**­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**

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](#_Toc53032619)

[1.1 ĐẶT VẤN ĐỀ 10](#_Toc53032620)

[1.2 MỤC TIÊU 10](#_Toc53032621)

[1.3 NỘI DUNG NGHIÊN CỨU 11](#_Toc53032622)

[1.4 GIỚI HẠN 11](#_Toc53032623)

[1.5 CẤU TRÚC ĐỒ ÁN 11](#_Toc53032624)

[Chương 2 CƠ SỞ LÝ THUYẾT 13](#_Toc53032625)

[2.1 GIỚI THIỆU 13](#_Toc53032626)

[2.2 GIỚI THIỆU PHẦN CỨNG 13](#_Toc53032627)

[2.2.1 Bộ điều khiển trung tâm Arduino Uno R3 13](#_Toc53032628)

[2.2.2 Relay 15](#_Toc53032629)

[2.2.3 Điện trở 16](#_Toc53032630)

[2.2.4 Đèn led 17](#_Toc53032631)

[2.2.5 Module cảm biến ánh sáng 18](#_Toc53032632)

[2.2.6 Động cơ điện Motor 19](#_Toc53032633)

[2.3 GIỚI THIỆU CÔNG CỤ LẬP TRÌNH 19](#_Toc53032634)

[2.3.1 Arduino IDE 19](#_Toc53032635)

[2.3.2 Visual Studio Code 25](#_Toc53032636)

[2.3.3 Android Studio 27](#_Toc53032637)

[Chương 3 PHÂN TÍCH & THIẾT KẾ HỆ THỐNG 31](#_Toc53032638)

[3.1 GIỚI THIỆU 31](#_Toc53032639)

[3.2 THIẾT KẾ SƠ ĐỒ KHỐI 31](#_Toc53032640)

[3.3 NGUYÊN LÝ HOẠT ĐỘNG 32](#_Toc53032641)

[3.4 LƯU ĐỒ THUẬT TOÁN 33](#_Toc53032642)

[3.4.1 Lưu đồ lấy trạng thái của các đèn và cảm biến từ Web server 33](#_Toc53032643)

[3.4.2 Lưu đồ điều khiển các đèn 34](#_Toc53032644)

[3.4.3 Lưu đồ điều khiển quạt 36](#_Toc53032645)

[3.4.4 Lưu đồ điều khiển đèn bởi cảm biến ánh sáng 37](#_Toc53032646)

[3.4.5 Lưu đồ điều khiển cửa 38](#_Toc53032647)

[Chương 4 TRIỂN KHAI THỰC HIỆN 39](#_Toc53032648)

[4.1 LẮP RÁP PHẦN CỨNG 39](#_Toc53032649)

[4.2 THIẾT KẾ PHẦN MỀM 39](#_Toc53032650)

[4.2.1 Chương trình cho Arduino Uno R3 39](#_Toc53032651)

[4.2.2 Xây dựng Web Server 39](#_Toc53032652)

[4.2.3 Ứng dụng di động Android 41](#_Toc53032653)

[Chương 5 KẾT QUẢ ĐẠT ĐƯỢC 42](#_Toc53032654)

[5.1 MÔ HÌNH NHÀ THÔNG MINH 42](#_Toc53032655)

[5.2 WEB SERVER 43](#_Toc53032656)

[5.3 ỨNG DỤNG DI ĐỘNG 45](#_Toc53032657)

[Chương 6 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 46](#_Toc53032658)

[6.1 KẾT LUẬN 46](#_Toc53032659)

[6.2 HẠN CHẾ 46](#_Toc53032660)

[6.3 HƯỚNG PHÁT TRIỂN 46](#_Toc53032661)

**DANH MỤC HÌNH**

Trang

[Hình 2.1 Thiết bị Arduino Uno R3 12](#_Toc53033457)

[Hình 2.2 Sơ đồ chân Arduino Uno R3 14](#_Toc53033457)

[Hình 2.3 Relay 15](#_Toc53033458)

[Hình 2.4 Sơ đồ chân Relay 16](#_Toc53033459)

[Hình 2.5 Điện trở 17](#_Toc53033460)

[Hình 2.6 Giá trị của điện trở 17](#_Toc53033461)

[Hình 2.7 Đèn led 17](#_Toc53033462)

[Hình 2.8 Module cảm biến ánh sáng 18](#_Toc53033463)

[Hình 2.9 Giao diện Code Arduino IDE 20](#_Toc53033464)

[Hình 2.10 Các phiên bản JRE 20](#_Toc53033465)

[Hình 2.11 Tải xuống Arduino IDE 21](#_Toc53033466)

[Hình 2.12 Cài đặt Driver cho Arduino IDE 22](#_Toc53033467)

[Hình 2.13 Vùng lệnh cơ bản trong Arduino IDE 23](#_Toc53033468)

[Hình 2.14 Vùng thông báo trong Arduino IDE 24](#_Toc53033469)

[Hình 2.15 Chọn cổng COM cho arduino 25](#_Toc53033470)

[Hình 2.16 Giao diện Code Visual Studio Code 26](#_Toc53033471)

[Hình 2.17 Cài đặt Visual Studio Code 26](#_Toc53033472)

[Hình 2.18 Cài các extension cơ bản 27](#_Toc53033473)

[Hình 2.19 Giao diện màn hình đầu của Android Studio 27](#_Toc53033474)

[Hình 2.20 Một project bình thường thì có dạng như thế này 28](#_Toc53033475)

[Hình 2.21 Cài đặt Android Studio 28](#_Toc53033476)

[Hình 2.22 Cài máy ảo trong Android Studio 30](#_Toc53033477)

[Hình 3.1 Sơ đồ khối hệ thống 31](#_Toc53033478)

[Hình 3.2 Sơ đồ nguyên lý hoạt động 32](#_Toc53033479)

[Hình 3.3 Lưu đồ lấy trạng thái của các đèn và cảm biến từ Web server 33](#_Toc53033480)

[Hình 3.4 Lưu đồ điều khiển các bóng đèn bởi Relay 35](#_Toc53033481)

[Hình 3.5 Lưu đồ điều khiển quạt 36](#_Toc53033482)

[Hình 3.6 Lưu đồ điều khiển đèn bởi cảm biến ánh sáng 37](#_Toc53033483)

[Hình 3.7 Lưu đồ điều khiển cửa 38](#_Toc53033484)

[Hình 4.1 Chương trình cho Arduino Uno R3 39](#_Toc53033485)

[Hình 4.2 Xây dựng Web Server 40](#_Toc53033486)

[Hình 4.3 Phần mềm điều khiển sử dụng Android Studio 41](#_Toc53033487)

[Hình 5.1 Mô hình nhà thông minh 42](#_Toc53033488)

[Hình 5.2 Trang chủ điều khiển thiết bị 43](#_Toc53033489)

# TỔNG QUAN

## ĐẶT VẤN ĐỀ

Thường ngày để điều khiển chiếu sáng trong nhà, bạn phải di chuyển và bật tắt lắm 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 úng 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, đãng 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.

## 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.

## 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 khối điều khiển qua relayvà khối nút nhấn thủ công và khối hoàn chỉnh
* Xây dựng WebServer
* Viết code cho Arduino Uno R3
* Xây dựng ứng dụng di đông (Android App)
* Thiết kế mô hình nhà và lắp ráp các board mạch
* Chạy thử và kiểm tra, sửa chữa lỗi

## 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

## 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.

# CƠ SỞ LÝ THUYẾT

## 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.

## GIỚI THIỆU PHẦN CỨNG

* Thiết bị đầu vào: module cảm biến ánh sánh, nguồn 5V DC và 220V AC.
* Thiết bị đầu ra: module Relay, các đèn, quạt, cửa nhà.
* Thiết bị vửa 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)

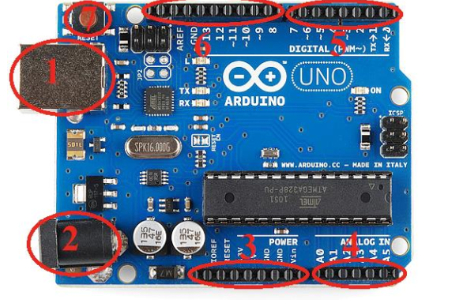
### Bộ điều khiển trung tâm Arduino Uno R3

#### Giới thiệu về Arduino Uno R3 Mạch Arduino Uno R3 | Arduinokit.vn

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.

#### Sơ đồ chân

7 6  5

1

!

2 3 4

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

### Relay

1. **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.

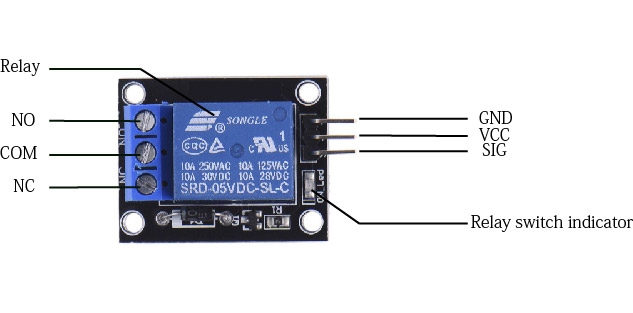
Module relay thích hợp cho các ứng dụng đóng ngắt điện thế cao AC hoặc DC, các thiết bị tiêu thụ dòng lớn, module thiết kế nhỏ gọn, có opto và transistor cách ly, kích đóng bằng mức thấp (0V) phù hợp với mọi loại MCU và thiết kế có thể sử dụng nguồn ngoài giúp cho việc sử dụng trở nên thật linh động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

1. **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

1. **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 Jupmer
   * Có đèn báo đóng ngắt trên mỗi Relay.

### Điện trở

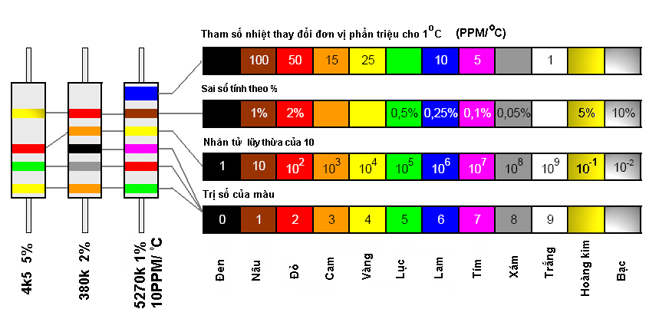
1. **Giới thiệu và tính năng**

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.



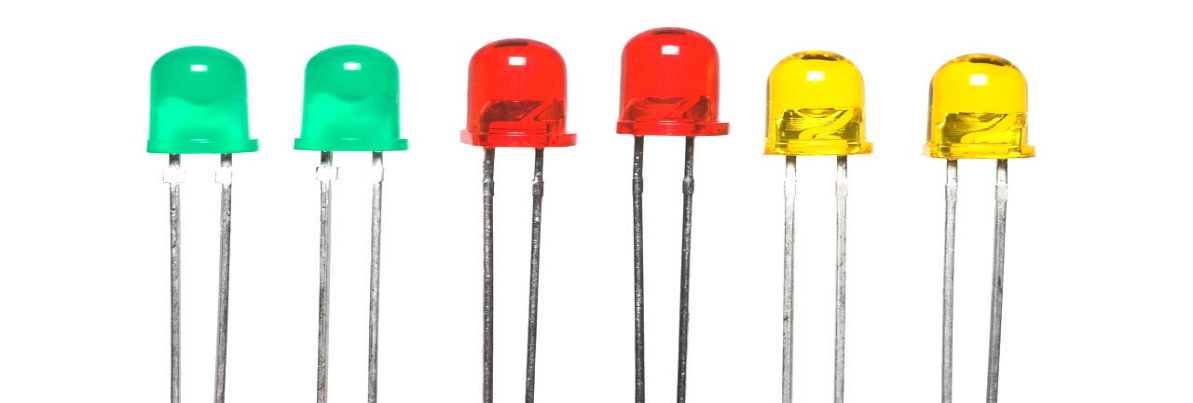
Hình 2.5 Điện trở

1. **Thông số**



Hình 2.6 Giá trị của điện trở

### Đèn led

Dùng để chiếu sáng, trong sản phẩm thực hiện dùng đèn AC 220V

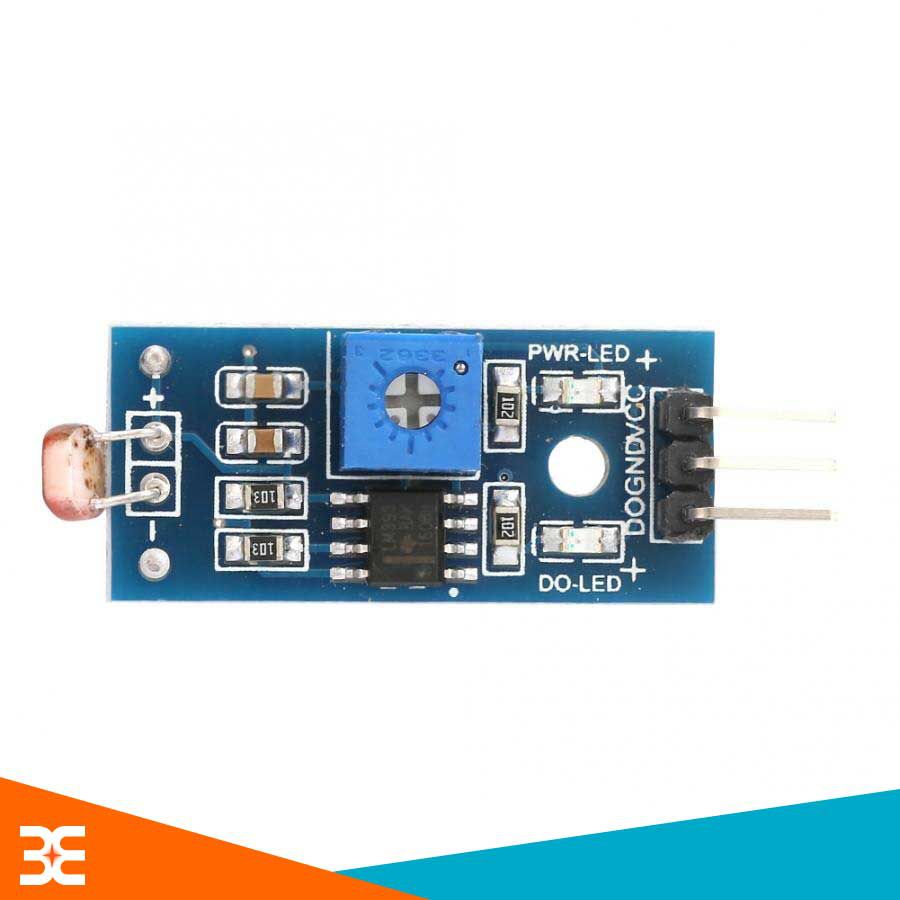
Hình 2.7 Đèn led

### Module cảm biến ánh sáng

1. **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 relaylà 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

1. **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.

### Động cơ điện Motor

1. **Giới thiệu và tính năng**

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

## GIỚI THIỆU CÔNG CỤ LẬP TRÌNH

### Arduino IDE

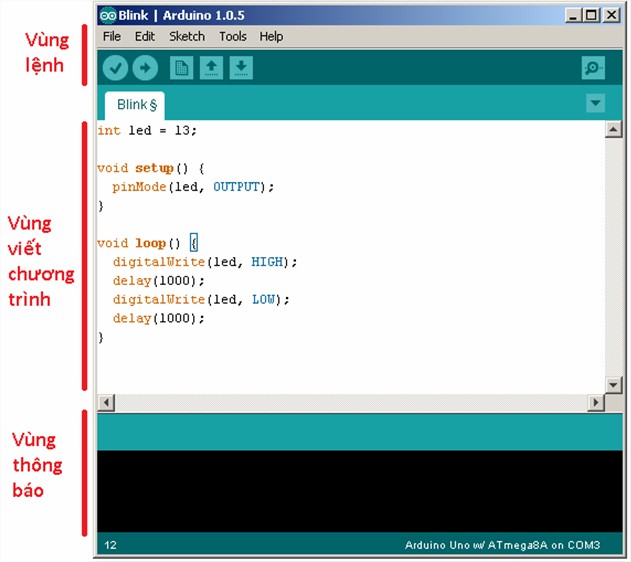
IDE viết tắt của cụm từ Integrated Development Environment là phần mềm cung cấp cho các lập trình viên một môi trường tích hợp bao gồm nhiều công cụ khác nhau như chương trình viết mã lệnh hay code editor, chương trình sửa lỗi hay debugger, chương trình mô phỏng ứng dụng khi chạy thực tế hay simulator…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

1. **Giao diện IDE**

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

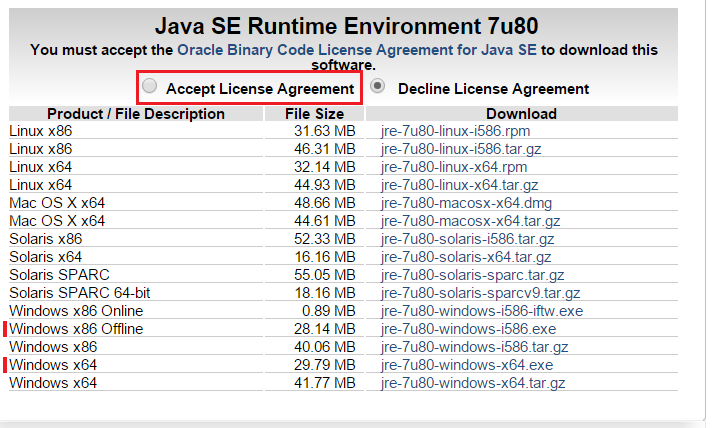
Một chương trình viết bởi Arduino IDE được gọi là sketch được lưu dưới dạng .ino.

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

1. **Cài đặt Arduino IDE**
   1. **Cài đặt Java Runtime Environment (JRE)**



Hình 2.10 Các phiên bản JRE

* 1. **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.





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

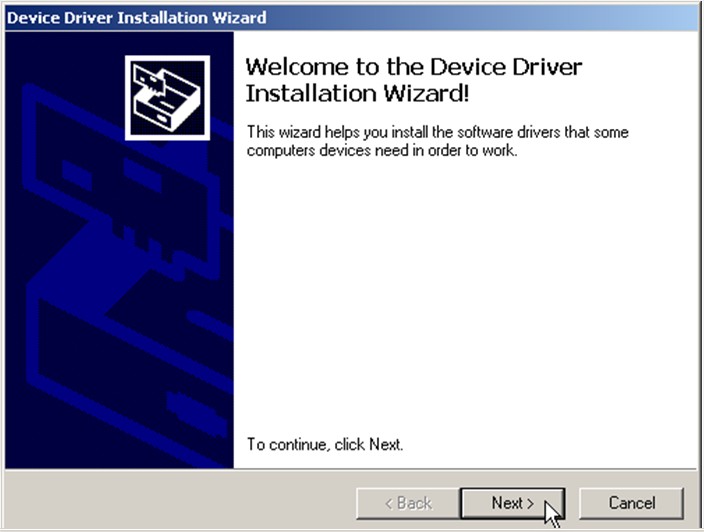
* 1. **Cài đặt Driver**

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

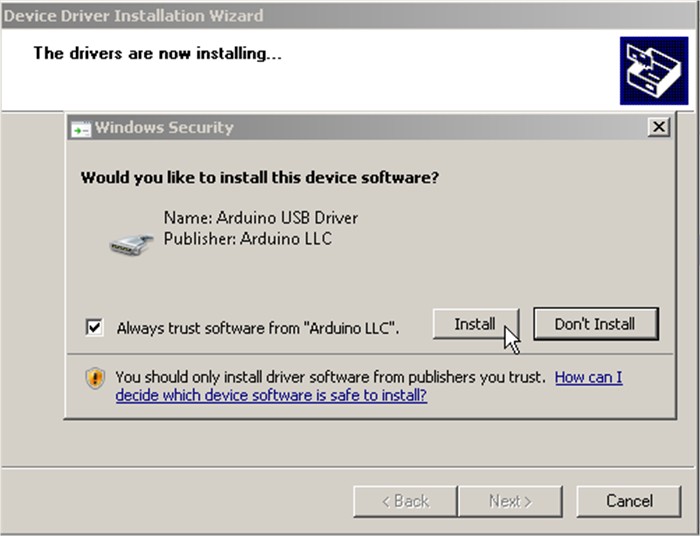
Xem hướng dẫn thực hiện tại bài viết Disabling Driver Signature on Windows 8 của SparkFun.

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.

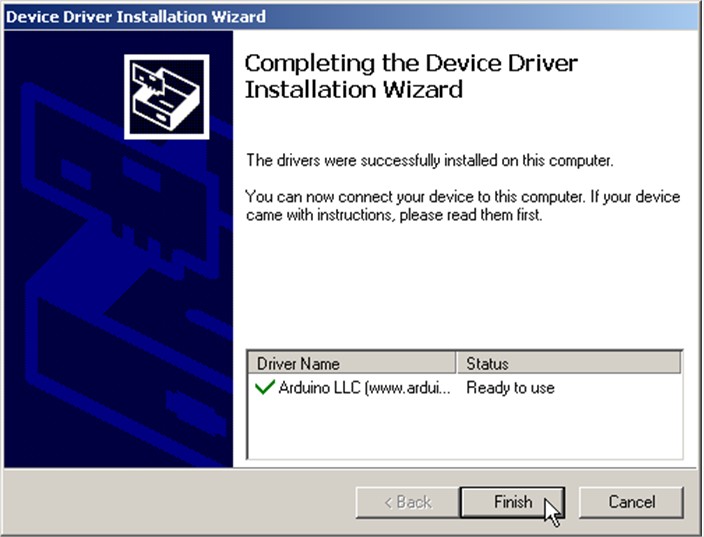


Hình 2.12 Cài đặt Driver cho Arduino IDE

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

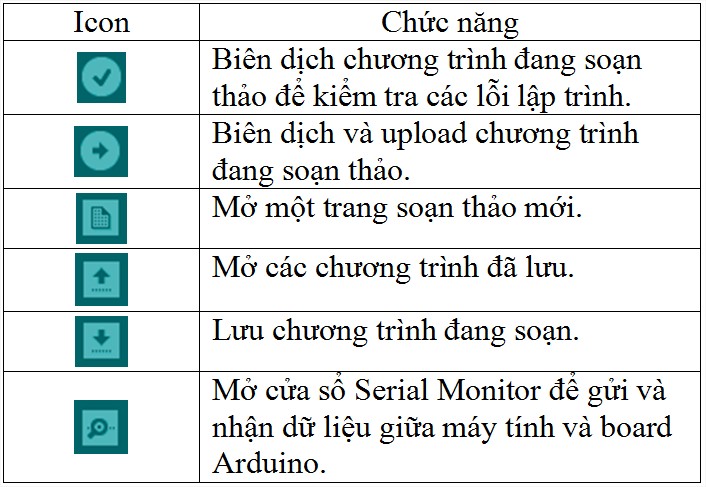


Đợi khoảng 10 giây trong lúc quá trình cài đặt diễn ra …



1. **Các chức năng cơ bản trong Arduino IDE**
   1. **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:



Hình 2.13 Vùng lệnh cơ bản trong Arduino IDE

* 1. **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.

* 1. **Vùng thông báo**

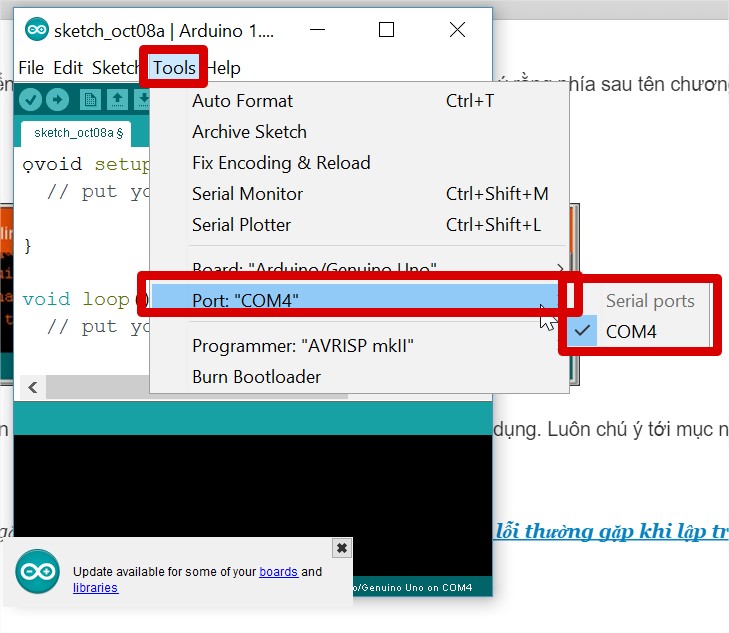
Những thông báo từ IDE sẽ được hiển thị tại đây. Để ý rằng góc dưới cùng bên phải hiển thị loại board Arduino và cổng COM được sử dụng. Luôn chú ý tới mục này bởi nếu chọn sai loại board hoặc cổng COM, bạn sẽ không thể upload được code của mình.



Hình 2.14 Vùng thông báo trong Arduino IDE

1. **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 đươc nhé.



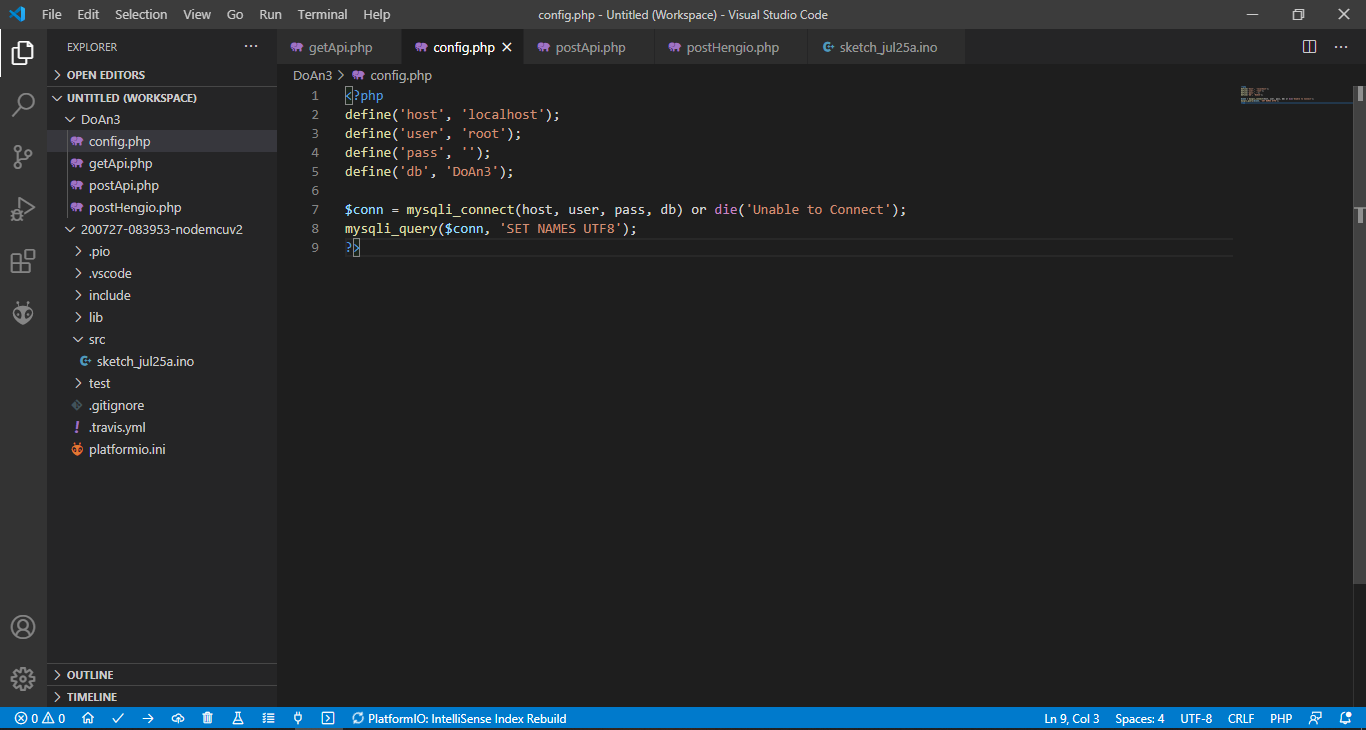
Hình 2.15 Chọn cổng COM cho arduino

### 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.

1. **Giao diện**

**Lưu ý**: Visual Studio Code không giống Visual Studio IDE

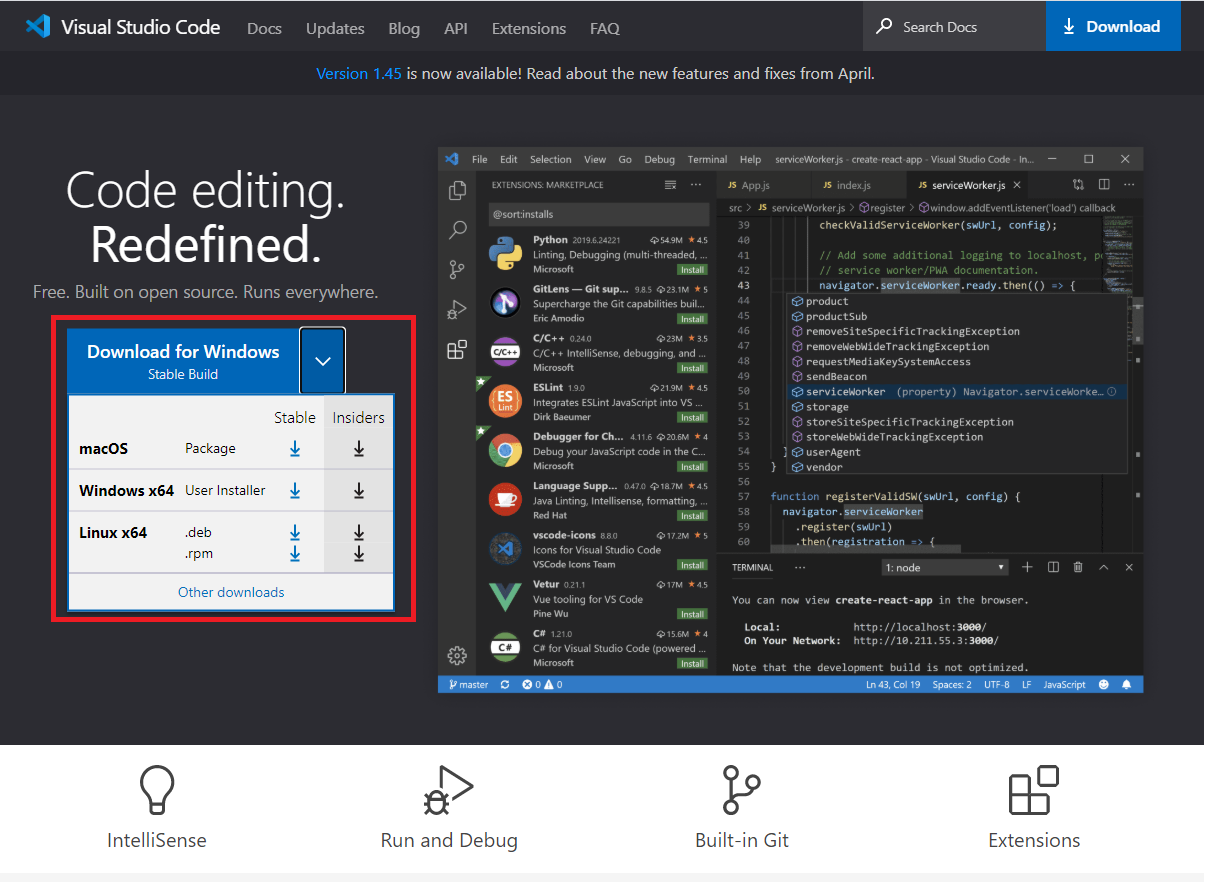
****

Hình 2.16 Giao diện Code Visual Studio Code

1. **Cài đặt**
   1. **Tải và cài đặt vscode**

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

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

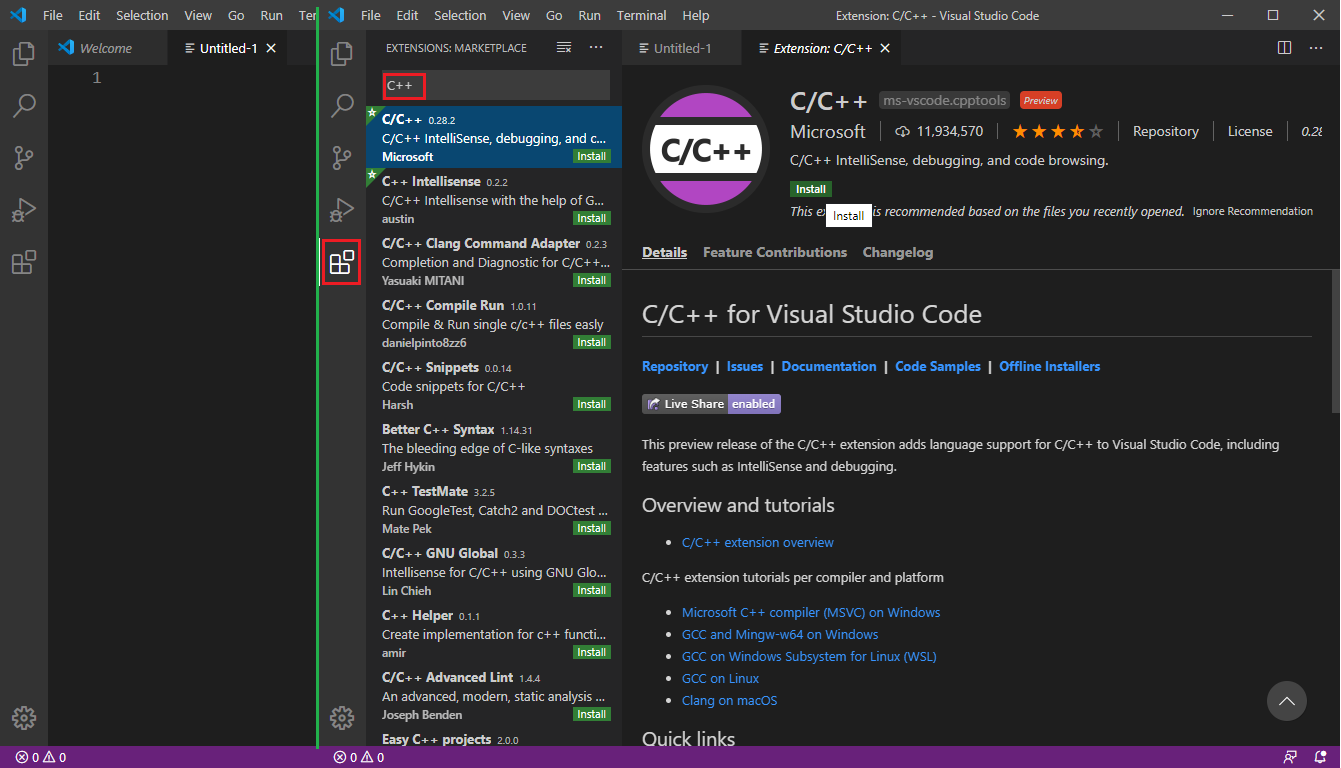


Hình 2.17 Cài đặt Visual Studio Code

* 1. **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.



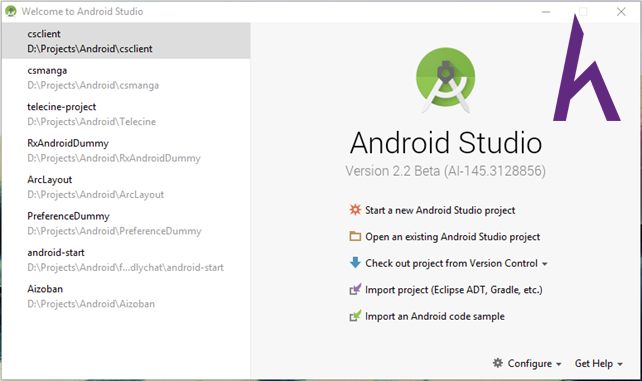
Hình 2.18 Cài các extension cơ bản

### Android Studio

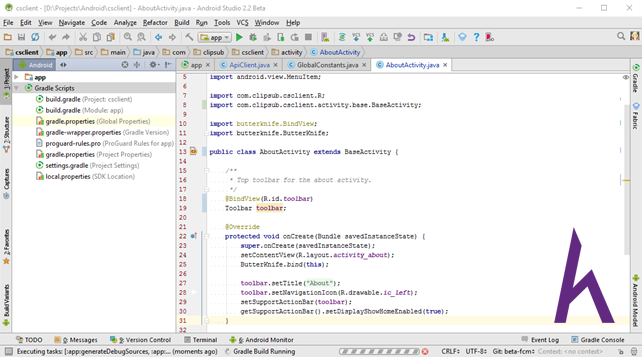
1. **Giới thiệu**

**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



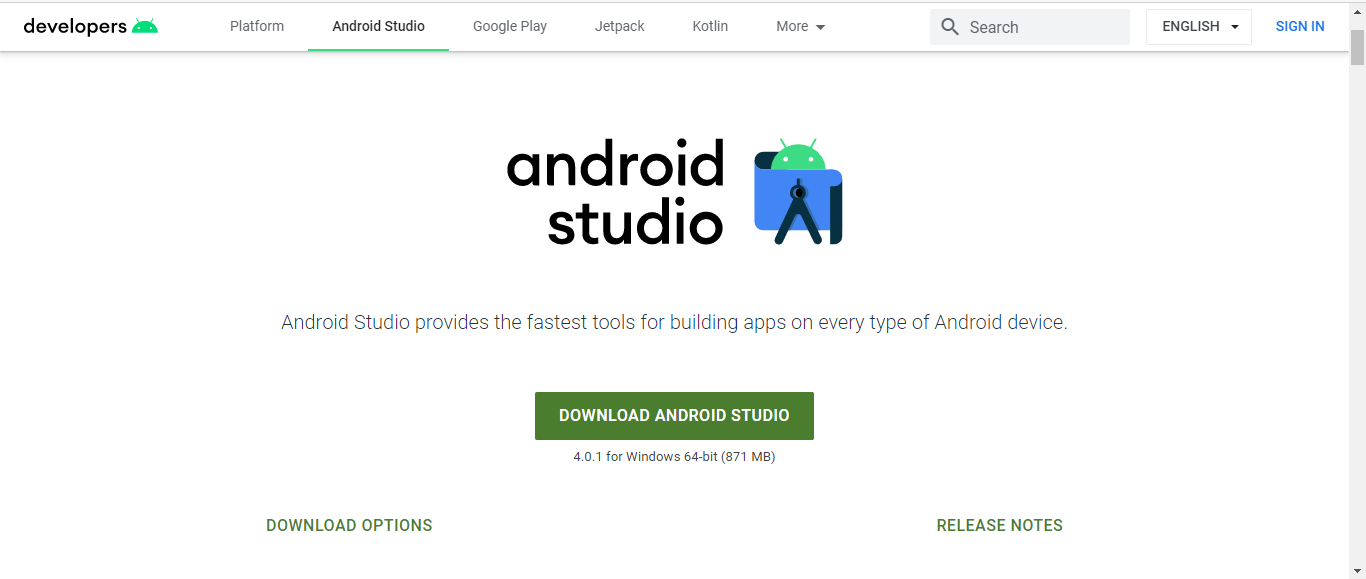
Hình 2.20 Một project bình thường thì có dạng như thế này

1. **Cài đặt**

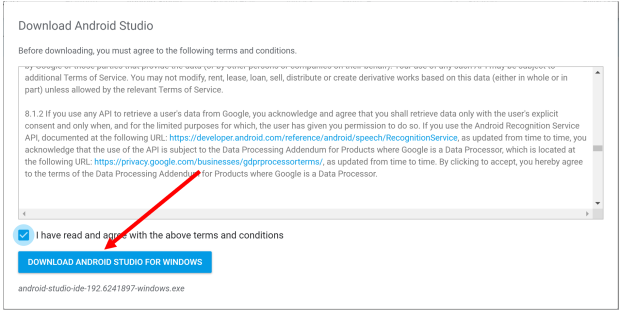
Bước 1: Chúng ta sẽ cần có JDK (Java Development Kit) trước khi làm bất cứ việc gì: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Bước 2: Chuẩn bị Android Studio và Android SDK.

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



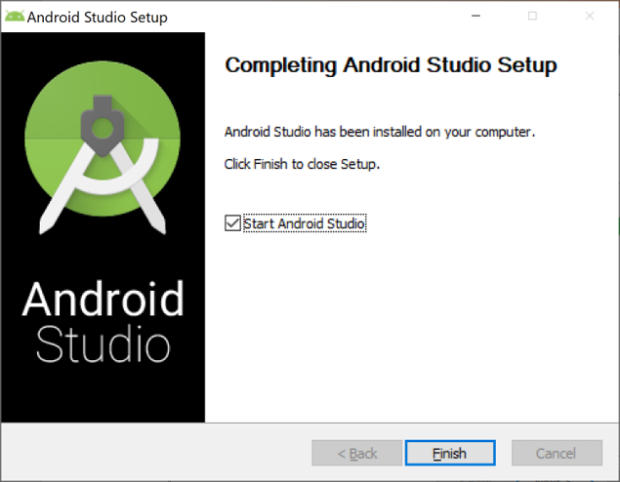
Hình 2.21 Cài đặt Android Studio



Hình 2.22 Cài đặt Android Studio

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

1. **Tạo Android Virtual Device**

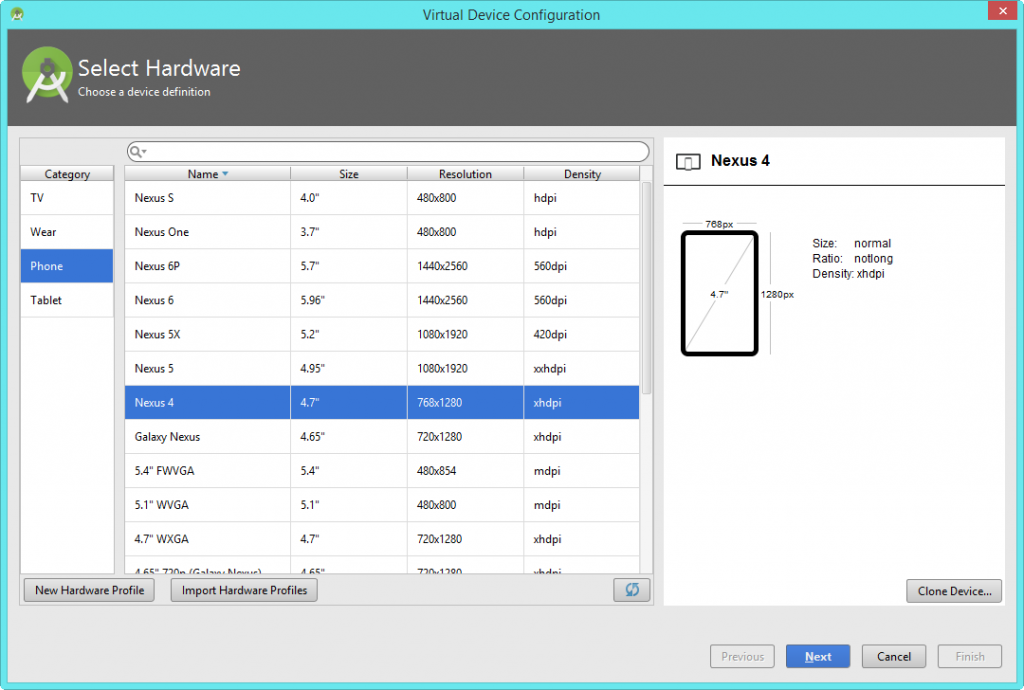
Nhấp chuột vào biểu tượng AVD Manager trong Android Studio

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

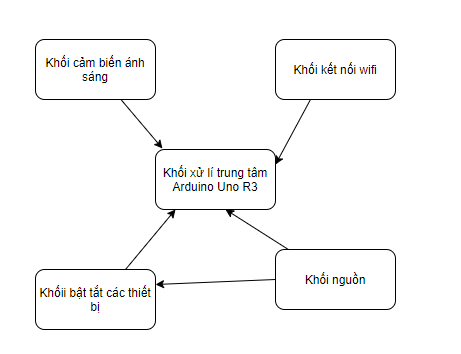
# PHÂN TÍCH & THIẾT KẾ HỆ THỐNG

## 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.

## THIẾT KẾ SƠ ĐỒ KHỐI

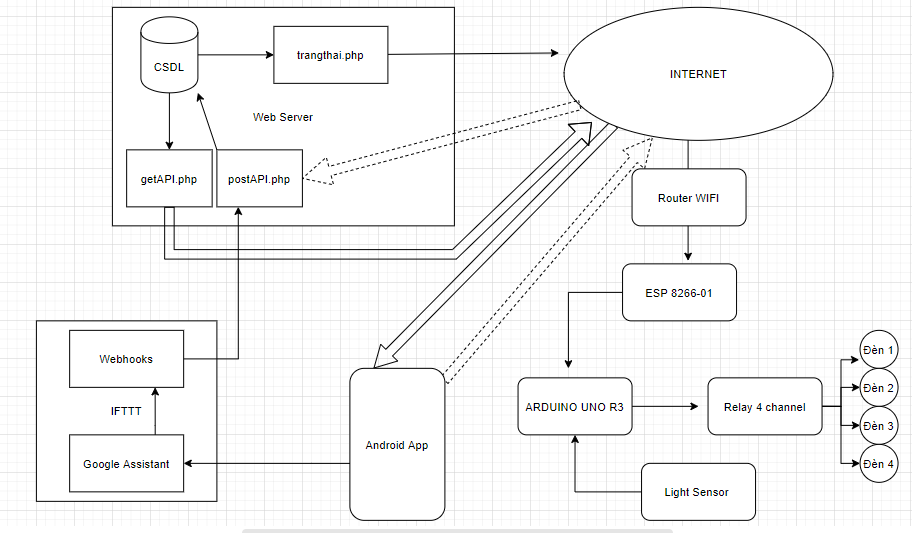
Với các yêu cầu đưa ra, nhóm đã dựng lên sơ đồ khối như sau:



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 biết bị tương úng.
  + 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.

## NGUYÊN LÝ HOẠT ĐỘNG

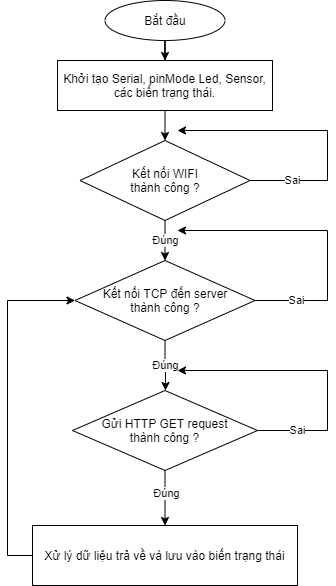


Hình 3.2 Sơ đồ nguyên lý hoạt động

* Nguyên lý hoạt động của hệ thống
  + Web server đóng vai trò quan trọng trong việc điều khiển tất cả thiết bị chiếu sang, 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.
  + Hổ 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 úng.
  + 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 theo yêu cầu người dùng.

## LƯU ĐỒ THUẬT TOÁN

### 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

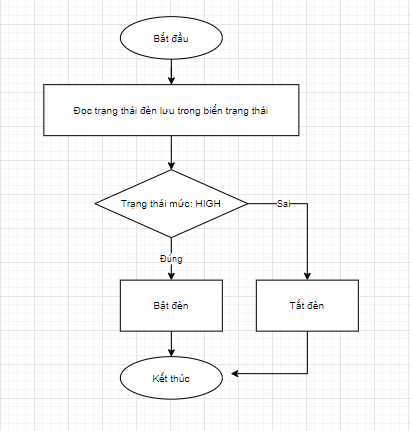
B3: Gửi yêu cầu kết nối tới WIFI, nếu thất bại quay lại B3

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

### Lưu đồ điều khiển các đèn



Hình 3.4 Lưu đồ điều khiển các bóng đèn bởi Relay

Giải thích lưu đồ thuật toán:

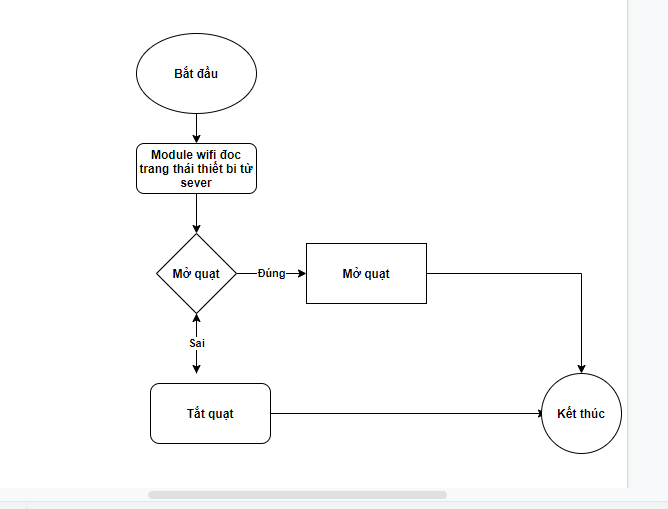
B1: Bắt đầu

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.

### Lưu đồ điều khiển quạt



Hình 3.5 Lưu đồ điều khiển quạt

Giải thích lưu đồ thuật toán:

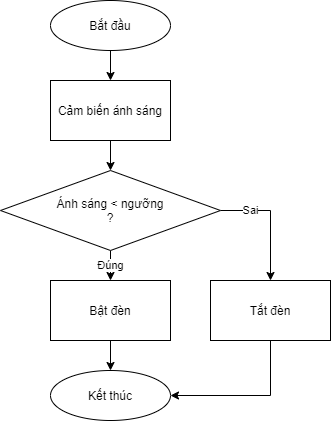
B1: Bắt đầu

B2: Đọc trạng thái của quạt

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

B4: Kết thúc.

### 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:

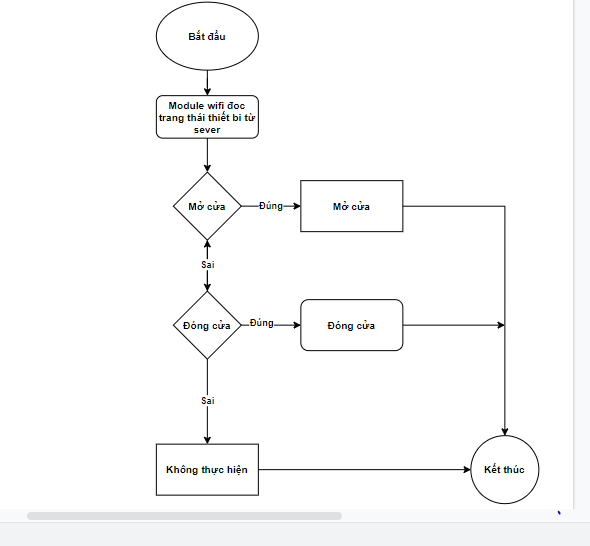
B1: Bắt đầu

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.

### 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.

# TRIỂN KHAI THỰC HIỆN

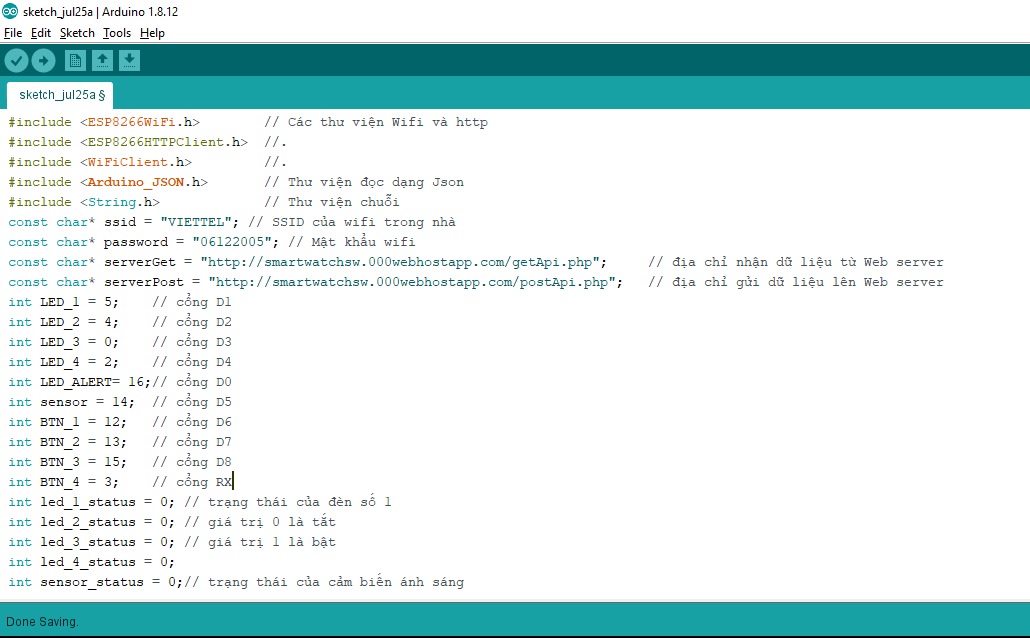
## LẮP RÁP PHẦN CỨNG

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.

## THIẾT KẾ PHẦN MỀM

### Chương trình cho Arduino Uno R3

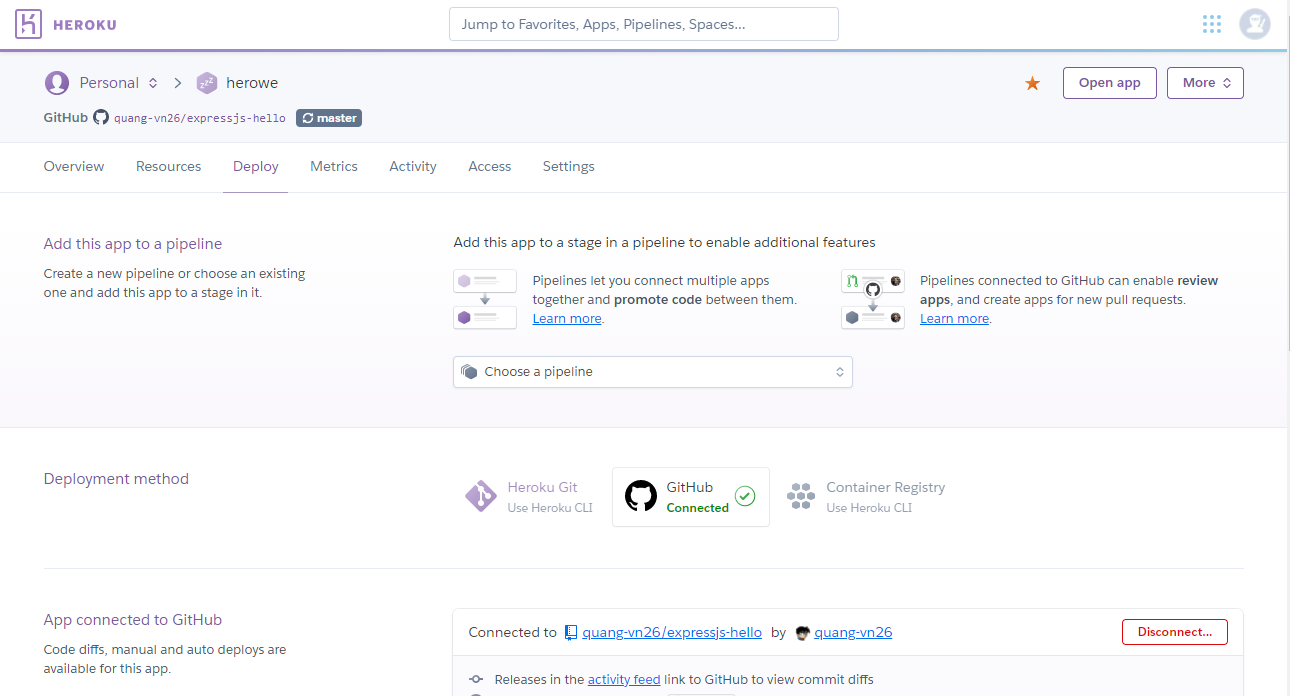
Chương trình được viết bằng ngôn ngữ lập trình C, trên môi trường lập trình Arduino IDE.



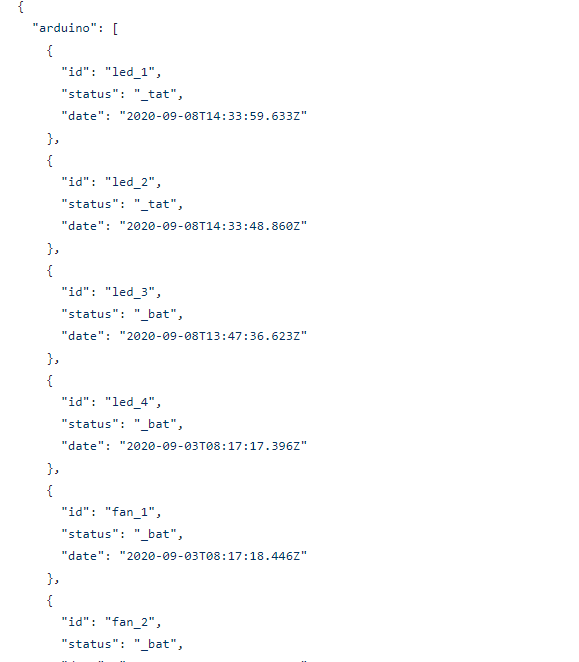
Hình 4.1 Chương trình cho Arduino Uno R3

### Xây dựng Web Server

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



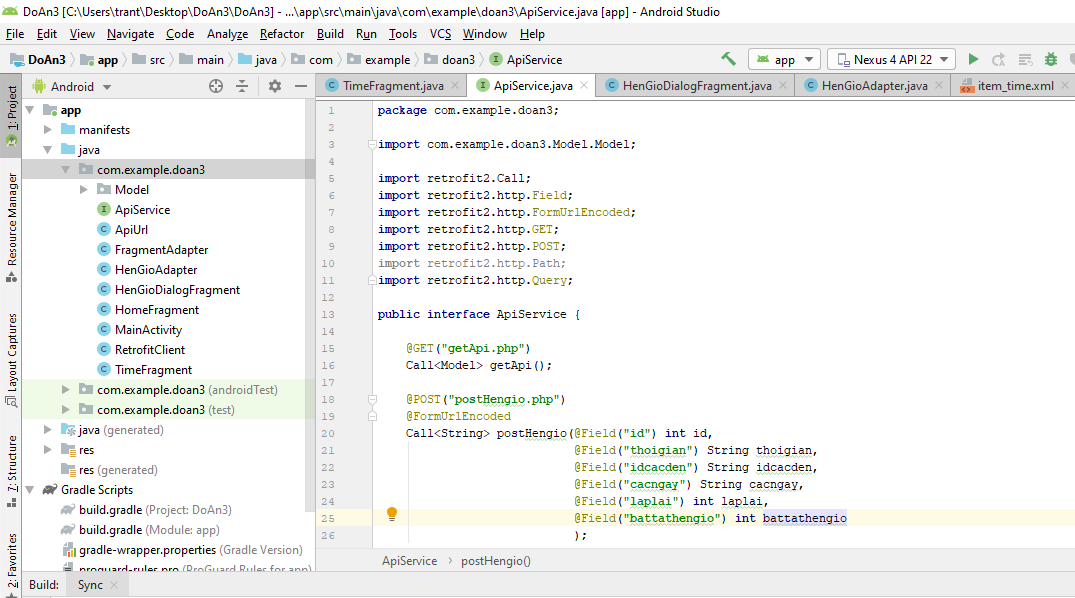
Hình 4.2 Xây dựng Web Server



Hình 4.3 Cài máy ảo trong Android Studio

### 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

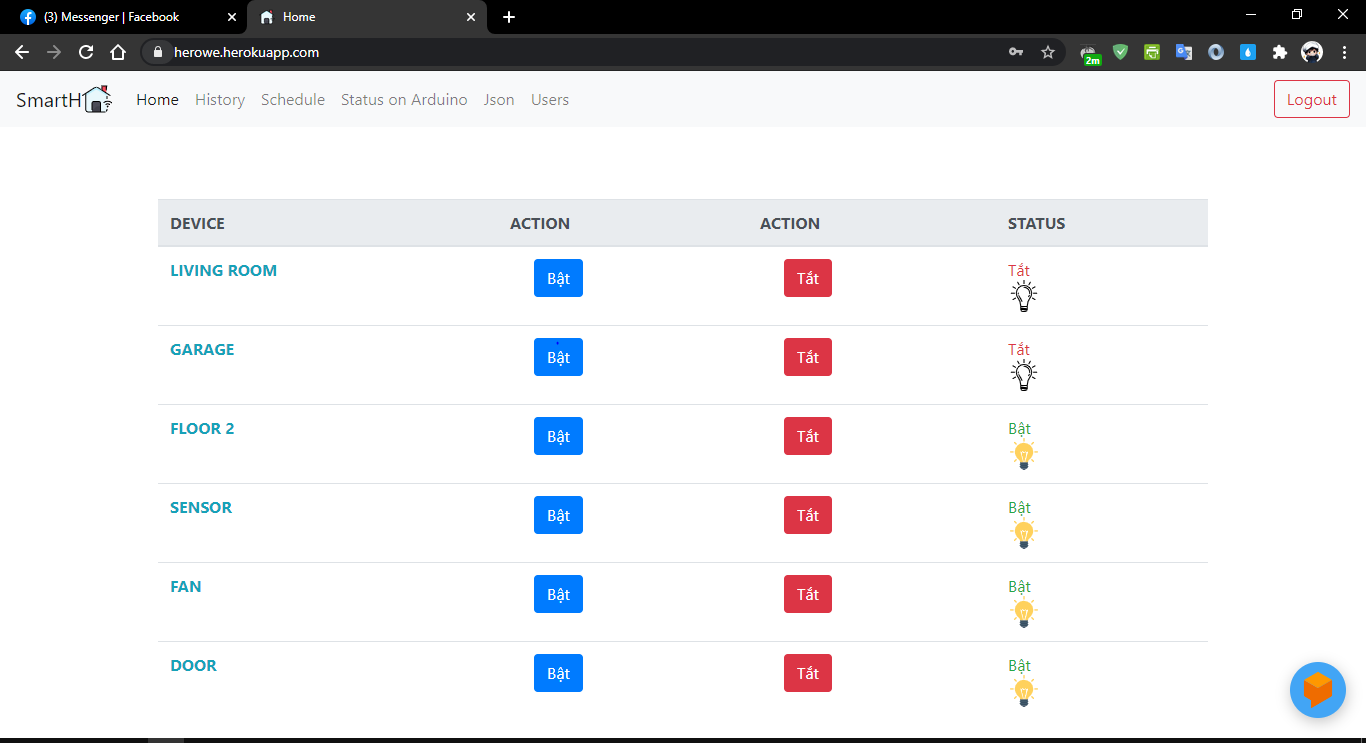
# KẾT QUẢ ĐẠT ĐƯỢC

## MÔ HÌNH NHÀ THÔNG MINH

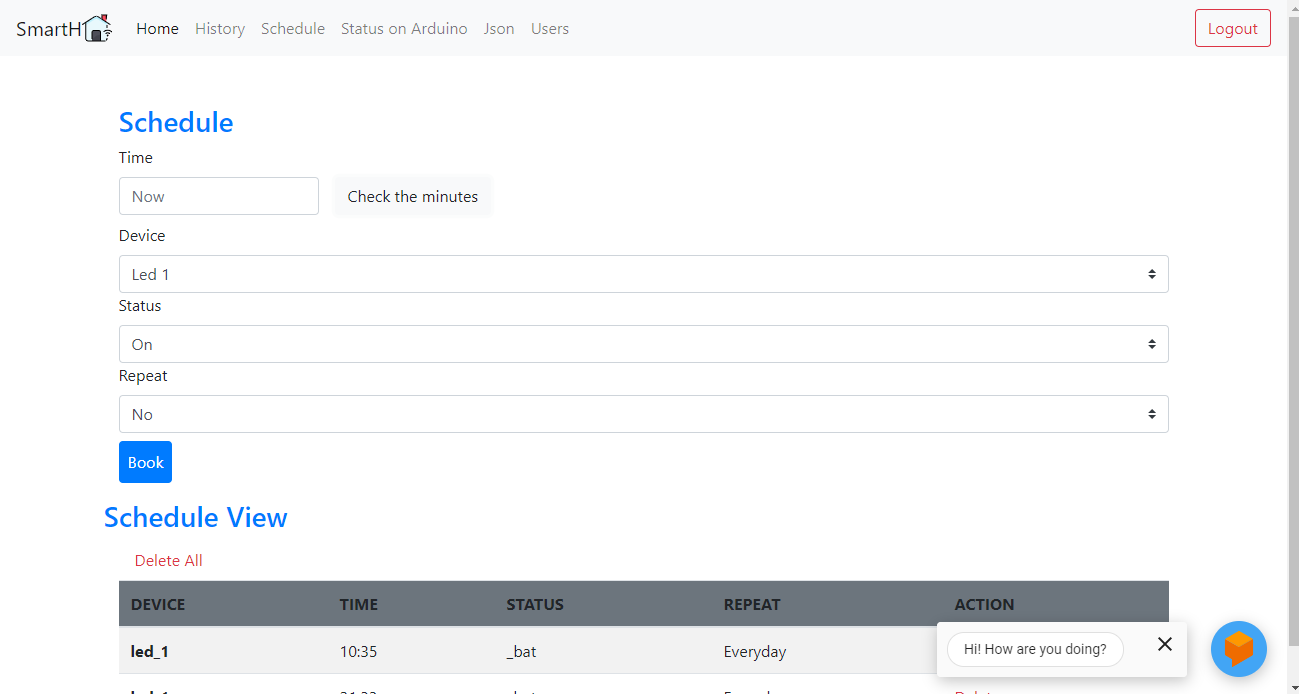


Hình 5.1 Mô hình nhà thông minh

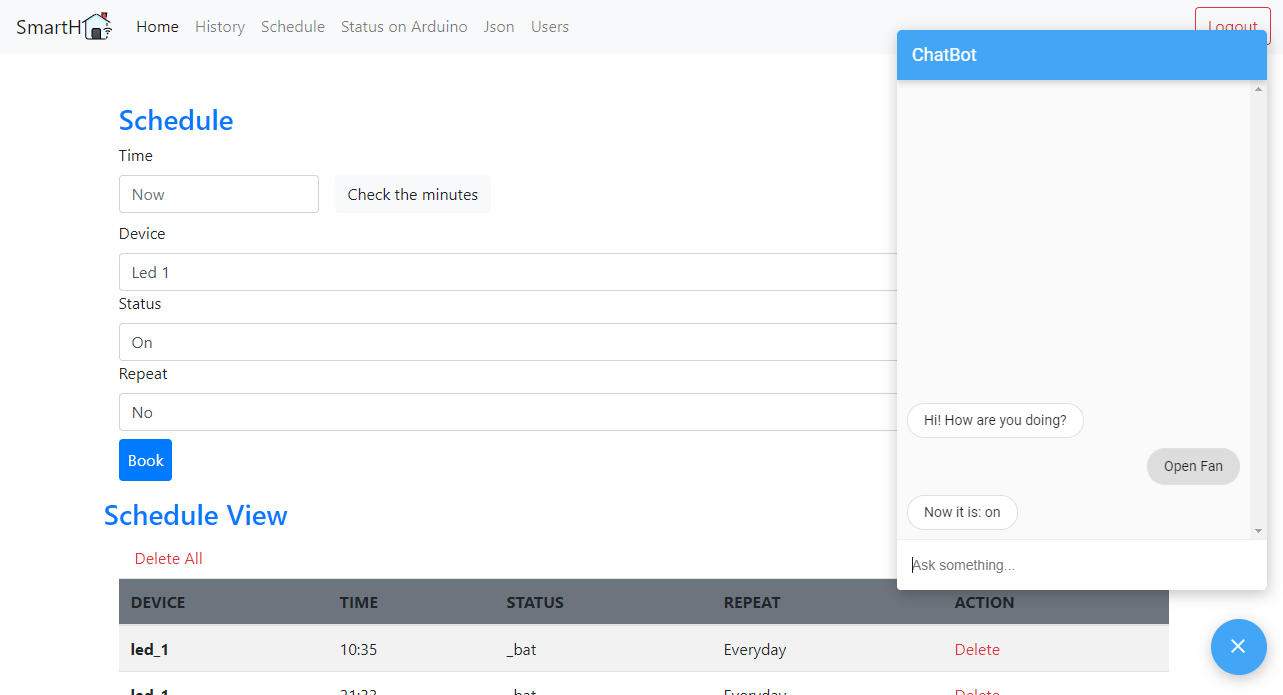
## WEB SERVER



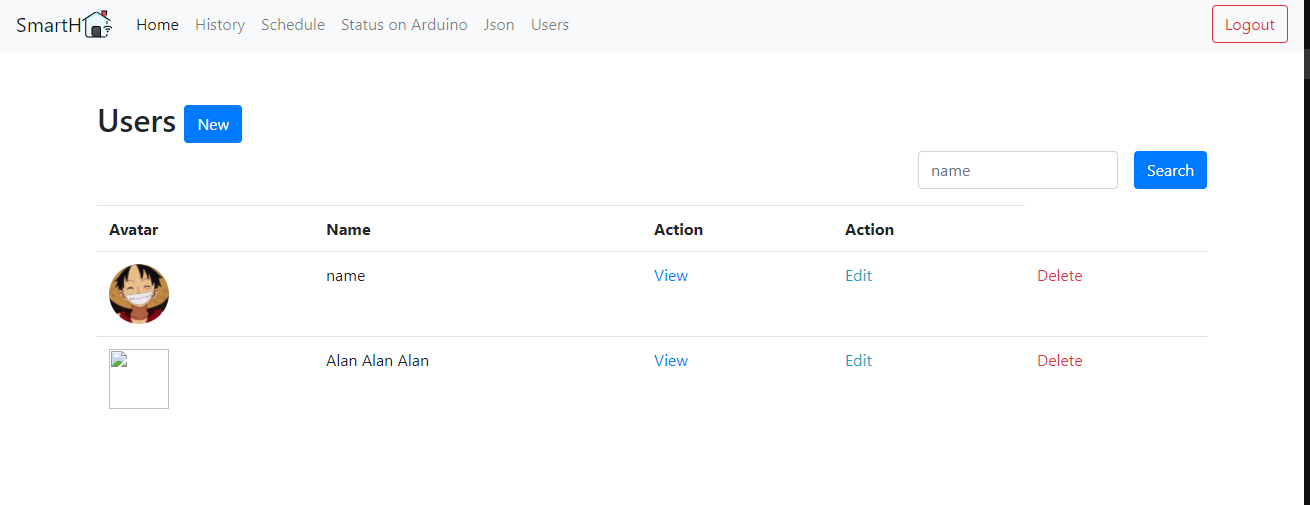
Hình 5.2 Trang chủ điều khiển thiết bị



Hình 5.3 Trang đặt lịch

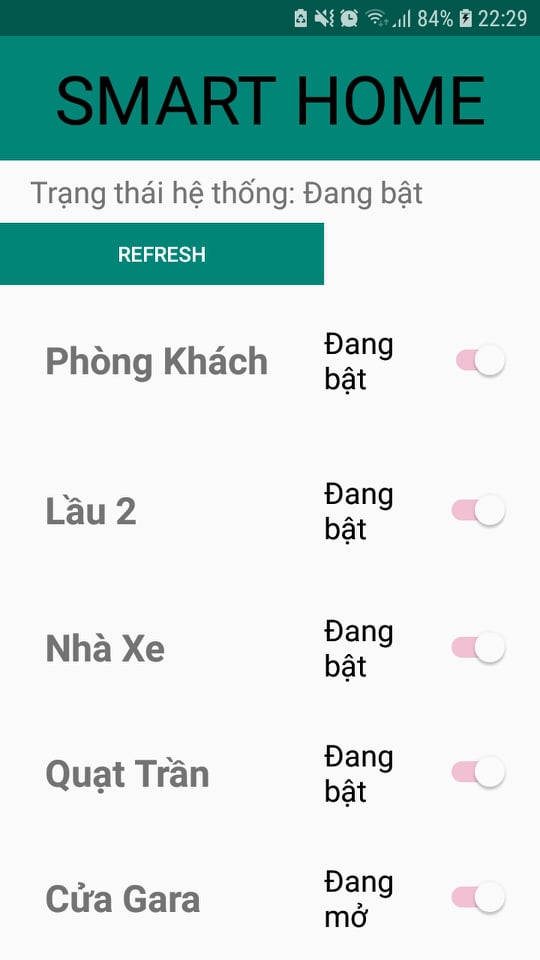


Hình 5.4 Trang chatbot



Hình 5.4 Trang người dùng

## ỨNG DỤNG DI ĐỘNG



Hình 5.5 Ứng dung di động điều khiền thiết bị chiếu sáng

## 

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## KẾT LUẬN

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

## 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.

## 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/](https://randomnerdtutorials.com/esp8266-nodemcu-http-get-post-arduino/)