẽ không có quy trình, có thể nhảy đi bỏ qua bước nào đó. Ích lợi dễ thấy nhất của bất đồng bộ là chương trình có thể xử lý nhiều công việc một lúc.

+ JavaScript là một ngôn ngữ lập trình nổi tiếng. Trong số rất nhiều chức năng của javascript là khả năng quản lý nội dung động của website và hỗ trợ tương tác với người dùng.

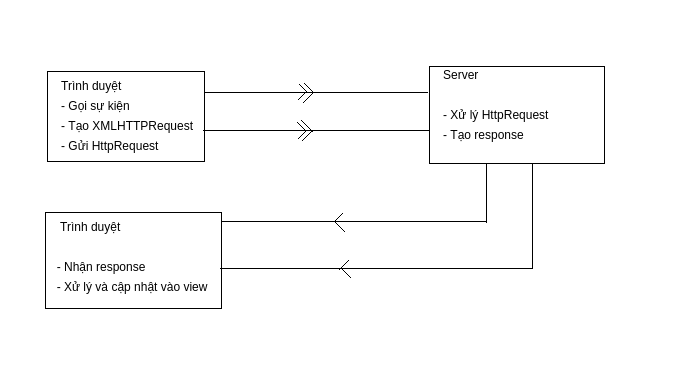
+ XML là một dạng của ngôn ngữ markup như HTML, chữ đầy đủ của XML là eXtensible Markup Language. Nếu HTML được dùng để hiển thị dữ liệu, XML được thiết kế để chứa dữ liệu.

Những lợi ích mà AJAX mang lại:

+ AJAX được sử dụng để thực hiện một callback. Được dùng để thực hiện việc truy xuất dữ liệu hoặc lưu trữ dữ liệu mà không cần phải reload lại toàn bộ trang web. Với những server nhỏ thì việc này cũng tiết kiệm được băng thông.

+ Cần gì thì chỉ gửi dữ liệu phần đó, load lại 1 phần nhỏ để cập nhật thông tin chứ không load cả trang. Bằng cách này thì có thể giảm thiểu được tốc độ tải trang giúp người dùng có trải nghiệm tốt hơn.

+ Trang web có sử dụng Ajax cũng sẽ đa dạng và động hơn.



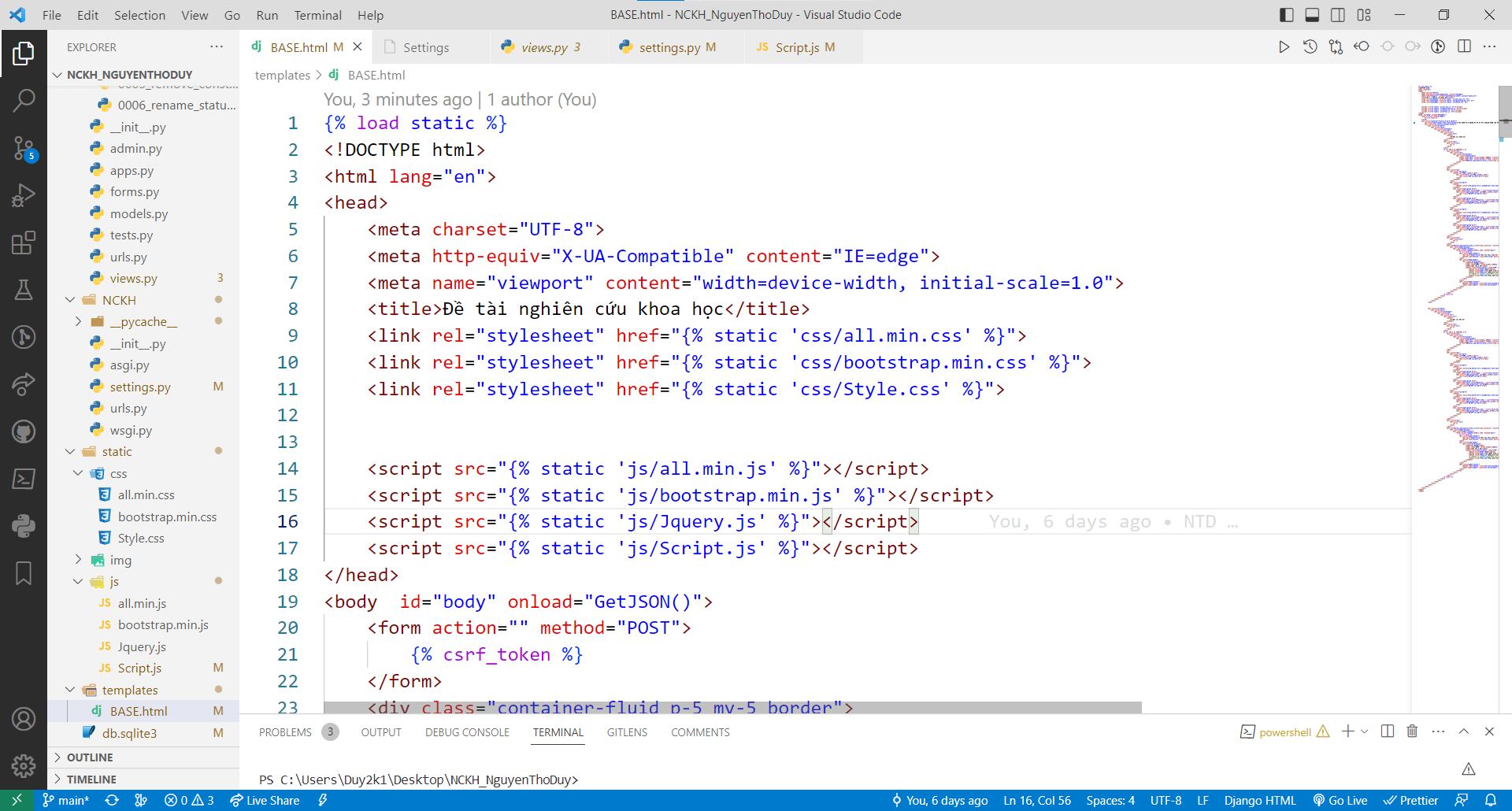
Hình 13. Sơ đồ tổ chức, hoạt động của Ajax

* Từ trình duyệt của client, ta có một sự kiện để gọi ajax. Khi đó javascript sẽ tạo nên một đối tượng XMLHttpRequest. Và đối tượng này sẽ được gửi một request đến server.

Ví dụ: Khi người dùng click vào ô input và chọn skill tìm việc làm của người dùng. Khi đó lập trình viên sẽ lấy thông tin đó và gửi đến server và cần trả về các việc làm tương ứng phù hợp với người dùng.

* Khi server nhận được HttpRequest từ đó sẽ xử lý request và trả về response về cho web. Server xử lí lấy ra các việc làm javascript chẳng hạn thuộc skill mà người dùng cần và trả về dữ liệu của các việc làm.
* Sau khi nhận được một response từ server, Javascript sẽ xử lý và cập nhật vào trang web.

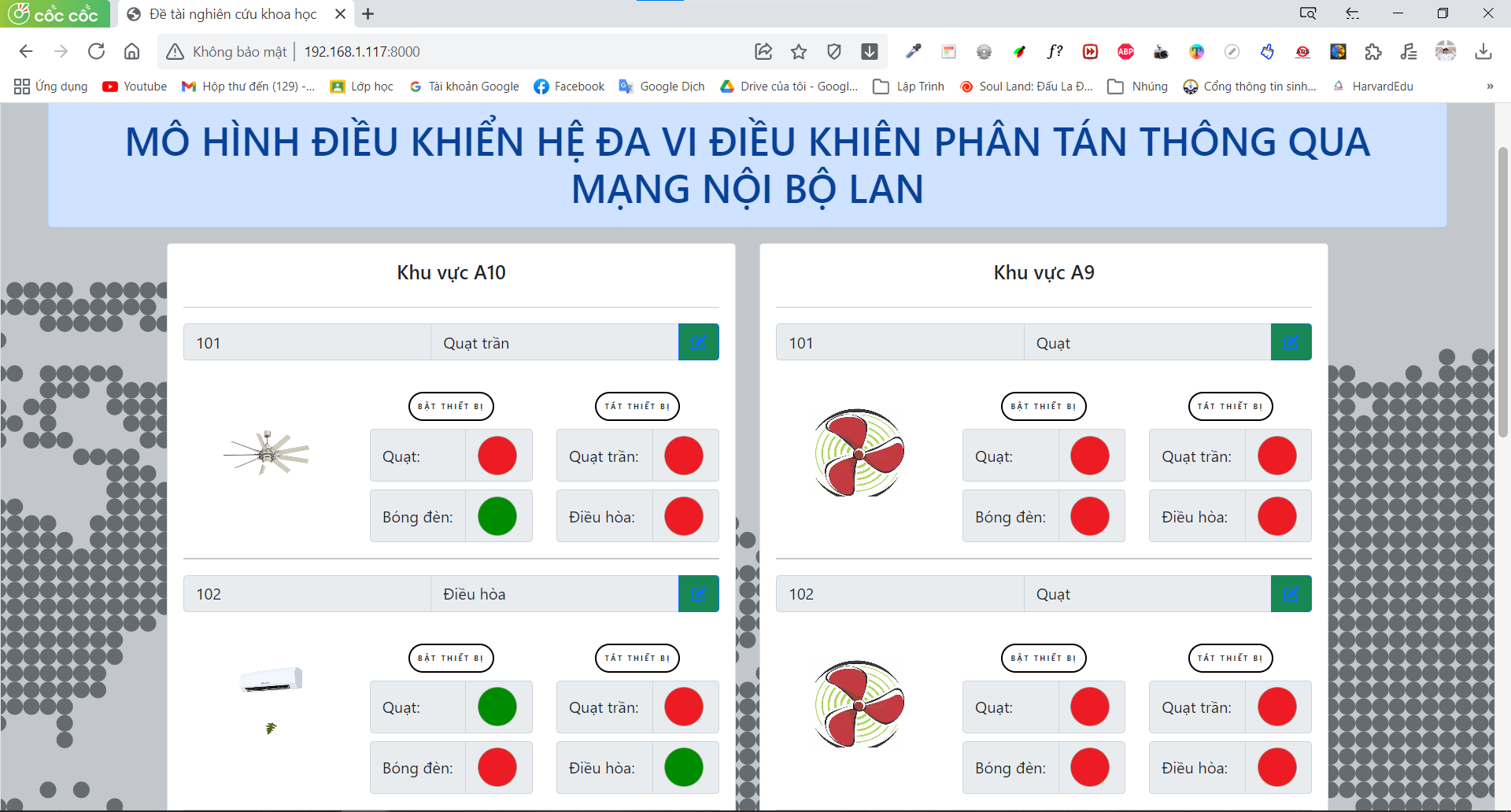
*3.2.2. Thiết kế giao diện*



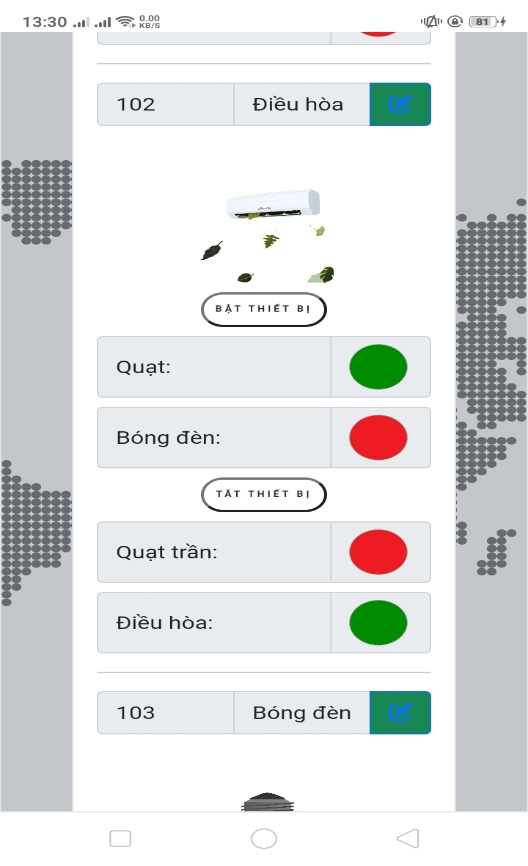
Hình 14. Thiết kế chương trình

Khi thực hiện thao tác “Bật/Tắt” một thiết bị, điều đầu tiên cần phải làm là xác định vị trí mà thiết bị điều khiển ở đâu trong vô số các thiết bị điều khiển. Tại đây, việc này được thực hiện thông qua hệ số ID được xác định sẵn đối với mỗi thiết bị. Khi có sự tương tác của người dùng. Chương trình sẽ gửi 1 request lên trên Server và nhận lại tín hiệu thông qua Response. Nếu tín hiệu gửi trả về nói rằng thiết bị đang ở trạng thái “On/Off” thì sẽ chỉ thực hiện xử lý khi người dùng thay đổi trạng thái của thiết bị. Ngược lại, khi người dùng cố thực hiện thao tác giữ nguyên của thiết bị (Ví dụ, cố bật thiết bị khi thiết bị đang trong trạng thái bật) thì để tránh ảnh hưởng đến thông tin dữ liệu. Thao tác của người dùng vẫn sẽ được tiếp nhận nhưng sẽ không thực hiện xử lý.

Tương tự với trường hợp như vậy, thay vì sử dụng trực tiếp với các thiết bị để giám sát và điều khiển. Cũng có thể kết nối ESP8266 với một vài thiết bị cảm biến như: Cảm biến nhiệt độ, cảm biến độ ẩm, cảm biến ánh sáng, … và cũng áp dụng việc kết nối với ESP để gửi dữ liệu lên trên Internet. Và kết quả là trên thiết bị người dùng đã có thể nhận được những thông tin mà cảm biến đo được.



Hình 15. Giao diện tổng quan các thiết bị



Hình 16. Giao diện tổng quan khi truy cập vào bằng điện thoại

**Các chức năng cơ bản trong giao diện thiết kế:**

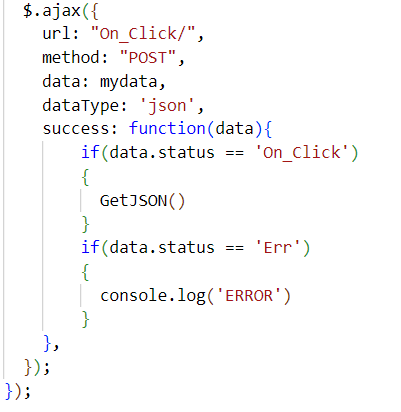
* Trang web điều khiển này được thiết kế chia làm 2 khu vực chính là toà nhà A9 tầng 1 và A10 tầng 1.
* Hiện tại do điều khiện phần cứng nên lắp đặt ở toà nhà A10 các phòng 101, 102, 103 và 104
* Ngoài ra, người dùng còn có thể tự do lựa chọn loại thiết bị điều khiển, ở trang Web này đã thiết kế để người dùng có thể lựa chọn loại thiết bị theo nhu cầu. Sau khi lựa chọn thiết bị thì hình ảnh tượng trưng cho tiết bị cũng sẽ thay đổi sao cho phù hợp với loại thiết bị đã được chọn trước đó. (Các thiết bị được setup sẵn ban đầu bao gồm: Quạt, Quạt trần, Điều hoà, Bóng đèn)

**Lưu ý:**

Sau khi thiết lập lại thông tin cho thiết bị điều khiển (Bao gồm tên thiết bị hoặc loại thiết bị) thì đồng thời, trạng thái điều khiển thiết bị trước đó cũng sẽ được resert về trạng thái ban đầu (Trường hợp nếu thiết bị đang được bật sẽ chyển sang trạng thái tắt) đồng thời, hình ảnh đại diện cho thiết bị cũng sẽ thay đổi thành hình ảnh theo loại thiết bị được chọn và đang ở trạng thái tắt.

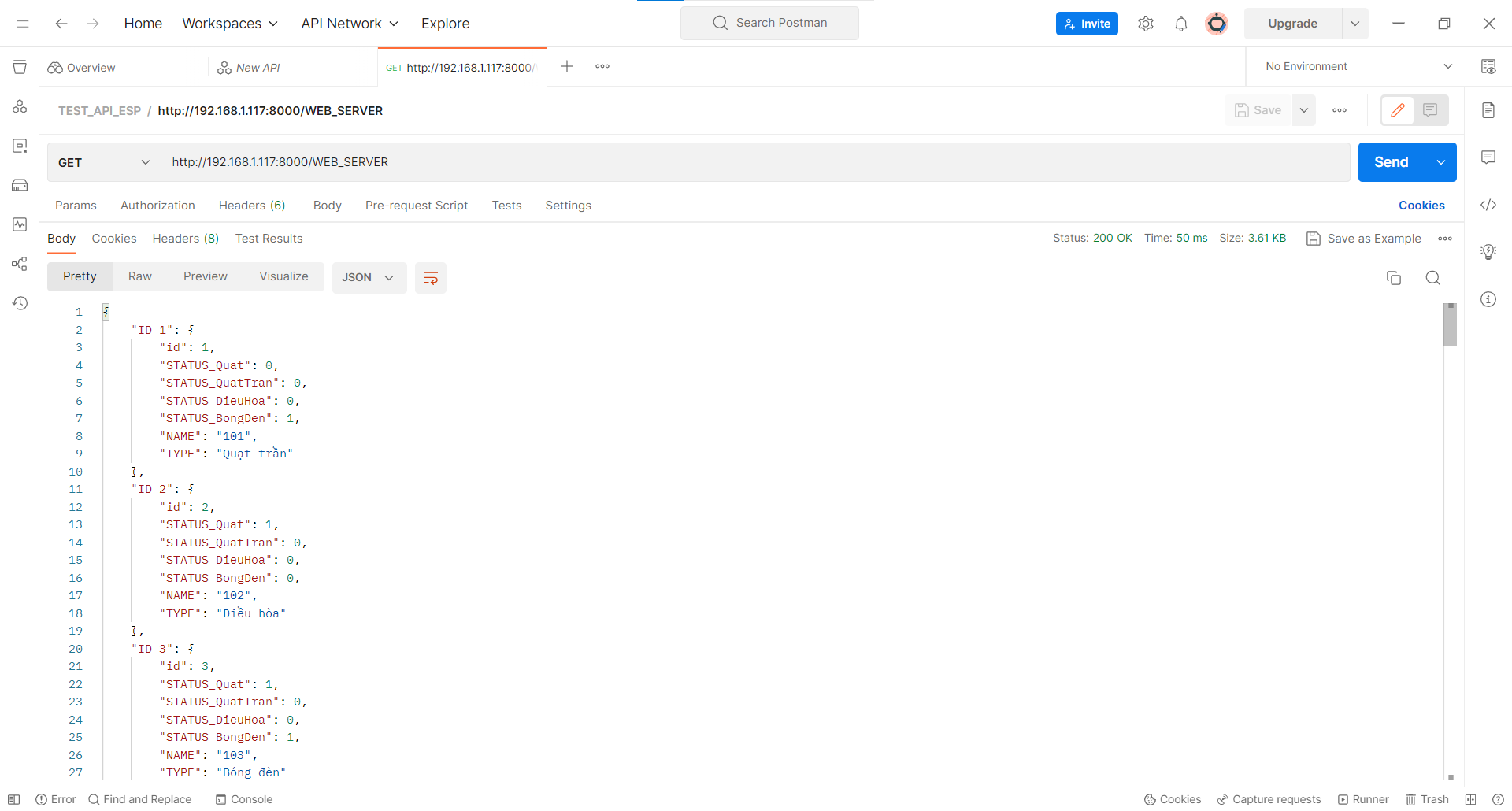
**Ưu điểm của trang web điều khiển:**

* Khác với những trang web thông thường khác, sau khi thay đổi thông số, thực hiện các chức năng bật/tắt thiết bị, trang web sẽ “không” phải thực hiện “tải lại trang (Load lại trang)” để có thể thấy được sự thay đổi. (Vì trang Web được sử dụng “Ajax - một bộ công cụ cho phép load dữ liệu từ server mà không yêu cầu tải lại trang”)



Hình 17. Sử dụng Ajax làm công cụ giúp tránh việc phải load lại trang

* Tốc độ sử lý truyền nhận tín hiệu diễn ra một cách nhanh chóng và ngay lập tức chỉ trong thời gian “click chuột”.
* Sử dụng kiểu dữ liệu “json” giúp bao đóng dữ liệu một cách nhanh chóng và tránh bị nhầm lẫn.
* Dữ liệu được đọc trực tiếp, không có khả năng sai sót hay nhầm lẫn dữ liệu với người dùng



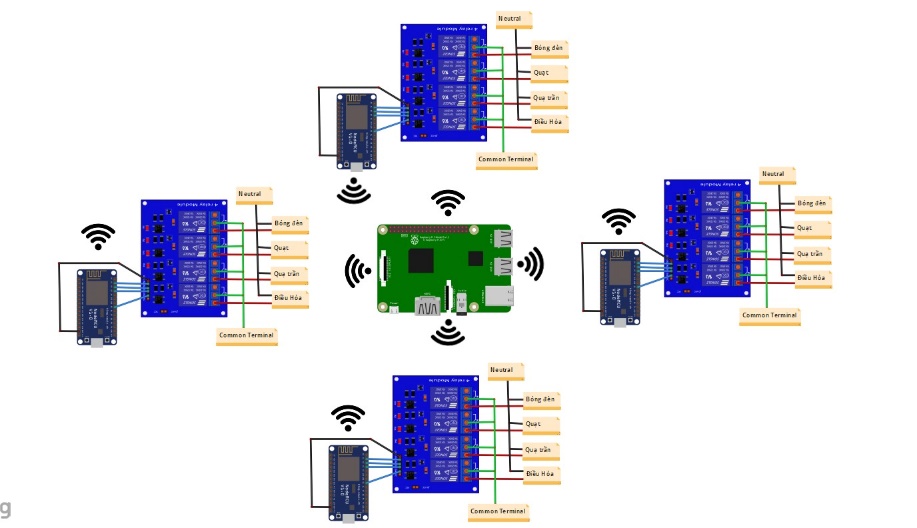
Hình 18. Sử dụng postmant để bắt dữ liệu kiểm tra

Toàn bộ thông tin dữ liệu sẽ được lưu trữ tập trung tại trên một trang Web được raspberry pi tạo ra và lưu trên một địa chỉ cố định, trang web này đóng vai trò như một Server chuyên dành cho các thiết bị. Dữ liệu này được lưu trữ dưới dạng một chuỗi json nhằm bao đóng toàn bộ thông tin dữ liệu.

Nhiệm vụ của Server này là nhằm lưu trữ, cung cấp và thực hiện sử lý dữ liệu. Bản chất, những tác động của người sử dụng (User) đối với các thiết bị, sự tương tác, quan sát, điều khiển, … hay thậm trí, dữ dữ liệu được Esp thu thập, xử lý, điều khiển, gửi dữ liệu, … Toàn bộ đề được thực hiện lên trên Server này. Mọi dữ liệu hiển thị lên trên thiết bị của người dùng (Browser) đều được đọc từ trên trang Server này. Chính vì vậy, những tác động của người dùng (User) lên trên trên thiết bị thông qua trang web có thể diễn ra một cách nhanh chóng và gần như ngay lập tức dù cho có nhiều người cùng truy cập trang web để quan sát hay điều khiển thiết bị. Bởi, thực chất mọi thao tác đều đang thực hiện trên Server này. Sau đó, thiết bị của người dùng mới thực hiện đọc dữ liệu từ Server và hiển thị lên cho người dùng.

**3.3. Thiết kế mạch phần cứng**

*3.3.1. Sơ đồ mạch nguyên lý*

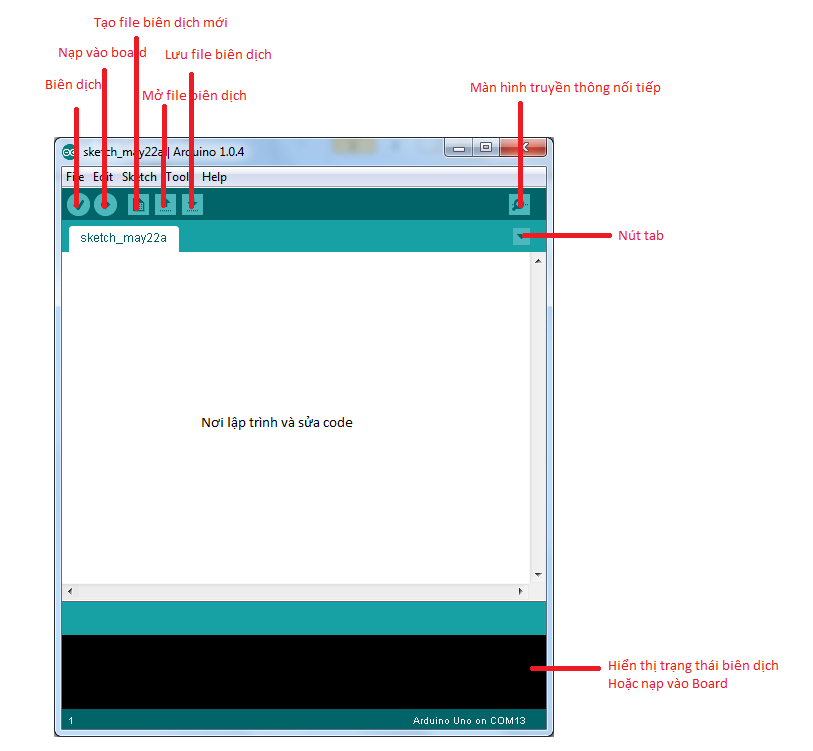


Hình 19. Sơ đồ mạch nguyên lý hoạt động

Đầu tiên, để có thể cho esp8266 hoạt động thì trước hết chúng ta cần cài đặt môi trường lập trình Arduino thì bước đầu tiên các bạn phải có phần mềm IDE chuyên dụng do nhà sản xuất Arduino cung cấp. Phần mềm có thể được download tại trang chủ theo link sau: [*http://arduino.cc/en/Main/Software*](http://arduino.cc/en/Main/Software)

Sau khi đã cài đặt phiên bản mới nhất của Arduino IDE, tục thực hiện các bước sau đây để tiến hành cài đặt thư viện và chức năng nạp code cho Arduino IDE.

* Mở cửa số Preferences từ Arduino IDE. Vào File → Preferences
* Nhập http://arduino.esp8266.com/stable/package\_esp8266com\_index.json vào ô “Additional Board Manager URLs” như được hiển thị trong hình bên dưới. Sau đó, nhấp vào nút “OK”:
* Tiếp theo vào Tools > Board > Boards Manager…
* Sau khi chương trình tìm kiếm xong, bạn kéo xuống dưới và nhấp vào ESP8266 by ESP8266 Community, click vào Install. Chờ phần mềm tự động download và cài đặt trong vài giây.



Hình 20. Chức năng các Menu chính

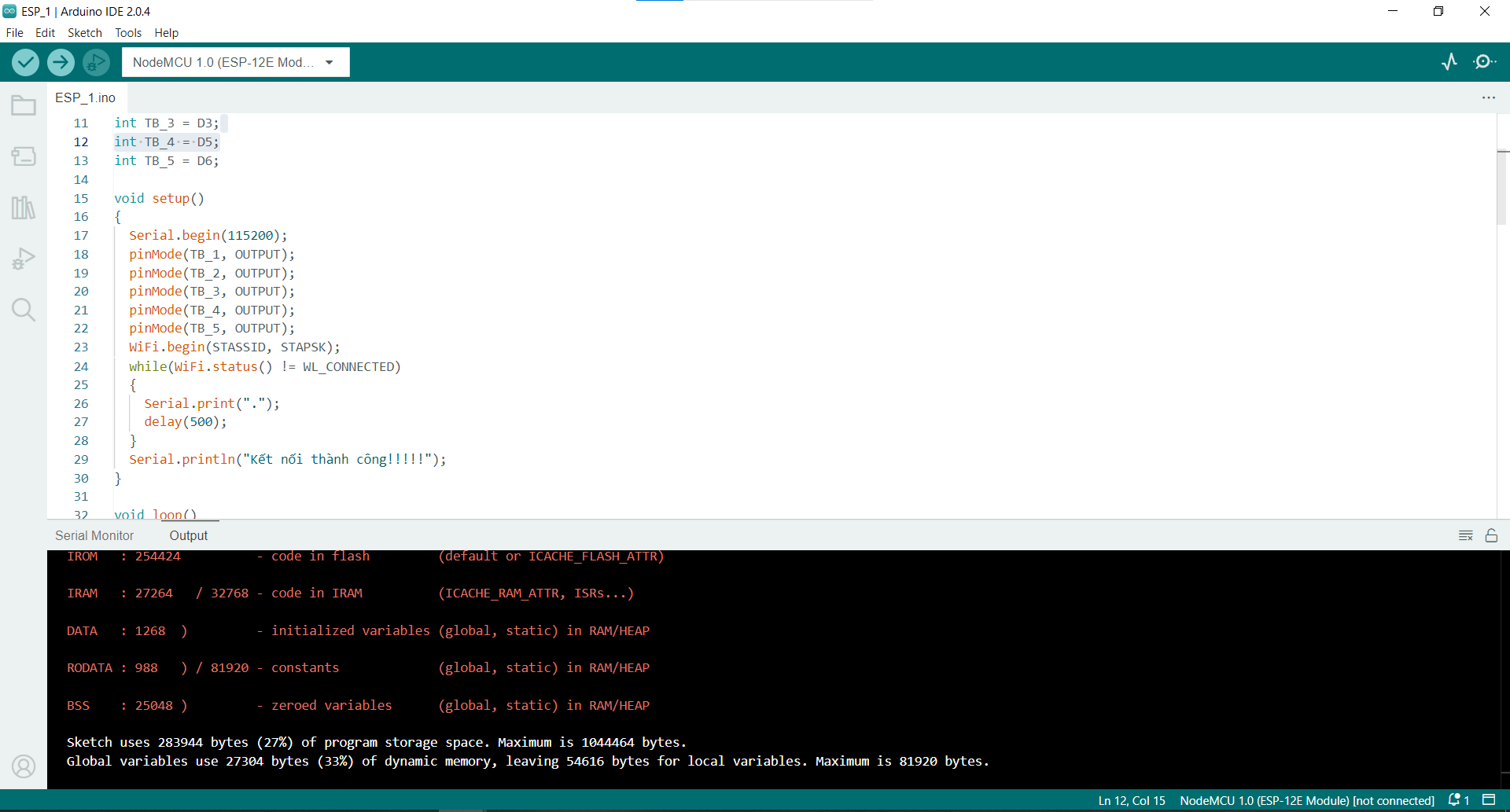
Trong giao diện lập trình căn bản khi lập trình cần chú ý các bước:

Bước 1: Tạo file biên dịch mới.

Bước 2: Lưu file code.

Bước 3: Lập trình code điều khiển.

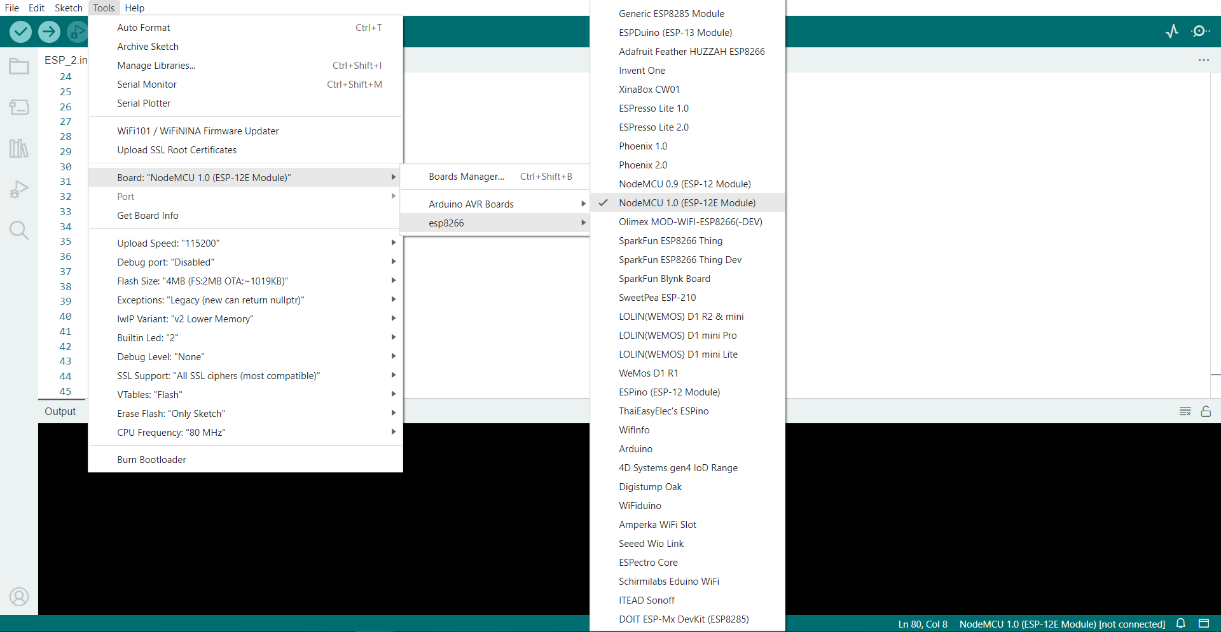
Bước 4: Biên dịch file để kiểm tra lỗi.



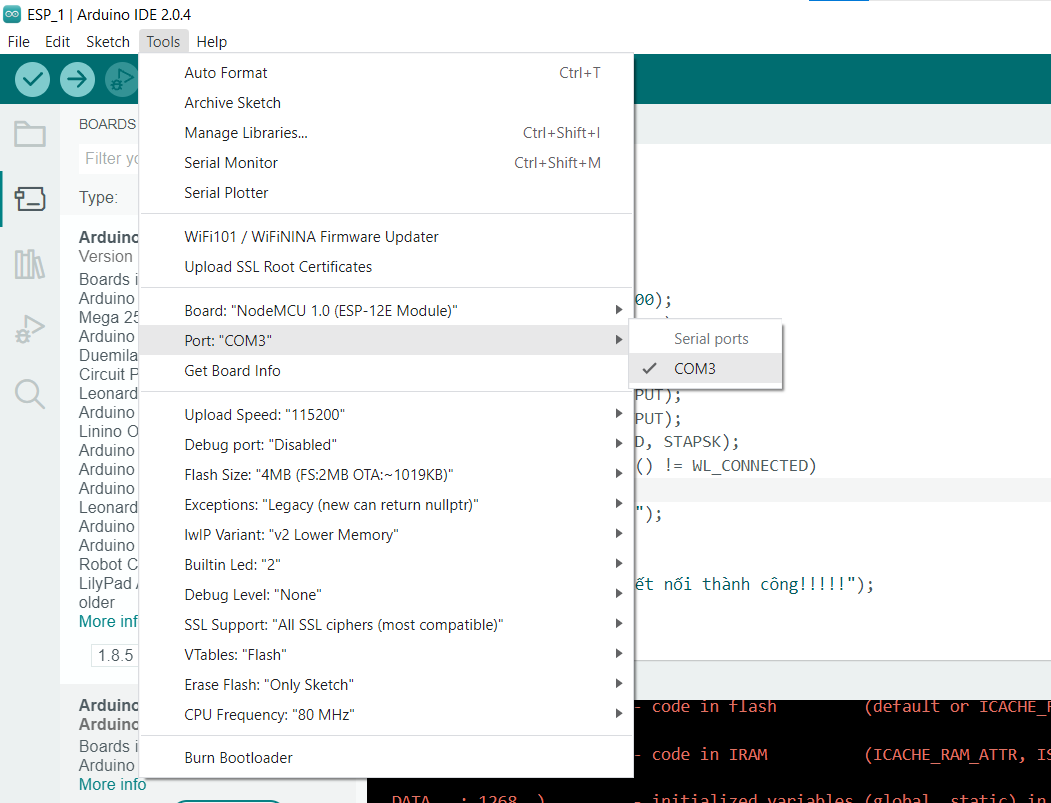
Hình 21. Thực hiện thao tác biên dịch chương trình

Bước 5: Nạp chương trình vào Board.

Cần chọn đúng board của esp8266 như dưới:



Hình 22. Chọn board thích hợp



Hình 23. Chọn Serial Port

Nạp chương trình và chạy ứng dụng: Cắm cáp nạp USB vào Board và máy tính kiểm tra trạng thái có cổng kết nối chưa tại phần Tools>Serial Port để chọn cổng com kết nối và Tool> Board để chọn board Arduino sử dụng. Sau đó nhấn Upload chương trình được nạp vào và chạy ứng dụng(cáp nạp USB nạp code cần chú ý loại cáp vì một số cáp không hỗ trợ việc truyền nhận dữ liệu). Arduino Toolbar:

****

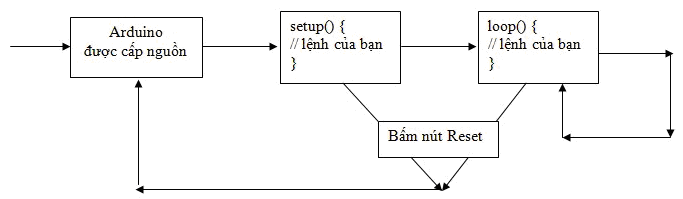
- Verify: Kiểm tra code có lỗi hay không

- Upload: Nạp code đang soạn thảo vào Arduino 

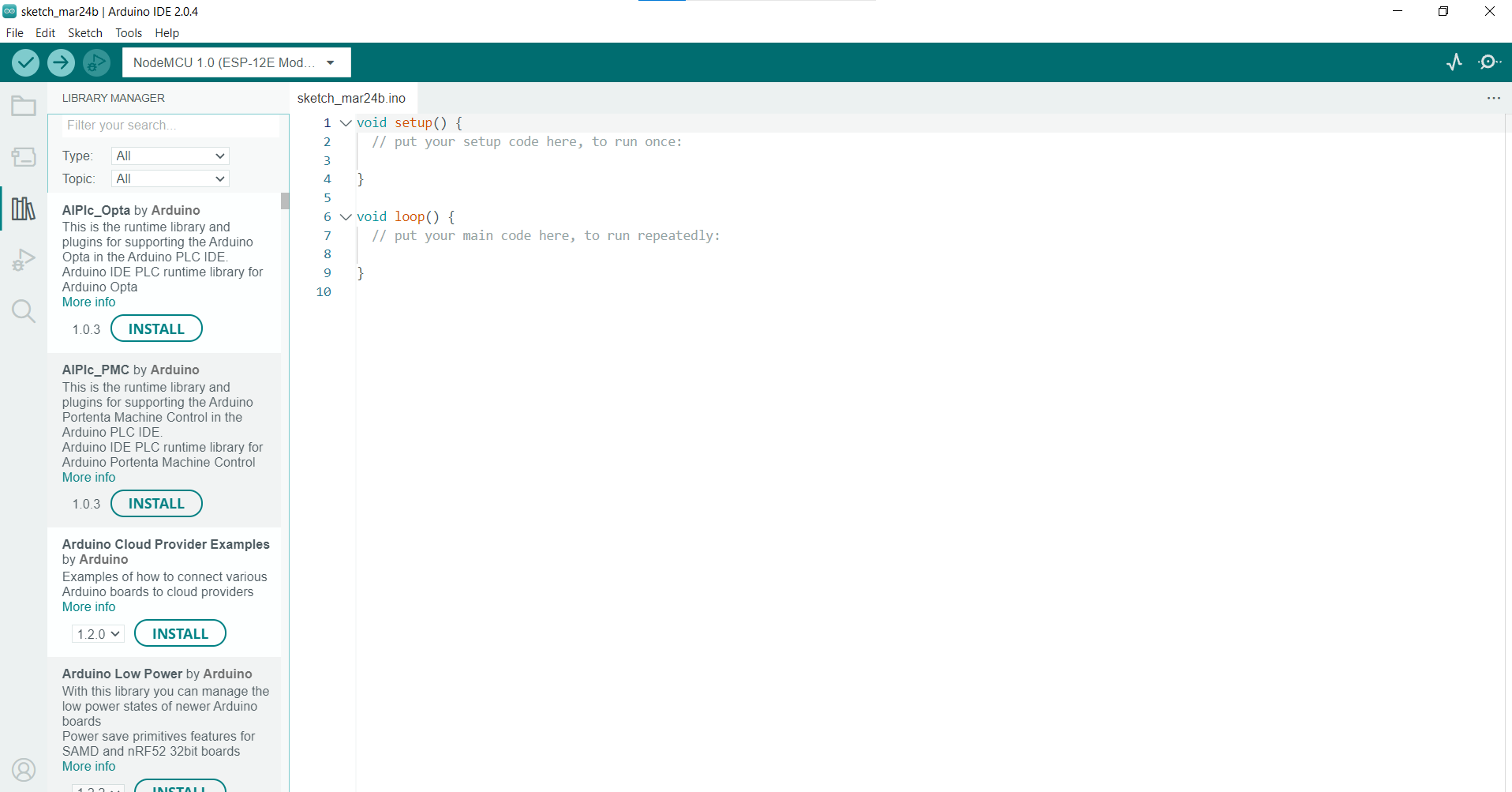
- New, Open, Save: Tạo mới, mở và Savesketch 

- Serial Monitor: Màn hình hiển thị dữ liệu từ Arduino gửi lên káy tính. 

*3.3.2. Chương trình kết nối giữa ESP và mạng Lan*



Hình 24. Sơ đồ thể hiện quá trình chạy chương trình trong Arduino IDE



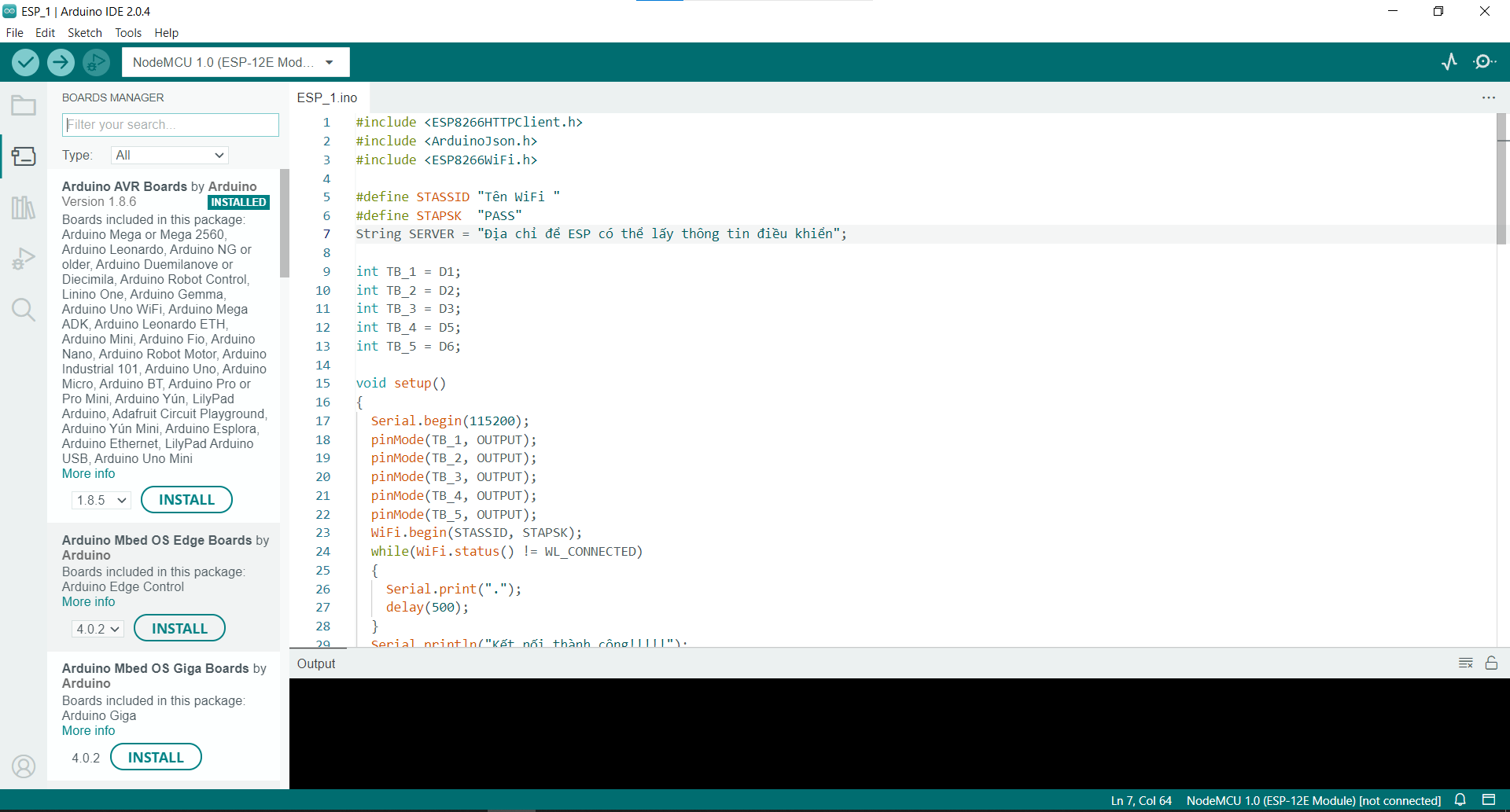
Hình 25. Giao diện ban đầu khi sử dụng Arduino IDE

Ở trạng thái mặc định, khi sử dụng IDE Arduino. Sẽ có 2 khu vực được viết sẵn dàng cho người dùng.

Hàm “setup” là nơi mà người dùng muốn mạch Arduino được nạp code sẽ chạy đầu tiên và chỉ chạy duy nhất 1 lần. Tại trong hàm setup này, có thể khai báo giá trị của những biến sẽ sử dụng, khai báo thư viện sẽ sử dụng, thiết lập các thông số cần thiết dành cho chương trình.

Hàm “loop” là hàm main chính của chương trình. Hàm này sẽ chạy sau khi setup chạy xong. Những câu lệnh, những chương trình được viết bên trong hàm loop này sẽ được chạy liên tục, lặp đi lặp lại cho đến khi người dùng ngắt nguồn của Board Arduino mới thôi.

Lưu ý: Bất cứ khi nào nhấn nút Reset trên thiết bị Arduino. Trương trình trên thiết bị sẽ trở lại trạng thái như khi Arduino mới được cấp nguồn.



Hình 26. Thực hiện viết chương trình điều khiển Arduino ESP

Trong đó:

* Trong hàm setup sẽ khai báo các chân kết nối thiết bị với ESP và trạng thái sẽ là OUTPUT (Xử lý các tín hiệu đầu ra). Khai báo cho phép thiết bị ESP kết nối mạng thông qua địa chỉ, mật khẩu wifi được cung cấp sẵn
* Trong hàm loop:

+ Thực hiện kiểm tra trạng thái kết nối thiết bị, nếu kết nối thành công (Kết nối thiết bị ESP với mạng Internet) sẽ thực hiện giao thức truyền siêu văn bản (HTTP) đây là một giao thức Request-Response giữa máy khách và máy chủ:

Ví dụ, trong trường hợp này:

ESP8266 (máy khách) gửi yêu cầu HTTP đến phía Server Raspberry Pi đang chạy

Server (Máy chủ) trả về phản hồi cho ESP8266 (máy khách)

Cuối cùng, phản hồi chứa thông tin trạng thái về yêu cầu và cũng có thể chứa nội dung được yêu cầu.

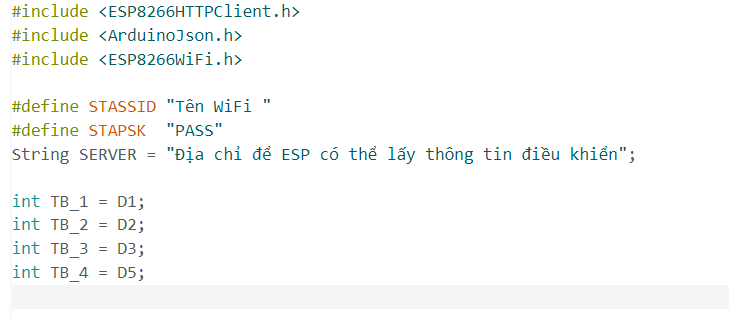
Phương thức được sử dụng ở đây là phương thức GET, với việc sử dụng phương thức này, tốc độ truyền nhận dữ liệu diễn ra một cách nhanh chóng.

Yêu cầu gửi đi của phương thức sử dụng trong trường hợp này là nhằm yêu cầu trả về một đối tượng JSON

Sau khi ESP nhận được JSON trả về, sẽ thực hiện phân tích dữ liệu nhằm xác định khu vực/vị trí của thiết bị đang nhận những câu lệnh nào và thực hiện theo những câu lệnh mà thiết bị nhận được từ SERVER.

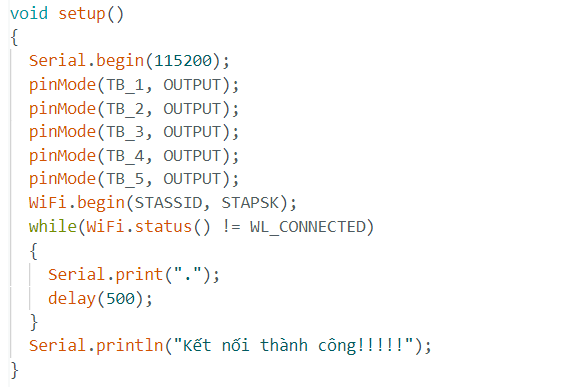
+ Nếu kiểm tra kết nối thất bại, thiết bị sẽ thực hiện kết nối lại cho đến khi kết nối thành công. Điều này đảm bảo thiết bị luôn ở trong trạng thái kết nối mạng để gửi nhận dữ liệu một cách nhanh chóng nhất.

Cụ thể chương trình được viết gồm những phần sau:



Hình 27. Khai báo mạng Lan sử dụng

Đoạn chương trình trên thể hiện việc khai báo các thử viện đã sử dụng ở đây là” ESP8266HTTPClient.h”, “ESP8266WiFi.h” và “ArduinoJson.h”. Tiếp theo là việc khai báo mạng lan gồm “STASSID” là tên wifi, “STAPSK” là mật khẩu wifi và “SERVER” chứa địa chỉ tcp/ip của web server được đặt trên rassberry pi. Cuối cùng là việc khai báo các chân ở đây sử dụng các chân D1, D2, D3 và D5 để điều khiển.



Hình 28. Hàm setup của chương trình

Hàm setup() làm những việc sau:

Khởi tạo giao tiếp nối tiếp với tốc độ truyền 115200 bit mỗi giây.

Đặt chế độ bốn chân (TB\_1, TB\_2, TB\_3, TB\_4) làm chân đầu ra, được sử dụng để gửi tín hiệu hoặc dữ liệu đến các thiết bị khác được kết nối với Arduino.

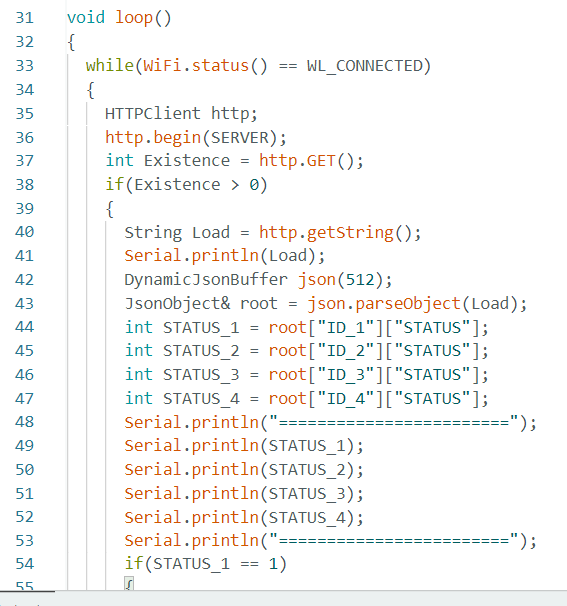
Bắt đầu quá trình kết nối với mạng Wi-Fi bằng cách gọi phương thức bắt đầu của thư viện WiFi và chuyển thông tin đăng nhập mạng (STASSID và STAPSK).

Nhập một vòng lặp while chờ kết nối được thiết lập bằng cách kiểm tra trạng thái của kết nối Wi-Fi bằng phương pháp trạng thái của thư viện WiFi. Nếu trạng thái không phải là WL\_CONNECTED, nghĩa là kết nối chưa được thiết lập, vòng lặp tiếp tục chạy.

Trong mỗi lần lặp lại vòng lặp, một dấu chấm được in ra màn hình nối tiếp để cho biết rằng chương trình vẫn đang cố gắng kết nối.

Chức năng trì hoãn khiến chương trình đợi trong 500 mili giây (nửa giây) trước khi kiểm tra lại trạng thái kết nối.

Khi kết nối được thiết lập, vòng lặp sẽ thoát và thông báo "Kết nối thành công!!!!!".



Hình 29. Hàm loop của chương trình

Hàm loop() làm những việc sau:

Khi kết nối Wi-Fi vẫn được thiết lập (vẫn còn kết nối đến mạng trong lan). Gửi yêu cầu “http.get” đến máy chủ bằng cách gọi phương thức GET của thư viện HTTPClient.

Sau đó kiểm tra mã phản hồi từ máy chủ (đã được lưu trữ) để xem liệu nó có lớn hơn 0 hay không, cho biết phản hồi thành công. Nếu phản hồi không thành công, chương trình sẽ thoát khỏi vòng lặp và đợi kết nối Wi-Fi được thiết lập lại.

Nếu phản hồi thành công, chương trình sẽ truy xuất nội dung phản hồi (được lưu trữ trong biến Load) bằng cách gọi phương thức getString của thư viện HTTPClient.

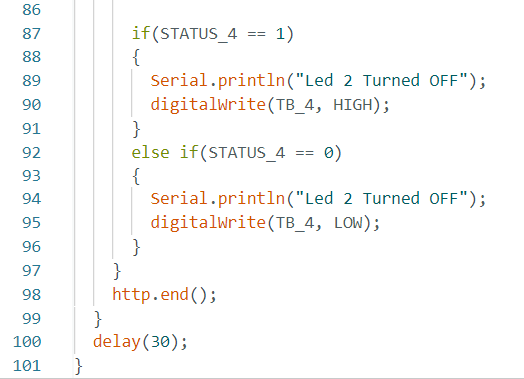
In nội dung phản hồi tới màn hình nối tiếp bằng phương thức Serial.println.

Tiếp đó phân tích nội dung phản hồi dưới dạng đối tượng JSON bằng thư viện DynamicJsonBuffer và JsonObject.

Trích xuất giá trị của "STATUS" cho từng thiết bị trong số bốn thiết bị ("ID\_1" đến "ID\_4") từ đối tượng JSON và lưu trữ chúng trong bốn biến số nguyên riêng biệt (STATUS\_1 đến STATUS\_4).

In các giá trị của bốn biến số nguyên ra màn hình nối tiếp.

Như vậy hàm loop() gửi một yêu cầu HTTP GET đến máy chủ và truy xuất phản hồi JSON chứa trạng thái của bốn thiết bị. Sau đó, nó trích xuất và in các giá trị trạng thái ra màn hình. Quá trình này lặp lại lặp lại miễn là kết nối Wi-Fi được thiết lập.

****

Hình 30. Kiểm tra trạng thái của thiết bị

Đây là quá trình kiểm tra trạng thái của thiết bị:

Kiểm tra giá trị của biến STATUS\_4 xem nó có bằng 1 không.

Nếu STATUS\_4 bằng 1, in thông báo "Led 2 Turned OFF" ra màn hình nối tiếp bằng phương thức Serial.println và đặt chân đầu ra TB\_4 thành “HIGH

Nếu STATUS\_4 không bằng 1 (tức là bằng 0), hãy in thông báo "Led 2 Turned ON" ra màn hình nối tiếp và đặt chân đầu ra TB\_4 thành ” LOW”. Điều này cũng diễn ra tương tự với các “STATUS” khác.

Với mỗi một các ESP8266 khác nhau thì việc chứa các “ID” của thiết bị là khác nhau. Các “ID” của thiết bị giúp điều khiển các thiết bị riêng biệt không xung đột lẫn nhau trong bất kỳ một trường hợp nào. Vậy nên, nếu một thiết bị hay thậm chí là cả một ESP8266 gặp sự cố thì những thiết bị khác vân hoạt động bình thường.

**CHƯƠNG 4. KIỂM THỬ VÀ ĐÁNH GIÁ**

**4.1. Kiểm thử trương trình và đánh giá kết quả**

Thông qua quá trình làm bài nghiên cứu này, em xin được đưa ra những đánh giá về kết quả nghiên cứu của mình như sau:

* **Về ưu điểm**: Với nội dung bài nghiên cứu là “Nghiên cứu và xây dựng mô hình điều khiển hệ đa vi điều khiển phân tán thông qua mạng nội bộ Lan”, em đã thực hiện lần lược các bước: Lên ý tưởng, phân tích, thiết kế và thực hiện xây dựng chương trình. Từ đó, em đã có thể đưa ra được những ưu điểm về đề tài của mình như:

+ Đáp ứng được tính đổi mới trong quá trình áp dụng tin học trong việc đưa ra giải quyết các bài toán thực tiễn.

+ Xây dựng thành công mô hình điều khiển các thiết bị phân tán thông qua mạng Lan.

+ Đảm bảo tốc độ thông tin không gây ảnh hưởng đến quá trình giám sát và điều khiển.

+ Việc điều khiển hệ thống được phân tán trên các vi điều khiển khác nhau. Nếu một vi điều khiển bị lỗi, hệ thống vẫn có thể tiếp tục hoạt động với các vi điều khiển khác.

+ Quá trình điều khiển diễn ra nhanh chóng và chính xác

+ …

* **Về nhược điểm:** Trong quá trình xây dựng và thiết kế hệ thống, mặc dù em đã cố gắng để có thể xây dựng một chương trình đáp ứng một cách tốt nhất có thể. Nhưng cũng không thể bỏ qua được hạn chế như:

+ Việc thiết kế hệ thống còn chưa được sâu sắc, các chức năng mới chủ yếu ở mức độ đơn giản, ….

+ …

Từ những ưu điểm và nhược điểm của mô hình điều khiển hệ đa vi điều khiển phân tán thông qua mạng nội bộ LAN, ta có thể thấy rằng mô hình này có nhiều ưu điểm, bao gồm tính linh hoạt, khả năng mở rộng, khả năng phân tán điều khiển và tiết kiệm chi phí. Tuy nhiên, cũng có một số nhược điểm, bao gồm độ trễ, sự phụ thuộc vào mạng LAN, và vấn đề bảo mật.

**4.2. Phương hướng phát triển**

Thông qua đề tài này, em có thể nhìn thấy tính khả thi của nội dung đề tài là vô cùng rộng lớn. Với chi phí không quá cao mà có thể kiến cho các thiết bị đều có thể dễ dàng quan sát và điều khiển. Việc đó sẽ khiến cho cuộc sống của mọi người trở nên dễ dàng hơn.

Tính áp dụng này không chỉ có thể dành cho các thiết bị trong gia đình, mà cũng có thể được mở rộng ra cho các nhà máy, các xí nghiệp. Thay vì sử dụng sức người để quan sát và điều khiển máy móc, có thể sử dụng “chính máy móc để điều khiển máy móc”. Từ đó dần cải thiện và nâng cao năng xuất lao động, …

Ngoài ra, việc hoàn thiện hệ thống có làm theo các hướng sau:

Phát triển các công cụ quản lý và phát triển ứng dụng: Cần phát triển các công cụ quản lý để giúp quản lý và phát triển các ứng dụng phân tán trên mô hình này một cách dễ dàng hơn.

Tăng tính linh hoạt: Cần phát triển các giải pháp để tăng tính linh hoạt của hệ thống, cho phép các vi điều khiển và thiết bị khác có thể được thêm hoặc loại bỏ khỏi mạng một cách dễ dàng.

Tăng tính bảo mật: Cần tăng tính bảo mật của mô hình để đảm bảo an toàn cho hệ thống, bằng cách sử dụng các giao thức bảo mật, mã hóa dữ liệu và xác thực người dùng.

**KẾT LUẬN VÀ ĐÁNH GIÁ**

Đề tài **Nghiên cứu và xây dựng mô hình điều khiển hệ đa vi điều khiển phân tán thông qua mạng nội bộ Lan** về cơ bản đã đạt được mục tiêu đề ra gồm những nội dung chính sau:

* Tìm hiểu về lập trình Web, ngôn ngữ lập trình C cho Arduino, trình biên dịch Arduino IDE.
* Xây dựng thành công mô hình điều khiển đa vi điều khiển phân tán qua mạng nội bộ Lan.
* Áp dụng thành công những kiến thức đã học trên trường, những kiến thức từ trong thực tế áp dụng giải quyết bài toán thực tiễn.

**TÀI LIỆU THAM KHẢO**

[1]. Nguyễn Ngọc Cương, Nguyễn Đình Nghĩa, Đỗ Quốc Huy, trần nghi Phú và Phạm Thành Công, 2011. *Giáo trình Ngôn ngữ lập trình C/C++*. Nhà xuất bản Thông Tin Và Truyền Thông. Hà Nội. 191 trang.

[2]. Nguyễn Tất Bảo Thiện - Lập Trình IOT Với Arduino - ESP8266 & XBEE

[3]. Tài liệu về HTML,CSS, Boostrap, JS: <https://w3school.com>

[4]. Tài liệu về xây dựng chương trình web theo ngôn ngữ python: <https://www.djangoproject.com>