**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA ĐIỆN – ĐIỆN TỬ

**BỘ MÔN ĐIỆN TỬ**

---------------o0o---------------

****

**ĐỒ ÁN MÔN HỌC**

**ĐIỀU KHIỂN THIẾT BỊ VÀ THU THẬP DỮ LIỆU ĐƠN GIẢN**

**TRONG BỐI CẢNH TRỒNG CÂY TỰ ĐỘNG HÓA**

**GVHD: ThS Trần Hoàng Quân**

**SVTH: Trần Đình Huy 1611332**

**Nguyễn Công Huy 1611275**

**TP. HỒ CHÍ MINH, THÁNG 6 NĂM 2019**

***LỜI CẢM ƠN***

Được sự phân công của nhà trường, khoa Điện - Điện tử và sự đồng ý của giáo viên hướng dẫn ThS. Trần Hoàng Quân, chúng em đã thực hiện đồ án chuyên ngành điện tử với đề tài “*Điều khiển thiết bị và thu thập dữ liệu đơn giản trong bối cảnh trồng cây tự động hóa”* . Qua trang viết này, em xin gửi lời cảm ơn tới những người đã giúp đỡ em trong thời gian học tập và nghiên cứu vừa qua.

Trước hết, chúng em xin gửi lời cảm ơn chân thành đến ThS. Trần Hoàng Quân. Thầy là người trực tiếp hướng dẫn và chỉ bảo tận tình trong quá trình nghiên cứu, tìm hiểu tài liệu, phân tích và xử lý vấn đề của đồ án. Nhớ thầy mà chúng em có thể hoàn thành đồ án môn học.

Ngoài ra, chúng em xin chân thành cảm ơn gia đình, bạn bè đã luôn động viên và hỗ trợ em trong suốt quá trình nghiên cứu và hoàn thành đồ án này.

Cuối cùng, chúng em xin chúc quý thầy, cô dồi dào sức khỏe và thành công trong sự nghiệp cao quý.

*Tp. Hồ Chí Minh, ngày 13 tháng 06 năm 2019 .*

**Sinh viên**

**TÓM TẮT ĐỒ ÁN**

Trong đồ án này, chúng em đề xuất mô hình sử dụng smart phone tự động giám sát và điều khiển thiệt bị bằng nodeMCU8266. Phần mềm trên smart phone giám sát điều kiện môi trường thông qua hệ thống cảm biến và điều khiển điều kiện môi trường bằng hệ thống cơ cấu chấp hành. Hệ thống hoạt động ở chế độ tự động và bằng tay thông qua giao diện màn hình smart phone. NodeMCU8266 nhận thông tin môi trường từ hệ thống cảm biến, xử lý và tác động lên hệ thống cơ cấu chấp hành. NodeMcu8266 gửi dữ liệu lên server là Firebase và nhận tín hiệu điều khiển từ smart phone qua Firebase. Phần mềm trên smart phone cập nhật thông số hệ thống cảm biến và trạng thái hệ thống cơ cấu chấp hành thời gian thực từ Firebase, đồng thời điều khiển hệ thống cơ cấu chấp hành theo chế độ bằng tay và tự động dựa trên cài đặt thông số môi trường.

**MỤC LỤC**

[**GIỚI THIỆU**](#_30j0zll)10

[Tổng quan](#_1fob9te) 10

[Nhiệm vụ đề tài](#_3znysh7) 11

[Phân chia công việc trong nhóm](#_2et92p0) 11

[**LÝ THUYẾT**](#_tyjcwt)12

[HTTP](#_n05kl0fv4pwx) 12

[**THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG**](#_3dy6vkm)13

[Yêu cầu thiết kế](#_mhv8i9okupog) 13

[Phân tích thiết kế](#_rasthpwnlgkr) 13

[MCU dùng để lập trình](#_cg703o2lkjl6) 14

[Cảm biến ánh sáng](#_7g3g4tadv2dq) 14

[Sơ đồ khối tổng quát](#_95mtnadfz5zt) 14

[Power supply](#_otvlgtasbs8g) 15

[Cảm biến nhiệt độ và độ ẩm](#_5bayuep6l94) 15

[Cảm biến độ ẩm đất](#_yj25ht6bfva4) 15

[Cảm biến ánh sáng](#_upeto52sfare) 16

[MCU](#_6mkls02ij9yn) 16

[Bơm](#_aatspqfc3hy) 16

[LED RGB](#_33xtulogw234) 16

[Firebase](#_fejayxz3p0fg) 16

[Sơ đồ khối chi tiết](#_4lnmzliimflm) 16

[Khối cấp nguồn](#_4krdsv9w0olq) 16

[Cảm biến nhiệt độ và độ ẩm](#_9bjwbjiz8n8i) 17

[Cảm biến độ ẩm đất](#_nvg1jhmy1o9q) 18

[Cảm biến ánh sáng](#_oy8uyaatwu9z) 18

[Bơm](#_nj4d0124kfp7) 19

[LED RGB](#_8xi3drq0ga5j) 19

[Sơ đồ mạch chi tiết](#_deczr8id1v5m) 20

[Khối nguồn](#_qi6ts5l3zioa) 20

[Cảm biến nhiệt độ & độ ẩm](#_2a3k1lufuokj) 21

[Cảm biến độ ẩm đất](#_oem5ucxisir1) 22

[Cảm biến ánh sáng](#_94txjha3468w) 23

[Khối điều khiển bơm](#_w6uqx7i8xw8o) 24

[Khối điều khiển LED RGB](#_g0x47r2in5b1) 25

[**THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM**](#_1t3h5sf)25

[Google Firebase](#_vjfk5ynartsy) 25

[MIT APP INVENTOR](#_v1yxb31nh3ue) 26

[Lưu đồ giải thuật](#_p5tr0ojbyiqr) 27

[**KẾT QUẢ THỰC HIỆN**](#_4d34og8)31

[Phần cứng](#_etm3lrwthhjv) 31

[3D](#_4m86qnpbnrda) 31

[Layout](#_mmi1n7lafg50) 32

[Mạch thực tế](#_43oe8jm4tf58) 33

[Phần mềm](#_d9x8c6fc4qcc) 34

[MIT APP INVENTOR](#_cayft5jvfl13) 34

[Google Firebase](#_xivbc57c49gs) 36

[Giá trị cảm biến và thời gian thực](#_b4ass0jhhran) 36

[Giá trị cảm biến](#_u1ufj57rm6ae) 36

[Thời gian thực](#_nvg4h5blkctn) 37

[ĐÁNH GIÁ TỔNG QUAN](#_zb57dy2k23m) 37

[Phần cứng](#_eajww8cjyn7l) 37

[Phần mềm](#_to2f4wqa0o2m) 37

[Kết quả làm việc nhóm](#_k3cimlybuuy0) 38

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN](#_26in1rg) 38

[Kết luận](#_lnxbz9) 38

[Hướng phát triển](#_35nkun2) 38

[TÀI LIỆU THAM KHẢO](#_1ksv4uv) 39

[PHỤ LỤC](#_44sinio) 39

**DANH SÁCH HÌNH MINH HỌA**

[Hình 1: Sơ đồ khối tổng quát 15](#_Toc11278073)

[Hình 2: Sơ đồ khối cấp nguồn 16](#_Toc11278074)

[Hình 3: Sơ đồ khối cảm biến nhiệt độ và độ ẩm 17](#_Toc11278075)

[Hình 4: Sơ đồ khối cảm biến độ ẩm đất 18](#_Toc11278076)

[Hình 5: Sơ đồ khối cảm biến ánh sáng 18](#_Toc11278077)

[Hình 6: Sơ đồ khối điều khiển bơm 19](#_Toc11278078)

[Hình 7: Sơ đồ khối điều khiển LED RGB 19](#_Toc11278079)

[Hình 8: Sơ đồ mạch chi tiết 20](#_Toc11278080)

[Hình 9: Sơ đồ mạch khối nguồn 20](#_Toc11278081)

[Hình 10: Sơ đồ kết nối DHT11 trên mạch 21](#_Toc11278082)

[Hình 11: Sơ đồ kết nối cảm biến đo độ ẩm trên mạch 22](#_Toc11278083)

[Hình 12: Sơ đồ kết nối cảm biến đo ánh sáng trên mạch 23](#_Toc11278084)

[Hình 13: Sơ đồ kết nối khối điều khiển bơm trên mạch 24](#_Toc11278085)

[Hình 14 Sơ đồ mạch điều khiển đèn RGB 25](#_Toc11278086)

[Hình 15: Lưu đồ giải thuật tổng quát. 27](#_Toc11278087)

[Hình 16: Login Mode 28](#_Toc11278088)

[Hình 17: Auto Mode 28](#_Toc11278089)

[Hình 18: Manual Mode 29](#_Toc11278090)

[Hình 19: Schedule Mode 29](#_Toc11278091)

[Hình 20: Tương tác giữa App và Firebase 30](#_Toc11278092)

[Hình 21: Hình ảnh 3D của mạch trên Altium 31](#_Toc11278093)

[Hình 22: Hình ảnh layout của mạch trên Altium 32](#_Toc11278094)

[Hình 23: Hình ảnh mạch thực tế 33](#_Toc11278095)

[Hình 24: Hình ảnh app 34](#_Toc11278096)

[Hình 25: Giao diện kéo thả 35](#_Toc11278097)

[Hình 26: Giao diện trên điện thoại 35](#_Toc11278098)

[Hình 27: Giao diện firebase. 36](#_Toc11278099)

[Hình 28: Giá trị cảm biến 36](#_Toc11278100)

[Hình 29: Thời gian thực 37](#_Toc11278101)

**DANH SÁCH BẢNG SỐ LIỆU**

[Bảng 1: Phân chia công việc trong nhóm 11](#_Toc11278306)

[Bảng 2: Lựa chọn MCU 14](#_Toc11278307)

[Bảng 3: Lựa chọn cảm biến ánh sáng 14](#_Toc11278308)

# GIỚI THIỆU

## Tổng quan

Ngày nay, chúng ta được chứng kiến vô số những thành tựu khoa học vô cùng ấn tượng trong khắp các lĩnh vực của đời sống. Và trong nông nghiệp, con người ngày càng trở nên nhàn hạ hơn, năng suất ngày càng được cải thiện nhờ những cỗ máy hiện đại và không thể không nhắc đến những mô hình canh tác nông nghiệp thông minh. Trong các mô hình thông minh này thì hệ thống tự động hóa chính là điểm mấu chốt, là cốt lõi tạo nên những giá trị to lớn.

So với ngành nông nghiệp tại những nước phát triển trên thế giới thì Việt Nam vẫn còn phải nỗ lực rất lớn để có thể theo kịp, tuy nhiên cho tới nay ngành nông nghiệp nước nhà cũng đã có được những bước phát triển rất đáng ghi nhận. Cụ thể đó chính là việc ứng dụng ngày càng rộng rãi các mô hình canh tác thông minh có tính tự động hóa cao vào trồng trọt.

Nhờ có ứng dụng hệ thống tự động hóa trong các nhà trồng thông minh mà các bà con nông dân có thể giám sát được các thông số về dinh dưỡng như: độ dẫn điện, độ pH, tỉ lệ các chất, lượng nước tưới cho cây trồng trong ngày,.. Không chỉ vậy hệ thống này còn có tính năng giúp giám sát các thông số vi khí hậu trong các nhà kính trồng cây như nhiệt độ, độ ẩm không khí, cường độ ánh sáng, độ ẩm của đất,.. Và thêm vào đó là các yếu tố liên quan đến thời tiết như: tốc độ gió, hướng gió, lượng mưa,..

Dựa vào sự phân tích các yếu tố tác động đó, hệ thống sẽ tự động điều chỉnh tăng hoặc giảm các thông số về dinh dưỡng cho cây trồng. Nhờ vậy chắc chắn bà con sẽ tiết kiệm được rất nhiều thời gian, công sức, tiền bạc và các khoản đầu tư cũng trở nên hiệu quả hơn, đem lại năng suất cây trồng cao hơn, chất lượng tốt hơn.

Như vậy chúng ta hoàn toàn có thể thấy rõ rằng hệ thống tự động trong các mô hình trồng trọt thông minh đã đem lại cho bà con nông dân nói riêng và nền nông nghiệp Việt Nam nói chung những bước đột phá vô cùng đáng ghi nhận. Đây quả thực là một ý tưởng canh tác nông nghiệp cần phải được khuyến khích ứng dụng rộng rãi hơn nữa trong các hộ gia đình nông nghiệp Việt Nam.

## Nhiệm vụ đề tài

Từ những nhận định trên, nhóm chúng em xin đề xuất xây dựng một hệ thống điều khiển thiết bị và thu thập dữ liệu đơn giản trong bối cảnh trồng cây tự động hóa.

Trong đồ án này, chúng em tiến hành nghiên cứu những nội dung sau:

**Nội dung 1:** Tìm hiểu cấu trúc, ngôn ngữ lập trình cho module ESP 8266

**Nội dung 2:** Tìm hiểu về module cảm biến ánh sáng (BH1750), module đo nhiệt độ & độ ẩm (DHT11), cảm biến độ ẩm đất.

**Nội dung 3:** Tìm hiểu về MIT inventor (trang web tạo app đơn giản cho android)

**Nội dung 4:** Tìm hiểu về firebase (Nơi lưu trữ dữ liệu miễn phí của google)

**Nội dung 5:** Đặt ra yêu cầu, trường hợp hoạt động thực tế: Thông qua yêu cầu xây dựng project requirement

**Nội dung 6:** Tiến hành thực hiện lắp rắp mạch trên thực tế.

## Phân chia công việc trong nhóm

|  |  |
| --- | --- |
| **Tên thành viên** | **Nhiệm vụ** |
| **Nguyễn Công Huy** | * Tìm hiểu giao thức mạng HTTP POST, GET * Tìm hiểu sử dụng các loại module cảm biến * Đề xuất mô hình phần cứng * Vẽ schematic và layout phần cứng |
| **Trần Đình Huy** | * Tìm hiểu về firebase (cách get và up dữ liệu) * Tạo một app đơn giản sử dụng MIT inventor * Tìm hiểu sử dụng các loại module cảm biến |

Bảng 1: Phân chia công việc trong nhóm

# LÝ THUYẾT

## HTTP

HTTP (HyperText Transfer Protocol - Giao thức truyền tải siêu văn bản) là một trong các giao thức chuẩn về mạng Internet, được dùng để liên hệ thông tin giữa Máy cung cấp dịch vụ (Web server) và Máy sử dụng dịch vụ (Web client).

HTTP hoạt động dựa trên mô hình Client – Server. Trong mô hình này, các máy tính của người dùng sẽ đóng vai trò làm máy khách (Client). Sau một thao tác nào đó của người dùng, các máy khách sẽ gửi yêu cầu đến máy chủ (Server) và chờ đợi câu trả lời từ những máy chủ này.

Các thành phần chính của HTTP:

**HTTP Request Method:** Là phương thức để chỉ ra hành động mong muốn được thực hiện trên tài nguyên đã xác định.

GET: GET được sử dụng để lấy lại thông tin từ Server đã cung cấp bởi sử dụng một URI đã cung cấp. Các yêu cầu sử dụng GET chỉ nhận dữ liệu và không có ảnh hưởng gì tới dữ liệu.

HEAD: Tương tự như GET, nhưng nó truyền tải dòng trạng thái và khu vực Header.

POST: Một yêu cầu POST được sử dụng để gửi dữ liệu tới Server, ví dụ, thông tin khách hàng, file tải lên, …, bởi sử dụng các mẫu HTML.

PUT: Thay đổi tất cả các đại diện hiện tại của nguồn mục tiêu với nội dung được tải lên.

DELETE: Gỡ bỏ tất cả các đại diện hiện tại của nguồn mục tiêu bởi URI.

**HTTP - Responses:**

Cấu trúc của một HTTP response:

Một Status-line = Phiên bản HTTP + Mã trạng thái + Trạng thái

Có thể có hoặc không có các trường header

Một dòng trống để đánh dấu sự kết thúc của các trường header

Tùy chọn một thông điệp

Mã trạng thái: Thông báo về kết quả khi nhận được yêu cầu và xử lí bên server cho client.

Các kiểu mã trạng thái:

1xx: Thông tin (100 -> 101) VD: 100 (Continue), ….

2xx: Thành công (200 -> 206) VD: 200 (OK) , 201 (CREATED), …

3xx: Sự điều hướng lại (300 -> 307) VD: 305 (USE PROXY), …

4xx: Lỗi phía Client (400 -> 417) VD: 403 (FORBIDDEN), 404 (NOT FOUND), …

5xx: Lỗi phía Server (500 -> 505) VD: 500 (INTERNAL SERVER ERROR)

* 1. **WiFi**

Access Point là gì?

Thiết bị kết nối vào mạng WIFI được gọi là station (trạm). Việc kết nối vào mạng Wifi được hỗ trợ bởi một access point (AP), một AP có chức năng như một hub nhưng dùng cho nhiều station. Một access point thông thường được kết nối vào một mạng dây để phát WIFI (tức là chuyển từ mạng dây sang WIFI). Do đó access point luôn được tích hợp vào router. Mỗi access point được nhận biết bằng một SSID (Service Set IDentifier), SSID cũng là tên của mạng hiển thị khi ta kết nối vào WIFI.

# THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG

## Yêu cầu thiết kế

* Kích thước nhỏ gọn: 60 x 90 mm.
* Giá thành hợp lý: khoảng 400.000 đồng.
* Tiết kiệm năng lượng.
* Truyền nhận dữ liệu tốt, ít bị nhiễu.
* Giao tiếp được nhiều cảm biến. (ánh sáng, nhiệt độ & độ ẩm, độ ẩm đất).
* Sai số từ cảm biến thấp.
* Hoạt động lâu dài, độ ổn định cao.

## Phân tích thiết kế

Từ yêu cầu đặt ra về mạch điện, các phương pháp được đề ra cho thiết kế như sau:

## 

### MCU dùng để lập trình

|  |  |  |
| --- | --- | --- |
| **Tên MCU** | **ESP v1** | **Node MCU** |
| **Giá thành** | Rẻ (68.000đ) | Vừa phải (98.000đ) |
| **Các chức năng IO** | GPIO, UART (Ít, cần đi kèm với MCU khác) | GPIO, UART, PWM, ADC, I2C |
| **Độ ổn định** | Thấp | Cao |
| **Kích thước (mm)** | 25x15 | 30x55 |

Bảng 2: Lựa chọn MCU

Từ bảng so sánh trên, ta chọn Node MCU vì có đủ các chức năng IO mà ta cần.

### Cảm biến ánh sáng

|  |  |  |
| --- | --- | --- |
| **Tên** | **Quang trở (LDR)** | **BH1750** |
| **Giá thành** | Rất rẻ (2.000đ) | Vừa phải (35.000đ) |
| **Độ chính xác** | Thấp | Cao |
| **Giao tiếp với MCU** | ADC | I2C |
| **Tầm đo** | 10 - 1000 lux | 0 - 65535 lux |

Bảng 3: Lựa chọn cảm biến ánh sáng

Do NodeMCU được chọn ở trên chỉ có 1 chân ADC. Chân này được dùng cho cảm biến độ ẩm đất. Cộng với việc tầm đo của BH1750 rộng hơn so với quang trở, độ chính xác cao hơn. Vậy nên ta chọn BH1750 làm cảm biến ánh sáng

## Sơ đồ khối tổng quát

Gồm có 8 khối chính như sau:

**CẢM BIẾN**

**NHIỆT ĐỘ, ĐỘ ẨM**

**CẢM BIẾN**

**ĐỘ ẨM ĐẤT**

**CẢM BIẾN**

**ÁNH SÁNG**

**MCU**

**LED RGB**

**BƠM**

**FIREBASE**

**POWER SUPPLY**

**Adapter 12V**

Hình 1: Sơ đồ khối tổng quát

### Power supply

*Mục đích:* Cấp nguồn cho toàn bộ hệ thống.

*Yêu cầu:* Điện áp ổn định (± 2V). Khả năng cung cấp dòng điện từ 2A - 5A.

### Cảm biến nhiệt độ và độ ẩm

*Mục đích:* Đo nhiệt độ và độ ẩm của môi trường xung quanh. Từ đó có thêm thông tin về môi trường hiện tại.

*Yêu cầu:* Độ chính xác cao (sai số trong khoảng 2-5%), thời gian trả kết quả nhanh (trễ tối đa 1 - 2s). Tầm đo 0 - 50 C đối với nhiệt độ và 20-90 % đối với độ ẩm.

### Cảm biến độ ẩm đất

*Mục đích:* Đo độ ẩm của đất.

*Yêu cầu:* Sai số ở mức cho phép (khoảng 10 - 20%). Ổn định lâu (sai lệch 2 lần đo liên tục không quá 10%). Bền với thời gian (sử dụng được 1 - 2 năm).

### Cảm biến ánh sáng

*Mục đích:* Đo độ sáng của môi trường hiện tại.

*Yêu cầu:* Tầm hoạt động từ 10 - 1000 lux. Sai số giữa 2 lần đo liên tiếp không quá 20 lux.

### MCU

*Mục đích:* Lấy dữ liệu điều khiển từ firebase và lưu dữ liệu nhiệt độ, độ ẩm, ánh sáng,.. lên firebase. Điều khiển các thiết bị.

*Yêu cầu:* Tích hợp wifi. Nhiều GPIO (khoảng 8 GPIO). Tối thiểu 1 ADC. Kích thước nhỏ.

### Bơm

*Mục đích:* Bơm nước tưới cây.

*Yêu cầu:* Nhỏ gọn. Điện áp hoạt động từ 9 - 12V. Công suất nhỏ.

### LED RGB

*Mục đích:* Thắp sáng cho cây khi ánh sáng mặt trời không đủ sáng.

*Yêu cầu:* Dải LED. Điện áp hoạt động từ 9 - 12V. Công suất thấp (12 - 24W). Có thể điều khiển được nhiều màu.

### Firebase

*Mục đích:* Lưu dữ liệu nhiệt độ, độ ẩm, cường độ ánh sáng, tín hiệu điều khiển, v.v..

*Yêu cầu:* Khả năng lưu trữ lớn (tối đa 1GB).

## Sơ đồ khối chi tiết

### Khối cấp nguồn

**ADAPTER 12/5A**

**DC - DC**

**12V**

**5V**

Hình 2: Sơ đồ khối cấp nguồn

Mạch được cấp nguồn 12V/5A. Nguồn 12V phục vụ cho bơm và LED RGB. Bộ DC-DC có tác dụng hạ áp xuống còn 5V phục vụ cho NodeMCU và các cảm biến.

### Cảm biến nhiệt độ và độ ẩm

**DHT 11**

**NODE MCU**

DATA

Hình 3: Sơ đồ khối cảm biến nhiệt độ và độ ẩm

Single – bus data format được dùng cho giao tiếp và đồng bộ giữa MCU và DHT11. Một tiến trình giao tiếp thì mất khoảng 4ms. Dữ liệu bao gồm cả phần thập phân và phần nguyên (kiểu Float). Dữ liệu truyền nhận hoàn chỉnh chiếm 40bits, và cảm biến gửi những bit cao trước (higher data bit first).

Định dạng dữ liệu: 8bits RH (độ ẩm) phần nguyên + 8 bits RH phần thập phân + 8bits T (nhiệt độ) phần nguyên + 8bits T phần thập phân + 8bits check sum. Nếu dữ liệu truyện nhận là chính xác thì phần checksum sẽ 8 bits cuối cùng của chuỗi.

### Cảm biến độ ẩm đất

**CẢM BIẾN**

**ĐỘ ẨM ĐẤT**

**NODE MCU**

ANALOG

Hình 4: Sơ đồ khối cảm biến độ ẩm đất

Cảm biến độ ẩm đất với ngõ ra là analog được kết nối đến chân A0 của NODE MCU.

### Cảm biến ánh sáng

**BH1750**

**NODE MCU**

SDA

SCL

Hình 5: Sơ đồ khối cảm biến ánh sáng

Cảm biến ánh sáng (BH1750) giao tiếp với NODE MCU thông qua giao tiếp I2C. Nguồn cấp BH1750 được lấy từ chân 3.3V của NODE MCU.

### Bơm

**NODE MCU**

**KHỐI**

**ĐIỀU KHIỂN RELAY**

**BƠM**

Hình 6: Sơ đồ khối điều khiển bơm

Do bơm hoạt động cần điện áp 12V/1A, nên ta sử dụng relay để đóng mở bơm. Tín hiệu điều khiển bơm từ NODE MCU được đưa đến khối điều khiển relay cho phép đóng mở relay. Từ đó đóng mở bơm.

### LED RGB

**NODE MCU**

**KHỐI**

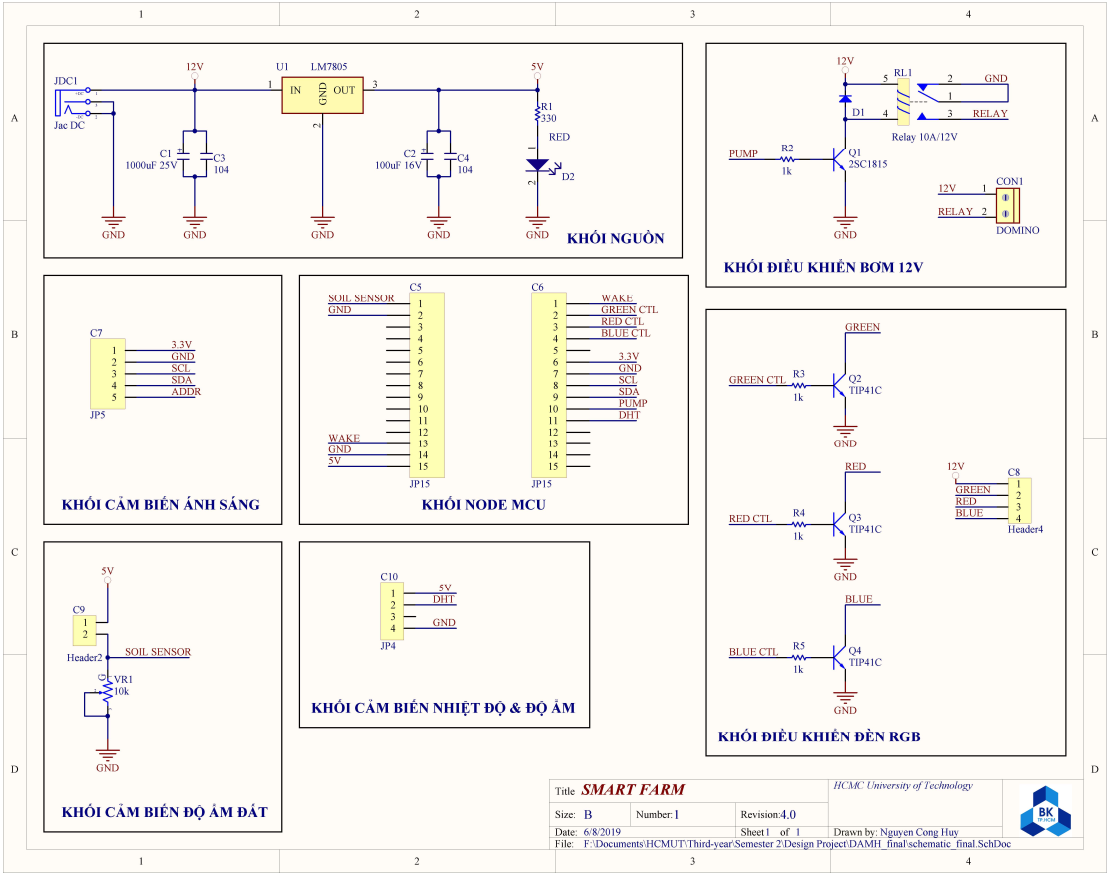
**ĐIỀU KHIỂN LED RGB**

**LED RGB**

Hình 7: Sơ đồ khối điều khiển LED RGB

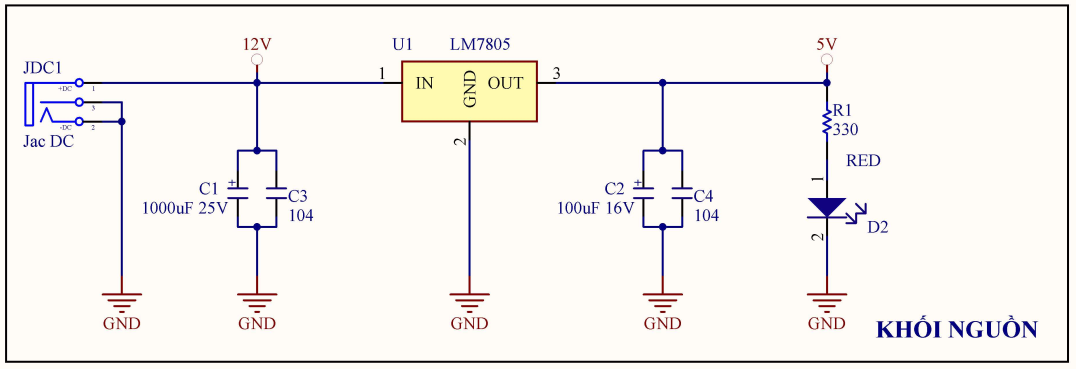
Tín hiệu điều khiển từ NODEMCU (dạng PWM) được đưa đến khối điều khiển LED RGB. Khối điều khiển này sử dụng các transistor với chức năng là switch có vai trò đóng mở dòng điều khiển LED.

## Sơ đồ mạch chi tiết

****

Hình 8: Sơ đồ mạch chi tiết

### Khối nguồn

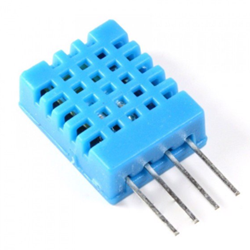


Hình 9: Sơ đồ mạch khối nguồn

Mạch sử dụng nguồn adapter 12V - 5A. Sử dụng tụ để làm phẳng điện áp đưa vào và lọc nhiễu, đảm bảo NODE MCU và các cảm biến đủ áp để có thể hoạt động ổn định.

Do NODE MCU hoạt động cần áp từ 3.3 - 20V, cảm biến nhiệt độ và độ ẩm (DHT11) cần áp 5V. Nên trong khối nguồn này em sử dụng LM7805 với áp đầu vào từ 7.0 - 25V và áp đầu ra từ 4.8 - 5.2V để cấp cho các module trên.

### Cảm biến nhiệt độ & độ ẩm

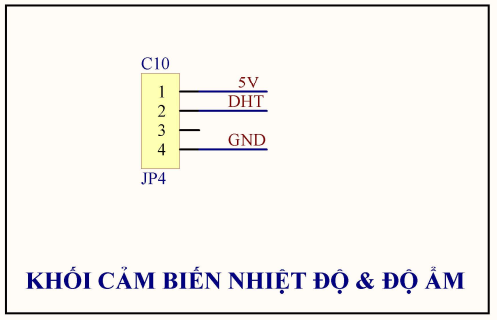


*Cảm biến DHT11*

*Thông số kỹ thuật:*

* Điện áp hoạt động 3.3V – 5.5V.
* Dải đo độ ẩm 0 - 100%.
* Sai số độ ẩm ±2%.
* Dải đo nhiệt độ - 40 – 80oC.
* Sai số nhiệt dộ ±0.5oC

*Kết nối trên mạch:*



Hình 10: Sơ đồ kết nối DHT11 trên mạch

Nguồn cấp 3 - 5.5V. Chân data (DHT) được kết nối với NODE MCU.

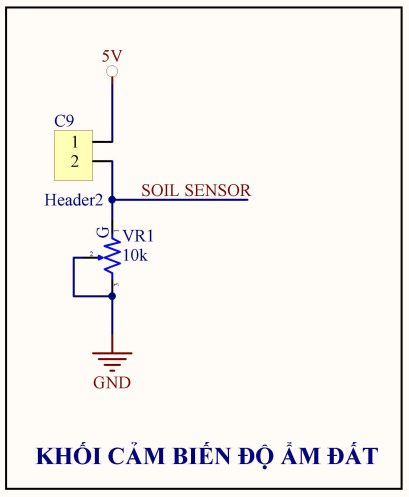
### Cảm biến độ ẩm đất



*Cảm biến độ ẩm đất*

*Thông số kỹ thuật:*

* Điện áp hoạt động 3 - 5.5V

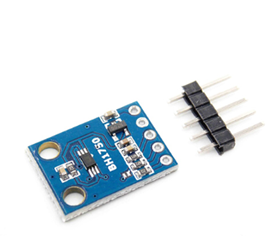


Hình 11: Sơ đồ kết nối cảm biến đo độ ẩm trên mạch

Sử dụng chân analog của NODE MCU để đo độ ẩm đất. Biến trở VR1 có tác dụng điều chỉnh điện trở cho phù hợp.

### Cảm biến ánh sáng

Cảm biến cường độ ánh sáng BH1750 được sử dụng để đo cường độ ánh sáng theo đơn vị lux, cảm biến có ADC nội và bộ tiền xử lý nên giá trị trả ra là giá trị trực tiếp cường độ ánh sáng lux mà không phải qua bất kỳ xử lý hay tính toán nào thông qua giao tiếp I2C.

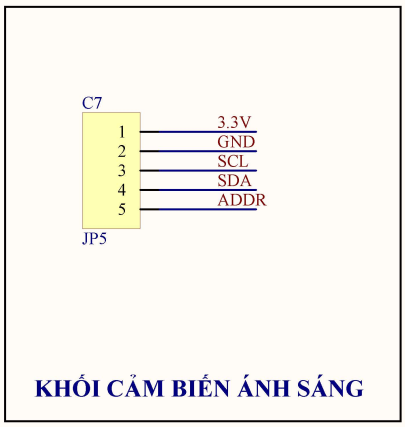
**

*Module BH1750*

*Thông số kỹ thuật:*

* Nguồn: 3~5VDC
* Giao tiếp: I2C
* Khoảng đo: 1 ➡ 65535 lux
* Kích cỡ: 21\*16\*3.3mm

*Kết nối trên mạch:*

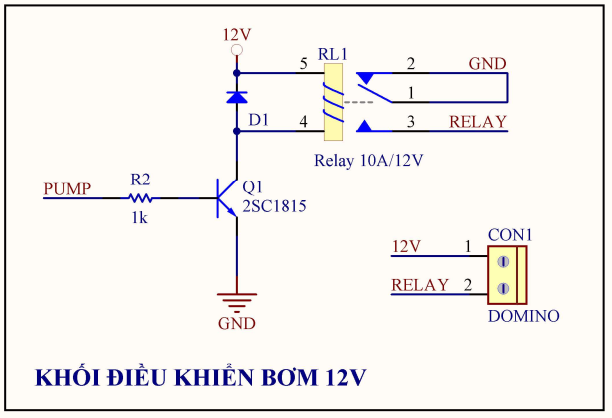


Hình 12: Sơ đồ kết nối cảm biến đo ánh sáng trên mạch

* Sử dụng I2C mềm để giao tiếp với cảm biến ánh sáng (BH1750)

### Khối điều khiển bơm

*Kết nối trên mạch:*

**

Hình 13: Sơ đồ kết nối khối điều khiển bơm trên mạch

* Sử dụng relay 12V/10A để có thể bật/tắt bơm với yêu cầu điện áp 12V và dòng 1A.
* Diode D1 có tác dụng dẫn dòng điện của điện áp cảm ứng (sinh ra khi transistor ngưng dẫn) phóng qua cuộn dây và qua diode làm cho từ trường bị triệt tiêu một cách nhanh chóng.
* Do điện áp ra tại chân NodeMCU không đủ để có thể kích relay, vì vậy ta sử dụng BJT C1815 (NPN) với chức năng là switch bật tắt relay.

### Khối điều khiển LED RGB

### 

Hình 14 Sơ đồ mạch điều khiển đèn RGB

Để LED có thể hoạt động tốt cần cấp nguồn 12V/2A. Do dòng lớn nên ta sử dụng NPN TIP41C với khả năng chịu dòng tối đa 6A (hoạt động ở trạng thái ON)

# THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM

## Google Firebase

Google firebase là gì?

Là một dịch vụ cơ sở dữ liệu thời gian thực hoạt động trên nền tảng đám mây được cung cấp bởi Google nhằm giúp các lập trình viên phát triển nhanh các ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.

Firebase Realtime Database:

Dữ liệu sẽ được cung cấp dưới dạng JSON và luôn đồng bộ thời gian thực đến mọi kết nối client.

Với các ứng dụng đa nền tảng, tất cả các client sẽ đề sử dụng chung 1 DB và luôn được tự động cập nhật dữ liệu mới nhất.

JSON(Javascript Oject Notation) , nó được dùng để truyền tải dữ liệu giữa server và client.

## MIT APP INVENTOR

MIT App Inventor dành cho Android là một ứng dụng web nguồn mở ban đầu được cung cấp bởi Google và hiện tại được duy trì bởi Viện Công nghệ Massachusetts (MIT).

Nền tảng cho phép nhà lập trình tạo ra các ứng dụng phần mềm cho hệ điều hành Android (OS). Bằng cách sử dụng giao diện đồ họa, nền tảng cho phép người dùng kéo và thả các khối mã (blocks) để tạo ra các ứng dụng có thể chạy trên thiết bị Android.

Những tính năng có trên MIT App Inventor là:

+ Cho phép xây dựng nhanh chóng những thành phần cơ bản (components) của một ứng dụng Android: Nút bấm, nút lựa chọn, chọn ngày giờ, ảnh, văn bản, thông báo, kéo trượt, trình duyệt web.

+ Sử dụng nhiều tính năng trên điện thoại: Chụp ảnh, quay phim, chọn ảnh, bật video hoặc audio, thu âm, nhận diện giọng nói, chuyển lời thoại thành văn bản, dịch.

+ Hỗ trợ xây dựng game bằng các components: Ball, Canvas, ImageSprite.

+ Cảm biến: đo gia tốc (AccelerometerSensor), đọc mã vạch, tính giờ, con quay hồi chuyển (gyroscopeSensor), xác định địa điểm (locationSensor), NFC, đo tốc độ (pedometer), đo khoảng cách xa gần với vật thể (proximitySensor).

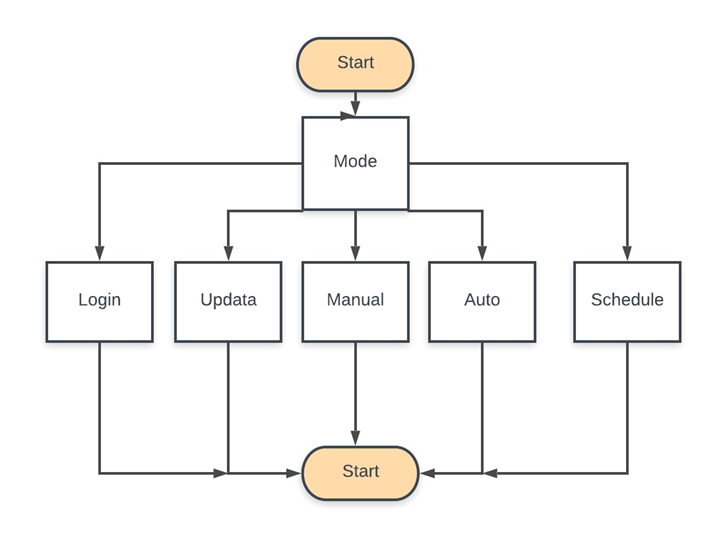
+ Kết nối: Danh bạ, email, gọi điện, chia sẻ thông qua các ứng dụng mạng xã hội khác trên thiết bị, nhắn tin, sử dụng twitter qua API, bật ứng dụng khác, bluetooth, bật trình duyệt.

+ Lưu trữ: đọc hoặc lưu tệp txt, csv, sử dụng FusiontablesControl, tạo cơ sở dữ liệu đơn giản trên điện thoại hoặc trên đám mây thông qua server tự tạo hoặc Firebase.

+ Điều khiển robot thông qua LegoMindstorms.

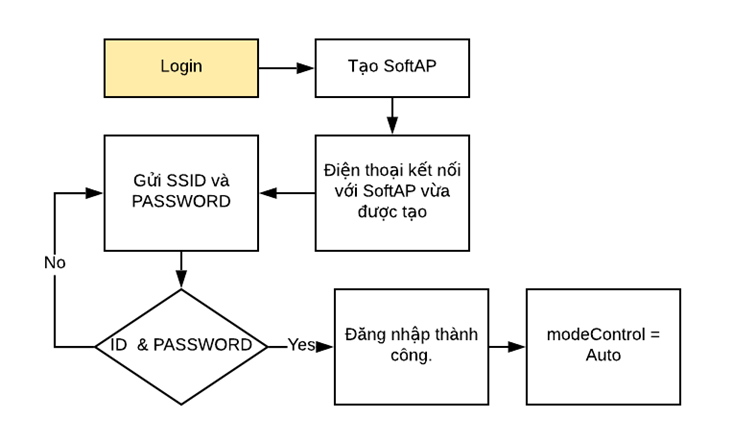
## Lưu đồ giải thuật

Lưu đồ giải thuật tổng quát:



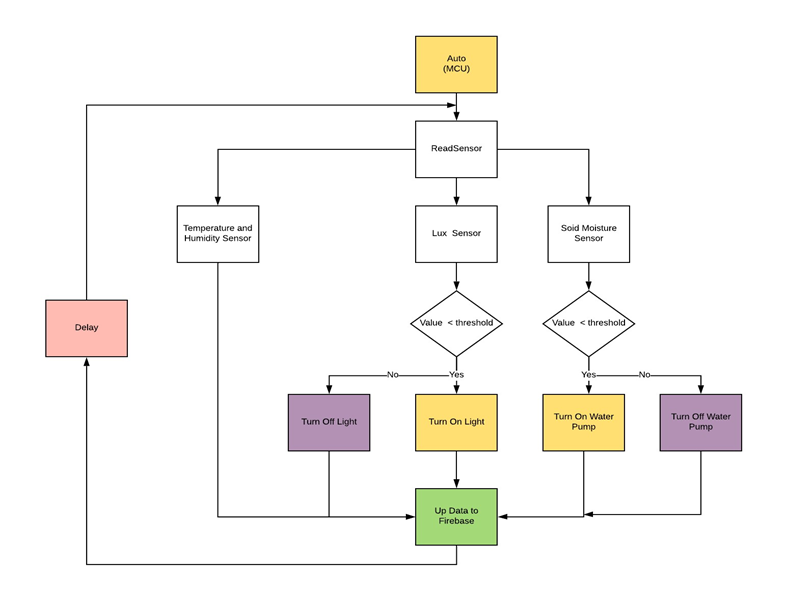
Hình 15: Lưu đồ giải thuật tổng quát.

Login Mode:



Hình 16: Login Mode

Auto Mode:



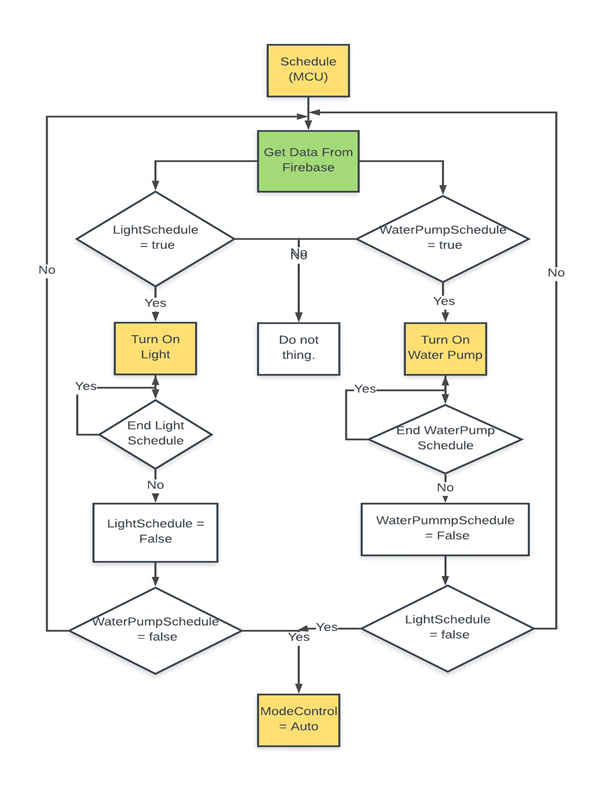
Hình 17: Auto Mode

Manual Mode:



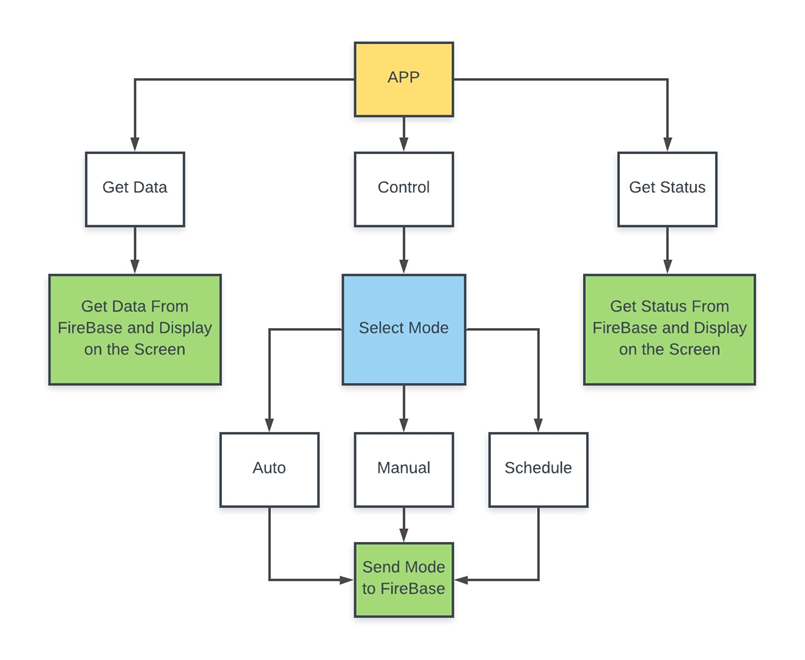
Hình 18: Manual Mode

Schedule Mode:



Hình 19: Schedule Mode

Tương tác giữa MIT APP INVENTOR và Firebase:

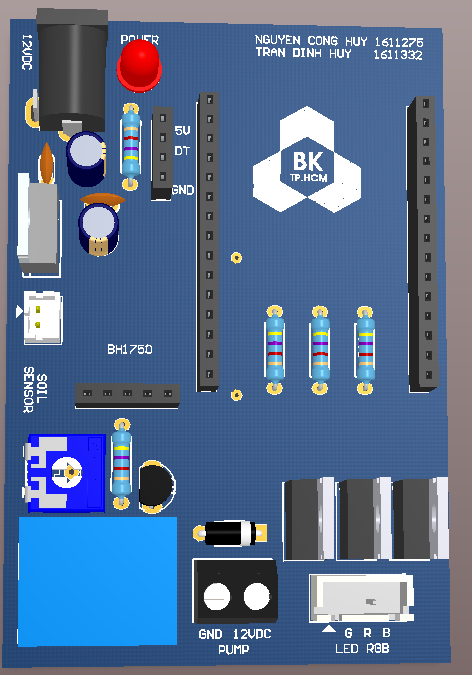


Hình 20: Tương tác giữa App và Firebase

# KẾT QUẢ THỰC HIỆN

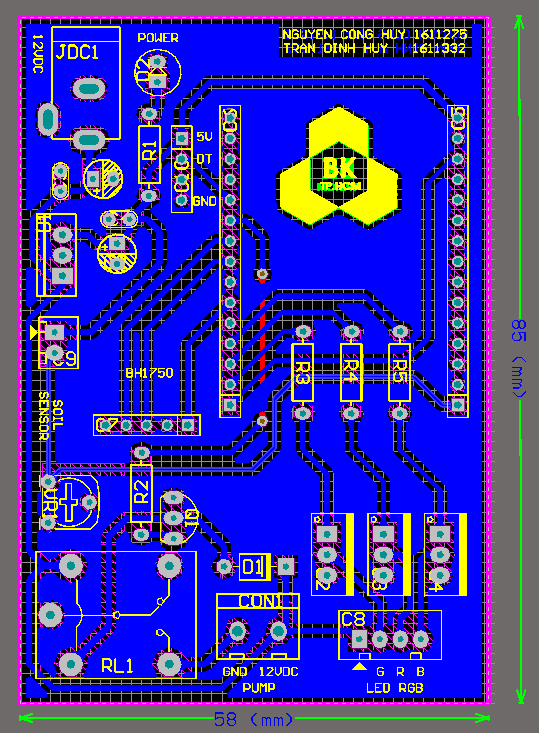
## Phần cứng

### 3D



Hình 21: Hình ảnh 3D của mạch trên Altium

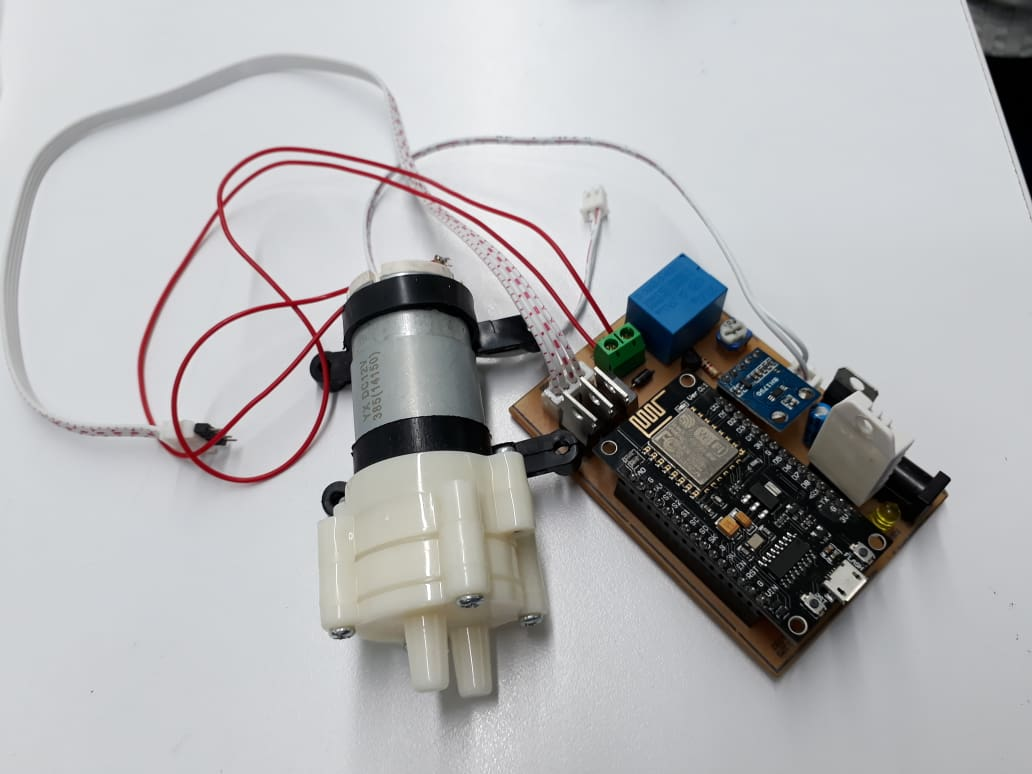
### Layout



Hình 22: Hình ảnh layout của mạch trên Altium

## 

### Mạch thực tế

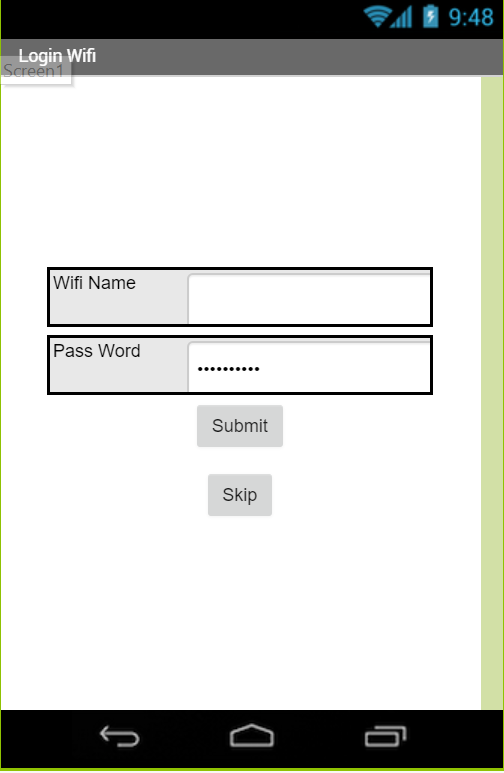


Hình 23: Hình ảnh mạch thực tế

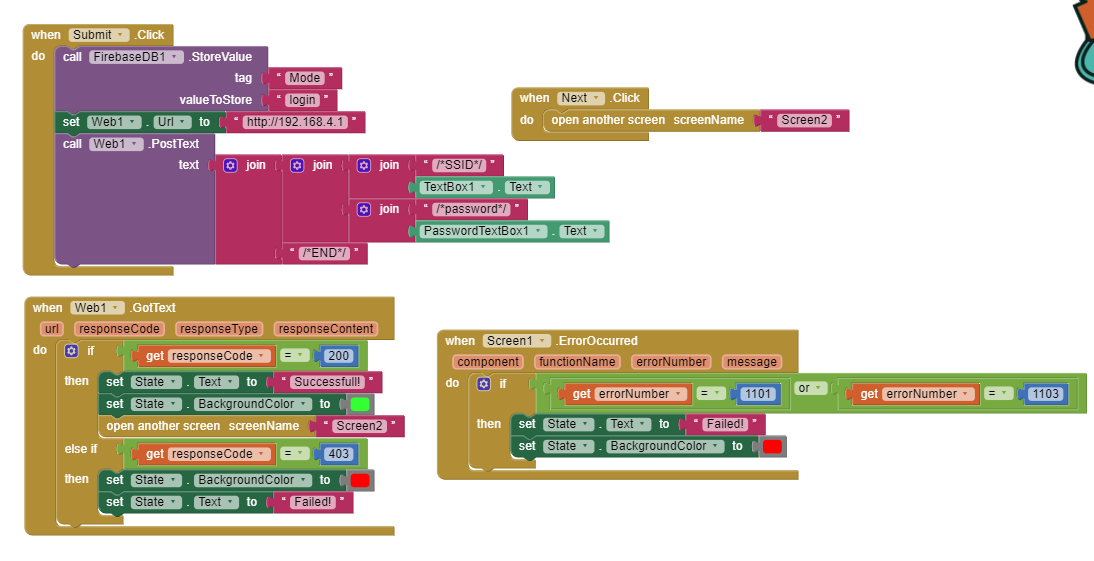
* Kích thước mạch 58x85 (mm).
* Độ rộng dây 12V, 5V và 3.3V vào khoảng 30 mil (0.762 mm).
* Độ rộng các chân tín hiệu vào khoảng 25 mil (0.635 mm).

## Phần mềm

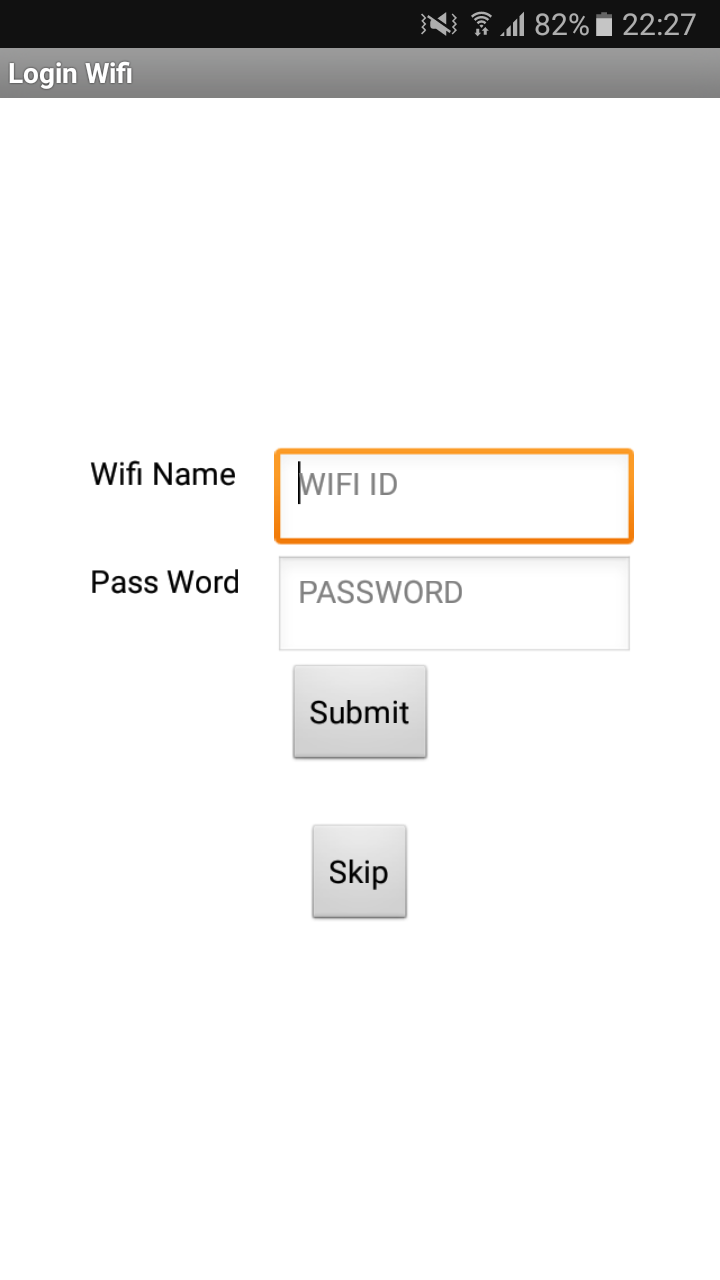
### MIT APP INVENTOR



Hình 24: Hình ảnh app

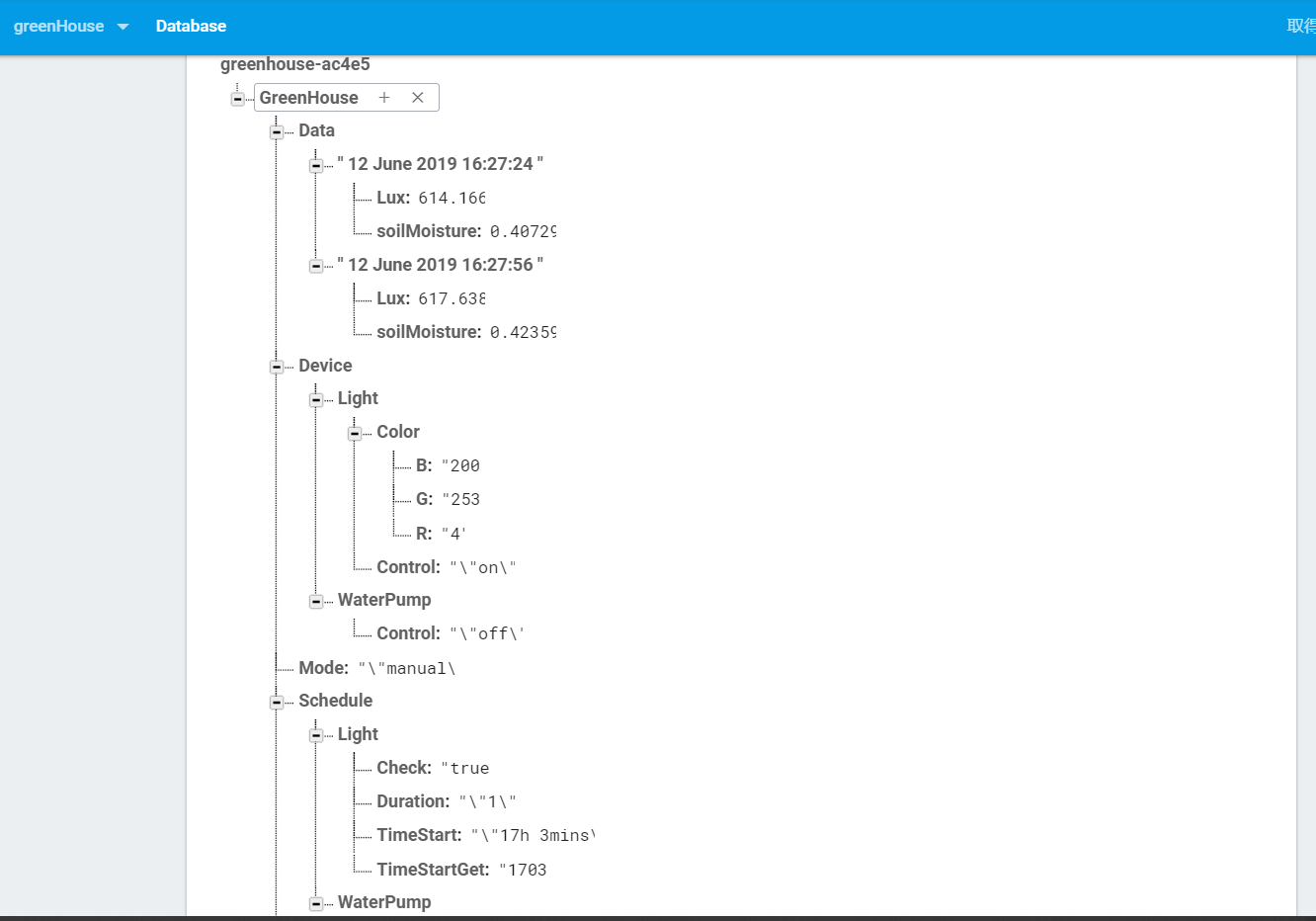


Hình 25: Giao diện kéo thả



Hình 26: Giao diện trên điện thoại

### Google Firebase

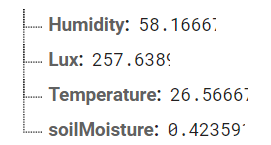


Hình 27: Giao diện firebase.

## Giá trị cảm biến và thời gian thực

### Giá trị cảm biến

Truy cập firebase để xem các giá trị của các cảm biến:



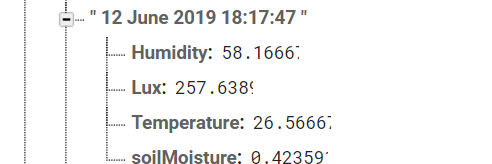
Hình 28: Giá trị cảm biến

***Phân tích đánh giá:***

* Dữ liệu nhiệt độ chênh lệch 1 độ C so với thực tế (kiểm tra nhiệt kế lắp trong phòng)
* Dữ liệu độ ẩm chênh lệch 2% so với thực tế (kiểm tra nhiệt kế lắp trong phòng)
* Đánh giá : dữ liệu nhiệt độ, độ ẩm đạt yêu cầu thiết kế.

### Thời gian thực

Truy cập firebase để xem thời gian thực:



Hình 29: Thời gian thực

Phân tích đánh giá:

* Dữ liệu thời gian thực trùng khớp với dữ liệu được kiểm tra từ đồng hồ

# ĐÁNH GIÁ TỔNG QUAN

## Phần cứng

* Kích thước, giá thành đáp ứng yêu cầu đặt ra
* Khối nguồn có khả năng chịu dòng tốt
* Tuy nhiên thỉnh thoảng vẫn xảy ra nhiễu khi kích relay hoạt động.

## Phần mềm

* MIT APP INVENTOR: giao diện chưa đẹp.
* Firebase: Cách tổ chức lưu trữ dữ liệu chưa tối ưu.
* Giải thuật: Đáp ứng chưa nhanh, còn nhiều bất cập.

## Kết quả làm việc nhóm

Sau khoảng thời gian cùng nhau làm việc nhóm, nhóm đã cùng nhau cố gắng nổ lực để hoàn thành các nhiệm vụ được GVHD đề ra và những gì chúng tôi tự đặt mục tiêu cho mình.

Làm việc nhóm khá hiệu quả, nhóm đã cùng nhau đóng góp ý kiến, giúp đỡ nhau hoàn thành. Nhóm biết cách phân công, sắp xếp thời gian để hoàn thành đúng tiến độ được giao.

Tuy nhiên, quá trình làm việc, nhóm vẫn chưa thực sự hoàn hảo vì còn khá nhiều thiếu xót như tính đúng giờ, khả năng hoàn thành công việc của mỗi người, khả năng cộng tác kế thừa và cùng hoàn thành hay độc lập riêng lẻ còn hạn chế.

Hi vọng trong những dự án sau, nhóm sẽ cố gắng khắc phục để cùng nhau tốt hơn.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Trong dự án này, nhóm đã đề xuất thiết kế một hệ thống điều khiển thiết bị và thu thập dữ liệu đơn giản từ cảm biến, lưu vào cơ sở dữ liệu và truy xuất một cách chủ động. Phần cứng được thiết kế đơn giản và nhỏ gọn. Giải thuật ứng dụng trên thiết bị giám sát đơn giản, có khả năng tối ưu. Giao diện ứng dụng đơn giản. Cơ sở dữ liệu thời gian thực được xây dựng sẵn, dễ dàng sử dụng và phát triển. Hệ thống hoạt động ổn định.

Tuy nhiên, hệ thống này vẫn có một số khuyết điểm. Các chức năng chưa thực sự thời gian thực. Phần cứng thiết kế chưa thực sự hoàn hảo, đôi khi xuất hiện nhiễu khi relay hoạt động. Mô hình chưa đặc tả được thực tế. Giải thuật chưa thực sự tối ưu, chưa phát triển tính năng sleep, việc thu thập dữ liệu còn nhiều khuyết điểm.

## Hướng phát triển

* Thêm một vài cảm biến đọc các thông số dinh dưỡng của cây như: độ pH, tỉ lệ các chất, lượng nước tưới trong một ngày…
* Phát triển trên nền tảng Web.
* Mở rộng quy mô…

# TÀI LIỆU THAM KHẢO

1. <https://www.espressif.com/en/products/hardware/esp8266ex/overview>
2. <https://firebase.google.com/docs/android/setup>
3. ESP8266 Non-OS SDK API Reference
4. ESP8266 Low Power Solutions

# PHỤ LỤC

Code: <https://github.com/rextran/nodeMcu.git>