

ĐẠI HỌC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



ĐỒ ÁN CƠ SỞ 3

ĐỀ TÀI:

**THIẾT KẾ MÔ HÌNH ĐIỀU KHIỂN THIẾT BỊ
TRONG NHÀ BẢNG SMARTPHONE VÀ WEB**

Giáo viên hướng dẫn : Ths. Nguyễn Anh Tuấn

Sinh viên thực hiện : Lương Tuấn Anh 18CE

Nguyễn Ngọc Quang 18IT5

Đà Nẵng, tháng 9 năm 2020

ĐẠI HỌC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

ĐỒ ÁN CƠ SỞ 3

ĐỀ TÀI:

**THIẾT KẾ MÔ HÌNH ĐIỀU KHIỂN THIẾT BỊ
TRONG NHÀ BẰNG SMARTPHONE VÀ WEB**

Đà Nẵng, tháng 9 năm 2020

MỞ ĐẦU

Ngày nay trên thế giới với sự bùng nổ của các ngành công nghệ thông tin, điện tử, nhúng v.v. Đã làm cho đời sống của chúng ta ngày càng hoàn thiện, tiến bộ hơn. Các thiết bị tự động hóa đã ngày càng được sử dụng rộng rãi đưa vào trong sản xuất và thậm chí là vào cuộc sống sinh hoạt hằng ngày. Nhà thông minh là một ví dụ điển hình. Các thiết bị giám sát, tự động, điều khiển từ xa v.v với đặc điểm nổi bật như sự chính xác cao, tốc độ nhanh, gọn nhẹ là những điều rất cần thiết cho cuộc sống của chúng ta ngày một tiến bộ hơn. Một trong những ứng dụng đó là kỹ thuật điều khiển thiết bị chiếu sáng, với tự động hóa hoàn toàn hoặc bán tự động trong việc quản lý, điều khiển. Hệ thống điện tử này giao tiếp với chủ nhà thông qua phần mềm trên di động, máy tính bảng, hoặc một giao diện web. Thông qua đó, ta có thể bật tắt tất cả thiết bị chiếu sáng trong nhà, hẹn giờ hay lập lịch trình tự động qua ứng dụng di động (Android). Thiết kế nhà thông minh mặc dù đều dựa trên nền tảng IoT, tuy nhiên có rất nhiều cách tiếp cận khác nhau trong việc thiết kế các mô hình nhà thông minh. Một số cách có thể kể đến là sử dụng các máy tính nhúng như Raspberry PI3, Orange Pi One, PIC hoặc Arduino. Ở đây nhóm đã sử dụng Arduino Uno R3 và xây dựng đề tài nghiên cứu với mục tiêu điều khiển các thiết bị trong nhà qua internet (wifi).

LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn đến thầy Nguyễn Anh Tuấn đã trực tiếp hướng dẫn, góp ý, chia sẻ nhiều kinh nghiệm quý báu, tận tình giúp đỡ và tạo điều kiện để chúng em hoàn thành tốt đề tài.

Xin chân thành cảm ơn!

NHẬN XÉT
(Của giáo viên hướng dẫn)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Chữ ký

MỤC LỤC

Chương 1	TỔNG QUAN.....	10
1.1	ĐẶT VẤN ĐỀ.....	10
1.2	MỤC TIÊU.....	10
1.3	NỘI DUNG NGHIÊN CỨU	11
1.4	GIỚI HẠN.....	11
1.5	CẤU TRÚC ĐỒ ÁN	11
Chương 2	CƠ SỞ LÝ THUYẾT	13
2.1	GIỚI THIỆU.....	13
2.2	GIỚI THIỆU PHẦN CỨNG.....	13
2.2.1	Bộ điều khiển trung tâm Arduino Uno R3	13
2.2.2	Relay	15
2.2.3	Điện trở	16
2.2.4	Đèn led.....	17
2.2.5	Module cảm biến ánh sáng	18
2.2.6	Động cơ điện Motor	19
2.3	GIỚI THIỆU CÔNG CỤ LẬP TRÌNH.....	19
2.3.1	Arduino IDE	19
2.3.2	Visual Studio Code	25
2.3.3	Android Studio	27
Chương 3	PHÂN TÍCH & THIẾT KẾ HỆ THỐNG.....	31
3.1	GIỚI THIỆU.....	31
3.2	THIẾT KẾ SƠ ĐỒ KHỐI	31
3.3	NGUYÊN LÝ HOẠT ĐỘNG	32
3.4	LƯU ĐỒ THUẬT TOÁN	33
3.4.1	Lưu đồ lấy trạng thái của các đèn và cảm biến từ Web server.....	33
3.4.2	Lưu đồ điều khiển các đèn.....	34
3.4.3	Lưu đồ điều khiển quạt.....	36
3.4.4	Lưu đồ điều khiển đèn bởi cảm biến ánh sáng	37
3.4.5	Lưu đồ điều khiển cửa	38
Chương 4	TRIỂN KHAI THỰC HIỆN.....	39
4.1	LẮP RÁP PHẦN CỨNG	39
4.2	THIẾT KẾ PHẦN MỀM.....	39
4.2.1	Chương trình cho Arduino Uno R3	39
4.2.2	Xây dựng Web Server	39
4.2.3	Ứng dụng di động Android.....	41
Chương 5	KẾT QUẢ ĐẠT ĐƯỢC.....	42
5.1	MÔ HÌNH NHÀ THÔNG MINH	42
5.2	WEB SERVER.....	43
5.3	ỨNG DỤNG DI ĐỘNG.....	45
Chương 6	KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	46
6.1	KẾT LUẬN	46
6.2	HẠN CHẾ	46
6.3	HƯỚNG PHÁT TRIỂN	46

DANH MỤC HÌNH

	Trang
Hình 2.1 Thiết bị Arduino Uno R3	12
Hình 2.2 Sơ đồ chân Arduino Uno R3	14
Hình 2.3 Relay	15
Hình 2.4 Sơ đồ chân Relay.....	16
Hình 2.5 Điện trở	17
Hình 2.6 Giá trị của điện trở	17
Hình 2.7 Đèn led	17
Hình 2.8 Module cảm biến ánh sáng.....	18
Hình 2.9 Giao diện Code Arduino IDE.....	20
Hình 2.10 Các phiên bản JRE	20
Hình 2.11 Tải xuống Arduino IDE	21
Hình 2.12 Cài đặt Driver cho Arduino IDE	22
Hình 2.13 Vùng lệnh cơ bản trong Arduino IDE.....	23
Hình 2.14 Vùng thông báo trong Arduino IDE	24
Hình 2.15 Chọn cổng COM cho arduino	25
Hình 2.16 Giao diện Code Visual Studio Code	26
Hình 2.17 Cài đặt Visual Studio Code.....	26
Hình 2.18 Cài các extension cơ bản.....	27
Hình 2.19 Giao diện màn hình đầu của Android Studio.....	27
Hình 2.20 Một project bình thường thì có dạng như thế này.....	28
Hình 2.21 Cài đặt Android Studio.....	28
Hình 2.22 Cài máy ảo trong Android Studio	30
Hình 3.1 Sơ đồ khối hệ thống	31
Hình 3.2 Sơ đồ nguyên lý hoạt động.....	32
Hình 3.3 Lưu đồ lấy trạng thái của các đèn và cảm biến từ Web server	33
Hình 3.4 Lưu đồ điều khiển các bóng đèn bởi Relay.....	35
Hình 3.5 Lưu đồ điều khiển quạt	36
Hình 3.6 Lưu đồ điều khiển đèn bởi cảm biến ánh sáng.....	37
Hình 3.7 Lưu đồ điều khiển cửa.....	38
Hình 4.1 Chương trình cho Arduino Uno R3	39
Hình 4.2 Xây dựng Web Server.....	40
Hình 4.3 Phần mềm điều khiển sử dụng Android Studio	41
Hình 5.1 Mô hình nhà thông minh	42
Hình 5.2 Trang chủ điều khiển thiết bị	43

Chương 1 TỔNG QUAN

1.1 ĐẶT VẤN ĐỀ

Thường ngày để điều khiển chiếu sáng trong nhà, bạn phải di chuyển và bật tắt lăm công tắc. Tuy nhiên, với ngôi nhà thông minh thì các kịch bản chiếu sáng được thiết lập sẵn cho từng hoạt cảnh chi tiết, chỉ một chạm là bạn được điều khiển khung chiếu sáng theo ý muốn.

Với những người mất khả năng đi lại, thì giải pháp này là lựa chọn tốt nhất. Qua bảng điều khiển trên ứng dụng, bạn có thể biết được đèn nào đang bật hay đã tắt và bạn 100% bật hoặc tắt thiết bị đó ngay trên điện thoại hay Website khi không cần thiết.

Nếu bạn là người hay quên, đăng trí thì chức năng hẹn giờ bật tắt đèn là giải pháp hiệu quả trong việc tiết kiệm điện, ví dụ như đèn ở ngoài trời tự động bật tắt vào ban ngày hoặc ban đêm nhờ cảm biến ánh sáng, hay hẹn giờ lập thời gian biểu cho các thiết bị trong nhà có thể tự động bật tắt theo giờ, theo ngày, hay cả tuần.

1.2 MỤC TIÊU

- Điều khiển hệ thống thiết bị trong gia đình qua các thiết bị có internet bất cứ ở đâu.
- Bật tắt tự động qua thiết lập thời gian biểu cho các thiết bị.
- Thay đổi tùy thuộc vào ánh sáng môi trường và điều chỉnh cho phù hợp.
- Xây dựng phần mềm quản lý trên nhiều nền tảng: Android, Website, IOS (sau này), v.v
- Giao diện quản lý thân thiện, đơn giản và dễ sử dụng.

=> Tiết kiệm năng lượng, hiệu quả sử dụng cao, an toàn, dễ sử dụng, tiện nghi và thông minh, đem lại sự hiện đại, sang trọng.

1.3 NỘI DUNG NGHIÊN CỨU

- Xác định mục tiêu và giới hạn nghiên cứu
- Tìm hiểu cơ sở lý thuyết
- Thiết kế khối cảm biến, khối điều khiển qua relay và khối nút nhấn thủ công và khối hoàn chỉnh
- Xây dựng WebServer
- Viết code cho Arduino Uno R3
- Xây dựng ứng dụng di động (Android App)
- Thiết kế mô hình nhà và lắp ráp các board mạch
- Chạy thử và kiểm tra, sửa chữa lỗi

1.4 GIỚI HẠN

- Hệ thống chỉ ở mức độ điều khiển đơn giản
- Chỉ hoạt động khi có WIFI trong nhà
- Domain cho WebServer miễn phí nên khó tránh rủi ro tốc độ và lâu dài

1.5 CẤU TRÚC ĐỒ ÁN

Với đề tài "THIẾT KẾ MÔ HÌNH ĐIỀU KHIỂN THIẾT BỊ TRONG NHÀ BẰNG SMARTPHONE VÀ WEB" thì bố cục đồ án như sau:

Chương 2: Cơ sở lý thuyết

Chương này trình bày giới thiệu phần cứng của hệ thống điều khiển, các công cụ hỗ trợ cho việc xây dựng phần mềm hệ thống

Chương 3: Phân tích và thiết kế hệ thống.

Chương này trình bày về sơ đồ khối, sơ đồ nguyên lý, sơ đồ hoạt động của các board mạch của hệ thống, lưu đồ thuật toán. Quy trình xây dựng WebServer và App Android và các chức năng liên quan.

Chương 4: Triển khai thực hiện

Chương này trình bày quy trình lắp ráp các board mạch và thiết kế mô hình nhà. Hoàn thiện WebServer và ứng dụng di động Android. Bên cạnh đó là hình ảnh thực tế, cũng như kết quả.

Chương 5: Kết quả đạt được

Chương này giới thiệu về sản phẩm thực hiện, mô hình, ứng dụng di động và web server.

Chương 6: Kết luận và hướng phát triển

Chương này trình bày quy trình kết quả mà đề tài đạt được, đồng thời đưa ra hướng phát triển để có được một đề tài hoàn thiện và đáp ứng được nhu cầu cho cuộc sống hiện đại như ngày nay.

Chương 2 CƠ SỞ LÝ THUYẾT

2.1 GIỚI THIỆU

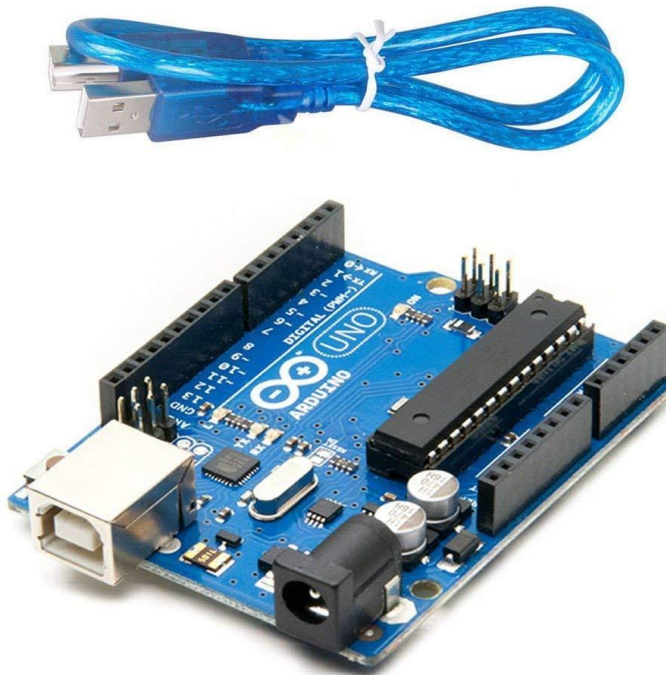
Trong chương này là các lý thuyết có liên quan đến các vấn đề mà đề tài sẽ dùng để thiết kế và triển khai thực hiện.

2.2 GIỚI THIỆU PHẦN CỨNG

- Thiết bị đầu vào: module cảm biến ánh sáng, nguồn 5V DC và 220V AC.
- Thiết bị đầu ra: module Relay, các đèn, quạt, cửa nhà.
- Thiết bị vừa là đầu vào và đầu ra: Arduino Uno R3.
- Thiết bị điều khiển: di động Android, máy tính, laptop có kết nối internet.
- Chuẩn giao thức mạng internet: HTTP (GET, POST METHOD)

2.2.1 Bộ điều khiển trung tâm Arduino Uno R3

a) Giới thiệu về Arduino Uno R3



Hình 2.1 Thiết bị Arduino Uno R3

Arduino thật ra là một bo mạch vi xử lý được dùng để lập trình tương tác với các thiết bị phần cứng như cảm biến, động cơ, đèn hoặc các thiết bị khác. Đặc điểm nổi bật của Arduino là môi trường phát triển ứng dụng cực kỳ dễ sử dụng, với một

ngôn ngữ lập trình có thể học một cách nhanh chóng ngay cả với người ít am hiểu về điện tử và lập trình.

b) Sơ đồ chân



Hình 2.1 Sơ đồ chân Arduino Uno R3

- USB (1): Arduino sử dụng cáp USB để giao tiếp với máy tính. Thông qua cáp USB chúng ta có thể Upload chương trình cho Arduino hoạt động, ngoài ra USB còn là nguồn cho Arduino.
- Nguồn (2 và 3): Khi không sử dụng USB làm nguồn thì chúng ta có thể sử dụng nguồn ngoài thông qua jack.
Chân 5V và chân 3.3V (Output voltage) : các chân này dùng để lấy nguồn ra từ nguồn mà chúng ta đã cung cấp cho Arduino
- GND: chân được xem là nguồn âm
- Input và Output (4, 5 và 6):
Arduino Uno có 14 chân digital với chức năng input và output sử dụng các hàm pinMode(), digitalWrite() và digitalRead()
- Serial : chân 0 (Rx), chân 1 (Tx). Hai chân này dùng để truyền (Tx) và nhận (Rx) dữ liệu nối tiếp TTL

-PWM (pulse width modulation): các chân 3, 5, 6, 9, 10, 11 trên bo mạch có dấu “~” là các chân PWM chúng ta có thể sử dụng nó để điều khiển tốc độ động cơ, độ sáng của đèn...

-SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), các chân này hỗ trợ giao tiếp theo chuẩn SPI.

- Reset (7): dùng để reset Arduino

2.2.2 Relay

a) Giới thiệu relay và tính năng

Rơ-le là một loại linh kiện điện tử thụ động rất hay gặp trong các ứng dụng thực tế.

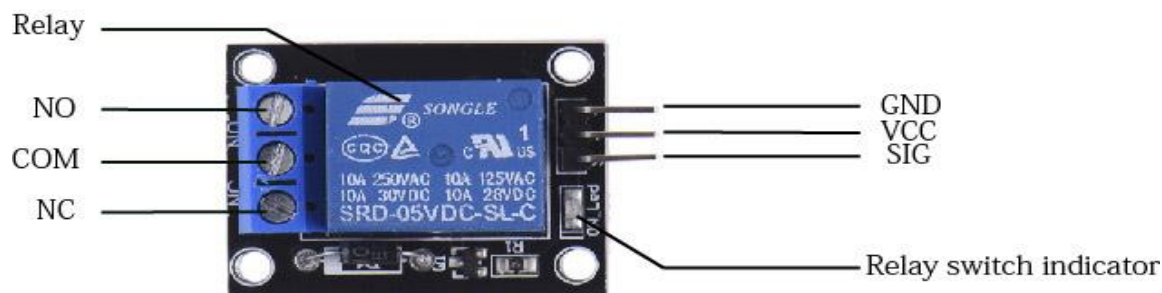
Module relay thích hợp cho các ứng dụng đóng ngắt điện thế cao AC hoặc DC, các thiết bị tiêu thụ dòng lớn, module thiết kế nhỏ gọn, có opto và transistor cách ly, kích hoạt bằng mức thấp (0V) phù hợp với mọi loại MCU và thiết kế có thể sử dụng nguồn ngoài giúp cho việc sử dụng trở nên thật linh động và dễ dàng!

Rơ-le là một công tắc (khóa K). Nhưng khác với công tắc ở một chỗ cơ bản, rơ-le được kích hoạt bằng điện thay vì dùng tay người.



Hình 2.3 Relay

b) Sơ đồ chân



Hình 2.4 Sơ đồ chân Relay

NO	=	Normally Open	: chân thường mở
COM	=	Common Pin	: chân chung
NC	=	Normally Closed	: chân thường đóng
GND	=	Ground (-)	: chân đất, mass – (DC-)
VCC	=	5V (+)	: chân nguồn dương + (DC+)
SIG	=	SIGNAL	: chân tín hiệu
Relay switch indicator = HIGH or LOW			: chế độ ban đầu cao hoặc thấp
Relay			: tải đóng ngắt (cuộn cảm)

c) Thông số

- Sử dụng điện áp nuôi 5VDC.
- 4 Relay đóng ngắt ở điện thế kích bằng 0V nên có thể sử dụng cho cả tín hiệu 5V hay 3v3 (cần cấp nguồn ngoài), mỗi Relay tiêu thụ dòng khoảng 80mA.
- Điện thế đóng ngắt tối đa: AC250V - 10A hoặc DC30V - 10A.
- Có thể kích mức 0 hoặc mức 1 thông qua Jumper
- Có đèn báo đóng ngắt trên mỗi Relay.

2.2.3 Điện trở

a) Giới thiệu và tính năng

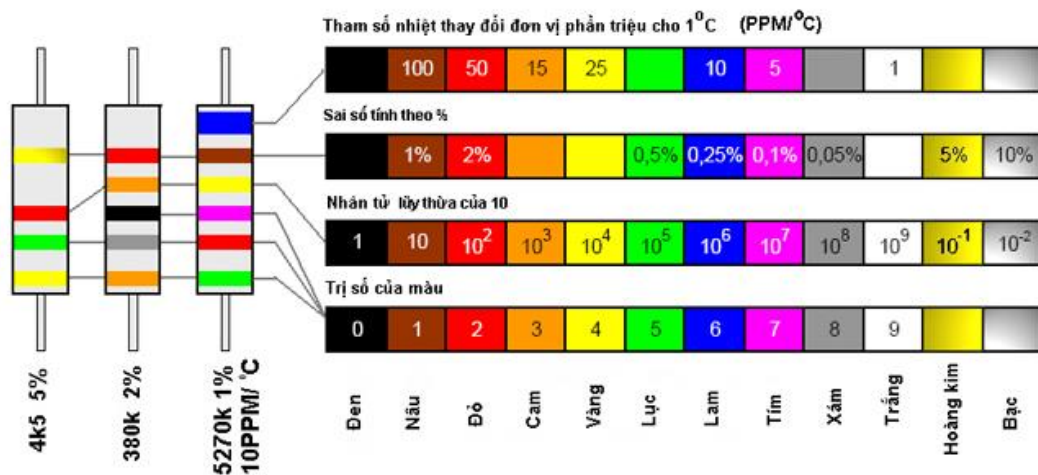
Điện trở hay Resistor là một linh kiện điện tử thụ động gồm 2 tiếp điểm kết nối, thường được dùng để hạn chế cường độ dòng điện chảy trong mạch, điều chỉnh mức độ tín hiệu, dùng để chia điện áp, kích hoạt các linh

kiện điện tử chủ động như transistor, tiếp điểm cuối trong đường truyền điện và có trong rất nhiều ứng dụng khác.



Hình 2.5 Điện trở

b) Thông số



Hình 2.6 Giá trị của điện trở

2.2.4 Đèn led

Dùng để chiếu sáng, trong sản phẩm thực hiện dùng đèn AC



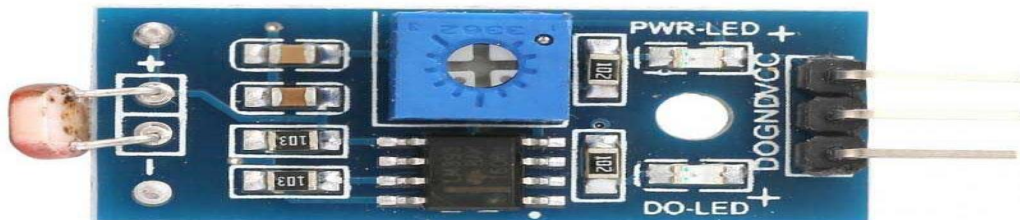
Hình 2.7 Đèn led

2.2.5 Module cảm biến ánh sáng

a) Giới thiệu về module cảm biến ánh sáng và tính năng

Module Cảm Biến Ánh Sáng dùng quang trở có ưu điểm:

- Giá thành rẻ.
- Điện áp sử dụng 3.3-5V, tương thích với các board Arduino
- Độ chính xác cao nhờ sử dụng IC so sánh áp (comparator) LM393.
- Nhỏ gọn
- Các thành phần phụ như điện trở, tụ điện... cần thiết cho mạch đã được gắn đầy đủ.
- Chỉ cần cấp nguồn, nối dây điều khiển vào relay là có thể tắt/mở bóng đèn theo ý muốn.



Hình 2.8 Module cảm biến ánh sáng

Tính năng:

- Cảm biến ánh sáng ban ngày và ban đêm
- Sử dụng ánh sáng điều khiển thiết bị điện
- Đèn sáng tự động khi trời tối
- Các ứng dụng quang học khác

b) Thông số

- Điện áp làm việc: 3V – 5V
- Kích thước module: 32mm x 11mm x 20mm
- 3(hoặc 4) chân ra : VCC- D0(I/O)- GND
- IC so sánh áp (comparator) LM393.

Theo sơ đồ mạch nguyên lý dưới: Khi module cảm biến rung được kích hoạt, khi đó sẽ có sự thay đổi điện áp tại đầu vào của Ic LM393. Ic này

nhận biết có sự thay đổi nó sẽ đưa ra một tín hiệu thấp để báo hiệu có sự rung động.

2.2.6 Động cơ điện Motor

a) Giới thiệu và tính năng

Motor được dùng để làm quạt quay và tạo ròng rọc để đẩy cánh cửa lên

2.3 GIỚI THIỆU CÔNG CỤ LẬP TRÌNH

2.3.1 Arduino IDE

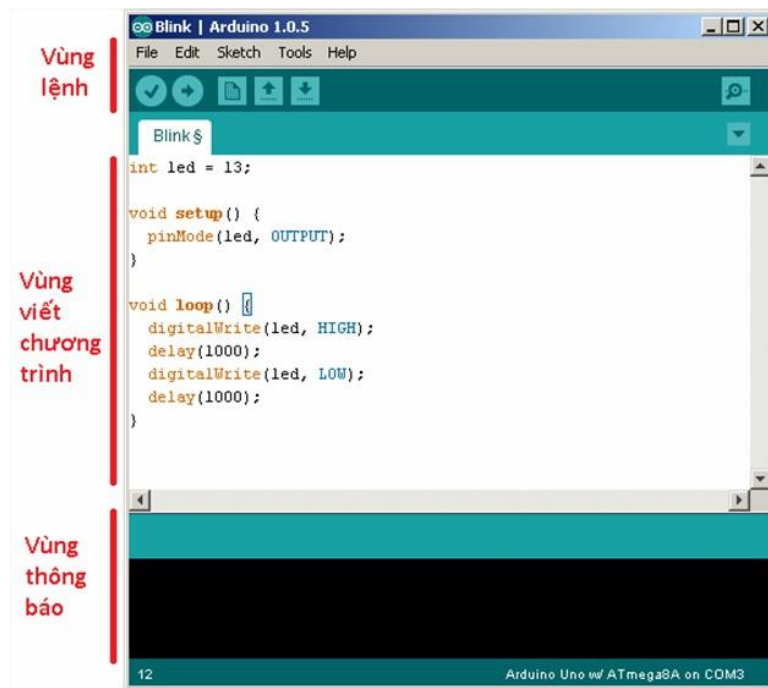
IDE viết tắt của cụm từ Integrated Development Environment là phần mềm cung cấp cho các lập trình viên một môi trường tích hợp bao gồm nhiều công cụ khác nhau như chương trình viết mã lệnh hay code editor, chương trình sửa lỗi hay debugger, chương trình mô phỏng ứng dụng khi chạy thực tế hay simulator... Nói cách khác thì IDE là một phần mềm bao gồm những gói phần mềm khác nhau giúp phát triển ứng dụng phần mềm.

a) Giao diện IDE

Arduino IDE là một trình soạn thảo văn bản, giúp bạn viết code để nạp vào bo mạch Arduino.

Một chương trình viết bởi Arduino IDE được gọi là sketch được lưu dưới dạng .ino.

Môi trường lập trình đơn giản dễ sử dụng, ngôn ngữ lập trình C hoặc C++ quen thuộc với người làm kỹ thuật. Số lượng thư viện code viết sẵn và chia sẻ bởi cộng đồng nguồn mở cực kỳ lớn.



Hình 2.9 Giao diện Code Arduino IDE

b) Cài đặt Arduino IDE

a. Cài đặt Java Runtime Environment (JRE)

Java SE Runtime Environment 7u80		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
<input type="radio"/> Accept License Agreement <input checked="" type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	31.63 MB	jre-7u80-linux-i586.rpm
Linux x86	46.31 MB	jre-7u80-linux-i586.tar.gz
Linux x64	32.14 MB	jre-7u80-linux-x64.rpm
Linux x64	44.93 MB	jre-7u80-linux-x64.tar.gz
Mac OS X x64	48.66 MB	jre-7u80-macosx-x64.dmg
Mac OS X x64	44.61 MB	jre-7u80-macosx-x64.tar.gz
Solaris x86	52.33 MB	jre-7u80-solaris-i586.tar.gz
Solaris x64	16.16 MB	jre-7u80-solaris-x64.tar.gz
Solaris SPARC	55.05 MB	jre-7u80-solaris-sparc.tar.gz
Solaris SPARC 64-bit	18.16 MB	jre-7u80-solaris-sparcv9.tar.gz
Windows x86 Online	0.89 MB	jre-7u80-windows-i586-iftw.exe
Windows x86 Offline	28.14 MB	jre-7u80-windows-i586.exe
Windows x86	40.06 MB	jre-7u80-windows-i586.tar.gz
Windows x64	29.79 MB	jre-7u80-windows-x64.exe
Windows x64	41.77 MB	jre-7u80-windows-x64.tar.gz

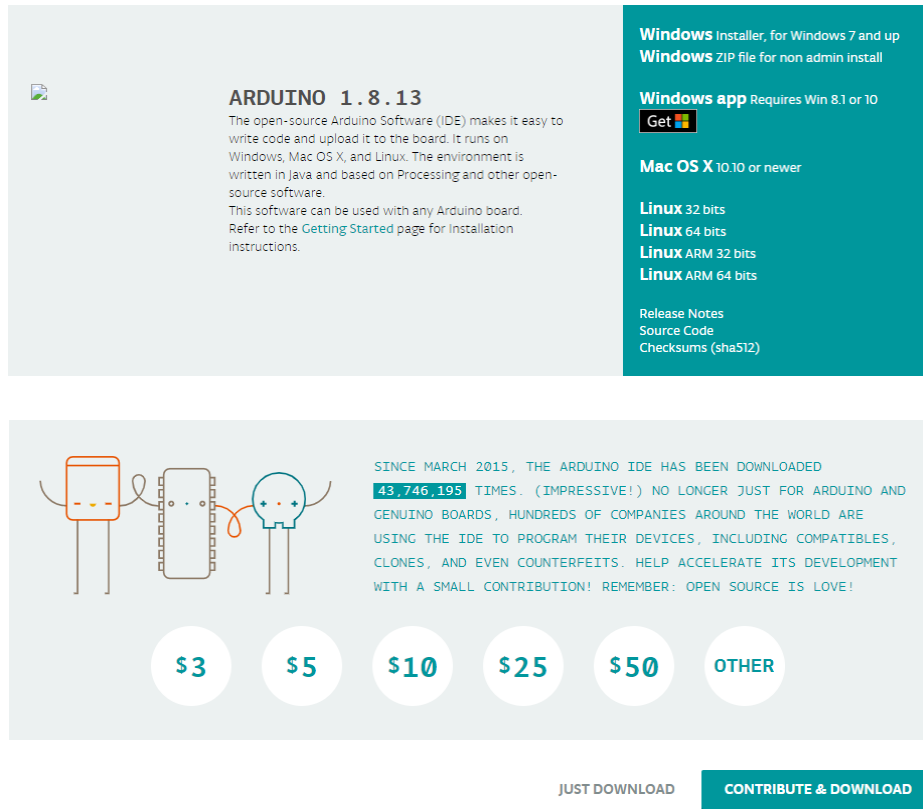
Hình 2.10 Các phiên bản JRE

b. Cài đặt Arduino IDE

Bước 1: Truy cập địa chỉ <http://arduino.cc/en/Main/Software>.

Đây là nơi lưu trữ cũng như cập nhật các bản IDE của Arduino. Bấm vào mục Windows ZIPfile for non admin install.

Download the Arduino IDE



ARDUINO 1.8.13
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.
This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

Windows Installer, for Windows 7 and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10
[Get](#)

Mac OS X 10.10 or newer

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **43,746,195** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3 **\$5** **\$10** **\$25** **\$50** **OTHER**

[JUST DOWNLOAD](#) [CONTRIBUTE & DOWNLOAD](#)

Hình 2.11 Tải xuống Arduino IDE

Bước 2: Sau khi download xong, các bạn bấm chuột phải vào file vừa download arduino-1.8.13-windows.zip và chọn “Extract here” để giải nén.

Bước 3: Copy thư mục arduino-1.8.13 vừa giải nén đến nơi lưu trữ.

Bước 4: Chạy file trong thư mục arduino-1.8.13\ để khởi động Arduino IDE

c. Cài đặt Driver

Để máy tính của bạn và board Arduino giao tiếp được với nhau, chúng ta cần phải cài đặt driver trước tiên.

Nếu bạn dùng Windows 8, trong một số trường hợp Windows không cho phép bạn cài Arduino driver (do driver không được kí bằng chữ kí số hợp lệ). Do vậy bạn cần vào Windows ở chế độ Disable driver signature enforcement thì mới cài được driver

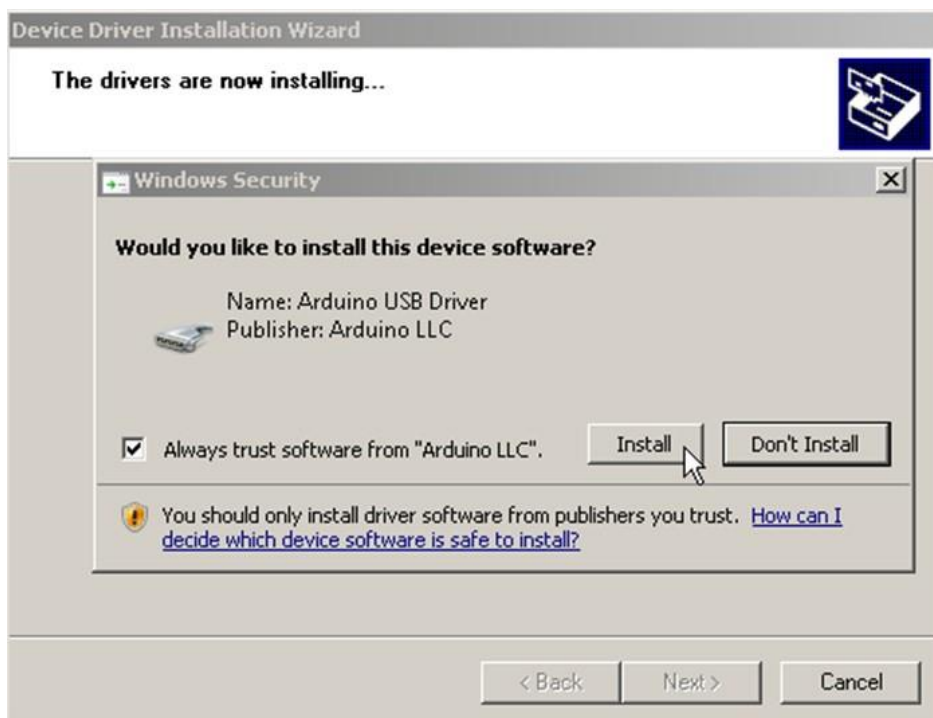
Xem hướng dẫn thực hiện tại bài viết Disabling Driver Signature on Windows 8 của SparkFun.

Đầu tiên, các bạn chạy file `arduino-1.6.4\drivers\dpinst-x86.exe` (Windows x86) hoặc `arduino-1.6.4\drivers\dpinst-amd64.exe` (Windows x64). Cửa sổ “Device Driver Installation Wizard” hiện ra, các bạn chọn Next để tiếp tục.



Hình 2.12 Cài đặt Driver cho Arduino IDE

Khi có yêu cầu xác nhận cài đặt driver, chọn “Install”



Đợi khoảng 10 giây trong lúc quá trình cài đặt diễn ra ...



c) Các chức năng cơ bản trong Arduino IDE

a. Vùng lệnh

Bao gồm các nút lệnh menu (File, Edit, Sketch, Tools, Help).

Phía dưới là các icon cho phép sử dụng nhanh các chức năng thường dùng của IDE được miêu tả như sau:

Icon	Chức năng
	Biên dịch chương trình đang soạn thảo để kiểm tra các lỗi lập trình.
	Biên dịch và upload chương trình đang soạn thảo.
	Mở một trang soạn thảo mới.
	Mở các chương trình đã lưu.
	Lưu chương trình đang soạn.
	Mở cửa sổ Serial Monitor để gửi và nhận dữ liệu giữa máy tính và board Arduino.

Hình 2.13 Vùng lệnh cơ bản trong Arduino IDE

b. Vùng viết chương trình

Bạn sẽ viết các đoạn mã của mình tại đây. Tên chương trình của bạn được hiển thị ngay dưới dãy các Icon, ở đây nó tên là “**Blink**”. Để ý rằng phía sau tên chương trình có một dấu “\$”. Điều đó có nghĩa là đoạn chương trình của bạn chưa được lưu lại.

c. Vùng thông báo

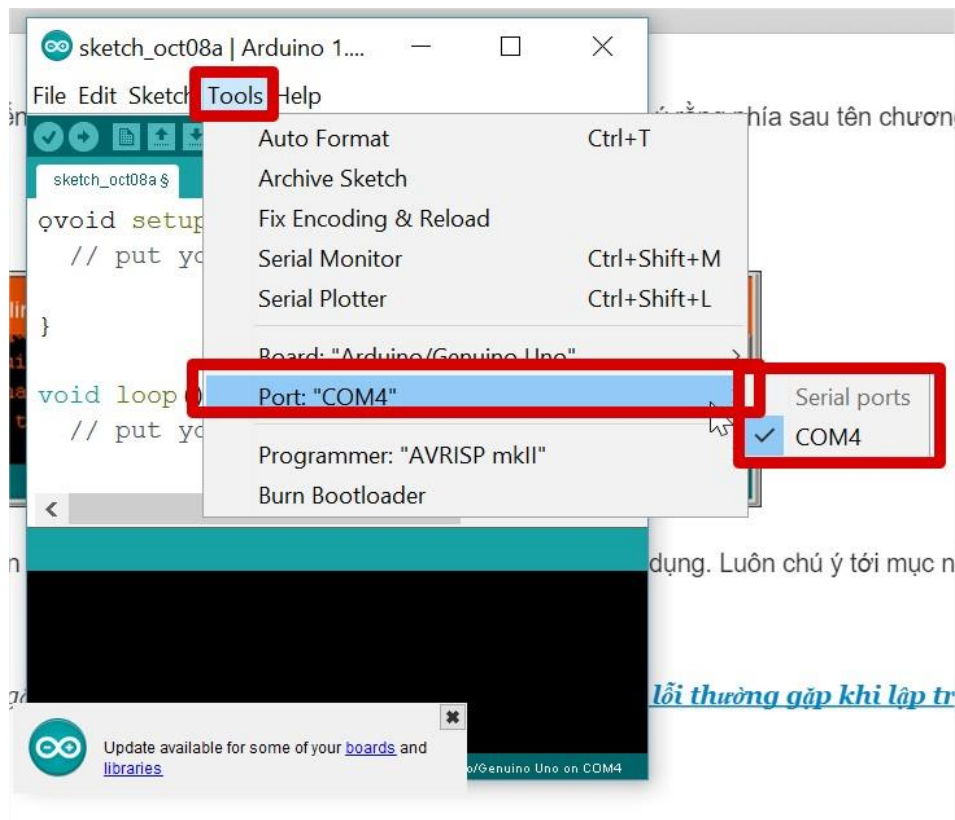
Những thông báo từ IDE sẽ được hiển thị tại đây. Để ý rằng góc dưới cùng bên phải hiển thị loại board Arduino và cổng COM được sử dụng. Luôn chú ý tới mục này bởi nếu chọn sai loại board hoặc cổng COM, bạn sẽ không thể upload được code của mình.



Hình 2.14 Vùng thông báo trong Arduino IDE

d) Một số lưu ý

Khi lập trình, các bạn cần chọn port (cổng kết nối khi gắn board vào) và board (tên board mà bạn sử dụng). Giả sử, bạn đang dùng mạch Arduino Uno R3, và khi gắn board này vào máy tính bằng cáp USB nó được nhận là COM4 thì bạn chỉnh như thế này là có thể lập trình được nhé.



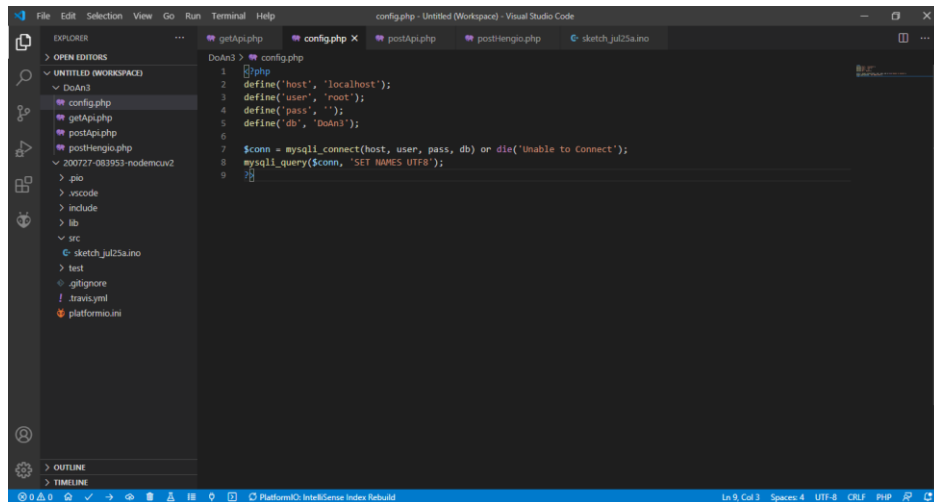
Hình 2.15 Chọn cổng COM cho arduino

2.3.2 Visual Studio Code

Visual Studio Code là một trình biên tập mã được phát triển bởi Microsoft dành cho Windows, Linux và macOS. Nó hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nó 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.

a) Giao diện

Lưu ý: Visual Studio Code không giống Visual Studio IDE



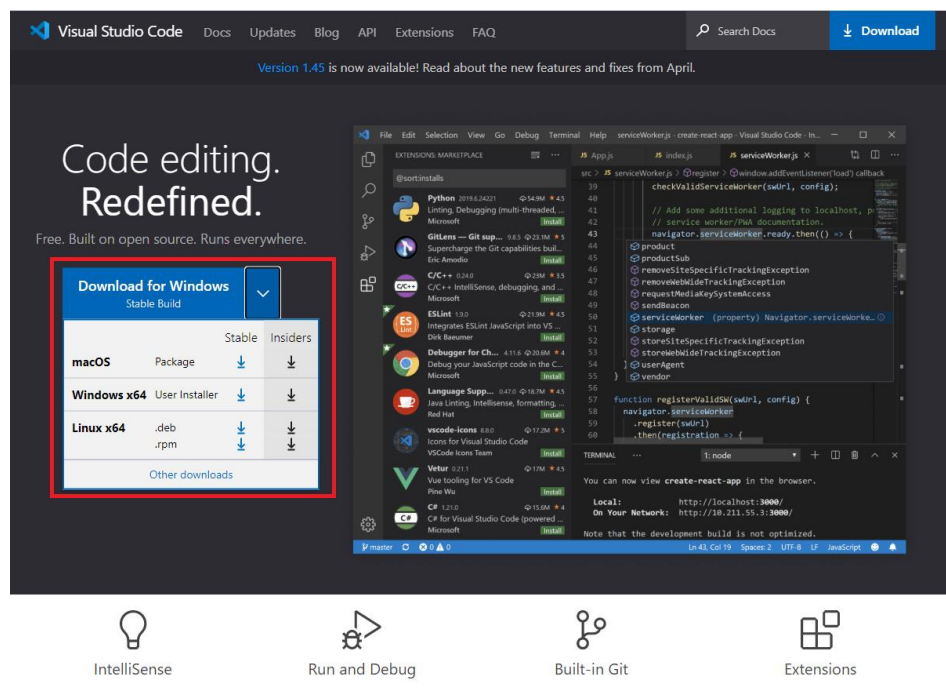
Hình 2.16 Giao diện Code Visual Studio Code

b) Cài đặt

a. Tải và cài đặt vscode

Chọn phiên bản phù hợp với thiết bị của các bạn và tải về.

<https://code.visualstudio.com/>

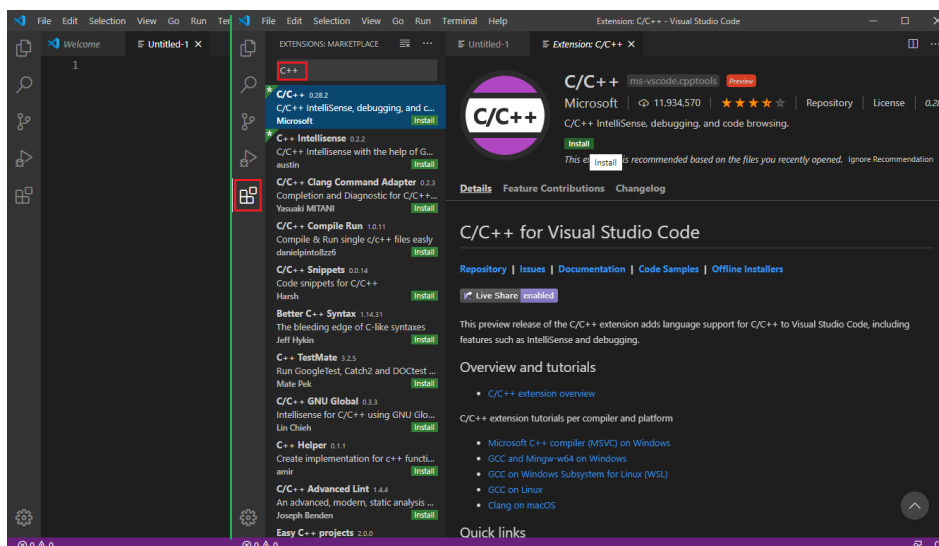


Hình 2.17 Cài đặt Visual Studio Code

b. Cài extension php

Các bạn ấn vào Extensions hoặc Ctrl + Shift + X, để mở giao diện như hình trên.

Ví dụ: các bạn gõ trên thanh tìm kiếm từ khóa “C++”, sau đó chọn extension C/C++ do Microsoft phát hành và ấn Install để cài đặt.



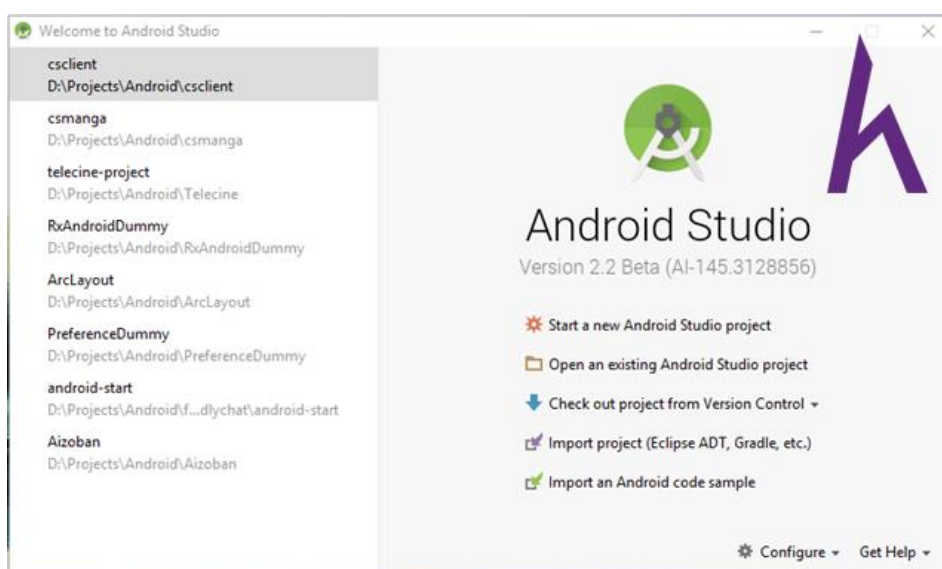
Hình 2.18 Cài các extension cơ bản

2.3.3 Android Studio

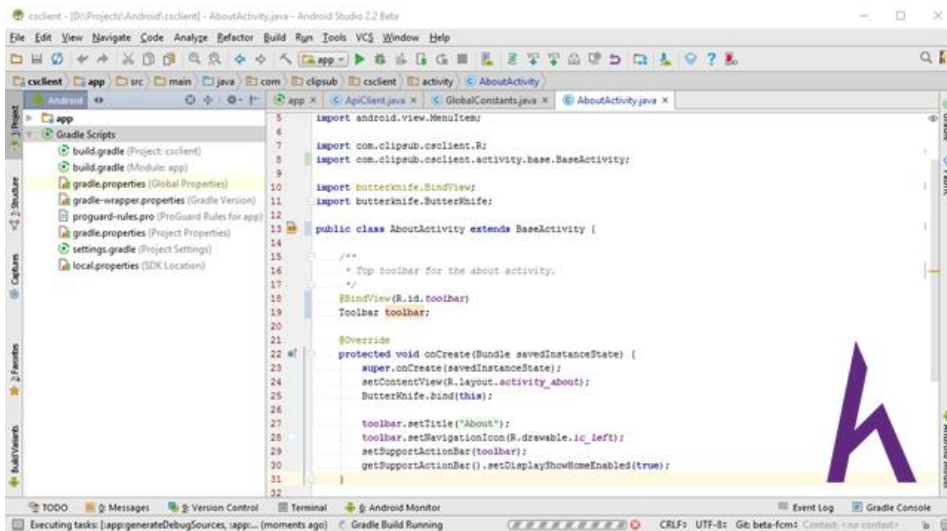
a) Giới thiệu

Android Studio là một môi trường phát triển ứng dụng tích hợp (IDE) dành riêng cho Android, mã nguồn mở, dựa trên IDE **Java IntelliJ** của hãng **JetBrains** (đối thủ với Eclipse và Netbeans, vốn khá quen thuộc với dân lập trình Java).

Android Studio chạy trên Windows, Mac và Linux, nhằm thay thế cho **Eclipse Android Development Tool (ADT)** vốn được sử dụng làm IDE chính trong các năm trước đó.



Hình 2.19 Giao diện màn hình đầu của Android Studio



Hình 2.20 Một project bình thường thì có dạng như thế này

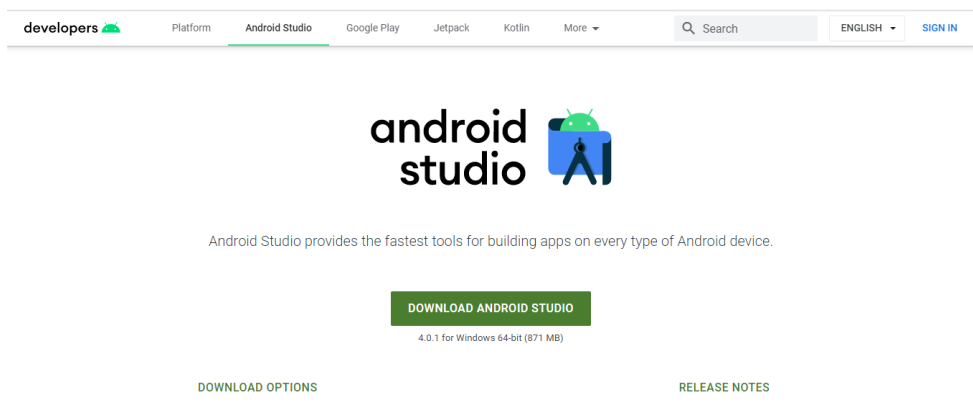
b) Cài đặt

Bước 1: Chúng ta sẽ cần có JDK (Java Development Kit) trước khi làm bất cứ việc gì:

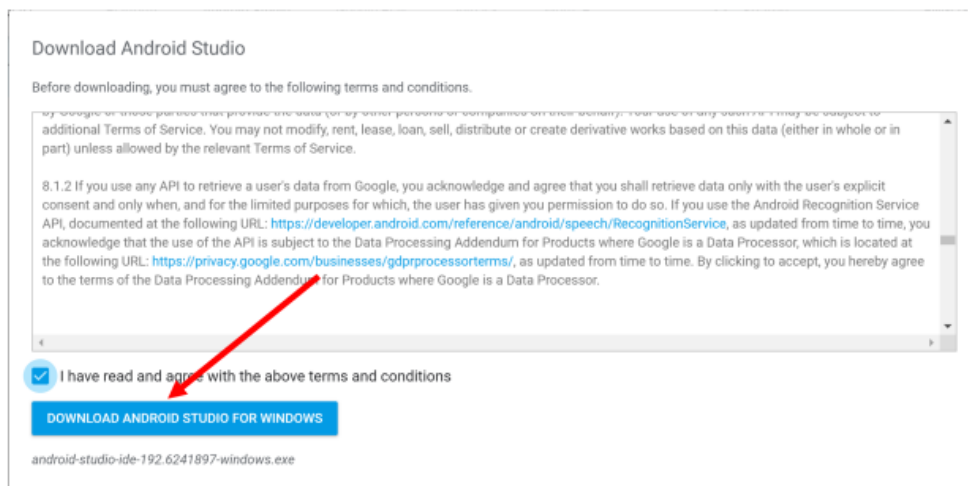
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Bước 2: Chuẩn bị Android Studio và Android SDK.

Truy cập vào trang: <https://developer.android.com/studio/index.html>



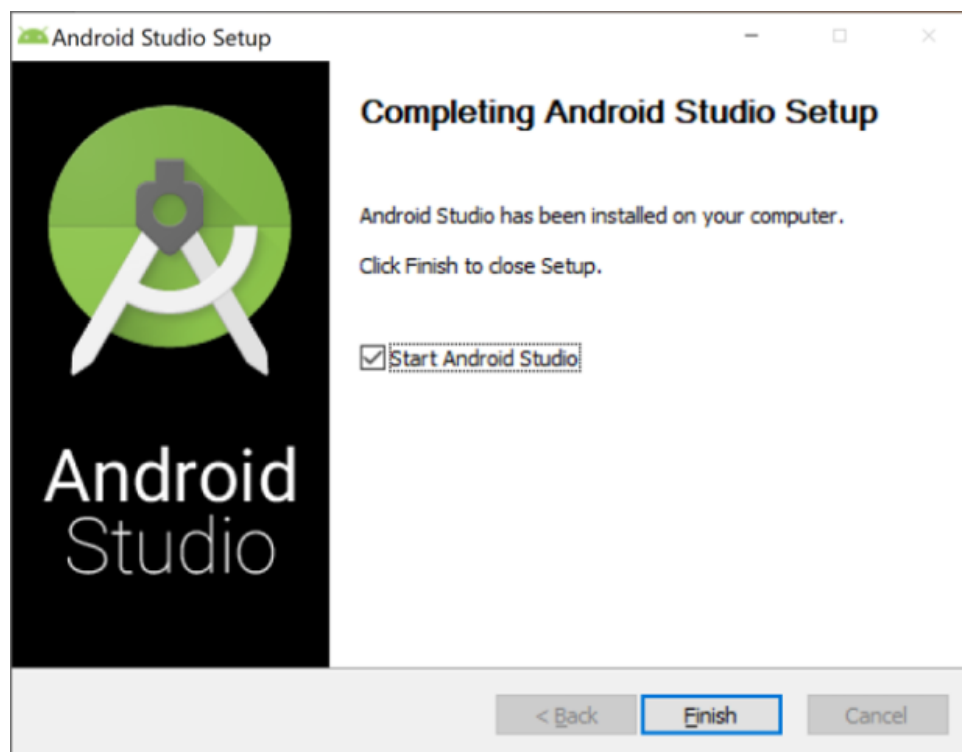
Hình 2.21 Cài đặt Android Studio



Hình 2.22 Cài đặt Android Studio

Bước 3: Chạy file .exe vừa tải về và tiến hành cài đặt.

Kết quả khi hoàn tất.



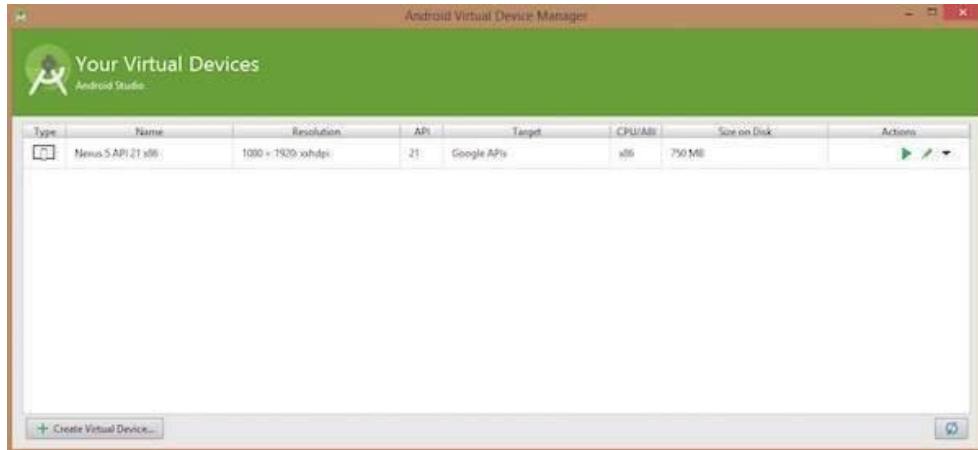
Hình 2.23 Cài đặt Android Studio

Bước 4: Nếu chương trình yêu cầu cài đặt SDK thì tiếp tục thực hiện theo chỉ dẫn.

c) Tạo Android Virtual Device

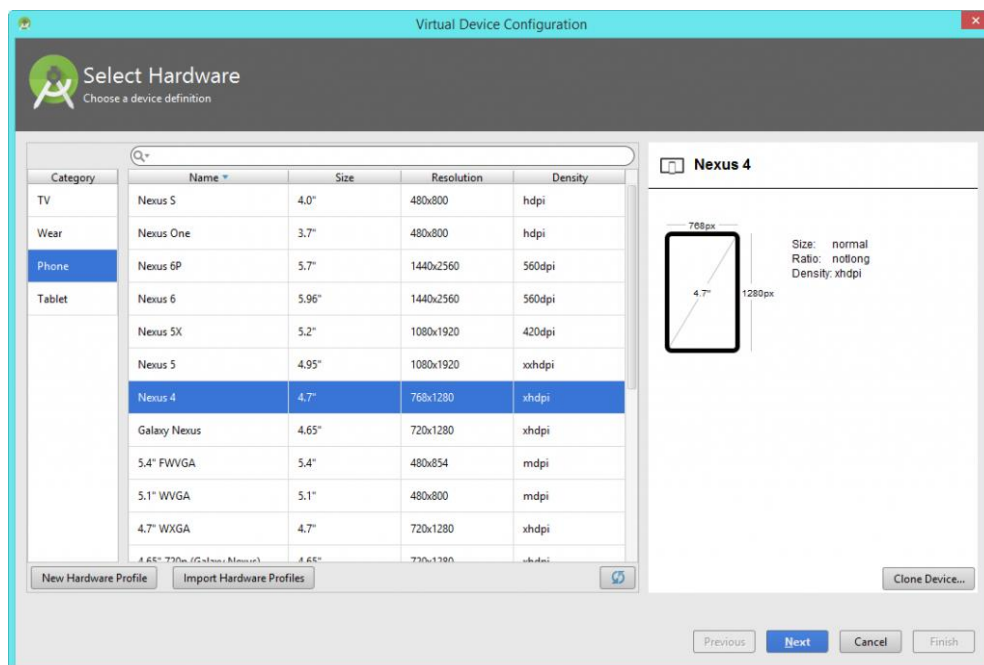
Nhấp chuột vào biểu tượng AVD Manager trong Android Studio

Chọn ‘Create Virtual Device’



Hình 2.24 Cài máy ảo trong Android Studio

Chọn thiết bị => Next



Hình 2.25 Cài máy ảo trong Android Studio

Chọn System Image (Theo đề xuất các bạn nên chọn là 1 trong các System Image trong thẻ Recommended). Trong hình tôi đang chọn ‘Lollipop với API Level 22’. => Next => Finish => Khởi động máy ảo

Chương 3 PHÂN TÍCH & THIẾT KẾ HỆ THỐNG

3.1 GIỚI THIỆU

Chương này trình bày về sơ đồ khối, sơ đồ nguyên lý, sơ đồ hoạt động của các board mạch của hệ thống, lưu đồ thuật toán. Quy trình xây dựng WebServer và App Android và các chức năng liên quan.

3.2 THIẾT KẾ SƠ ĐỒ KHỐI

Với các yêu cầu đưa ra, nhóm đã dựng lên sơ đồ khối như sau:



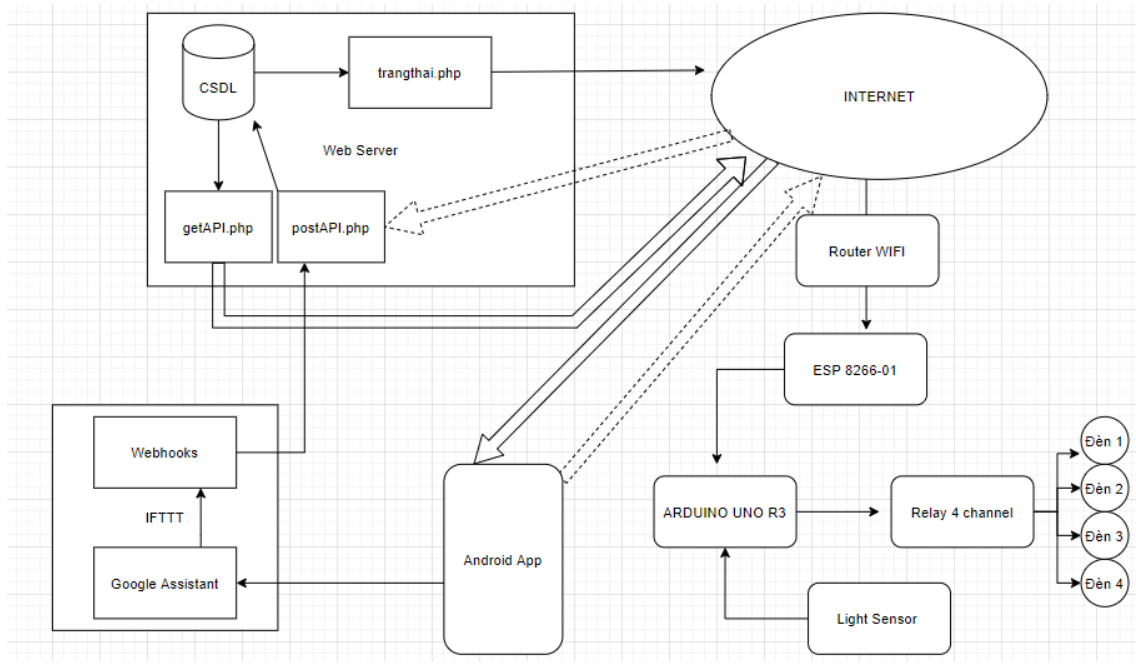
Hình 3.1 Sơ đồ khối hệ thống

❖ Chức năng các khối:

- Khối nguồn: cung cấp nguồn hoạt động cho hệ thống.
- Khối cảm biến ánh sáng: (cảm quang) tự động nhận biết mức sáng môi trường và gửi tín hiệu về Arduino Uno R3 xử lý.
- Khối tín hiệu: cảnh báo một hành động như kết nối, bật tắt, lỗi xảy ra trong quá trình hoạt động.
- Khối bật tắt các thiết bị: nhận tín hiệu điều khiển từ Arduino Uno R3 và thực hiện bật hay tắt các thiết bị tương ứng.

- Khối xử lý trung tâm Arduino Uno R3 toàn mạch: nhận tín hiệu từ WebServer thông qua module wifi sau đó sẽ gửi tín hiệu đến các khối để bật tắt và nhận tín hiệu từ các khối khác để xử lý.

3.3 NGUYÊN LÝ HOẠT ĐỘNG



Hình 3.2 Sơ đồ nguyên lý hoạt động

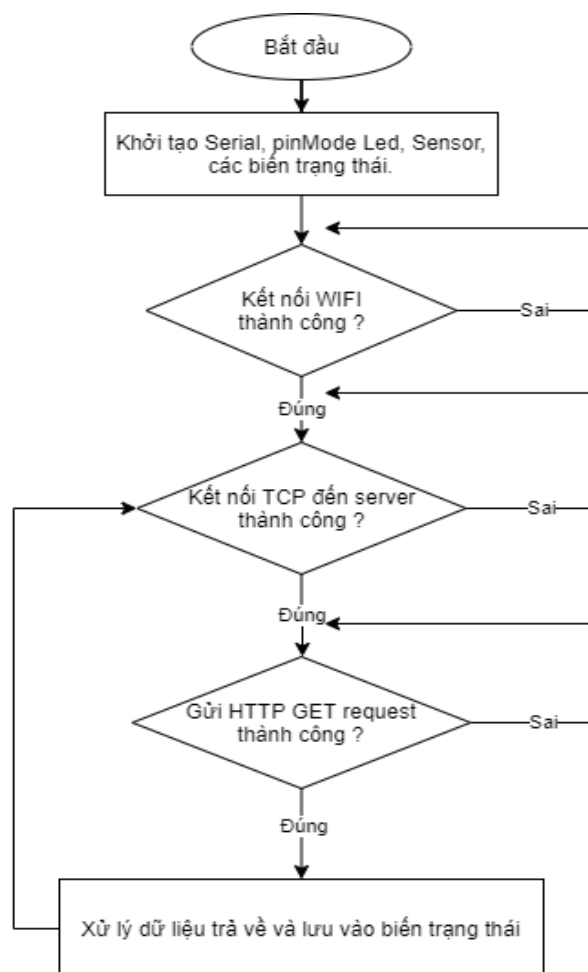
❖ Nguyên lý hoạt động của hệ thống

- Web server đóng vai trò quan trọng trong việc điều khiển tất cả thiết bị chiếu sáng, quạt và cửa. Web server được viết bằng ngôn ngữ lập trình Javascript và cơ sở dữ liệu MongoDB, LowDB sau đó được up lên host heroku.
- ESP-01 nhận dữ liệu trạng thái yêu cầu từ Web server, tùy theo điều kiện mà bật hay tắt cho phù hợp.
- Cảm biến ánh sáng gửi tín hiệu (HIGH/LOW) về Arduino để điều khiển đèn số 4 (chân D4).
- Relay nhận tín hiệu từ Arduino Uno R3 để điều khiển các thiết bị.
- Khi nhấn vào nút trong bảng điều khiển, đèn tương ứng sẽ đổi trạng thái (ví dụ: đang bật xong nhấn sẽ tắt) và gửi tín hiệu lên server đồng bộ dữ liệu.

- Hỗ trợ chatbot nhận yêu cầu từ người dùng và gửi tín hiệu đến Web Server:
 - Nền tảng Dialogflow: nhận diện yêu cầu xong rồi thực hiện yêu cầu bởi Webhooks.
 - Nền tảng Webhooks: gửi một yêu cầu qua API đến Web server với trạng thái của thiết bị tương ứng.
- Ứng dụng di động với các chức năng bật tắt điều khiển thông qua Web server bằng API.
- Đặc biệt, với chức năng hẹn giờ giúp quản lý, lập thời gian biểu bật tắt các thiết bị theo yêu cầu người dùng.

3.4 LƯU ĐỒ THUẬT TOÁN

3.4.1 Lưu đồ lấy trạng thái của các đèn và cảm biến từ Web server



Hình 3.3 Lưu đồ lấy trạng thái của các đèn và cảm biến từ Web server

Giải thích lưu đồ thuật toán:

B1: Bắt đầu

B2: Khởi tạo Serial, port các đèn và cảm biến, chế độ pinMode, các biến lưu trạng thái của đèn và cảm biến

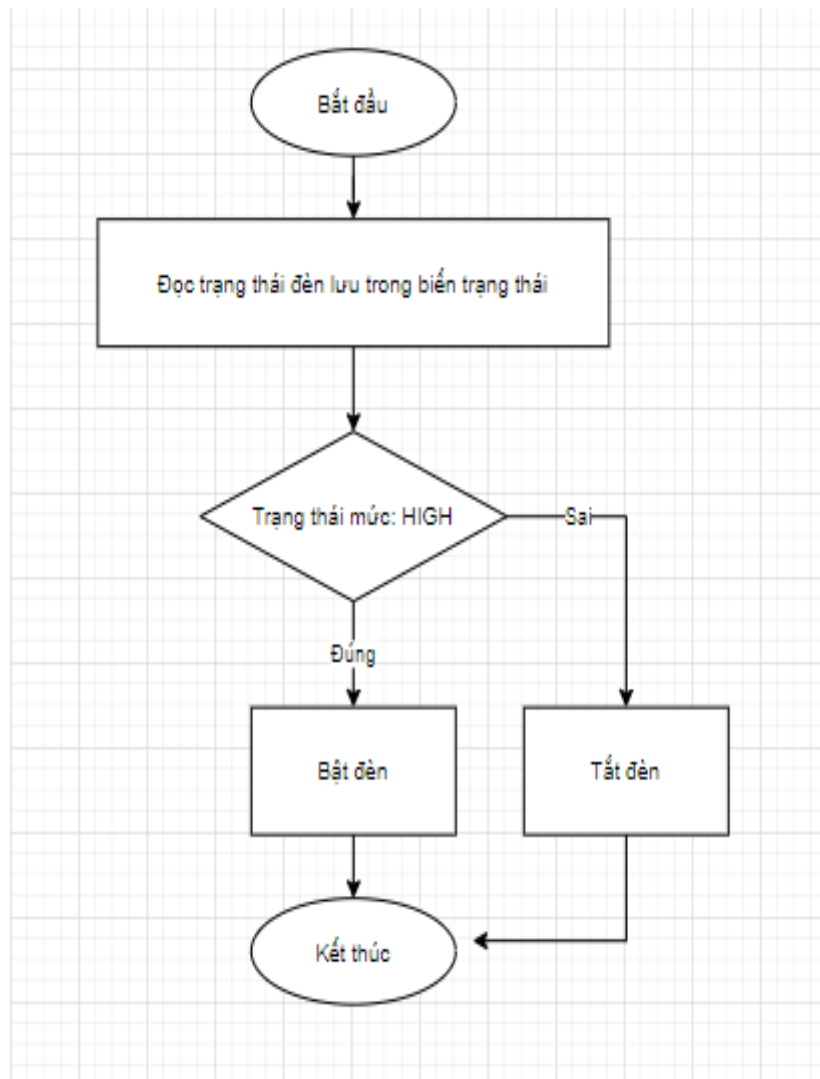
B3: Gửi yêu cầu kết nối tới WIFI, nếu thất bại quay lại B3

B4: Gửi kết nối tới Web Server phương thức TCP qua cổng mặc định 80, nếu thất bại quay lại B4

B5: Gửi một yêu cầu lên Server với phương thức HTTP GET METHOD, nếu mã yêu cầu trả về khác 200 OK thì quay lại B4

B6: Xử lý dữ liệu trả về và gán trạng thái đèn cho biến trạng thái, rồi quay lại B4

3.4.2 Lưu đồ điều khiển các đèn



Hình 3.4 Lưu đồ điều khiển các bóng đèn bởi Relay

Giải thích lưu đồ thuật toán:

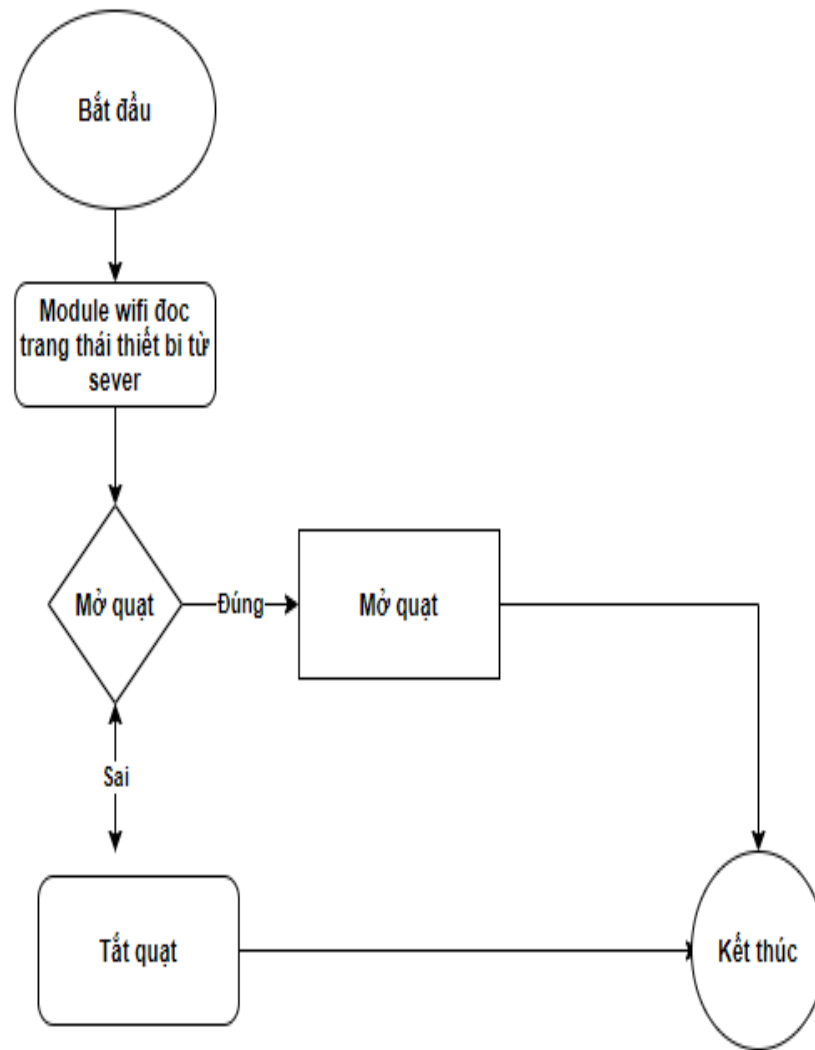
B1: Bắt đầu

B2: Đọc trạng thái đèn được lưu trong các biến trạng thái ở lưu đồ trên

B3: Nếu trạng thái đèn ở mức HIGH (cao) thì bật đèn, mức LOW(thấp) tắt đèn

B4: Kết thúc.

3.4.3 Lưu đồ điều khiển quạt



Hình 3.5 Lưu đồ điều khiển quạt

Giải thích lưu đồ thuật toán:

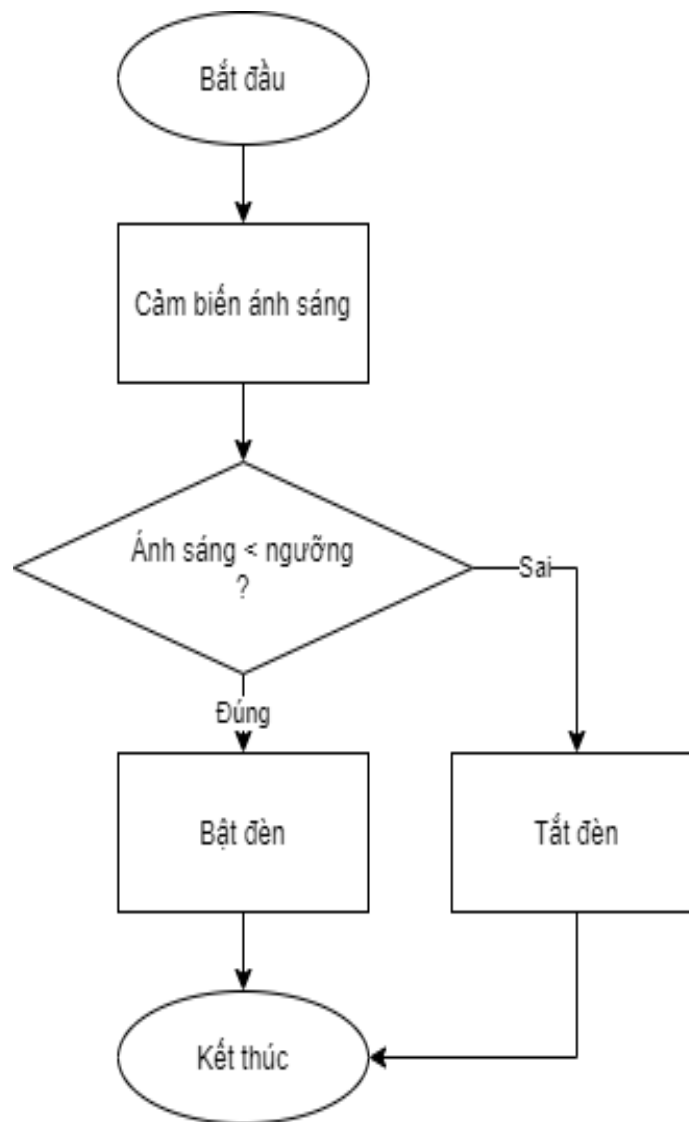
B1: Bắt đầu

B2: Đọc trạng thái của quạt

B3: Nếu trạng thái quạt là mở thì mở quạt, nếu không thì tắt quạt

B4: Kết thúc.

3.4.4 Lưu đồ điều khiển đèn bởi cảm biến ánh sáng



Hình 3.6 Lưu đồ điều khiển đèn bởi cảm biến ánh sáng

Giải thích lưu đồ thuật toán:

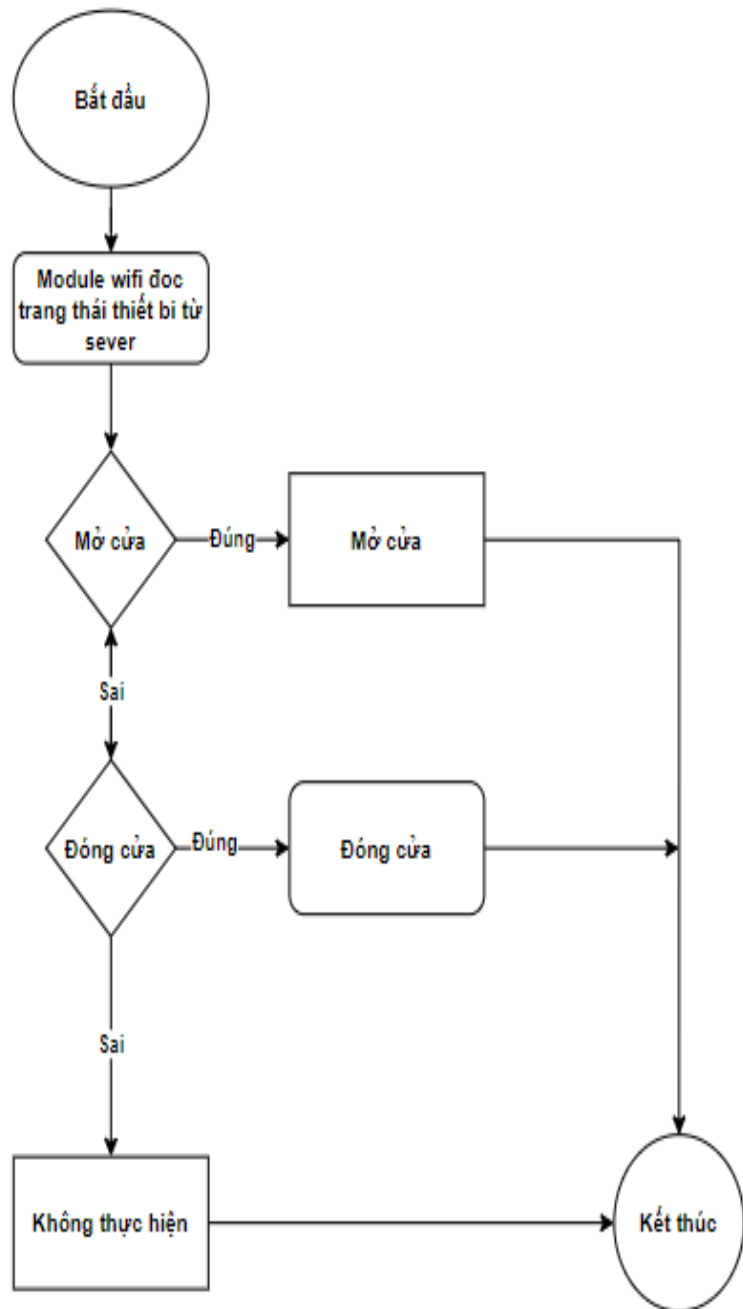
B1: Bắt đầu

B2: Đọc giá trị của cảm biến ánh sáng

B3: Nếu mức ánh sáng vượt ngưỡng thì bật đèn, chưa vượt thì tắt đèn

B4: Kết thúc.

3.4.5 Lưu đồ điều khiển cửa



Hình 3.7 Lưu đồ điều khiển cửa

Giải thích lưu đồ thuật toán:

B1: Bắt đầu

B2: Đọc trạng thái được lưu trong các biến trạng thái ở lưu đồ trên

B3: Nếu trạng thái cửa ở trạng thái mở, thì motor sẽ quay theo chiều mở cửa và ngược lại

B4: Kết thúc.

Chương 4 TRIỂN KHAI THỰC HIỆN

4.1 LẮP RÁP PHẦN CỨNG

Từ sơ đồ khối và sơ đồ toàn mạch chúng em lắp ráp mạch thiết bị và thiết kế mô hình nhà từ bìa cứng để xây dựng một mô hình hoàn chỉnh.

4.2 THIẾT KẾ PHẦN MỀM

4.2.1 Chương trình cho Arduino Uno R3

Chương trình được viết bằng ngôn ngữ lập trình C, trên môi trường lập trình Arduino IDE.



```
sketch_jul25a | Arduino 1.8.12
File Edit Sketch Tools Help

sketch_jul25a $

#include <ESP8266WiFi.h> // Các thư viện Wifi và http
#include <ESP8266HTTPClient.h> //.
#include <WiFiClient.h> //.
#include <Arduino_JSON.h> // Thư viện đọc dạng Json
#include <String.h> // Thư viện chuỗi
const char* ssid = "VIETTEL"; // SSID của wifi trong nhà
const char* password = "06122005"; // Mật khẩu wifi
const char* serverGet = "http://smartwatchsw.000webhostapp.com/getApi.php"; // địa chỉ nhận dữ liệu từ Web server
const char* serverPost = "http://smartwatchsw.000webhostapp.com/postApi.php"; // địa chỉ gửi dữ liệu lên Web server

int LED_1 = 5; // cổng D1
int LED_2 = 4; // cổng D2
int LED_3 = 0; // cổng D3
int LED_4 = 2; // cổng D4
int LED_ALERT = 16; // cổng D0
int sensor = 14; // cổng D5
int BTN_1 = 12; // cổng D6
int BTN_2 = 13; // cổng D7
int BTN_3 = 15; // cổng D8
int BTN_4 = 3; // cổng RX

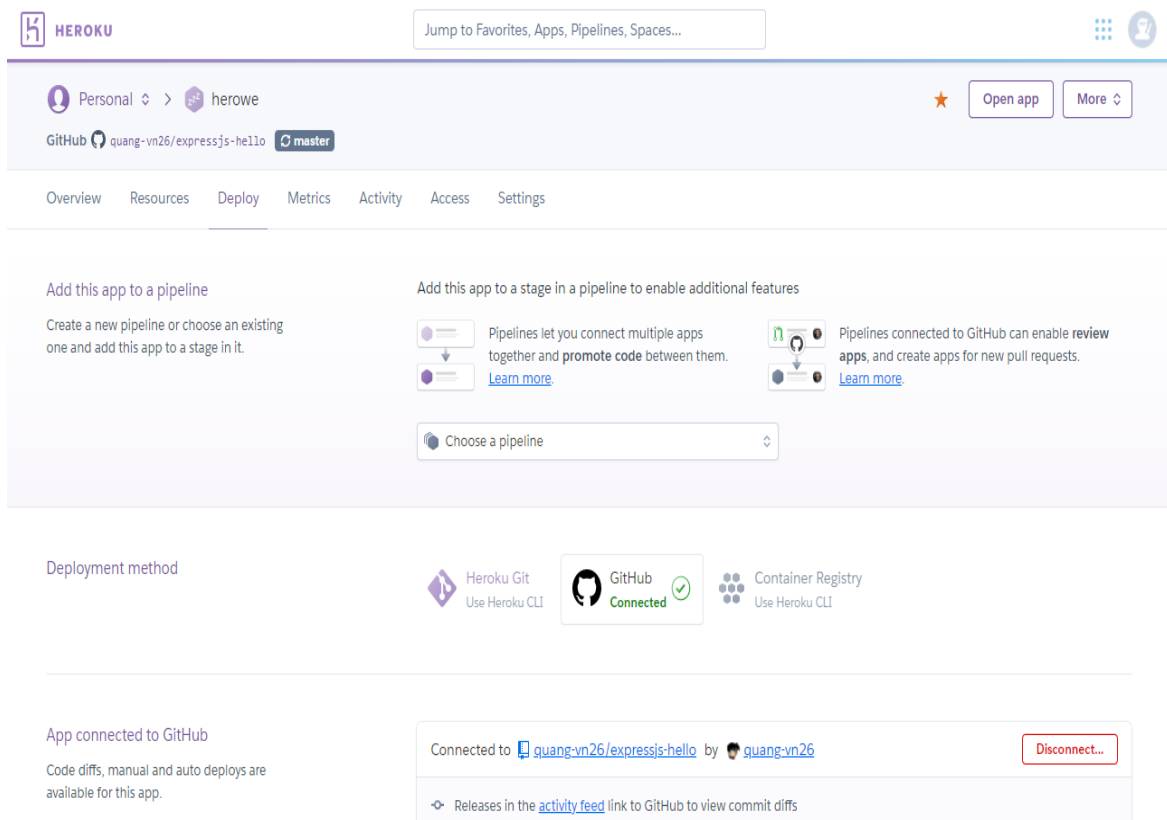
int led_1_status = 0; // trạng thái của đèn số 1
int led_2_status = 0; // giá trị 0 là tắt
int led_3_status = 0; // giá trị 1 là bật
int led_4_status = 0;
int sensor_status = 0; // trạng thái của cảm biến ánh sáng

Done Saving.
```

Hình 4.1 Chương trình cho Arduino Uno R3

4.2.2 Xây dựng Web Server

Web server được viết bằng ngôn ngữ lập trình javascript và up lên heroku, một tên miền miễn phí.



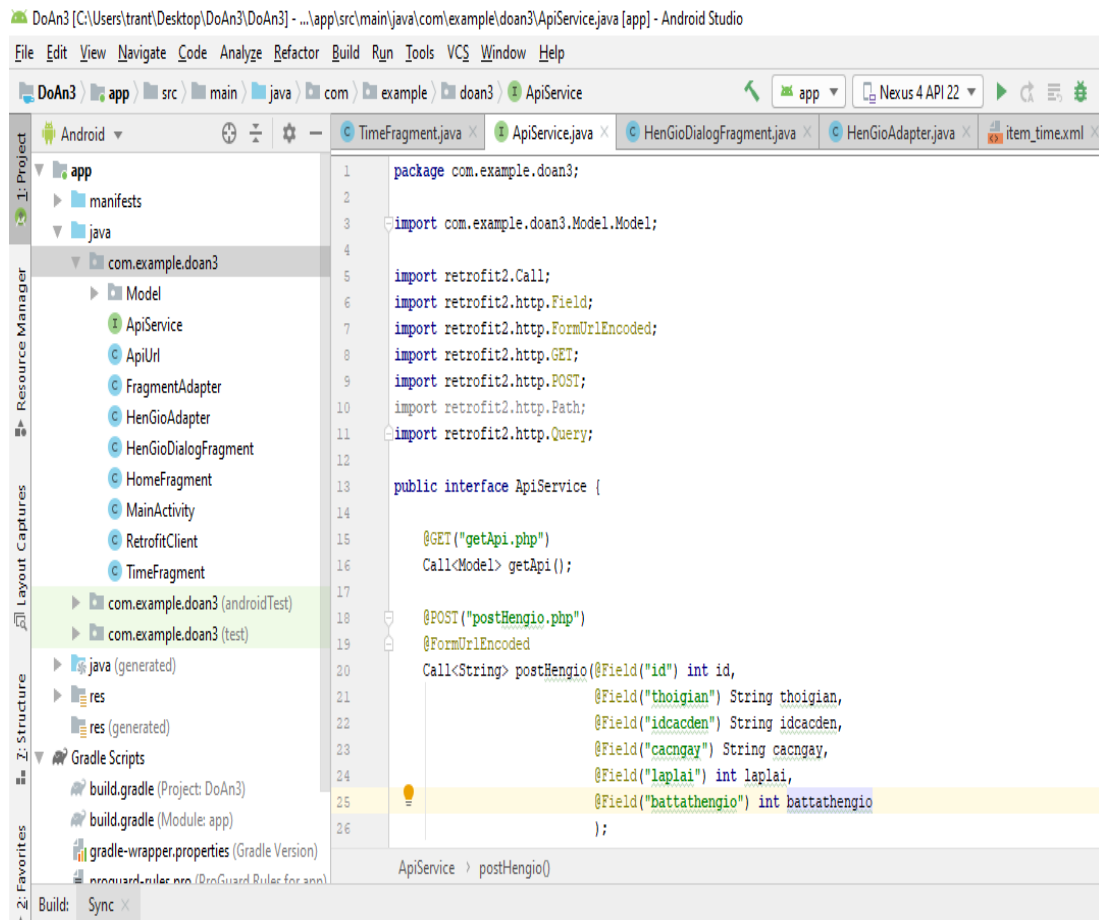
Hình 4.2 Xây dựng Web Server

```
{
  "arduino": [
    {
      "id": "led_1",
      "status": "_tat",
      "date": "2020-09-08T14:33:59.633Z"
    },
    {
      "id": "led_2",
      "status": "_tat",
      "date": "2020-09-08T14:33:48.860Z"
    },
    {
      "id": "led_3",
      "status": "_bat",
      "date": "2020-09-08T13:47:36.623Z"
    },
    {
      "id": "led_4",
      "status": "_bat",
      "date": "2020-09-03T08:17:17.396Z"
    },
    {
      "id": "fan_1",
      "status": "_bat",
      "date": "2020-09-03T08:17:18.446Z"
    },
    {
      "id": "fan_2",
      "status": "_bat",
      "date": "2020-09-03T08:17:18.446Z"
    }
  ]
}
```

Hình 4.3 Cài máy ảo trong Android Studio

4.2.3 Ứng dụng di động Android

Được lập trình bởi ngôn ngữ Java, có thể điều khiển các thiết bị qua wifi



Hình 4.4 Phần mềm điều khiển sử dụng Android Studio

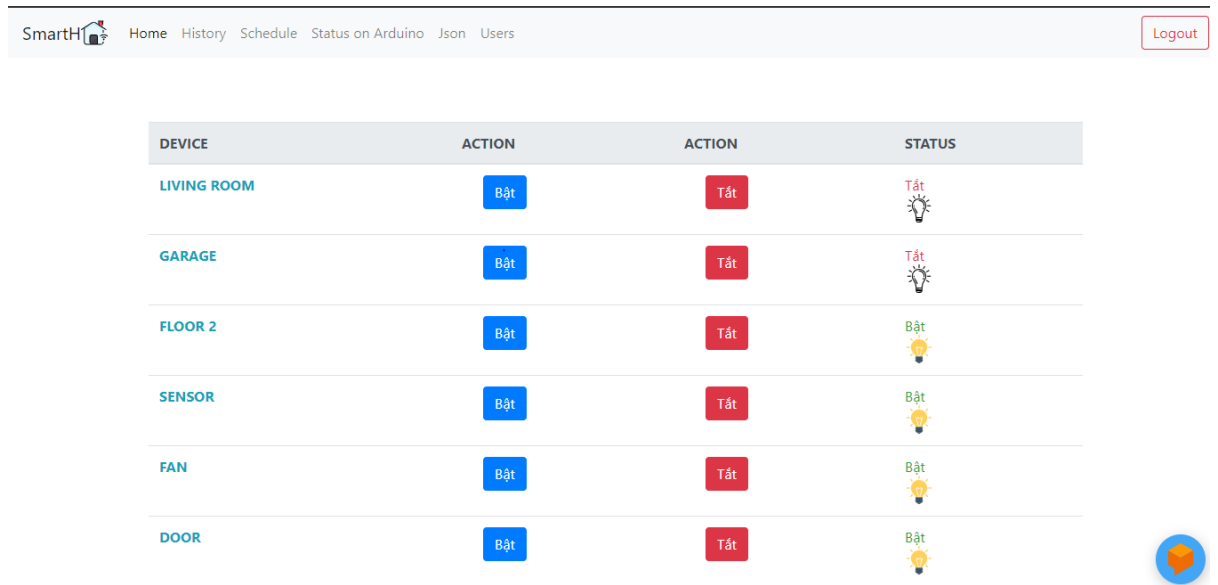
Chương 5 KẾT QUẢ ĐẠT ĐƯỢC

5.1 MÔ HÌNH NHÀ THÔNG MINH

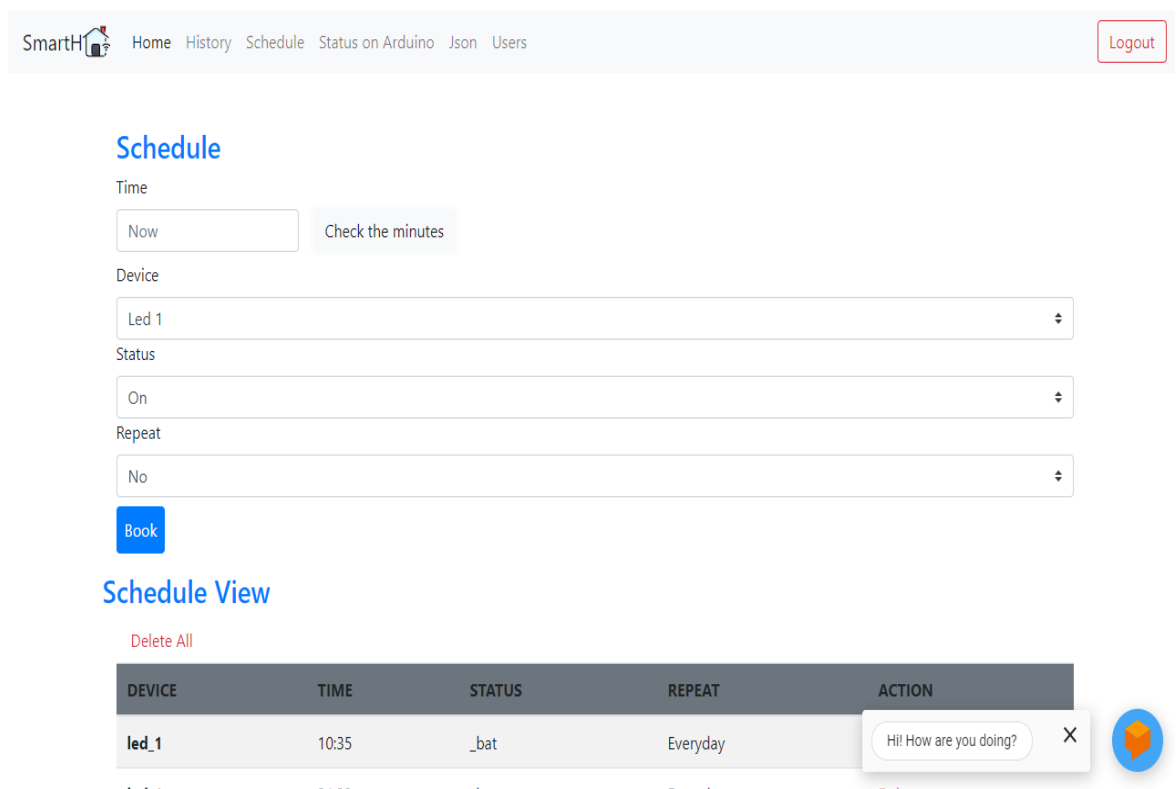


Hình 5.1 Mô hình nhà thông minh

5.2 WEB SERVER



Hình 5.2 Trang chủ điều khiển thiết bị



Hình 5.3 Trang đặt lịch

SmartH

[Home](#)
[History](#)
[Schedule](#)
[Status on Arduino](#)
[Json](#)
[Users](#)

Logout

Schedule

Time

Device

Status

Repeat

Schedule View

Delete All

DEVICE	TIME	STATUS	REPEAT	ACTION
led_1	10:35	_bat	Everyday	Delete
led_1	11:22	_bat	Everyday	Delete

ChatBot

Hi! How are you doing?

Now it is: on

Ask something...

Open Fan

X

Hình 5.4 Trang chatbot

SmartH

[Home](#)
[History](#)
[Schedule](#)
[Status on Arduino](#)
[Json](#)
[Users](#)

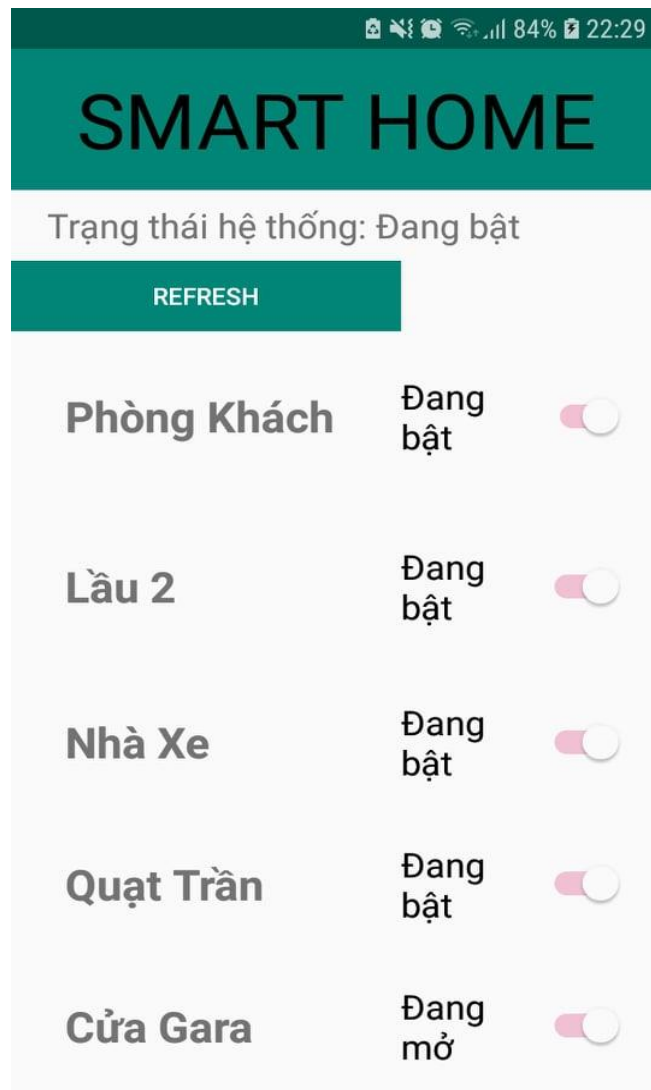
Logout

Users

Avatar	Name	Action	Action
	name	View	Edit Delete
	Alan Alan Alan	View	Edit Delete

Hình 5.4 Trang người dùng

5.3 ỨNG DỤNG DI ĐỘNG



Hình 5.5 Ứng dụng di động điều khiển thiết bị chiếu sáng

Chương 6 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1 KẾT LUẬN

- ❖ Hoàn tất các mục tiêu đã đề ra của đề tài
- ❖ Điều khiển được các thiết bị qua Smartphone Android và trang Web
- ❖ Hệ thống trong quá trình hoạt động không xảy ra lỗi lầm
- ❖ Mô hình thiết kế có tính thẩm mỹ
- ❖ Đánh giá, so sánh với các hệ thống thực tế khoảng 65%.

6.2 HẠN CHẾ

- ❖ Tốc độ đáp ứng chưa được nhanh
- ❖ Server được tạo ra là server miễn phí nên có nhiều hạn chế trong tốc độ truyền tải và sự ổn định
- ❖ Các chức năng chỉ đáp ứng ở mức độ nghiên cứu nền tảng
- ❖ Giao diện web và ứng dụng di động còn đơn giản, chưa có đầy đủ các chức năng, còn hạn chế.

6.3 HƯỚNG PHÁT TRIỂN

- ❖ Thêm các cảm biến như: cảm biến chuyển động, cảm biến nhiệt, cảm biến tiệm cận, v.v
- ❖ Sử dụng domain có băng thông và tốc độ xử lý nhanh
- ❖ Thêm điều khiển qua Bluetooth
- ❖ Chế độ hẹn giờ thực hiện offline qua module thời gian thực RTC DS3231
- ❖ Tối ưu hóa mã nguồn và thêm các chức năng mới đáp ứng được nhu cầu quản lý.

DANH MỤC TÀI LIỆU THAM KHẢO

Trang web tham khảo:

- [1] Cộng đồng Arduno Việt Nam, “Giới thiệu Arduino Uno R3”, 26/02/2017
<http://arduino.vn/bai-viet/1496-esp8266-ket-noi-internet-phan-1-cai-dat-esp8266-lam-mot-socket-client-ket-noi-toi>
- [2] Cộng đồng Arduno Việt Nam, “Giới thiệu cảm biến ánh sáng”, 18/06/2014
<http://arduino.vn/bai-viet/180-gioi-thieu-cam-bien-anh-sang-va-cach-lap-trinh>
- [3] Cộng đồng Arduno Việt Nam, “Cách sử dụng Relay”, 02/09/2014
<http://arduino.vn/bai-viet/302-module-relay-cach-su-dung-ro-le-va-nhung-ung-dung-hay-cua-no>
- [4] Retrofit, “A type-safe HTTP client for Android and Java”, 05/08/2020
<https://square.github.io/retrofit/>
- [5] RANDOM NERD TUTORIALS, “ESP8266 Arduino Uno R3”, 05/06/2020
<https://randomnerdtutorials.com/esp8266-Arduino Uno R3-http-get-post-arduino/>