

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ NH  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
**KHOA ĐIỆN TỬ VIỄN THÔNG**

---oOo---

**ĐÀO THỊ HỒNG NHI**

**HỆ THỐNG THU THẬP THÔNG TIN  
TỪ CẢM BIẾN NHIỆT ĐỘ VÀ ĐỘ ẨM TRONG NÔNG NGHIỆP**

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN NGÀNH ĐIỆN TỬ - VIỄN THÔNG

CHUYÊN NGÀNH: VIỄN THÔNG - MẠNG

NGƯỜI HƯỚNG DẪN KHOA HỌC:

**ThS. NGÔ ĐẮC THUẦN**

**Thành phố Hồ Chí Minh – 2021**

**ĐẠI HỌC QUỐC GIA TP. HCM**  
**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**KHOA ĐIỆN TỬ - VIỄN THÔNG**

**HỌ VÀ TÊN HỌC VIÊN**

**ĐÀO THỊ HỒNG NHI**

**ĐỀ TÀI:**

**HỆ THỐNG THU THẬP THÔNG TIN**  
**TỪ CẢM BIẾN NHIỆT ĐỘ VÀ ĐỘ ẨM TRONG NÔNG NGHIỆP**

**KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN NGÀNH ĐIỆN TỬ - VIỄN THÔNG**  
**CHUYÊN NGÀNH: VIỄN THÔNG – MẠNG**

**NGƯỜI HƯỚNG DẪN KHOA HỌC:**

**Th.S NGÔ ĐẮC THUẦN**

**TP. Hồ Chí Minh – Năm 2021**

## LỜI CẢM ƠN

Lời đầu tiên, em xin chân thành bày tỏ lòng cảm ơn và kính trọng sâu sắc đến **Thầy, ThS. Ngô Đắc Thuận**, người đã tạo điều kiện tốt nhất để em hoàn thành tốt luận văn này. Luận văn này sẽ không thể hoàn thành nếu không có sự hướng dẫn, tin tưởng và tận tình chỉ bảo của thầy.

Em xin chân thành cảm ơn quý công ty TNNN Minh Huy và anh Lê Quốc Trung đã hướng dẫn tận tình. Cảm ơn bạn Cao Hoàng Minh đã hỗ trợ thiết kế phần cứng, và báo cáo này bao gồm cả phần cứng và phần mềm. Hai phần này hỗ trợ cùng phát triển để hoàn thành đề tài hoàn thiện nên cả hai phần đều không thể thiếu trong đề tài.

Em xin thể hiện lòng biết ơn đến **Quý Thầy Cô** trong **Khoa Điện tử - Viễn Thông** đã tận tình giảng dạy, truyền đạt những kiến thức quý báu cho em trong những năm học vừa qua, giúp em có được một nền tảng kiến thức vững chắc.

Nhân đây, con xin được bày tỏ lòng cảm ơn sâu sắc đến **gia đình**, cảm ơn những tình cảm, sự vất vả, và hi sinh của **Cha Mẹ** đã giúp con có thể theo đuổi con đường học tập cho đến ngày hôm nay.

Cuối cùng, xin được nói lời cảm ơn chân thành đến các **anh chị** và **các bạn** đã giúp đỡ, khích lệ cũng như phê bình, góp ý, giúp em hoàn thành công việc một cách tốt nhất.

Người thực hiện đề tài

**Đào Thị Hồng Nhi**

## MỤC LỤC

### Mục lục

LỜI CẢM ƠN.....	1
DANH MỤC CÁC HÌNH VẼ.....	4
LIỆT KÊ BẢNG.....	5
CHƯƠNG 1: TỔNG QUAN.....	6
1.1 ĐẶT VẤN ĐỀ.....	6
1.2 MỤC TIÊU.....	6
1.3 NỘI DUNG NGHIÊN CỨU.....	7
1.4 GIỚI HẠN.....	7
1.5 PHẠM VI ỨNG DỤNG.....	7
Chương 2: CƠ SỞ LÝ THUYẾT.....	8
2.1 TỔNG QUAN VỀ IoT.....	8
2.2 CÔNG NGHỆ WI-FI.....	12
2.3.1 Khái niệm Wi-Fi.....	12
2.3.2 Các đặc điểm của Wi-Fi.....	13
2.3 PHẦN CỨNG.....	13
2.3.1 Giới thiệu về ESP8266 NodeMCU.....	13
2.3.2 Module DHT11.....	15
2.3.3 Giao tiếp UART.....	17
2.4 HỆ ĐIỀU HÀNH ANDROID.....	17
2.3.1 Khái niệm Android.....	17
2.3.2 Lịch sử phát triển.....	17
2.3.3 Phiên bản của hệ điều hành.....	18
2.3.4 Ưu khuyết điểm của hệ điều hành Android.....	21
2.3.5 Kiến trúc hệ điều hành.....	21
2.3.6 Chu kì ứng dụng trên Android.....	24
2.5 GOOGLE FIREBASE.....	25
2.5.1 Giới thiệu về Firebase.....	25

2.5.2	Lịch sử của Firebase.....	25
2.5.3	Chức năng của Firebase .....	25
2.5.4	Thành phần của Firebase.....	26
CHƯƠNG 3: TÍNH TOÁN THIẾT KẾ .....		27
3.1	GIỚI THIỆU VỀ ĐỀ TÀI.....	27
3.2	THIẾT KẾ SƠ ĐỒ HỆ THỐNG .....	27
3.2.1	Thiết kế sơ đồ khối hệ thống .....	27
3.2.2	Tính toán và thiết kế.....	28
3.2.2.1	Khối xử lý trung tâm .....	28
3.2.3	Sơ đồ nguyên lý toàn mạch .....	35
CHƯƠNG 4: THI CÔNG HỆ THỐNG .....		36
4.1	GIỚI THIỆU .....	36
4.2	THI CÔNG HỆ THỐNG .....	36
4.2.1	Board điều khiển .....	36
4.2.1	Phần mềm Arduino IDE.....	38
4.2.2	Firebase .....	44
4.2.3	App điều khiển .....	44
4.3	HƯỚNG DẪN SỬ DỤNG .....	53
CHƯƠNG 5: KẾT QUẢ_NHẬN XÉT _ĐÁNH GIÁ .....		60
5.1	KẾT QUẢ ĐẠT ĐƯỢC .....	60
	Tổng quan kết quả đạt được.....	60
	Thống kê kiểm thử hệ thống .....	64
5.2	Nhận xét_ Đánh giá.....	64
CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....		65
6.1	KẾT LUẬN.....	65
6.1.1	Ưu điểm .....	65
6.1.2	Nhược điểm .....	65
6.2	HƯỚNG PHÁT TRIỂN.....	65
Tài liệu tham khảo .....		66

## DANH MỤC CÁC HÌNH VẼ

<b>Hình</b>	<b>Trang</b>
Hình 2.1: Kiến trúc IoT. ....	8
Hình 2.2: Smart home.....	9
Hình 2.3: Smart watch. ....	9
Hình 2.4: Smart city.....	10
Hình 2.5: Internet trong công nghiệp. ....	10
Hình 2.6: Smart car.....	11
Hình 2.7: Telemedicine. ....	11
Hình 2.8: IoT trong nông nghiệp. ....	12
Hình 2.9: ESP 8266. ....	13
Hình 2.10: Sơ đồ chân ESP8266. ....	14
Hình 2.11: Tín hiệu DHT11 vào vi điều khiển.....	16
Hình 2.12: Kiến trúc hệ điều hành Android. ....	22
Hình 2.13: Vòng đời Activity.....	24
Hình 2.14: Trang chủ Firebase. ....	25
Hình 3.1: Sơ đồ khối hệ thống.....	27
Hình 3.8: Ảnh module hạ áp XL4015. ....	32
Hình 4.13: Code định hướng các nút trong tab. ....	49
Hình 4.14: Code lấy dữ liệu cảm biến đo được từ Firebase. ....	50
Hình 4.15: Code nhận dữ liệu và hiển thị thành biểu đồ. ....	50
Hình 4.16: Nút điều khiển thiết bị trên App.....	51
Hình 4.17: Code gửi giá trị điều khiển và ngưỡng của máy bơm lên Firebase.....	51
Hình 4.18: Sơ đồ khối App điều khiển Android.....	52
Hình 4.19: Lưu đồ cập nhật dữ liệu.....	53
Hình 4.21: Màn hình đăng nhập. ....	54
Hình 4.22: Màn hình trang chủ.....	55
Hình 4.23: Đồ thị nhiệt độ và độ ẩm. ....	55
Hình 4.25: Màn hình thêm vườn. ....	56
Hình 4.26: Màn hình thiết bị. ....	57

## LIỆT KÊ BẢNG

<b>Bảng</b>	<b>Trang</b>
Bảng 1. 1: So sánh đề tài với sáng chế phương pháp và hệ thống kiểm soát tưới tiêu. ....	6
Bảng 2. 1: Thông số kỹ thuật DHT11. ....	15
Bảng 3. 1: Thông số kỹ thuật Module 4 Relay rắn SSR. ....	29
Bảng 3. 2: Thông số kỹ thuật của nguồn tổ ong. ....	32
Bảng 3. 3: Thông số kỹ thuật của mạch hạ áp XL 4015. ....	33
Bảng 3. 4: Dòng điện các linh kiện sử dụng trong mạch điều khiển ....	33
Bảng 3. 5: Bảng thống kê linh kiện điều khiển. ....	37
Bảng 5.1: Thống kê kết quả thực tế. ....	75

## CHƯƠNG 1: TỔNG QUAN

### 1.1 ĐẶT VẤN ĐỀ

Hiện nay, với sự phát triển không ngừng của công nghệ 4.0. Hệ thống IoT (Internet of Things) dần trở nên phổ biến và được ứng dụng trong nhiều lĩnh vực khác nhau như nông nghiệp, y tế, giáo dục, giao thông,... nhằm nâng cao chất lượng sống cho người dân. Nhiều dự án, nghiên cứu về lĩnh vực điều khiển từ xa ngày càng phát triển vượt bậc.

Đặc biệt là trong lĩnh vực nông nghiệp, các sản phẩm công nghệ của IoT đã giúp kết nối ngành nông nghiệp với các công nghệ điều khiển từ xa. Nó giúp cho quá trình giám sát và điều khiển trở nên thuận tiện ở vị trí xa vườn. Từ nhu cầu trên em đã làm một đề tài “*Hệ thống thu thập thông tin từ cảm biến nhiệt độ và độ ẩm trong nông nghiệp*”.

Hệ thống tối ưu trong việc đo giá trị nhiệt độ và độ ẩm trong vườn từ đó thực hiện việc điều khiển các thiết bị. Thiết bị thực hiện các chức năng tự vận hành hoặc điều khiển của người dùng thông qua App android. Người dùng có thể hẹn giờ, đặt giá trị ngưỡng giá trị nhiệt độ và độ ẩm cho thiết bị và bật tắt thiết bị theo ý muốn khi điện thoại có kết nối Internet. Ứng dụng điện thoại có những chức năng tương tác với người dùng để nâng cao năng suất.

Tên đề tài	Hệ thống thu thập thông tin từ cảm biến nhiệt độ và độ ẩm trong nông nghiệp	Các phương pháp và hệ thống kiểm soát tưới tiêu
Mục tiêu	Chăm sóc rau hiệu quả	Chăm sóc các cây về cảnh quan
Thiết bị	Đèn, quạt, phun sương, bơm	Hệ thống tưới nhỏ giọt
Dữ liệu	Cảm biến	Thời tiết lịch sử khu vực
Cách thức truyền	Mạng không dây	Mạng không dây
Chức năng	Điều khiển, đặt ngưỡng, hẹn giờ	Điều khiển, đặt giờ

*Bảng 1. 1: So sánh đề tài với sáng chế phương pháp và hệ thống kiểm soát tưới tiêu.*

Bảng biểu thị điểm mạnh và yếu của đề tài và nếu xét trên các phương diện khác nhau thì các đề tài đều phát huy điểm mạnh của nó. Từ những module sẵn có để thực hiện những yêu cầu. Đề tài được so sánh với Các phương pháp và hệ thống kiểm soát tưới tiêu là một sáng chế của Bulut F. Ersavas, Semih Pekol, Atakan Bodur.

Công ty TNNN Minh Huy hiện đang thi công các hệ thống cảnh quan dựa trên sáng chế phương pháp và hệ thống kiểm soát tưới tiêu khá thành công và sử dụng app Hydrawise của công ty Hunter Industries. Và công ty TNNN Minh Huy mở rộng sang mảng chăm sóc cây trồng, đó là điều kiện để em làm đề tài này.

### 1.2 MỤC TIÊU

- Thiết bị Nhận giá trị nhiệt độ, độ ẩm đo được từ cảm biến.
- Hệ thống có chức năng giám sát và điều khiển từ xa thông qua internet.
- App dùng để đặt giờ, đảm bảo điều kiện sống của cây, quản lý thông tin người dùng.



- Thực hiện đồ án trên vườn thực tế hoặc mô hình.
- Giá thành phù hợp và hệ thống ổn định.

### 1.3 NỘI DUNG NGHIÊN CỨU

Mục tiêu xây dựng app ***“Hệ thống thu thập thông tin từ cảm biến nhiệt độ và độ ẩm trong nông nghiệp”***, em sẽ thực hiện những nội dung như sau:

- Nội dung 1: Nghiên cứu tài liệu về KIT NodeMCU ESP8266, nhiệt độ và độ ẩm DHT11 chuẩn giao tiếp Wi-Fi và mạng Internet, Phần mềm Android Studio.
- Nội dung 2: Tìm hiểu về cách gửi dữ liệu đo được từ cảm biến đến phần mềm App.
- Nội dung 3: Thiết kế và tính toán thiết kế mạch phần cứng, thiết kế App.
- Nội dung 4: Thi công phần cứng, thử nghiệm và hiệu chỉnh phần cứng.
- Nội dung 5: Thử nghiệm và điều chỉnh hệ thống. Đánh giá các thông số so với thực tế.
- Nội dung 6: Viết báo cáo thực hiện.
- Nội dung 7: Bảo vệ khóa luận.

### 1.4 GIỚI HẠN

- Sử dụng KIT NodeMCU ESP8266, cảm biến DHT11.
- Tập trung vào thiết kế hệ thống trung tâm.
- Dùng app để giám sát thiết bị hiển thị trên điện thoại.

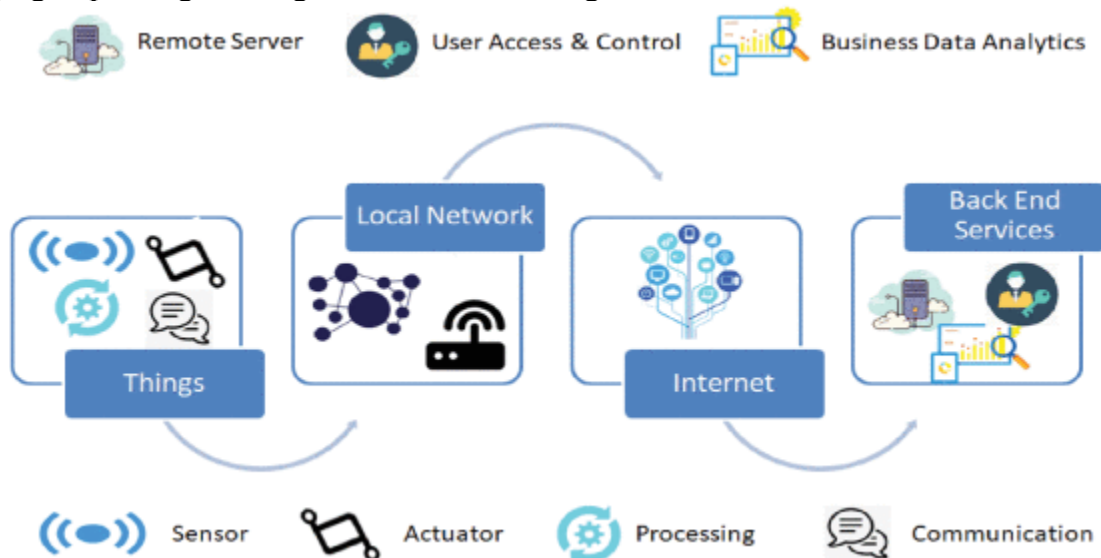
### 1.5 PHẠM VI ỨNG DỤNG

Đề tài là mô hình thu nhỏ, nhưng vẫn có thể áp dụng trên các vườn nhỏ.

## Chương 2: CƠ SỞ LÝ THUYẾT

### 2.1 TỔNG QUAN VỀ IoT

Internet of things là sự kết nối giữa các thiết bị vật lý khác nhau để thu thập, kiểm soát, phân tích và chia sẻ dữ liệu trong thời gian thực. Từ đó nâng cao chất lượng cuộc sống con người. IoT có tác động hầu hết các ứng dụng như chăm sóc sức khỏe, nhà thông minh, nông nghiệp thông minh, giám sát cơ sở hạ tầng,...



Hình 2.1: Kiến trúc IoT.

#### Các thành phần chính trong một hệ thống IoT:

- **Thiết bị:** Là cảm biến vật lý và thiết bị truyền động. Cảm biến đo các thông số khác nhau và chuyển chúng thành dữ liệu điện hoặc dữ liệu kỹ thuật số. Các cảm biến này được kết nối với thiết bị chủ hoặc được tích hợp vào thiết bị chủ.
- **IoT- Platform:** Là bộ điều phối của toàn bộ giải pháp IoT và thường được lưu trữ trên đám mây. Khối này chịu trách nhiệm giao tiếp với các thiết bị và tiêu thụ một lượng lớn tốc độ dữ liệu.
- **Gateway:** Là thiết bị biên có thể giao tiếp với hệ thống. Thường là các mô-đun giao tiếp tích hợp như Wi-Fi hoặc GSM.
- **Ứng dụng:** Giao diện người dùng.

#### Ứng dụng thực tế của IoT:

- **Nhà thông minh:** Là ứng dụng phổ biến và quan trọng trong đời sống hiện nay. Cứ mỗi tháng thì số lượng người tìm kiếm về khái niệm này tăng khoảng 60.000 người. Các công ty lớn đang đầu tư rất nhiều vào sản phẩm về nhà thông minh. Hứa hẹn sẽ có cuộc bức phá trong dự án nhà thông minh.



*Hình 2.2: Smart home.*

- **Sản phẩm có thể đeo được:** Là chủ đề nóng trong các ứng IoT tiềm năng. Được người tiêu dùng săn đón. Các hãng công nghệ lớn như Samsung, Redmi,... đang nghiên cứu và dần hoàn thiện sản phẩm của mình một cách tốt nhất.



*Hình 2.3: Smart watch.*

- **Thành phố thông minh:** Cần một sự đổi mới lớn và một sự phối hợp giữa các thiết bị rất lớn. Từ phân phối nước đến quản lý giao thông, giám sát môi trường, an ninh đô thị. Đây là khó khăn của ứng dụng này.



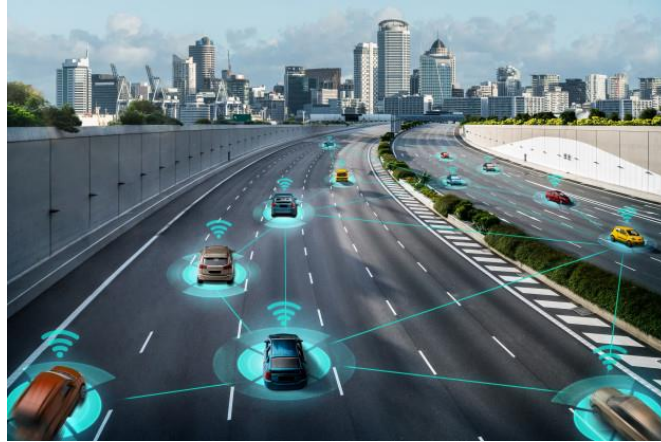
*Hình 2.4: Smart city.*

- **Internet công nghiệp:** Là kết nối các máy và thiết bị trong ngành lại với nhau. Nó cũng được dùng trong các tình huống mà thời gian máy ngừng dẫn đến lỗi hệ thống có thể dẫn đến các tình huống đe dọa. Lúc này là lúc cần những thiết bị tự động.



*Hình 2.5: Internet trong công nghiệp.*

- **Xe được kết nối:** Là một hệ thống gồm nhiều cảm biến, ăng-ten, phần mềm nhúng và công nghệ hỗ trợ giao tiếp đến người dùng. Nó đóng vai trò rất quan trọng. Tính tin cậy cao khi đưa vào sử dụng để đảm bảo tính an toàn với người dùng.



*Hình 2.6: Smart car.*

- **Sức khỏe được kết nối (Telemedicine):** Là một ứng dụng quan trọng vì lĩnh vực y tế sức khỏe là một vấn đề người dùng rất quan tâm. Nó bao gồm các thiết bị điều khiển từ xa, bộ cảm ứng tiên tiến để tích hợp vào thiết bị. Thiết bị sẽ thay thế một số công việc của bác sĩ trong thời gian dài và liên tục tương tác với bác sĩ của họ. Nó giải quyết được vấn đề nhân lực ở các nước phát triển. Từ cảm biến thể dục cá nhân đến robot phẫu thuật.



*Hình 2.7: Telemedicine.*

- **Nông nghiệp thông minh:** Là một ứng dụng tiềm năng. Với sự gia tăng dân số liên tục nên nhu cầu về lương thực tăng lên. Nông dân được học tập thêm các công nghệ về sinh học mới để nâng cao năng suất, đồng thời kết hợp với công nghệ mới để tăng sản lượng phù hợp với dân số. Ứng dụng giúp giải quyết hầu hết các vấn đề khó khăn mà người nông dân nước ta đang gặp phải.



*Hình 2.8: IoT trong nông nghiệp.*

Như các ví dụ về ứng dụng ở trên ta dễ dàng thấy được những lợi ích mà IoT mang lại cho người dùng. Nó cải thiện nâng cao đáng kể mức sống của người dân. Nếu một quốc gia sở hữu các công nghệ một cách hoàn thiện quốc gia đó sẽ phát triển với tốc độ cực kỳ nhanh.

## 2.2 CÔNG NGHỆ WI-FI

### 2.3.1 Khái niệm Wi-Fi

Wi-Fi (Wireless Fidelity) là mạng kết nối internet không dây, sử dụng sóng vô tuyến để truyền tín hiệu. Hầu hết các thiết bị điện tử ngày nay như máy tính, laptop, điện thoại, máy tính bảng,... đều có thể kết nối Wi-Fi.

Sóng Wi-Fi hiện nay sử dụng chuẩn kết nối 802.11 đi kèm một hoặc trong thư viện IEEE là chữ viết tắt của Institute of Electrical and Electronic Engineers, tạm dịch là Hiệp hội các kỹ sư Điện và Điện Tử, cơ quan có trách nhiệm phê chuẩn cấu hình cũng như thúc đẩy sự phát triển của Wi-Fi. Các chuẩn mạng Wi-Fi được sử dụng rộng rãi bao gồm:

- **Chuẩn Wi-Fi 802.11:** Được IEEE giới thiệu trên thị trường vào năm 1997, là Wi-Fi thế hệ thứ nhất. Chuẩn hỗ trợ băng thông mạng cực đại 2Mbps, tốc độ rất chậm đối với hầu hết các ứng dụng.
- **Chuẩn Wi-Fi 802.11b:** Là chuẩn Wi-Fi 802.11 được nâng cấp năm 1999, là Wi-Fi thế hệ thứ hai. Chuẩn dùng băng thông lên đến 11Mbps ở dải tần 2.4 GHz, tương đương với Ethernet truyền thống.
- **Chuẩn Wi-Fi 802.11a:** Được ra mắt cùng năm với chuẩn Wi-Fi 802.11b. Chuẩn dùng băng thông 54Mbps ở dải tần 5GHz cao hơn chuẩn Wi-Fi 802.11b nên nó thường được dùng trong mạng doanh nghiệp.
- **Chuẩn Wi-Fi 802.11g:** Wi-Fi thế hệ thứ ba, được hoàn thành năm 2003. Chuẩn này là sự kết hợp ưu điểm giữa chuẩn Wi-Fi 802.11b và chuẩn Wi-Fi 802.11a nên có băng thông 54Mbps ở dải tần 2.4GHz.



- **Chuẩn Wi-Fi 802.11n:** Wi-Fi thế hệ thứ tư, được giới thiệu trên thị trường năm 2009. Tốc độ tối đa 600Mbps. Chuẩn này có thể hoạt động trên cả hai băng tần 2,4GHz và 5GHz và nếu router hỗ trợ thì hai băng tần này có thể được phát song song nhau.
- **Chuẩn Wi-Fi 802.11ac:** Tốc độ tối đa hiện là 1730Mbps (sẽ còn tăng tiếp) và chỉ chạy ở băng tần 5GHz. Một số mức tốc độ thấp hơn (ứng với số luồng truyền dữ liệu thấp hơn) bao gồm 450Mbps và 900Mbps.

## 2.3.2 Các đặc điểm của Wi-Fi

### 2.2.2.1 Ưu điểm của Wi-Fi

**Sự tiện lợi:** Hệ thống đơn giản gọn nhẹ so với dùng cable truyền thống. Người dùng có thể truy cập ở bất cứ vị trí nào trong vùng phủ sóng mà tại đó Router làm trung tâm.

**Dễ dàng sửa đổi và nâng cấp:** người sử dụng có thể tăng băng thông truy cập, tăng số lượng người sử dụng mà không cần nâng cấp thêm Router hay dây cáp như các kết nối bằng dây vật lý.

**Tính thuận tiện:** người truy cập có thể duy trì kết nối kể cả khi đang di chuyển, một ví dụ cụ thể là các Router Wi-Fi đặc lắp trên các xe khách đường dài. Bên cạnh đó, tính bảo mật của mạng Wi-Fi tương đối cao Repeater hay access point, dẫn đến chi phí gia tăng.

### 2.2.2.2 Khuyết điểm

**Bảo mật:** Môi trường kết nối không dây nên người dùng rất dễ bị tấn công.

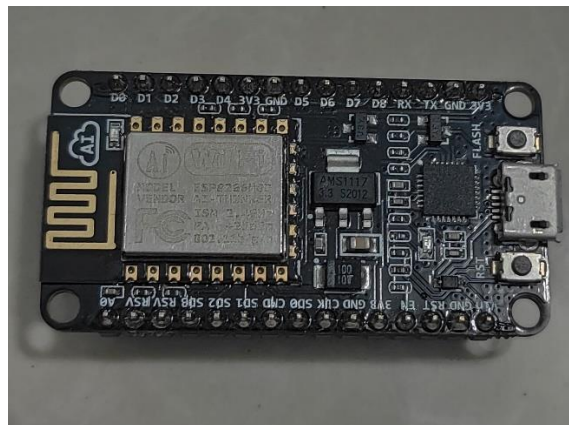
**Phạm vi:** Phạm vi truyền của Wi-Fi là có giới hạn trong vài chục mét. Nó phù hợp cho một căn nhà, nhưng đối với một doanh nghiệp lớn thì không thể. Để khắc phục vấn đề thì người dùng phải mua thêm

**Độ tin cậy:** Vì dùng sóng vô tuyến để truyền nên việc bị nhiễu, tín hiệu bị giảm tác động của các thiết bị khác là không thể tránh khỏi.

## 2.3 PHẦN CỨNG

### 2.3.1 Giới thiệu về ESP8266 NodeMCU

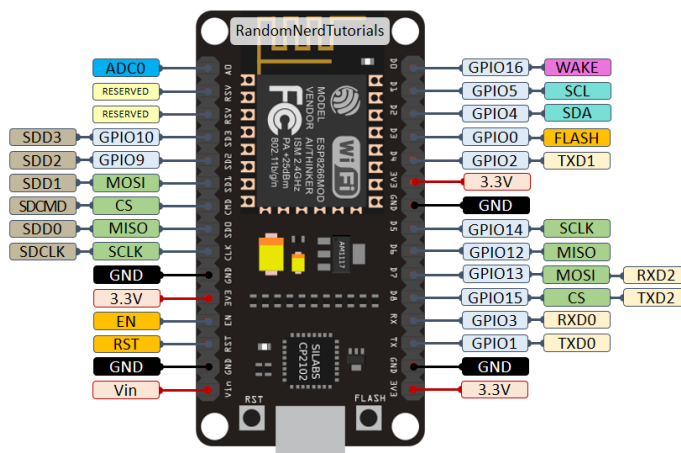
ESP 8266 là tên của vi điều khiển được thiết kế bởi Espressif Systems. Espressif là công ty Trung quốc có trụ sở ở Thượng Hải. ESP8266 được giới thiệu là giải pháp, thiết bị cầu nối của vi điều khiển và Wi-Fi . Nó cũng có thể chạy độc lập trên ứng dụng nạp sẵn.



Hình 2.9: ESP 8266.

ESP 8266 tích hợp với Wi-Fi 2.4GHz và rất ít linh kiện đi cùng. Với chi phí rẻ so với tính năng và khả năng ESP8266 làm được. ESP8266 được xem là rất tiềm năng và hiện được các nhà phát triển trên thế giới tiếp cận và phát triển.

### 2.3.1.1 Cấu tạo ESP8266



Hình 2.10: Sơ đồ chân ESP8266.

Module ESP8266 có các chân dùng để cấp nguồn và thực hiện kết nối. Chức năng các chân như sau:

- VCC: 3.3V.
- GND: Chân nối đất.
- Tx: Chân Tx của giao thức UART .
- Rx: Chân Rx của giao thức UART.
- RST: Chân reset, kéo xuống mass để reset.
- 10 chân GPIO từ D0-D8, có chức năng PWM, IIC, giao tiếp SPI, 1-Wire và ADC trên chân A0.
- Kết nối mạng Wi-Fi (có thể sử dụng như điểm truy cập vào trạm máy chủ lưu trữ, máy chủ Web), kết nối internet để lấy hoặc tải lên dữ liệu.

### 2.3.1.2 Thông số kỹ thuật NODEMCU ESP8266

Thông số kỹ thuật:

- Phiên bản firmware: NodeMCU Lua.
- Chip nạp và giao tiếp UART: CP2102.
- Điện áp hoạt động: 3.3V.
- Điện áp vào thông qua cổng USB: 5V.
- Số chân I/O: 11 chân (các chân I/O đều có Interrupt/PWM/I2C/One-wire, trừ chân D0).
- Tích hợp Led báo trạng thái, nút Reset, Flash.
- Số chân Analog Input: 1 chân (điện áp vào tối đa 3.3V).
- Bộ nhớ Flash: 4MB.
- Tương thích với Arduino IDE.



- Giao tiếp : Cable Micro USB.
- Hỗ trợ bảo mật: WPA/WPA2.
- Tích hợp giao thức: TCP/IP.
- Chuẩn nối tiếp UART.

### 2.3.2 Module DHT11

Trên thị trường hiện nay có rất nhiều loại cảm biến nhiệt độ trên thị trường như LM35, DHT11, Pt100, DHT22... được sáng chế ra với nhiều mục đích khác nhau. Để phù hợp với nhu cầu về đo nhiệt độ và độ ẩm phù hợp với giá cả hợp lý thì DHT11 là một lựa chọn đúng.

DHT11 là cảm biến đo nhiệt độ và độ ẩm có độ chính xác cao và giá thành rẻ. Cảm biến DHT11 có bộ điều chỉnh đầu ra tín hiệu số được hiệu chuẩn qua bộ tiền xử lý tín hiệu tích hợp trong cảm biến giúp đảm bảo độ tin cậy và ổn định của cảm biến. Các kết quả được lưu dưới dạng chương trình trong bộ nhớ OTP, khi cảm biến nhận được tín hiệu truyền, nó sẽ trích dẫn hệ số từ bộ nhớ. Cảm biến được kết nối với bộ vi điều khiển 8 bit tốc độ cao giúp cung cấp chất lượng tốt, phản ứng nhanh, chống nhiễu tốt.

Với kích thước nhỏ và mức tiêu thụ thấp và khoảng cách truyền dài (20m) cho phép DHT11 phù hợp ứng dụng trong môi trường khắc nghiệt.

#### 2.3.2.1 Thông số kỹ thuật Module DHT11

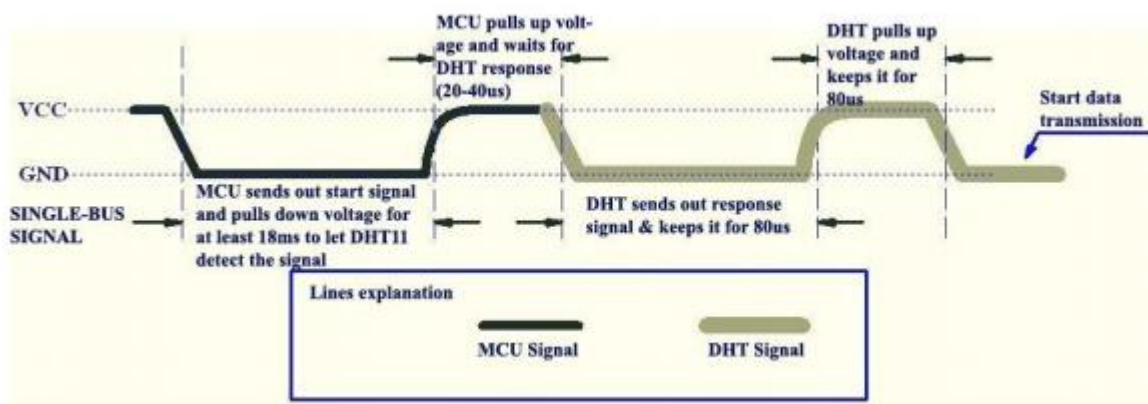
Thông số kỹ thuật DHT11:

Model	DHT11
Điện áp hoạt động	3.3-5V
Dải nhiệt độ hoạt động	0-50°C, sai số $\pm 2^{\circ}\text{C}$
Dải độ ẩm hoạt động	20-80%, sai số $\pm 5\%$
Tần số lấy mẫu	1Hz
Kích thước	15.5mm x 12mm x 5.5mm
Dòng tối đa	2.5mA
Khoảng cách truyền	20m

*Bảng 2. 1: Thông số DHT11.*

Như vậy thông qua các thông số trên thấy được những ưu điểm của DHT11 để dùng đo nhiệt độ và độ ẩm của môi trường.

### 2.3.2.2 Nguyên lí hoạt động



Hình 2.11: Tín hiệu DHT11 vào vi điều khiển.

DHT11 sử dụng một dây để giao tiếp giữa MCU và DHT11. Quá trình giao tiếp được chia làm 3 bước: MCU gửi yêu cầu đến cảm biến, cảm biến sẽ gửi xung phản hồi và gửi dữ liệu đến vi điều khiển.

- **MCU gửi yêu cầu đến cảm biến:** Mức điện áp đầu vào là mức điện áp cao. Khi giao tiếp giữa MCU và DHT11 bắt đầu, chương trình của MCU sẽ chuyển đổi mức điện áp của dây dữ liệu từ mức cao xuống mức thấp và thời gian diễn ra quá trình này khoảng hơn 1ms để đảm bảo DHT11 có thể kết nối với MCU, sau đó mất 20-40µs để phản hồi của DHT11.
- **Cảm biến DHT11 gửi xung phản hồi đến MCU:** Khi Cảm biến nhận được tín hiệu từ MCU, nó sẽ gửi tín hiệu ở mức điện áp thấp và tín hiệu này kéo dài 80µs dưới dạng tín hiệu phản hồi, sau đó chương trình DHT11 chuyển đổi mức điện áp của dây truyền dữ liệu từ mức thấp lên mức cao và 80µs cuối cùng để DHT11 chuẩn bị dữ liệu gửi.
- **Dữ liệu được truyền đến MCU:** Mỗi bit lúc này truyền với mức điện áp với thời gian là 50µs. Tùy thuộc vào tín hiệu ở mức cao mà bit được truyền vô là bit “1” hay “0”.

Đọc giá trị trên DHT11: DHT11 sẽ trả về giá trị 5 byte.

Byte 1: Byte cao của giá trị độ ẩm.

Byte 2: Byte thấp của giá trị độ ẩm.

Byte 3: Byte cao giá trị nhiệt độ.

Byte 4: Byte thấp giá trị nhiệt độ.

Byte 5: Kiểm tra tổng.

Byte 5 = (8 bit)(Byte1 + Byte2 + byte3 + byte4) => Đúng

⇒ Đây là giá trị của nhiệt độ, độ ẩm.

Byte 5 ≠ (8 bit)(Byte1 + Byte2 + byte3 + byte4) => sai

⇒ Giá trị không có nghĩa.

- DHT11 sẽ gửi liên tiếp 40 bit 0 hoặc 1 về MCU, tương ứng chia thành 5 byte .Đó là kết quả đo được.
- Công thức tính giá trị: (Byte cao \*256 + Byte thấp)/10= Số thực

### **2.3.3 Giao tiếp UART**

**UART (Universal Asynchronous Receive/ Transmit)** là chuẩn giao tiếp truyền nhận dữ liệu không đồng bộ. Truyền UART gồm có phần phát và phần thu. Phần máy phát bao gồm ba khối là thanh ghi giữ truyền, thanh ghi dịch chuyển và logic điều khiển. Phần máy thu gồm một thanh ghi giữ, thanh ghi thay đổi và logic điều khiển. Hai lần này thường được kết nối bởi tốc độ baud. Thanh ghi giữ trong máy phát bao gồm byte dữ liệu được truyền. Các thanh ghi thay đổi trong máy phát và máy thu di chuyển các bit sang phải hoặc trái cho đến khi một byte dữ liệu được truyền hoặc nhận.

Tốc độ truyền của vi điều khiển là 115200.

#### **Nguyên lý hoạt động:**

- Việc truyền dữ liệu của UART được thực hiện bằng cách nối dây song song giữa bên thiết bị truyền và thiết bị nhận.
- Start-bit: Bit vào đầu tiên. Bit được đặt trước dữ liệu thực tế. Truyền UART chuyển đường truyền từ mức điện áp cao ở đây là bit 1 về mức điện áp thấp là bit 0.
- Stop-bit: Bit cuối cùng. Dùng để dừng sóng, UART giữ đường dữ liệu ở mức điện áp cao.
- Bit chẵn lẻ: Bit kiểm tra. Đây là hệ thống kiểm tra lỗi cấp thấp . Trên thực tế, bit này không được sử dụng rộng rãi.
- Bit dữ liệu: Bit được truyền từ thiết bị gửi đến thiết bị nhận. Độ dài Khung chứa bit trong khoảng từ 5 đến 8. Nếu có thêm bit chẵn lẻ thì có thể dài đến 9 bit.

## **2.4 HỆ ĐIỀU HÀNH ANDROID**

### **2.3.1 Khái niệm Android**

Android là một hệ điều hành có mã nguồn mở dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google, sau này được chính Google mua lại vào năm 2005. Chiếc điện thoại đầu tiên chạy Android là HTC Dream được bán vào ngày 22 tháng 10 năm 2008.

Chính mã nguồn mở cùng với giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên được điều chỉnh và phân phối Android một cách tự do.

### **2.3.2 Lịch sử phát triển**

Android, Inc. được thành lập tại Palo Alto, California vào tháng 10 năm 2003 bởi Andy Rubin (đồng sáng lập công ty Danger), Rich Miner (đồng sáng lập Tổng công ty

Viễn thông Wildfire), Nick Sears (từng là Phó giám đốc T-Mobile), và Chris White (trưởng thiết kế và giao diện tại WebTV) để phát triển. Là một hệ điều hành cho các thiết bị cầm tay dựa trên lõi Linux. Công ty sau đó được Google mua lại vào năm 2005 và bắt đầu xây dựng Android Platform.

Google đã công bố thiết bị chạy Android đầu tiên của mình ra thị trường vào năm 2007.

Phiên bản SDK lần đầu tiên phát hành vào tháng 11 năm 2007, hãng T-Mobile cũng công bố chiếc điện thoại Android đầu tiên. Tháng 10 năm 2008, Google được cấp giấy phép mã nguồn mở cho Android Platform.

Vào cuối năm 2008, Google cho phát hành một thiết bị cầm tay được gọi là Android Dev Phone 1 có thể chạy được các ứng dụng Android mà không bị ràng buộc vào các nhà cung cấp mạng điện thoại di động.

Hiện nay, hệ điều hành Android trở nên phát triển và phổ biến trên thế giới. Với các nhà phát triển ứng dụng việc Google công bố Android sử dụng mã nguồn mở giúp cho các nhà phát triển dễ dàng tiếp cận và phát triển nó. Tất cả các chương trình ứng dụng được viết bằng ngôn ngữ java kết hợp với XML nên có khả năng chuyển cao.

### 2.3.3 Phiên bản của hệ điều hành

Tính đến thời điểm hiện tại, Android đã trở thành hệ điều hành dành cho nền tảng di động phổ biến nhất thế giới (chiếm đến 85%). Android phổ biến tới mức mặc dù được thiết kế để chạy trên điện thoại và máy tính bảng, nó còn được các nhà sản xuất tùy chỉnh để cài đặt trên các Smart TV, máy chơi game và các thiết bị điện tử khác. Hệ điều hành Android ra mắt đầu tiên năm 2008 và đã trải qua hàng loạt các thay đổi nâng cấp để được như hiện nay:

- **Android 1.0:** Ra mắt ngày 23-11-2008, lúc đầu hệ điều hành rất nguyên sơ nhưng vẫn có các tính năng người dùng cần như thanh thông báo kéo từ trên xuống cho phép người dùng xem nhanh các thông tin ngày giờ, tin nhắn, cuộc gọi... Màn hình chính và Widget: Màn hình chính gồm các biểu tượng chương trình người dùng hay truy cập và các Widget là các ứng dụng nhỏ trên màn hình chính, hoạt động và cung cấp thông tin liên tục. Tích hợp chặt chẽ với Gmail.
- **Android 1.1:** Ra mắt ngày 9-2-2009, là phiên bản cập nhật và sửa các lỗi gặp phải trên phiên bản 1.0. Có thêm tính năng cập nhật phần mềm tự động qua OTA( Over The Air).
- **Android 1.5 Cupcake:** Được ra mắt 30-4-2009, được Google đặt tên theo món ăn. Với phiên bản này Google nâng cấp tính mượt, thẩm mỹ cho giao diện. Xuất hiện bàn phím ảo, hỗ trợ khả năng quay phim cho camera. Thêm chức năng copy, cut, paste.
- **Android 1.6 Donut:** Ra mắt ngày 30-9-2009, Donut khắc phục các chức năng trong Cupcake. Bổ xung các tính năng có thể chạy trên nhiều độ phân giải và tỉ lệ màn hình khác nhau, cho phép những thiết bị có nhiều độ phân giải hơn là 320x480. Tính năng Quick Search Box được phổ biến, người dùng có thể dễ dàng tìm kiếm danh bạ, ứng

- dụng, nhạc,... Android Market với thiết kế mang tông màu xanh trắng đặc trưng của Android, có thể hiển thị các ứng dụng miễn phí và trả phí hàng đầu.
- **Android 2.0 và 2.1 Éclair:** Ra mắt tháng 11-2009, Android 2.0 đã được ra mắt trên chiếc Motorola Droid cùng các tính năng mới. Hỗ trợ nhiều tài khoản người dùng: lần đầu tiên, nhiều tài khoản Google có thể được đăng nhập trên cùng một thiết bị chạy Android. Nó giúp rất nhiều trong việc đồng bộ hóa dữ liệu. Trình duyệt mới: hỗ trợ HTML5. Nâng cấp một số tính năng để hỗ trợ người dùng hiệu quả hơn. Android 2.1 ra mắt vào tháng 1 năm 2010. Ra mắt chủ yếu để sửa lỗi và thêm hàm API để lập trình viên can thiệp sâu hơn vào hệ thống.
  - **Android 2.2 Froyo: Được ra mắt năm 2010.** Nexus One là chiếc điện thoại đầu tiên được nâng cấp lên Android 2.2. Phiên bản 2.2 (Froyo) mang Adobe Flash đến Android. Người dùng có thể xem video clip nền Flash như Youtube. Dãy nút kích hoạt nhanh chế độ gọi điện, web và App Drawer cũng đã xuất hiện.
  - **Android 2.3 Gingerbread:** Ra ngày 6-12-2010, bản nâng cấp thêm vào hệ thống quản lý tập tin, cho phép theo dõi và truy xuất đến các tập tin đã tải về máy. Công cụ quản lý pin và ứng dụng: vì Android cho phép các ứng dụng chạy nền thực thụ chứ không bị ngắt lại, vấn đề pin bị người dùng phàn nàn rất nhiều. Một công cụ mới đã ra mắt với biểu đồ nhằm theo dõi mức độ tiêu thụ pin cùng việc quản lý ứng dụng đang chạy sẽ giúp đỡ người dùng phần nào. Hỗ trợ kết nối NFC.
  - **Android 3.0/3.1 Honeycomb:** Ra mắt ngày 22-2-2011. Đây không chỉ là một phiên bản, mà có thể xem là một thế hệ Android đầu tiên dành riêng cho máy tính bảng (tablet), ra mắt cùng tablet Motorola XOOM. Android 3.0 đặt nền móng quan trọng cho thế hệ Android 4.x hợp nhất, khắc phục sự phân mảnh của Android (có các phiên bản riêng dành cho smartphone và tablet)
  - **Android 4.0 Ice Cream Sandwich :** Ra mắt ngày 19-10-2011. Ice Cream Sandwich là thế hệ Android được mong đợi nhất đến nay, ra đời cùng dòng smartphone "bom tấn" Samsung Galaxy Nexus, thế hệ smartphone đầu tiên trang bị ICS. Android 4.0 thay đổi hoàn toàn độ phân giải màn hình, thay đổi các tính năng để giúp người dùng tốt hơn. Phiên bản cũng nhắm đến việc duyệt web nhanh hơn, tối ưu hóa hiệu suất hoạt động của thiết bị, kéo dài thời lượng pin,...
  - **Android 4.1 Jelly Bean:** Ra mắt 9-7-2012. Ở màn hình Lockscreen, trượt sang phải để mở khóa máy, trượt trái để mở camera, trượt lên trên sẽ thấy chữ "Google". Đây chính là cách thức để kích hoạt Google Now. Người dùng có thể tìm kiếm bằng giọng nói. Không chỉ tìm kiếm theo yêu cầu, Google Now còn tự chủ động thu thập thông tin liên tục dựa vào lịch sử tìm kiếm của bạn (thói quen người dùng), dựa vào danh sách lịch hẹn để biết bạn sẽ đi đâu, cần thông tin gì.

- **Android 4.2 vẫn là Jelly Bean:** Ra mắt tháng 11-2012, chỉ sau gần năm tháng ra mắt Android 4.1, Google tiếp tục bồi thêm sức nặng cho Android với phiên bản 4.2 và vẫn mang tên mã Jelly Bean. Không chỉ tìm kiếm theo yêu cầu, Google Now còn tự chủ động thu thập thông tin liên tục dựa vào lịch sử tìm kiếm của bạn (thói quen người dùng), dựa vào danh sách lịch hẹn để biết bạn sẽ đi đâu, cần thông tin gì.
- **Android 4.3 Jelly Bean :** Ra mắt tháng 7 - 2013, Google đã chính thức ra mắt hệ điều hành này song song với chiếc Nexus 7 (2013). Cùng với tính năng mới như hỗ trợ kết nối Bluetooth Smart, bộ API OpenGL ES 3.0, bổ sung tính năng sử dụng Wi-Fi để định vị ngay cả khi người dùng tắt kết nối này đi cùng nhiều thay đổi lớn nhỏ khác. Google bổ sung thêm một tính năng mới cho Android 4.3 đó là Restricted Profile. Mỗi thành viên sẽ có một “hồ sơ” riêng của mình và chỉ có thể làm được những gì được chỉ định trong “hồ sơ”.
- **Android 4.4 KitKat:** Ra mắt tháng 11 - 2013, Mục tiêu của Google đó là đem Android 4.4 lên mọi thiết bị, kể cả những dòng máy có cấu hình rất thấp đi chăng nữa, đây cũng là dịp họ muốn rũ bỏ cái danh “hệ điều hành ngón cái hình” hàng đầu thế giới. Bổ xung nhiều tính năng và các hàm API mới cho việc phát triển các ứng dụng hiện đại.
- **Android 5.0 Lollipop:** Ra mắt tháng 11 – 2014 đã đánh dấu một bước phát triển vượt bậc của hệ điều hành này. Android trên mọi màn hình: Trên điện thoại, tablet, laptop, thậm chí là TV, ô tô, đồng hồ và đồ điện gia dụng, hệ điều hành Android mới học theo cách tiếp cận của Microsoft khi cố gắng hợp nhất mọi nền tảng về một mái nhà duy nhất mang tên Windows. Hỗ trợ chip 64 bit. Cải thiện thời lượng pin. Mở khóa dựa trên bối cảnh: Mở khóa riêng tư là một ý tưởng thông minh, cho phép người dùng mở khóa thiết bị bằng một smartphone hoặc tablet đang kết nối Bluetooth với thiết bị gốc, bằng vị trí hiện tại của người dùng hay thậm chí là giọng nói của họ.
- **Android 6.0 Marshmallow:** Ra mắt 18-8-2015, cải thiện phân quyền cho ứng dụng một cách dễ dàng hơn. Tối ưu việc dùng pin của các ứng dụng. Giúp người dùng chia sẻ nhanh hơn. Người dùng chủ động xem được dung lượng Ram tiêu thụ của máy mà không cần phần mềm hỗ trợ.
- **Android 7.0 Nougat:** Được phát hành 9-3-2016, hệ điều hành có thể thực hiện chế độ đa cửa sổ, có nhiều thông báo hữu ích cho người dùng, tiện lợi trong tùy chọn cài đặt dễ dàng và nhanh hơn, các tính năng ẩn của System UI Tuner.
- **Android 8.0 Oreo:** Phát hành chính thức vào năm 2017, phiên bản được nâng cấp từ phiên bản trước: Trải nghiệm người dùng, độ an toàn bảo mật, nền tảng xây dựng. Chi tiết hơn là Google đặt giới hạn ứng dụng chạy nền giúp cho các nhà phát triển ứng dụng tạo nên ứng dụng ít ảnh hưởng xấu đến thiết bị và tiết kiệm pin.
- **Android 9.0 Pie:** Ra mắt năm 2018, hệ điều hành có những cải tiến hơn so với phiên bản trước là: Thao tác điều hướng bằng cử chỉ, ra mắt tính năng Adaptive Battery để tối

ưu hóa tuổi thọ pin điện thoại. Nâng cấp bảo mật và hạn chế ứng dụng chạy nền trên thiết bị.

- **Android 10.0 Q:** Ra mắt vào ngày 3-9-2019, Google nâng cấp tương tác với người dùng khi ra mắt chế độ tối toàn diện. Nâng cấp trợ lý với việc tương tác linh hoạt hơn. Hệ điều hành hỗ trợ tốt cho màn hình gập và HDR 10+ .
- **Android 11.0 R:** Ra mắt vào 11-06-2020, Google thêm một thanh đề xuất ứng dụng cho người dùng. Hỗ trợ quay video màn hình mà không cần ứng dụng hỗ trợ. Công bố một tính năng mới là điều khiển nhà thông minh cụ thể là các thiết bị gia dụng thông minh của Google. Tăng cường tính bảo mật cho thiết bị hơn. Đặc biệt hệ điều hành tối ưu và hỗ trợ cho mạng 5G.

### 2.3.4 Ưu khuyết điểm của hệ điều hành Android

#### Ưu điểm:

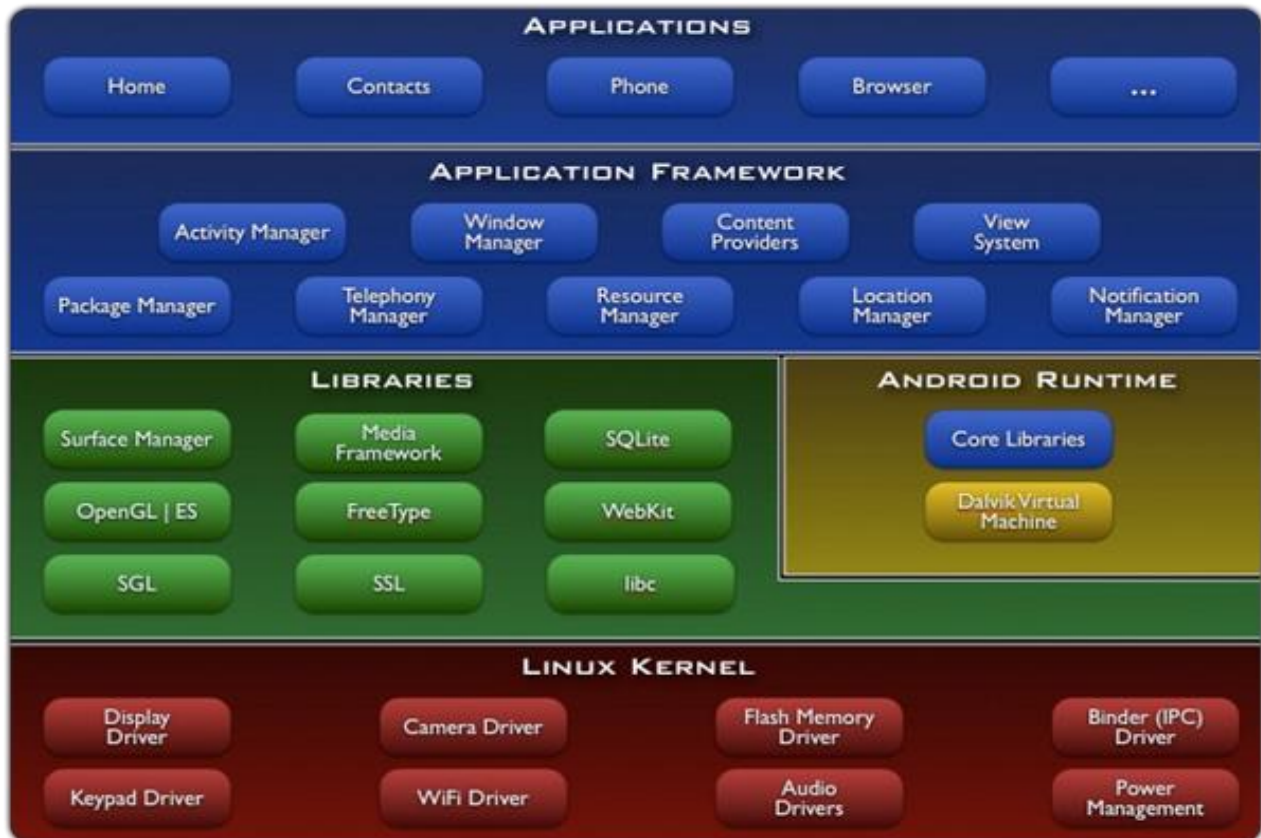
- + Là hệ điều hành có mã nguồn mở nên khả năng tùy biến cao, có thể tùy ý chỉnh sửa mà không có sự can thiệp hay cấm cản từ Google.
- + Đa dạng sản phẩm, rất nhiều hãng điện thoại, thiết bị công nghệ đã ưu ái chọn Android cho thiết bị của họ, giá cả thì hợp lý từ bình dân đến cao cấp.
- + Kho ứng dụng Google Play Store đồ sộ.
- + Thân thiện và dễ sử dụng.
- + Khả năng đa nhiệm, chạy cùng lúc nhiều ứng dụng cao.

#### Nhược điểm:

- + Dễ nhiễm phần mềm độc hại và virus. Do tính chất mã nguồn mở, nhiều phần mềm không được kiểm soát có chất lượng không tốt hoặc lỗi bảo mật vẫn được sử dụng.
- + Kho ứng dụng quá nhiều dẫn đến khó kiểm soát chất lượng, thiếu các ứng dụng thật sự tốt.
- + Sự phân mảnh lớn.
- + Cập nhật không tự động với tất cả thiết bị. Khi một phiên bản hệ điều hành mới ra mắt, không phải tất cả sản phẩm đều được cập nhật, thậm chí nếu muốn trải nghiệm bạn thường xuyên phải mua mới thiết bị.

### 2.3.5 Kiến trúc hệ điều hành

Kiến trúc hệ điều hành Android gồm 4 tầng, từ dưới lên trên là tầng hạt nhân Linux (v2.6), tầng Libraries & Android Runtime , tầng Application Framework và trên cùng là tầng Application. Mô hình tổng thể của các thành phần bên trong hệ thống sử dụng Android được minh họa .



Hình 2.12: Kiến trúc hệ điều hành Android.

#### 2.3.4.1 Tầng hạt nhân Linux (Linux Kernel)

Đây chính là nền tảng của Android, hệ điều hành được phát triển dựa trên nhân của hệ điều hành Linux. Việc sử dụng nhân Linux cho phép Android có nhiều đặc tính bảo mật và cho phép các nhà sản xuất thiết bị phát triển các bộ điều khiển (driver) cho thiết bị tốt nhất.

Tầng này có các thành phần chủ yếu:

- Display Driver: Điều khiển việc hiển thị lên màn hình cũng như thu nhận những điều khiển của người dùng lên màn hình (di chuyển, cảm ứng...).
- Camera Driver: Điều khiển hoạt động của camera, nhận luồng dữ liệu từ camera trả về.
- Bluetooth Driver: Điều khiển thiết bị phát và thu sóng Bluetooth.
- USB driver: Quản lý hoạt động của các cổng giao tiếp USB
- Keypad driver: Điều khiển bàn phím.
- Wi-Fi Driver: Chịu trách nhiệm về việc thu phát sóng Wi-Fi.



- Audio Driver: điều khiển các bộ thu phát âm thanh, giải mã các tín hiệu dạng audio thành tín hiệu số và ngược lại.
- Binder IPC Driver : Chịu trách nhiệm về việc kết nối và liên lạc với mạng vô tuyến như CDMA, GSM, 3G, 4G, E để đảm bảo những chức năng truyền thông được thực hiện.
- M-System Driver: Quản lý việc đọc ghi... lên các thiết bị nhớ như thẻ SD, flash.
- Power Management: Giám sát việc tiêu thụ điện năng.

#### 2.3.4.2 Tầng Library và Runtime

##### **Tầng Library:**

- Nhiều thành phần hệ thống nhân Android và các dịch vụ như ART và HAL như nêu ở trên được xây dựng từ native code, những dịch vụ này được viết bằng ngôn ngữ C/C++. Android cung cấp các Java Framework API để giúp ứng dụng viết bằng Java có thể sử dụng được những thư viện/ dịch vụ Native này.
- Nếu viết một ứng dụng bằng ngôn ngữ C/C++ bạn có thể sử dụng Android NDK thay vì Android SDK để truy cập một số thư viện native này một cách trực tiếp.

**Tầng Runtime:** Với những thiết bị chạy hệ điều hành Android phiên bản 5.0 trở lên, thì mỗi ứng dụng sẽ chạy riêng một tiến trình với một instance (cụ thể) của Android Runtime. ART được viết để chạy được trên nhiều máy ảo ở cấp độ thấp bằng việc thực thi các file DEX. ART sẽ giúp các ứng dụng trên Android chạy nhanh hơn ( chúng ta sẽ thấy rõ điều này khi sử dụng Android 5.0 trở lên và Android phiên bản nhỏ hơn, cùng cấu hình phần cứng ). Đặc biệt khi bạn viết các ứng dụng yêu cầu nhiều hỗ trợ từ phần cứng như xử lý ảnh cần bộ nhớ, camera, tốc độ CPU thì bạn sẽ thấy sự khác biệt đáng kể cỡ nào.

#### 2.3.4.3 Tầng Application Framework

Lớp Android Framework cung cấp các dịch vụ cấp độ cao hơn cho các ứng dụng dưới dạng các lớp Java. Các nhà phát triển ứng dụng được phép sử dụng các dịch vụ này trong ứng dụng của họ.

Android Framework bao gồm các dịch vụ chính sau:

- Activity Manager - Kiểm soát tất cả khía cạnh của vòng đời ứng dụng và ngăn xếp các Activity.
- Content Providers - Cho phép các ứng dụng chia sẻ dữ liệu với các ứng dụng khác.
- Resource Manager - Cung cấp quyền truy cập vào các tài nguyên như các chuỗi, màu sắc, các layout giao diện người dùng...
- Notifications Manager - Cho phép các ứng dụng hiển thị cảnh báo và các thông báo cho người dùng.
- View System - Tập các thành phần giao diện (view) được sử dụng để tạo giao diện người dùng.



- Nếu sự hiển thị của Activity khác làm cho Activity mà chúng ta đang nói đến không còn nhìn thấy nữa thì onStop() sẽ được gọi ngay sau đó. Nếu Activity đã vào onPause() rồi, tức là đang bị Activity khác đè lên, mà người dùng sau đó quay về lại Activity cũ, thì onResume() được gọi.
- Còn nếu Activity đã vào onStop() rồi, mà người dùng quay về lại Activity cũ thì onStart() được gọi.
- Trong cả hai trường hợp Activity rơi vào onPause() hoặc onStop(), nó sẽ rất dễ bị hệ thống thu hồi (tức là bị hủy) để giải phóng tài nguyên, khi này nếu quay lại Activity cũ, onCreate() sẽ được gọi chứ không phải onResume() hay onStart().
- Cuối cùng, nếu một Activity bị hủy một cách có chủ đích, chẳng hạn như người dùng nhấn nút Back ở System Bar, hay hàm finish() được gọi,... thì onDestroy() sẽ được kích hoạt và Activity kết thúc vòng đời của nó.

## **2.5 GOOGLE FIREBASE**

### **2.5.1 Giới thiệu về Firebase**



*Hình 2.14: Trang chủ Firebase.*

Firebase là platform do Google cung cấp và hoạt động trên nền tảng đám mây, nhằm hỗ trợ việc tạo ra các web application, mobile application với chất lượng cao.

### **2.5.2 Lịch sử của Firebase**

Năm 2011, Firebase ra đời với tên gọi là Envolv bởi James Tamplin và Andrew Lee. Mục đích Envolv cung cấp cho developer API là để tích hợp chức năng trò chuyện trực tuyến vào trang web. Năm 2014, Google mua lại Firebase. Sau đó, Firebase nhanh chóng phát triển thành ứng dụng đa năng của nền tảng di động và web hiện nay.

### **2.5.3 Chức năng của Firebase**

Firebase Analytics là dùng để phân tích sự tương tác của người dùng với ứng dụng. Kiểm tra được tình trạng ứng dụng hoạt động.

Firebase Hosting hỗ trợ các nhà phát triển tạo lập ứng dụng một cách đơn giản.

Firebase Cloud Messaging cung cấp việc tạo ra điểm nhận diện khi có thông báo đối tượng mong muốn gửi tin.

Firebase Authentication là tính năng xác thực người dùng bằng password, số điện thoại, tài khoản Google,...

Firebase Cloud Storage quản lý, chia sẻ hình, video mà người dùng tải lên và sử dụng tài nguyên đó.

Firebase Remote Config là tính năng giúp cho ứng dụng luôn được cập nhật một cách tự động.

Firebase Test Lab là chức năng để kiểm tra hoạt động của ứng dụng trên Cloud, trên những thiết bị khác.

Firebase Console là chức năng dùng để điều khiển và cấp quyền cho các chức năng khác.

#### **2.5.4 Thành phần của Firebase**

Firebase Realtime Database:

- Người dùng sẽ được cung cấp một hệ cơ sở dữ liệu dưới dạng Json, các client đều sử dụng chung một cơ sở dữ liệu. Dữ liệu sẽ được lưu lại local nếu không kết nối mạng và sẽ được cập nhật lại trên sever khi có kết nối mạng.
- Hỗ trợ: IOS, Android, Web, C++ , Unity.

Firebase Authentication: Dùng để xác thực thông tin tài khoản người dùng bằng Email, Facebook, Google, GitHub.

Firebase Hosting: Dùng để lưu trữ cơ sở dữ liệu đa năng và có khả năng bảo mật dữ liệu tốt.

## CHƯƠNG 3: TÍNH TOÁN THIẾT KẾ

### 3.1 GIỚI THIỆU VỀ ĐỀ TÀI

Đề tài này yêu cầu thiết kế hệ thống giám sát nhiệt độ, độ ẩm. Khi tương tác với App trên thiết bị android. Dữ liệu được gửi đến vi xử lý, và vi xử lý sẽ thực hiện các lệnh.

- Khi module Wi-Fi ESP8266 gửi giá trị nhiệt độ, độ ẩm thì giá trị hiển thị ra dạng biểu đồ đường cho người dùng giám sát trên giao diện người dùng.
- Thiết kế các relay có thể điều khiển được thiết bị thông qua tương tác các nút ấn thông qua app android trên smartphone.
- Thiết kế bộ quản lý đặt hẹn giờ và ngưỡng giá trị cho thiết bị thông qua giao diện người dùng. App sẽ lưu trữ giá trị đã đặt và truyền giá trị lên Firebase thông qua môi trường không dây ở đây là Wi-Fi, sau đó được truyền xuống vi xử lý để thực hiện.

Bộ xử lý trung tâm:

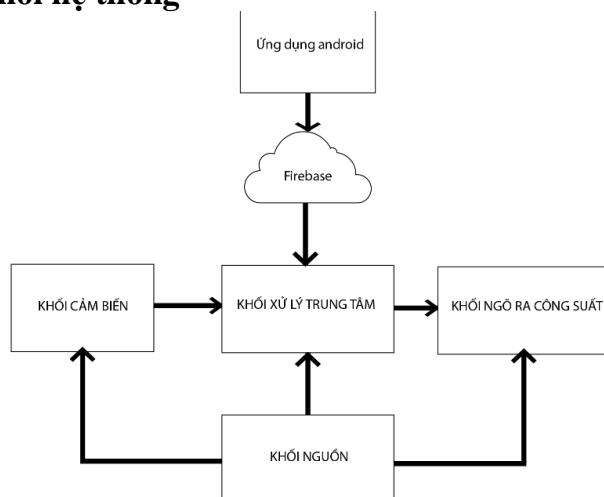
- Điện áp 5VDC.
- Giao tiếp ứng dụng Android.
- Giao tiếp với Firebase bằng Wi-Fi.
- Thiết kế nhỏ gọn, đảm bảo tính an toàn.

Ứng dụng Android

- Giao diện đăng nhập người dùng.
- Giám sát nhiệt độ, độ ẩm liên tục tại các thời điểm trong ngày.
- Điều khiển được nhiều thiết bị cùng một lúc.
- Chức năng phân tích hỗ trợ người dùng.

### 3.2 THIẾT KẾ SƠ ĐỒ HỆ THỐNG

#### 3.2.1 Thiết kế sơ đồ khối hệ thống



Hình 3.15: Sơ đồ khối hệ thống.

**Khối xử lý trung tâm:** Đây là khối trung tâm của toàn bộ hệ thống có nhiệm vụ điều khiển. Khối nhận tín hiệu từ ứng dụng Android, sau đó xử lý tín hiệu và chuyển nó đến khối

**Khối nguồn:** Nguồn cung cấp cho toàn mạch. Khối nguồn ở đây dùng 2 nguồn là 12VDC cung cấp cho khối xử lý trung tâm, 5VDC cấp cho mạch Relay, cảm biến, khối xử lý trung tâm.

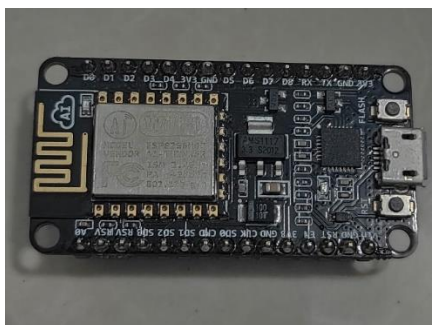
**Khối cảm biến:** Khối có nhiệm vụ giám sát nhiệt độ, độ ẩm của môi trường để đảm bảo an toàn cho hệ thống.

**Ứng dụng Android:** nhận tín hiệu từ khối xử lý trung tâm, sau đó xử lý tín hiệu đó với người dùng. Người dùng nhận và gửi tín hiệu yêu cầu đến khối xử lý trung tâm.

**Khối ngõ ra công suất:** Nhận tín hiệu từ khối xử lý và thực hiện đóng ngắt các tiếp điểm Relay, dựa vào đó để điều khiển các thiết bị.

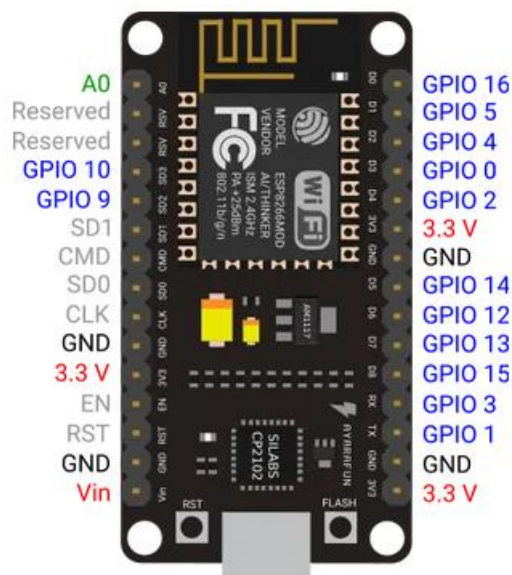
### 3.2.2 Tính toán và thiết kế

#### 3.2.2.1 Khối xử lý trung tâm



Hình 3.2: Ảnh ESP8266 thực tế.

Nguồn Board được cấp ở đây là 5VDC.



Hình 3.3: Sơ đồ chân ESP8266.

Thông số kỹ thuật:

Nguyên lý nối chân ESP8266:

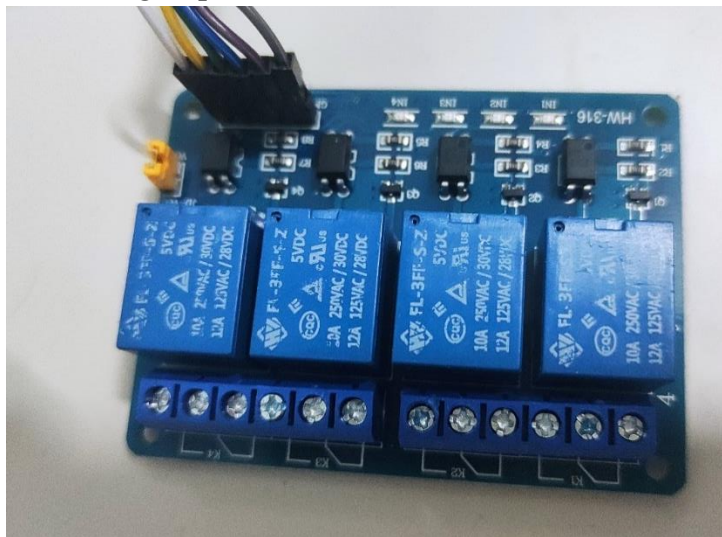
- Chân nguồn Vin và GND được nối với mạch nguồn cung cấp 5VDC.
- Các chân GPIO D0, D1, D2, D3 lần lượt nối với các ngõ relay tương ứng.
- Chân D6 nối với cảm biến DHT11.

### 3.2.2.2 Khối ngõ ra công suất

Với yêu cầu bài toán đặt ra phải đảm bảo tính an toàn khi sử dụng các thiết bị điện, cần cách ly mạch điều khiển với các thiết bị điện. Để giải quyết các vấn đề đó em chọn mạch Relay 4 kênh có opto cách ly.

Relay là một thiết bị chuyển mạch bằng điện. Dòng điện chạy qua cuộn dây của rơ-le tạo ra một từ trường hút lõi sắt non làm thay đổi công tắc chuyển mạch. Dòng điện qua cuộn dây có thể được bật hoặc tắt, vì thế rơ-le có hai vị trí chuyển mạch qua lại. Relay thường được sử dụng ở các mạch tự động.

Mạch cấp nguồn 5VDC và dòng vào 80mA cho các relay là tiếp điểm có thể đóng ngắt khi được kích. Nguồn ra tối đa ở đây là 12VDC an toàn do có opto cách ly. Dùng để chuyển điện thành tải trong nhiều trường hợp.



Hình 3.4: Mạch Module trạng thái rắn chuyển tiếp 4 kênh.

Relay SSR có cấu tạo gồm Diot phát quang, và bộ Tri-ac.

Model	4 kênh Relay rắn SSR
Điện áp kích hoạt	5VDC
Kích thước	76x56mm
Dòng tiêu thụ	80mA/1 relay
Dòng AC tối đa	250 VAC
Dòng DC tối đa	30VDC
Số chân	6 chân
Mạch cách ly	opto
Led báo trạng thái	4 led

Bảng 3. 1: Thông số kỹ thuật Module 4 Relay rắn SSR.

Relay 4 kênh 5V gồm 4 rơ le hoạt động tại điện áp 5VDC, chịu được hiệu điện thế lên tới 250V- 10A và được cách ly nên an toàn. Relay 4 kênh 5V được thiết kế chắc chắn, khả năng cách điện tốt. Trên module có transistor dùng để kích relay. Có sẵn header rất tiện lợi khi kết nối với vi điều khiển. Module hoạt động với 2 mức:

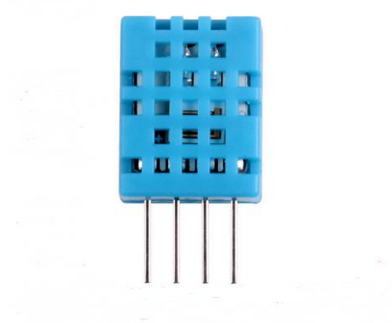
- Tín hiệu kích mức thấp (0VDC) thì lúc đó relay đóng.
- Tín hiệu kích mức cao (5VDC) thì lúc đó relay bật.

Module được kết nối với các board điều khiển bằng 6 chân header như sau:

- VCC cấp nguồn opto.
- GND nối GND của board điều khiển.
- IN1, IN2, IN3, IN 4 dùng để điều khiển relay 1, relay 2, relay 3, relay 4 tích cực mức thấp.

### 3.2.2.3 Khối cảm biến

Khối cảm biến sử dụng cảm biến nhiệt độ ẩm hiển thị lên app android. DHT 22 là lựa chọn tốt nhất trong trường hợp này. DHT11 là cảm biến đo nhiệt độ, độ ẩm với độ chính xác cao, chi phí rẻ. Mức điện áp cảm biến sử dụng là 3-6VDC, dòng cung cấp 2.5mA phù hợp với bộ xử lý.

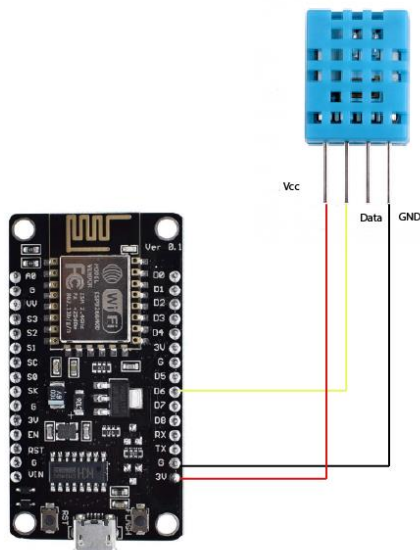


*Hình 3.5: Cảm biến DHT11.*

Cảm biến DHT11 gồm 4 chân được nối như sau:

- Chân VCC được nối với nguồn 5VDC.
- Chân Data được nối với chân D6 của ESP8266.
- Chân GND được nối với chân GPOP của vi điều khiển (ESP8266).



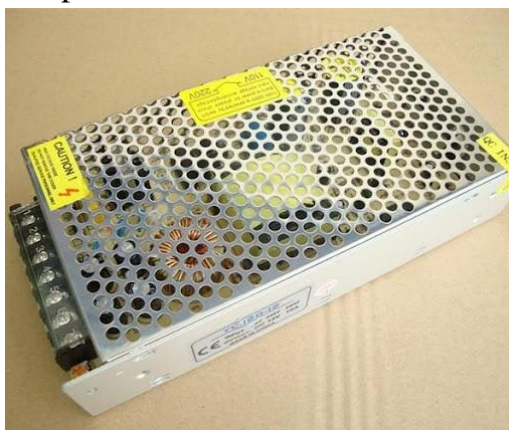


Hình 3.6: Sơ đồ kết nối cảm biến DHT11 với ESP8266.

#### 3.2.2.4 Khối nguồn

Khối nguồn sử dụng trong mạch là nguồn 5VDC. Nên em chọn nguồn tổ ong có điện áp đầu ra là 12V DC và mạch hạ áp XL 4015 dùng để hạ áp từ 12V xuống 5V.

Nguồn tổ ong hay gọi là nguồn xung. Nguồn được dùng để biến đổi dòng điện xoay chiều (220 VAC) sang nguồn điện một chiều. Nguồn điện công suất lớn sẽ có hai cổng vào 220VAC cho pha L-N, 3 cổng 0V, 3 cổng điện áp (+), cổng đất cho thiết bị. Nguồn tổ ong có kích thước vừa phải là lựa chọn tốt nhất vì nguồn tuyến tính cỡ điện có kích thước cồng kềnh và giá thành cao không phù hợp với đề tài.



Hình 3.7: Ảnh thực tế của nguồn tổ ong.

Cấu tạo của nguồn tổ ong:

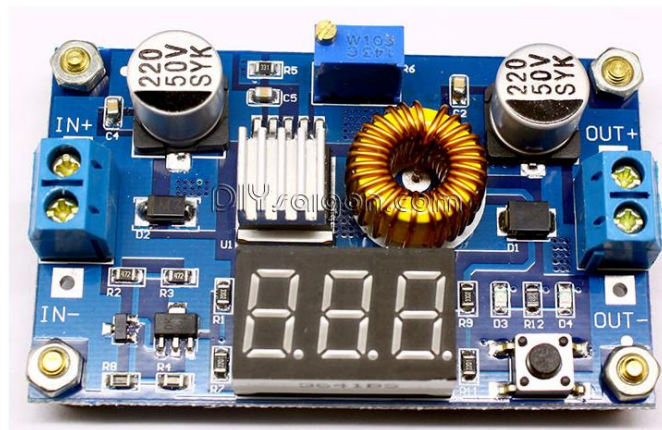
- Cầu chì: Linh kiện có tác dụng bảo vệ mạch nguồn bị ngắn mạch.

- Cuộn chống nhiễu, tụ lọc sơ cấp, diot chỉnh lưu: Chuyển dòng xoay chiều 220VAC thành điện áp 1 chiều. Sau đó tích điện trên tụ lọc sơ cấp để cung cấp năng lượng cho cuộn sơ cấp của máy biến áp xung.
- Sò công suất: Phần bán dẫn dùng như một công tắc chuyển mạch trong bộ xung.
- Tụ lọc nguồn thứ cấp: Dùng để trữ năng lượng điện từ cuộn thứ cấp của biến áp xung để cấp cho tải tiêu thụ. Khi cuộn sơ cấp được đóng cắt điện liên tục bằng sò. Chúng ta đều biết sẽ xuất hiện trường biến thiên dẫn đến cuộn thứ cấp cũng xuất hiện một điện áp ra. Điện áp sẽ được chỉnh lưu qua các điốt rồi đưa vào tụ lọc thứ cấp để san phẳng điện áp.
- IC quang và IC TL431: hai linh kiện này có nhiệm vụ tạo ra điện áp ổn định để khống chế điện áp ra bên thứ cấp ổn định theo mong muốn người dùng. Chúng cần phải khống chế dao động đóng cắt điện vào cuộn sơ cấp để điện áp ra bên thứ cấp đạt yêu cầu.

**Thông số kĩ thuật:**

Model	Nguồn tổ ong 12V – 30A
Điện áp ra	12V
Cường độ dòng điện	30A
Điện áp vào	90-120VAC/ 180-264VAC
Hiệu suất	75%
Công suất	360W
Nhiệt độ hoạt động	(-20°C) - 60°C
Nhiệt độ bảo quản	(-20°C) – 85°C
Tiêu chuẩn chất lượng	Rohs

*Bảng 3. 2: Thông số kĩ thuật của nguồn tổ ong.*



*Hình 3.8: Ảnh module hạ áp XL4015.*

Mạch hạ áp XL 4015 là mạch chuyên dùng để hạ áp DC-DC xuống mức thấp hơn ở đây là 12V xuống 5V. Mạch sử dụng phương pháp giảm áp xung để giảm áp DC với dòng đầu ra. Mạch có tích hợp đồng hồ led và phím chức năng chọn hiển thị áp đầu vào.

Thông số kĩ thuật :

Model	Hạ áp XL 4015
Điện áp đầu vào	4VDC – 38VDC
Điện áp ra	1.25V – 36V
Dòng ngõ ra	0A – 5A
Công suất ra	75W
Nhiệt độ hoạt động	(-40°C) - 85°C
Tần số hoạt động	180KHz
Hiệu suất chuyển đổi	96%

*Bảng 3. 3: Thông số kĩ thuật của mạch hạ áp XL 4015.*

STT	Tên linh kiện	Số lượng	Dòng tiêu thụ (mA)	Tổng dòng điện (mA)
1	ESP 8266	1	150	0.15
2	DHT11	1	2.5	0.0025
3	Relay	4	80	0.32

*Bảng 3. 4: Dòng điện các linh kiện sử dụng trong mạch điều khiển*

Từ bảng trên tổng dòng tiêu thụ cho toàn bộ mạch điều khiển là 0,4725AA vì vậy ta sử dụng nguồn có 12V-10A là an toàn cho mạch điều khiển.

### 3.2.2.5 Ứng dụng android

Yêu cầu của đề tài:

- Có thể bật tắt thiết bị.
- Thiết bị đặt thời gian và giá trị ngưỡng.
- Giao diện đăng nhập.
- Giao diện thân thiện với người dùng, dễ sử dụng.

Thiết kế App Android:

- Ứng dụng Android có giao diện thiết kế như sau:

Đèn      Quạt

Phun sương      Máy bơm

Ghi chú:

Quản lý thiết bị

*Hình 3.9: Thiết kế giao diện màn hình điều khiển.*

Giao diện sử dụng gồm: Đèn, quạt, phun sương, máy bơm. Các nút nhấn đèn, quạt, phun sương, máy bơm nhấn vào sẽ bật tắt thiết bị. Phần ghi chú sẽ ghi cách sử dụng quản lý thông số và trạng thái thiết bị. Khi ấn vào nút quản lý thiết bị sẽ chuyển qua trang quản lý để quản lý thiết bị.

Đèn

Nhiệt độ:       Bật ☐      Tắt ☐

Hẹn giờ:       Bật ☐      Tắt ☐

Quạt

Nhiệt độ:       Bật ☐      Tắt ☐

Hẹn giờ:       Bật ☐      Tắt ☐

Phun sương

Nhiệt độ:       Bật ☐      Tắt ☐

Độ ẩm:       Bật ☐      Tắt ☐

Hẹn giờ:       Bật ☐      Tắt ☐

Máy bơm

Nhiệt độ:       Bật ☐      Tắt ☐

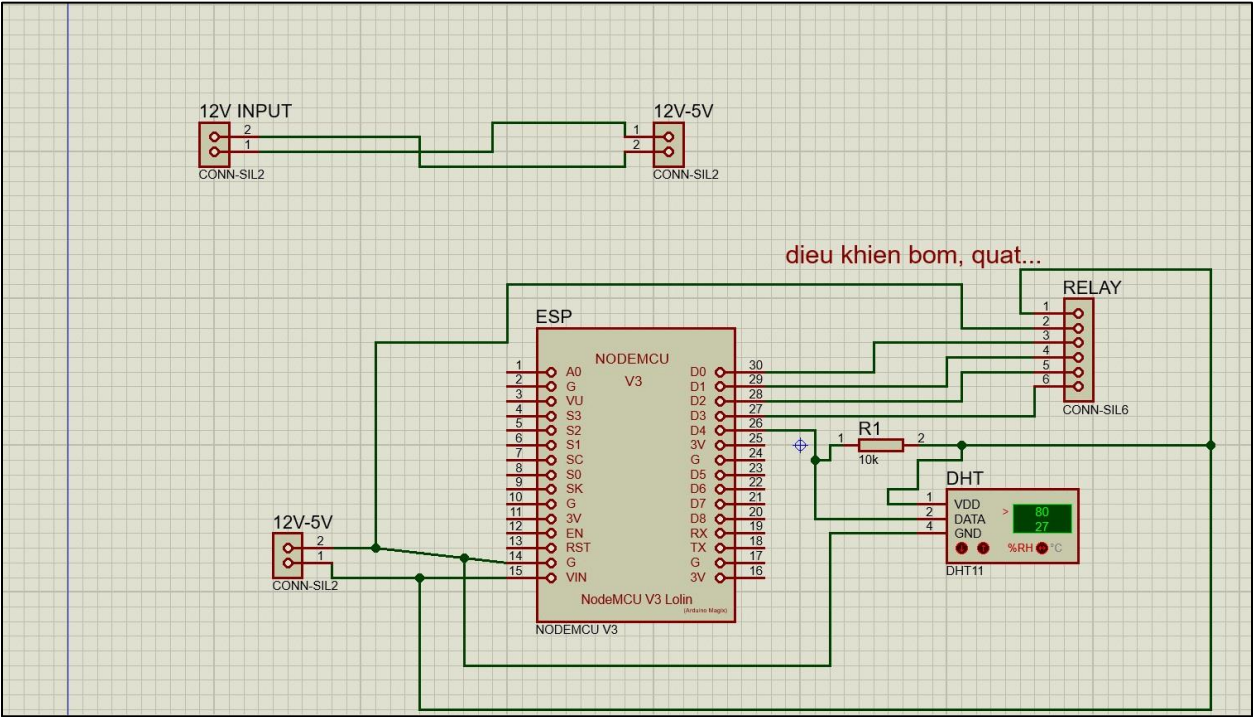
Hẹn giờ:       Bật ☐      Tắt ☐

Cập nhật thông số

*Hình 3.10: Quản lý thông số thiết bị.*

Tại trang quản lý thông số thiết bị, người dùng chủ động nhập thông số mà người dùng mong muốn theo một quy tắc thiết lập thông thường.

3.2.3 Sơ đồ nguyên lý toàn mạch



Hình 3.11: Sơ đồ nguyên lý toàn mạch.

## CHƯƠNG 4: THI CÔNG HỆ THỐNG

### 4.1 GIỚI THIỆU

Trong chương này trình bày quá trình PCB, lập trình, phần cứng và kiểm tra mạch. Bên cạnh đó là hình vẽ và hình thực tế lấy từ hệ thống.

### 4.2 THI CÔNG HỆ THỐNG

Dựa vào sơ đồ khối đã thiết kế, thực hiện thi công hệ thống gồm 2 phần:

- Board điều khiển.
- App điều khiển và tương tác người dùng.

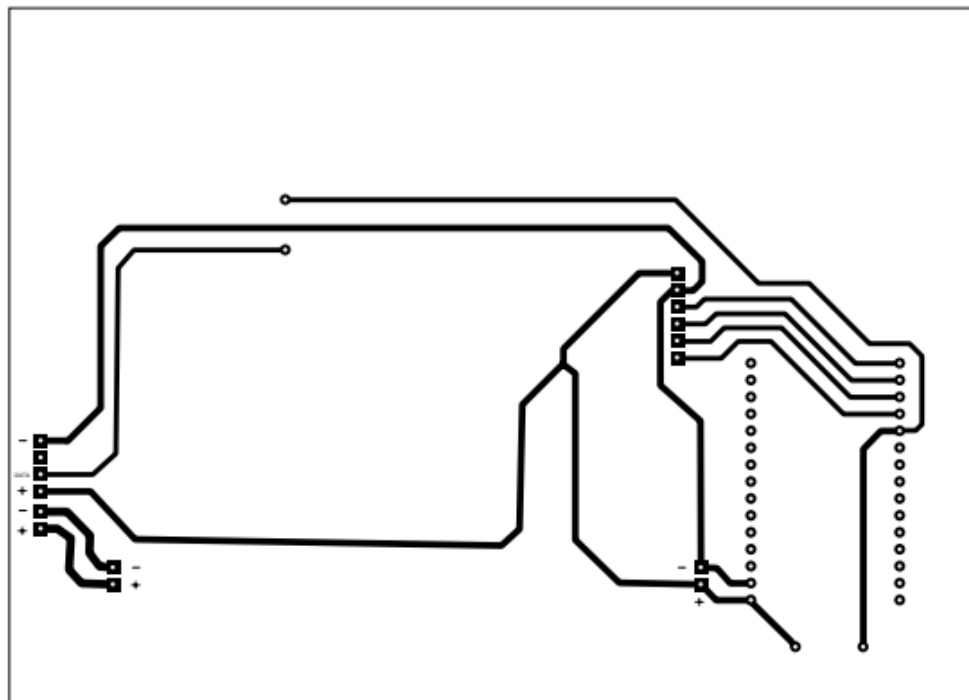
#### 4.2.1 Board điều khiển

Vật tư : Thước, phíp đồng, bàn ủi, cưa mini, cây bút vẽ mạch, chì hàn, mỏ hàn, bộ khoan, bột sắt để rửa mạch, kìm bấm chân linh kiện, VOM để đo thông mạch và một số dụng cụ khác.

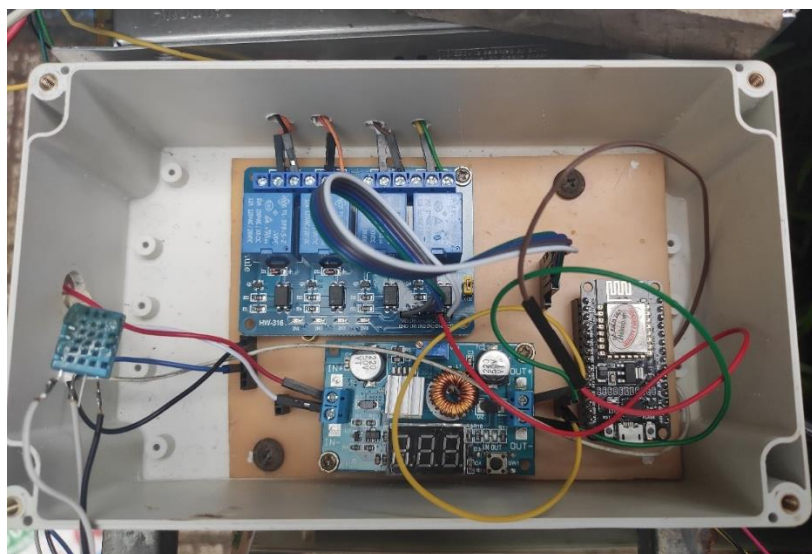
Vẽ mạch in bằng phần mềm Proteus xuất file PDF.

In mạch vào giấy bóng ảnh, sau đó cắt phíp đồng vừa với mạch in áp mạch in vào mặt có mực in của giấy, giữ thật chặt, để bàn ủi ở chế độ nóng nhất rồi ủi lên mặt sau của tờ giấy bóng cho đến khi mực in thấm ra mặt sau của giấy thì dừng.

Sau đó đợi 3 phút cho nguội rồi gỡ lớp giấy in ra một cách nhẹ nhàng để tránh bị mất đường mạch, nếu đường mạch bị đứt thì lấy bút vẽ mạch vẽ lại những đường bị đứt. Sau đó ngâm mạch với bột sắt pha với nước, cho đến khi lớp đồng còn lại bị mất hết. Sau đó ta rửa sạch lớp mực, ta được board mạch hoàn chỉnh. Dùng VOM đo xem mạch có thông nhau không, nếu có lỗi thì hàn lại cho thông mạch.



Hình 4.1: Sơ đồ mạch in.



Hình 4.2: Ảnh board mạch hoàn chỉnh.

STT	Tên linh kiện	Giá trị	Chú thích
1	Module nguồn	12V-30A	Nguồn tổ ong
2	Module hạ áp	5V	
3	ESP8266 NodeMCU	5V	
4	Module 4 Relay	5V	

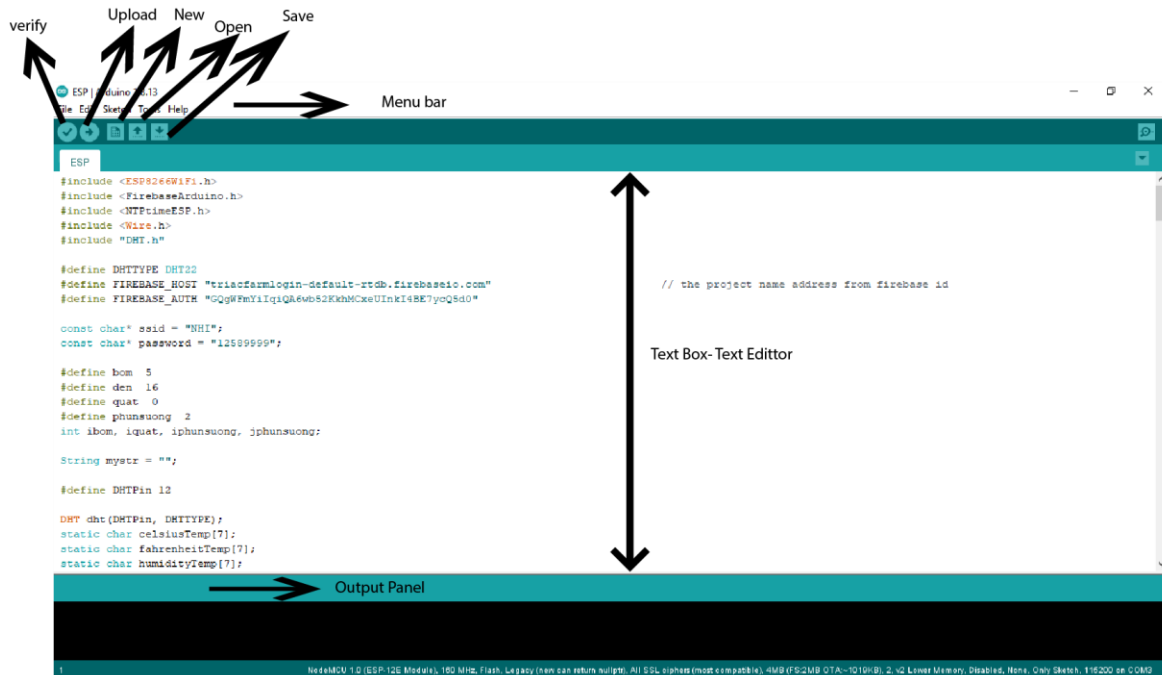
Bảng 3.6: Bảng thống kê linh kiện điều khiển.

Lắp ráp và kiểm tra:

Sau khi đã hoàn thành phần mạch in ta bắt đầu khoan lỗ, lắp linh kiện để hàn.

Kiểm tra mạch xem hệ thống bị đứt hay chập mạch dựa vào đồng hồ VOM. Kiểm tra chân nguồn và chân điều khiển.

## 4.2.1 Phần mềm Arduino IDE



Hình 4.3: Giao diện phần mềm Arduino IDE.

Arduino IDE là một phần mềm không tính phí phần tải và phần bản quyền. Đây là công cụ viết và code và nạp code các bo mạch Arduino cũng như các mạch NodeMCU. Ưu điểm của phần mềm là người dùng dễ dàng tiếp cận và sử dụng. Tương thích với hầu hết các hệ điều hành hiện nay.

Môi trường IDE gồm 2 thành phần cơ bản: Trình chỉnh sửa và trình biên dịch.

- + Trình chỉnh sửa: Dùng để viết mã theo yêu cầu.
- + Trình biên dịch: Dùng biên dịch và tải mã lên module Arduino.

Phần mềm sử dụng ngôn ngữ C với C++.

Để sử dụng phần mềm hỗ trợ theo nhu cầu mỗi người. Ta cần tải các thư viện phù hợp với vi điều khiển đang sử dụng.

### Phần code điều khiển của ESP8266 NodeMCU:

Để giao tiếp giữa ESP8266 với Firebase ngôn ngữ được sử dụng là C. Ngôn ngữ C là ngôn ngữ bậc thấp được phát triển bởi Dennis M. Ritchie để phát triển hệ điều hành UNIX sau đó được sử dụng để phát triển các phần mềm như IDE Arduino.

Phân khai báo thư viện Wi-Fi, giao tiếp với Firebase.

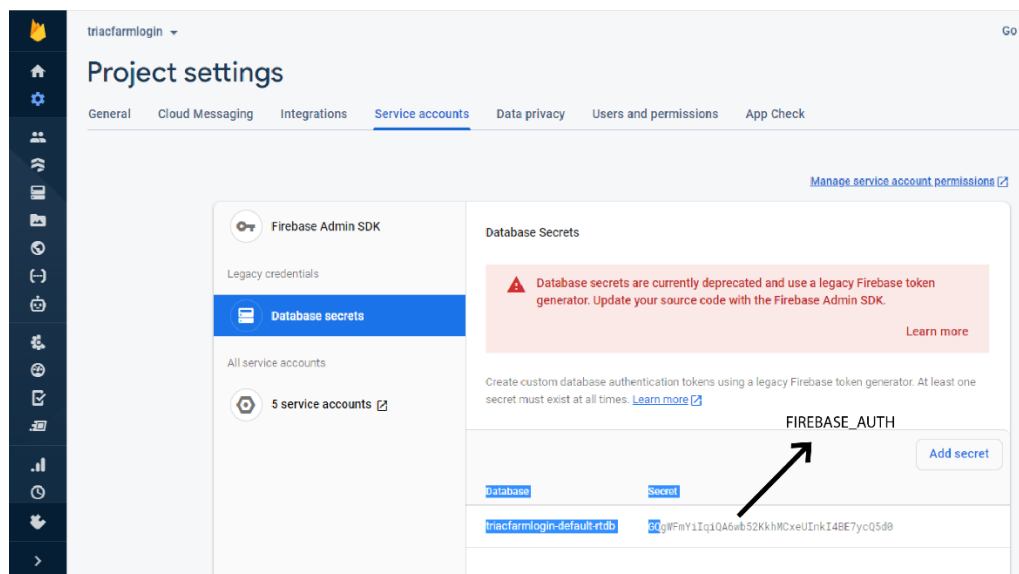
```
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
```



Khai báo URL, database secret của firebase, khai báo ID và Password của Wi-Fi đang dùng.

```
#define FIREBASE_HOST ""
#define FIREBASE_AUTH ""
Khai báo thư viện cảm biến DHT11.
#include "DHT.h"
#define DHTTYPE DHT11.
```

Để lấy mã của Firebase AUTH cho chương trình ESP8266, chúng ta vào Project trong Firebase **Setting > Project Setting > SERVICE ACCOUNTS > DATABASE SECRETS**.



Hình 4.4: Hình hướng dẫn lấy mã Firebase AUTH.

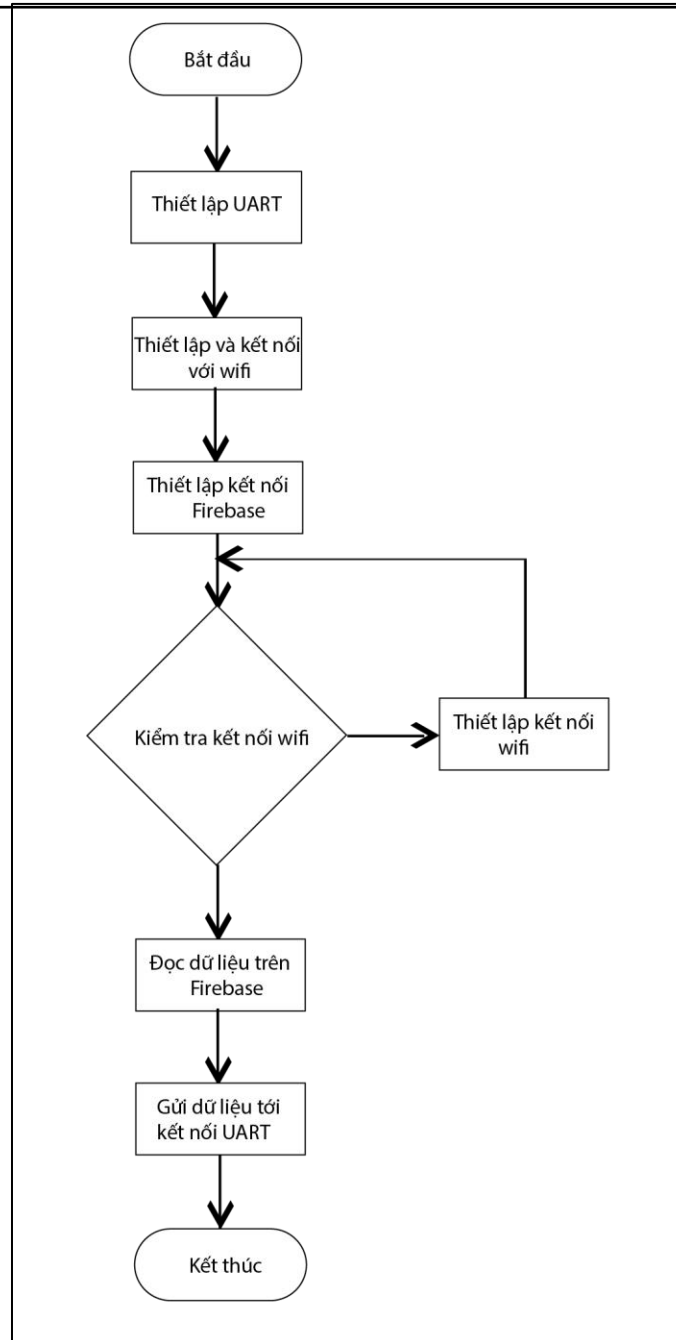
Thiết lập kết nối Wi-Fi và giao tiếp dữ liệu nhiệt độ, độ ẩm với Firebase.

```
void setup()
{
  Serial.begin(115200);
  delay(10);
  Serial.println();
  Serial.print("Đang kết nối đến mạng... ");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
```

```

}
Serial.println("");
Serial.println("Đã kết nối WiFi");
Serial.println("Firebase đang khởi động. Vui lòng đợi địa chỉ IP...");
delay(1000);
Serial.println(WiFi.localIP());
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
dht.begin();

```



Hình 4.5: Lưu đồ module ESP8266.

### Giải thích lưu đồ:

Bắt đầu, hệ thống sẽ thiết lập UART của ESP8266, thiết lập kết nối đến Wi-Fi mà ta đã thiết lập, trong lúc đó nó tạo thêm một dữ liệu gửi đến Firebase.

Tiếp theo kiểm tra đường truyền kết nối Wi-Fi, nếu có kết nối thì sẽ đọc dữ liệu trên Firebase và nếu không có kết nối với Wi-Fi nó sẽ tự động kết nối lại.

Sau khi đọc, nhận dữ liệu từ Firebase, ESP8266 NodeMCU sẽ truyền dữ liệu điều khiển đến các cổng điều khiển cổng Relay cấp nguồn cho thiết bị.

- Điều khiển thiết bị bằng tay thông qua App

```
#define DHTTYPE DHT11
#define bom 5
#define den 16
#define quat 0
#define phunsuong 2
void loop()
{
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  if (Firebase.failed()) {
    Serial.println("Firebase Error" + Firebase.error());
    return;
  }
  if (isnan(t) || isnan(h)) {
  } else {
    Firebase.setFloat("/DHT/data/nhiệtDo", t);
    Firebase.setFloat("/DHT/data/doAm", h);
    Serial.print("nhiệtDo: ");
    Serial.println(t);
    Serial.print("doam: ");
    Serial.println(h);
  }
}
```

- Giá trị ngưỡng, đặt hẹn của nhiệt độ và độ ẩm:

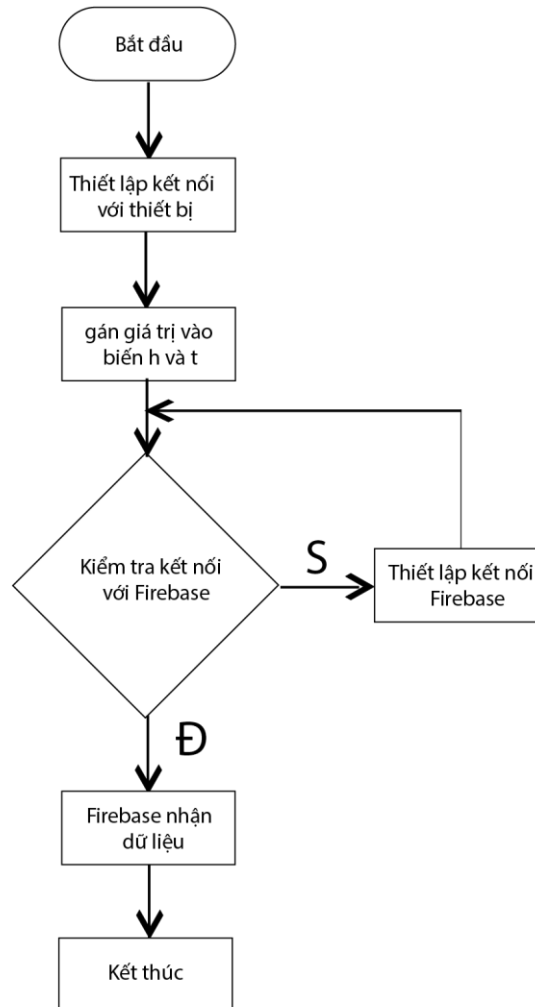
Các biến thời gian được đặt độc lập với các biến còn lại nếu đến đúng giờ thì thiết bị được bật và ngược lại

Giá trị về ngưỡng và chỉnh tay tự động sẽ ảnh hưởng đến nhau. Nếu tín hiệu thiết bị, ở đây ta xét là MayBom được nhận từ Firebase là 0 thì sẽ xét thêm điều kiện ở trong là MayBom và giá trị ngưỡng của nó và ngược lại .

Các thiết bị khác tương tự như MayBom.

```
if(thgian == MBtimeOn)
{
    Firebase.setInt("MayBom/MayBom", 1);
    Firebase.setInt("MayBom/DKBom", 1);
}
if(thgian == MBtimeOff)
{
    Firebase.setInt("MayBom/MayBom", 0);
    Firebase.setInt("MayBom/DKBom", 0);
}

if(Firebase.getInt("MayBom/DKBom") == 0)
{
    if(Firebase.getInt("MayBom/ibom") == 0 && t>MBon.toInt())
    {
        Firebase.setInt("MayBom/MayBom", 1);
        Firebase.setInt("MayBom/ibom", 1);
    }
    if(Firebase.getInt("MayBom/ibom") == 1 && t<MBoff.toInt())
    {
        Firebase.setInt("MayBom/MayBom", 0);
        Firebase.setInt("MayBom/ibom", 0);
    }
}
```



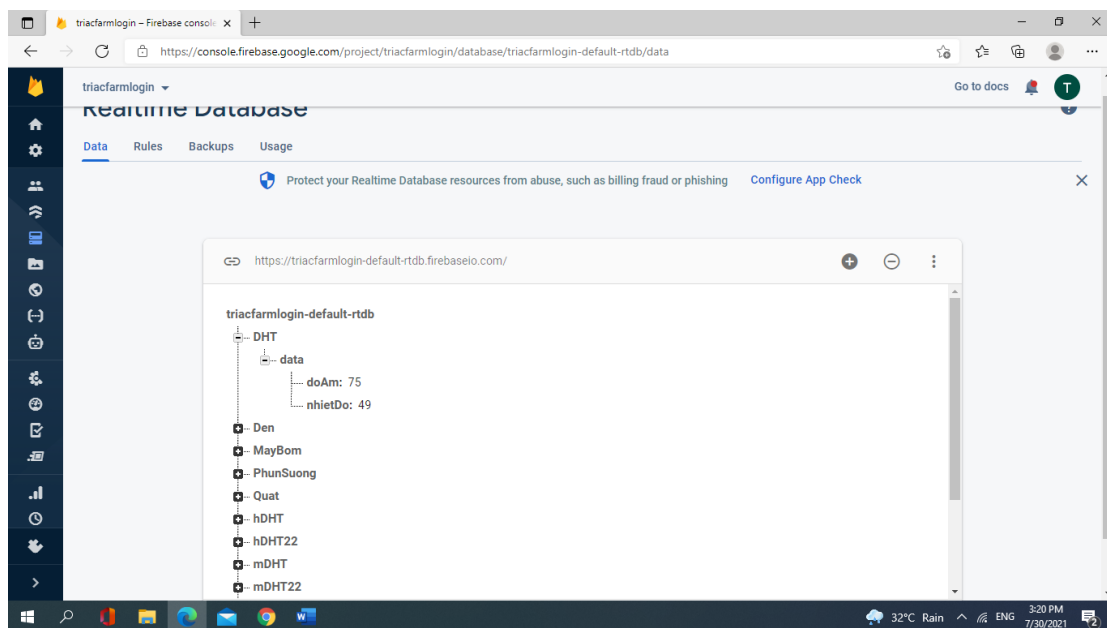
Hình 4.2: Sơ đồ thuật toán ESP8266 gửi dữ liệu lên Firebase.

**Giải thích lưu đồ thuật toán:**

Ban đầu DHT11 đo giá trị nhiệt độ và độ ẩm từ môi trường. Giá trị sẽ được mã hóa và được gửi đến Firebase.

ESP8266 tín hiệu cho Firebase nếu được phản hồi thì báo giá trị được gửi thành công và Firebase sẽ ghi nhận. Ngược lại, thì Firebase sẽ gửi tín hiệu và ESP8266 sẽ gửi lại tín hiệu đo được.

## 4.2.2 Firebase



Hình 4.7: Firebase Realtime.

## 4.2.3 App điều khiển

### 4.2.2.1 Phần mềm Android Studio



Hình 4.8: Logo phần mềm Android.

Android Studio là một trong những công cụ để phát triển Android. Đây là IDE chính thức của nền tảng android. Phần mềm thiết kế đơn giản dễ sử dụng. Các nhà phát triển ứng dụng mobile chạy trên hệ điều hành Android thường chọn phần mềm này để làm việc. Phần mềm tương thích với tất cả các hệ điều hành như Windows, Mac OSX và Linux.

Phần mềm dùng ngôn ngữ lập trình java. Java là một **một ngôn ngữ lập trình hiện đại, bậc cao, hướng đối tượng, bảo mật và mạnh mẽ** và là một **Platform**. Ngôn ngữ lập trình Java ban đầu được phát triển bởi **Sun Microsystems** do **James Gosling** khởi xướng và phát hành vào năm 1995.

Hệ điều hành Android là hệ điều hành có mã nguồn mở dựa trên nền tảng Linux do chính Google phát hành hệ điều hành có lượng người sử dụng đứng đầu thế giới, với hơn 85% thị phần di động ở thời điểm hiện tại. Hệ điều hành đã tích hợp trên điện thoại di động, máy tính bảng, smart tivi, smartwatch, và các thiết bị khác trong gia đình.

Chức năng của Android Studio rất đa dạng. Giải quyết dễ dàng các giao diện để tạo ứng dụng và các file phức tạp. Trong dự án người dùng có nhiều mô-đun bao gồm: mô-

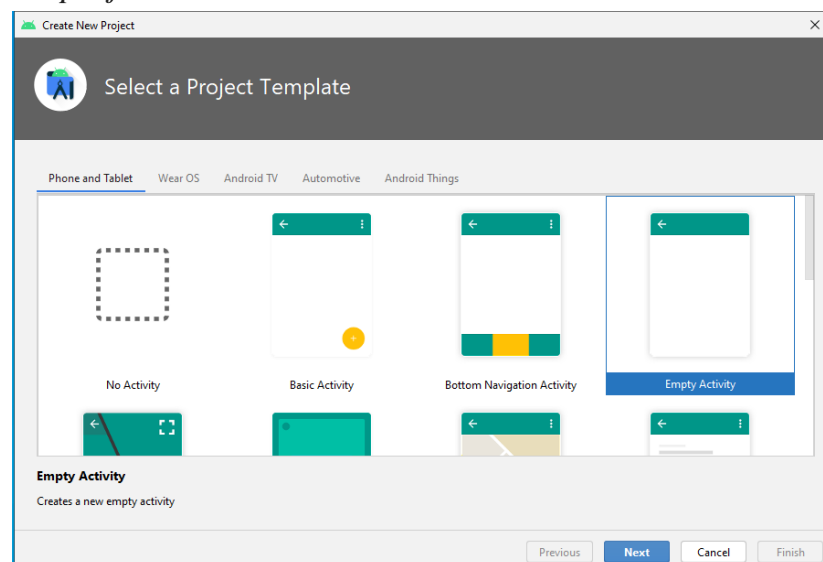
đun ứng dụng Android, các mô-đun thư viện, các mô-đun của Google App Engine. Phần mềm sử dụng ngôn ngữ java để giao tiếp.

▪ **Cách truy cập và tạo Project:**



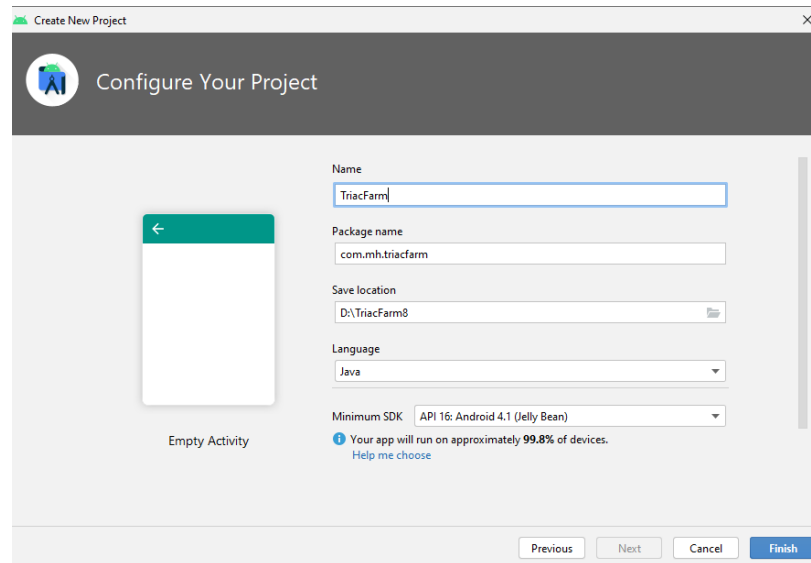
Hình 4.9: Giao diện create new project.

Khởi động Android Studio và trên cửa sổ *Android Studio Setup Wizard*, bạn chọn *Start a new Android Studio project*.



Hình 4.10: Cửa sổ Add an activity to Mobile.

Tiếp theo là cửa sổ *Add an activity to Mobile*, bạn chọn *Basic Activity*. Hãy dành khoảng nửa phút để xem hết tất cả các tùy chọn. Cửa sổ này cho ta có một cái nhìn tổng quan về các **layout template**.



Hình 4.11: cửa sổ *Customize the Activity*.

Trên cửa sổ *Customize the Activity* như màn hình bên dưới, bạn sẽ có các tùy chọn để thay đổi *Activity Name*, *Layout Name*, *Title* và *Menu Resource Name*. Sau đó nhấn finish để tạo project làm việc.

#### ▪ Chương trình điều khiển:

Yêu cầu ứng dụng:

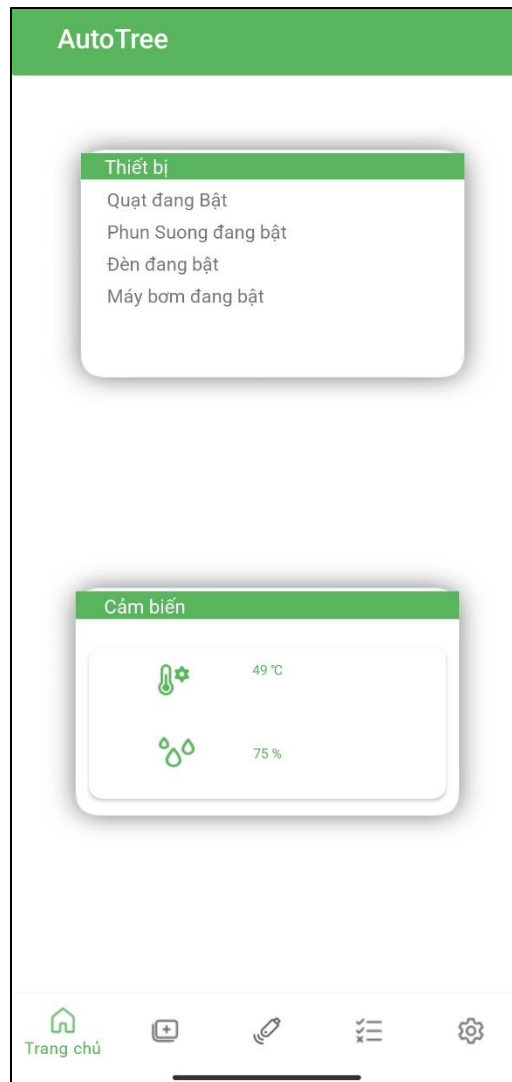
- Yêu cầu đăng nhập tài khoản người dùng.
- Kết nối với trung tâm điều khiển thông qua Wi-Fi tới Firebase.
- Điều khiển thiết bị thông qua nút bấm trên giao diện.
- Hiện biểu đồ nhiệt độ, độ ẩm để người dùng dễ dàng giám sát được điều kiện môi trường.
- Giao tiếp bằng một số lệnh đơn giản.
- Thông báo nhiệt độ, độ ẩm trên ứng dụng.
- Đồng bộ trạng thái các thiết bị theo thời gian thực.
- Nhiều người dùng có thể điều khiển cùng lúc.

#### ▪ Giao diện và các thành phần:

Từ những yêu cầu trên ta có thể thiết kế giao diện của ứng dụng gồm 5 màn hình: Màn hình đăng nhập, màn hình tương tác.

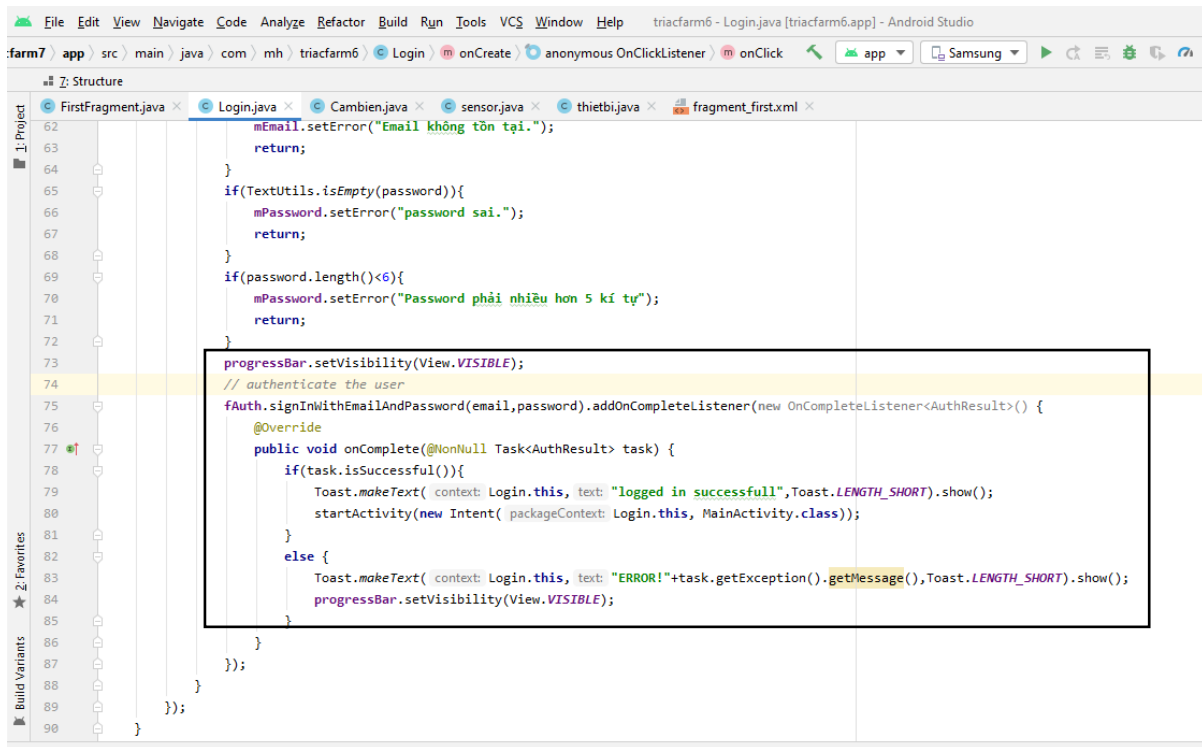


- Màn hình đăng nhập tài khoản (Login):



*Hình 4.12: Màn hình đăng nhập APP.*

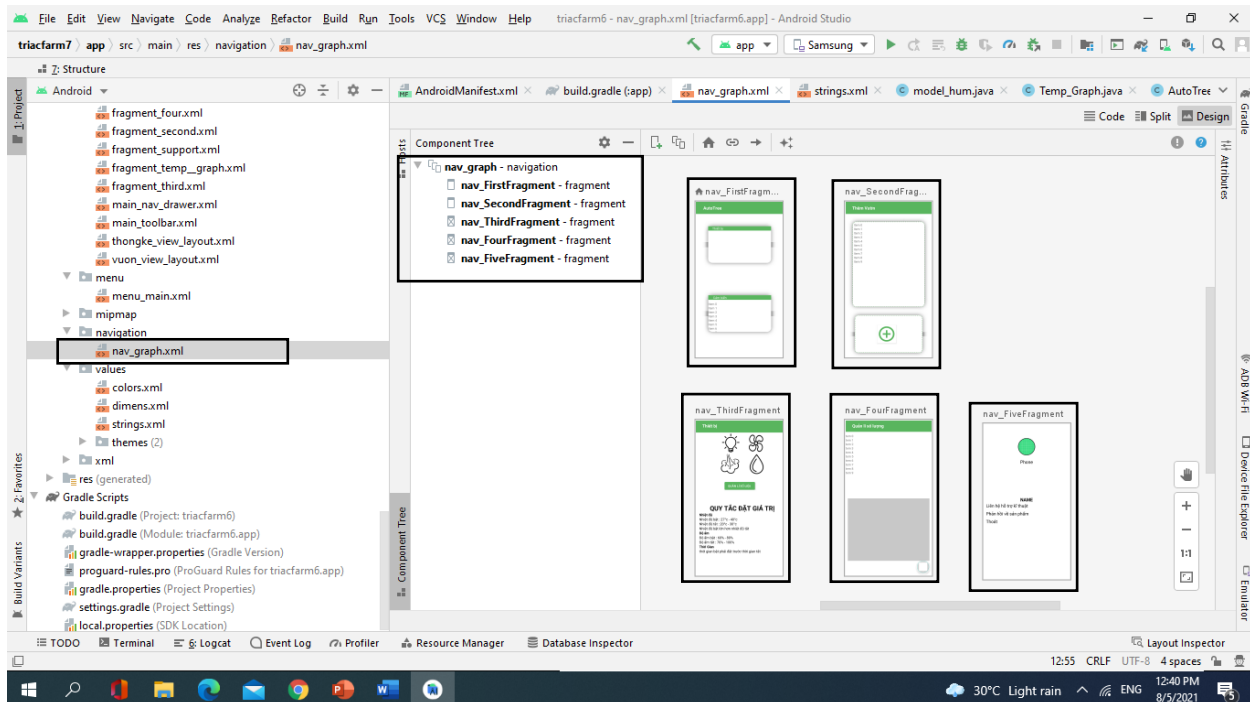
Màn hình đăng nhập gồm có edittext Email, edittext Password người dùng, và một button Đăng nhập.



Hình 4.13: Code kiểm tra mail và password.

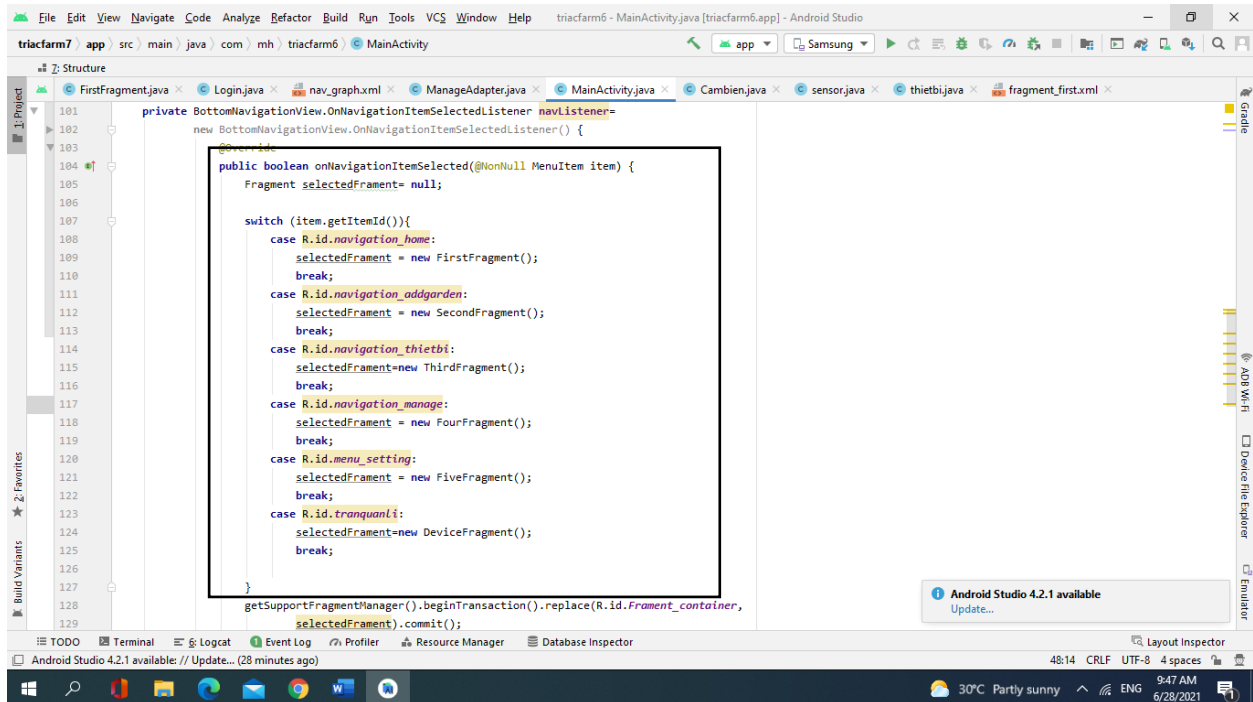
Đoạn code kiểm tra tài khoản đăng nhập có hợp lệ và chính xác. Chỉ những tài khoản được cấp quyền mới được đăng nhập vào hệ thống điều khiển thiết bị.

- Màn hình chức năng (MainActivity):



Hình 4.14: Code định hướng mở các Fragment tương ứng.

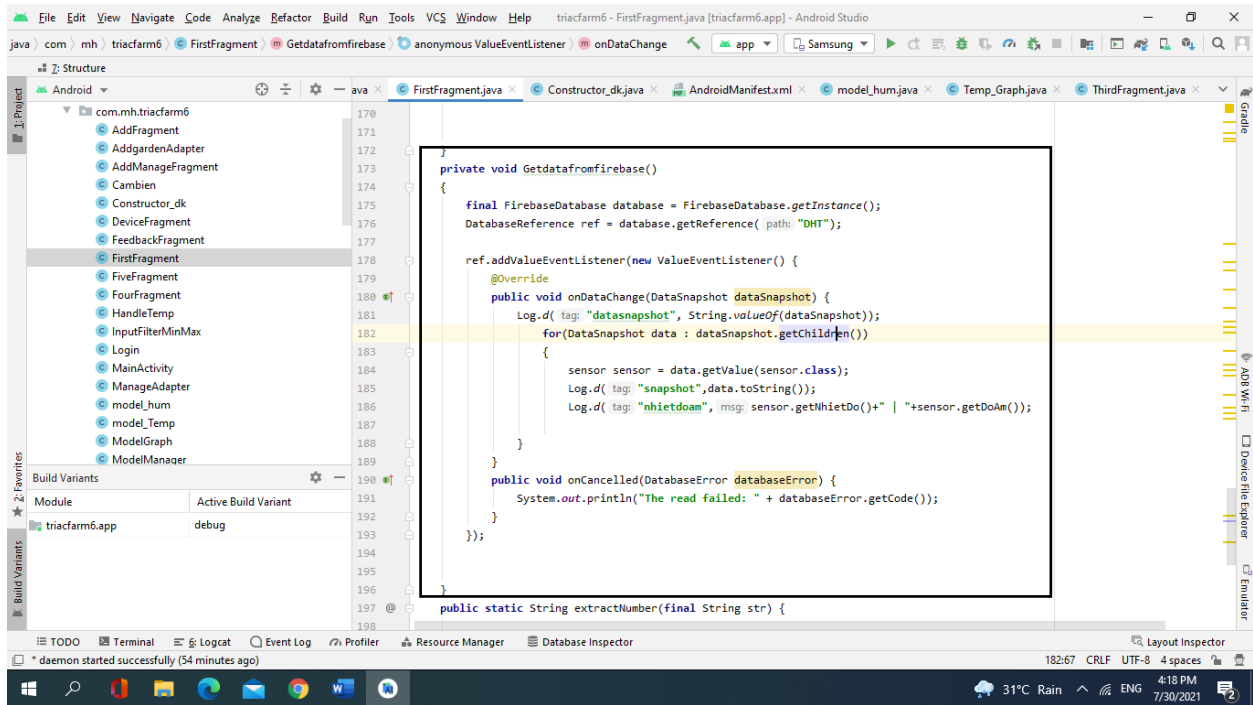
Sau khi đăng nhập người dùng hệ thống sẽ chuyển hướng vào MainActivity. Để tùy chỉnh cho Menu chuyển tab ở đây **nav\_graph.xml** là và chọn **Firstfragment** để làm màn hình mở đầu khi người dùng vào.



Hình 4.15: Code định hướng các nút trong tab.

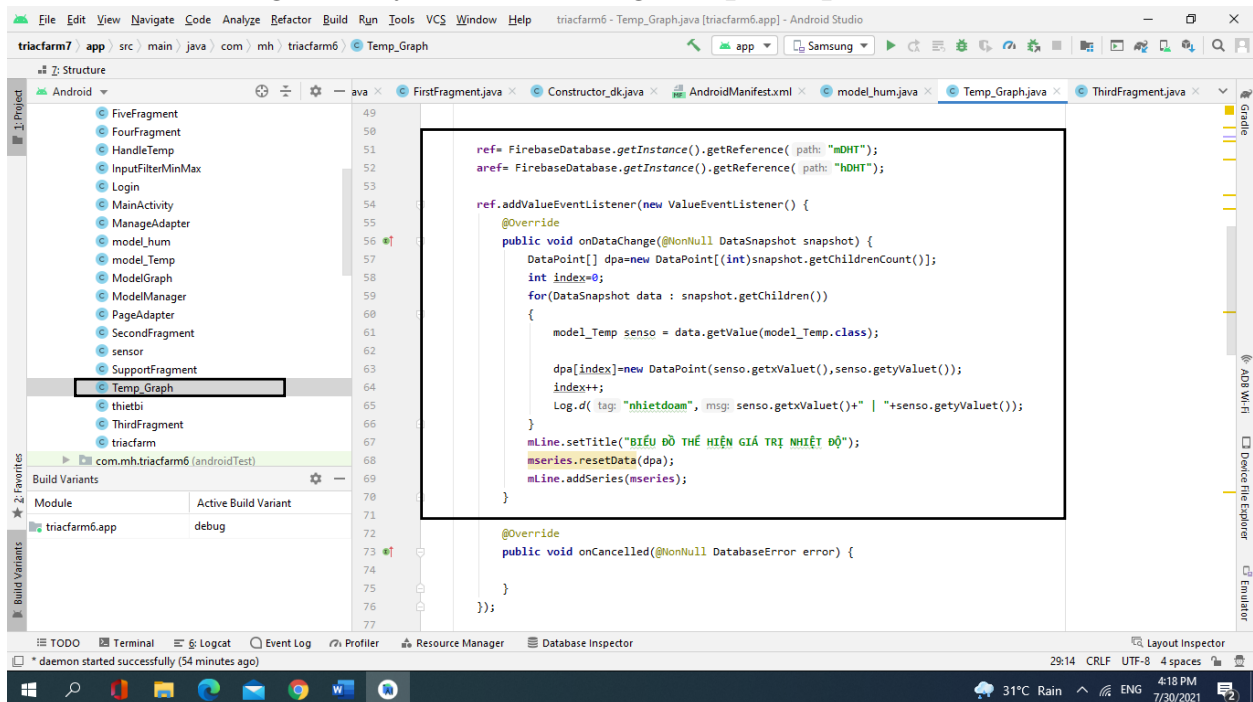
Tạo sự kiện lắng nghe bên trong MainActivity.java.

- Thiết lập Firebase:
- + Đọc dữ liệu từ Firebase Realtime Database:



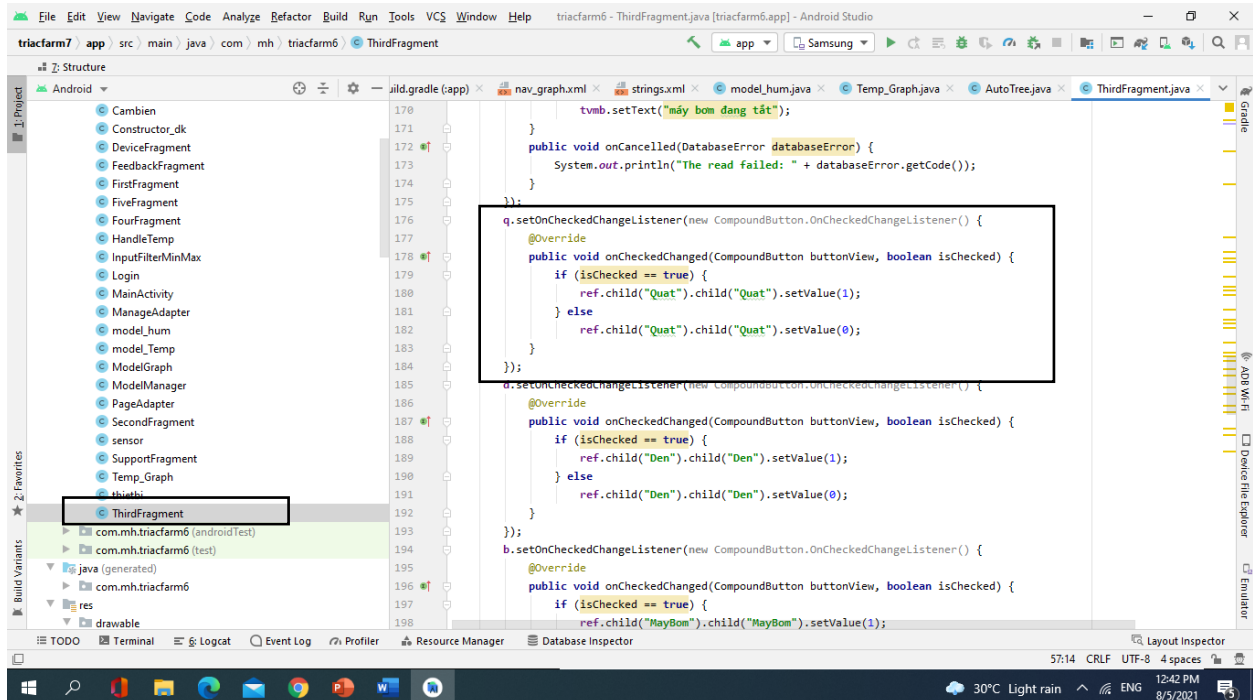
Hình 4.16: Code lấy dữ liệu cảm biến đo được từ Firebase.

Dữ liệu sau khi được tải lên Firebase sẽ được app nhận và hiển thị cho người dùng theo dõi một cách dễ dàng. Người dùng chỉ cần nhấn vào giá trị nhiệt độ hoặc độ ẩm. Lúc đó biểu đồ của hai giá trị này sẽ hiện lên trong **Temp\_Graph.class**.



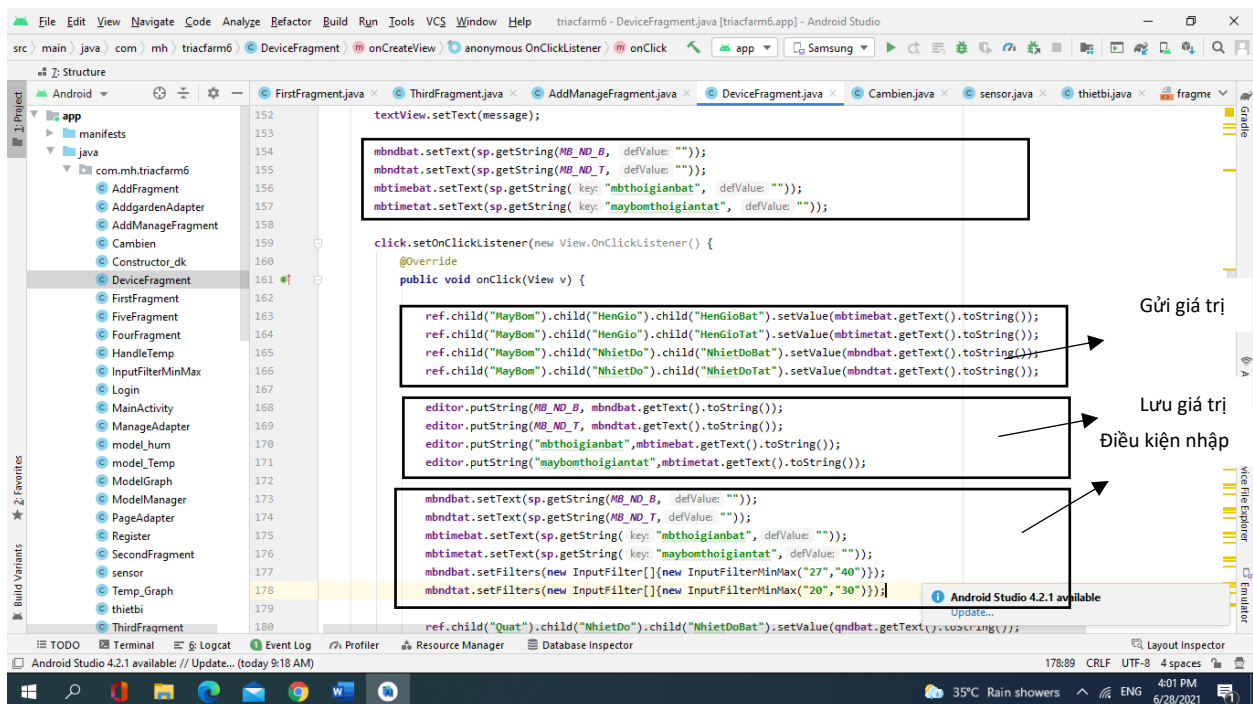
Hình 4.17: Code nhận dữ liệu và hiển thị thành biểu đồ.

## + Điều khiển thiết bị:



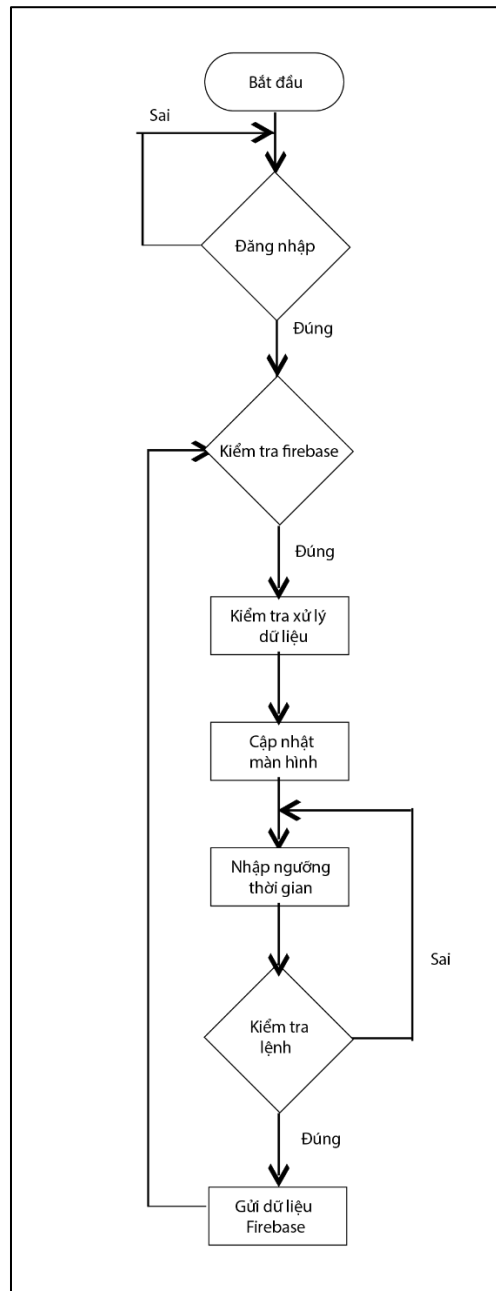
Hình 4.18: Nút điều khiển thiết bị trên App.

Để đảm bảo tính thẩm mỹ em dùng toggleButton để lắng nghe trạng thái thay đổi của nút.

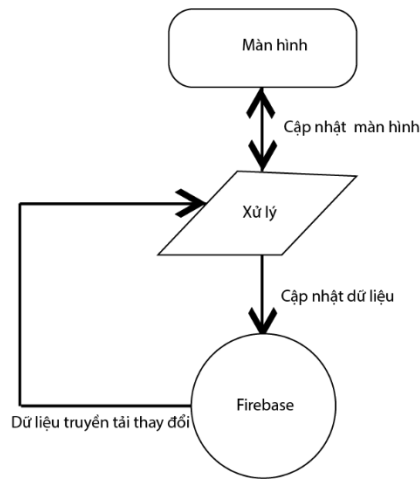


Hình 4.19: Code gửi giá trị điều khiển và ngưỡng của máy bơm lên Firebase.

Các giá trị ngưỡng được đặt một điều kiện nhất định để nhập tránh trường hợp người dùng nhập nhầm thì các thiết bị sẽ ảnh hưởng đến hiệu suất làm việc. Người dùng sau khi nhập giá phù hợp thì ấn nút cập nhật để gửi giá trị lên Firebase thì dữ liệu đó vẫn được giữ lại sau khi App đã tắt.



Hình 4.20: Sơ đồ khởi App điều khiển Android.



*Hình 4.21: Lưu đồ cập nhật dữ liệu.*

#### **Giải thích lưu đồ:**

Bắt đầu khi mở App android, mở đầu sẽ là trang logo của App. Sau đó sẽ là màn hình đăng nhập vào app để sử dụng và tương tác với thiết bị. Người dùng sẽ thấy và tương tác với dữ liệu. Đồng thời điều khiển thiết bị thông qua các nút trên màn hình đã thiết kế. Việc tự động hóa cũng được diễn ra một cách đồng thời với các nút ấn sau mỗi 0,3s, hệ thống sẽ cập nhật dữ liệu một lần và được gửi lên Firebase.

Quá trình cập nhật sẽ diễn ra liên tục, các giá trị sẽ liên tục được cập nhật và gửi lên Firebase sau đó được xử lý. Đồng thời quá trình này cũng sẽ được cập nhật liên tục và xử lý nó.

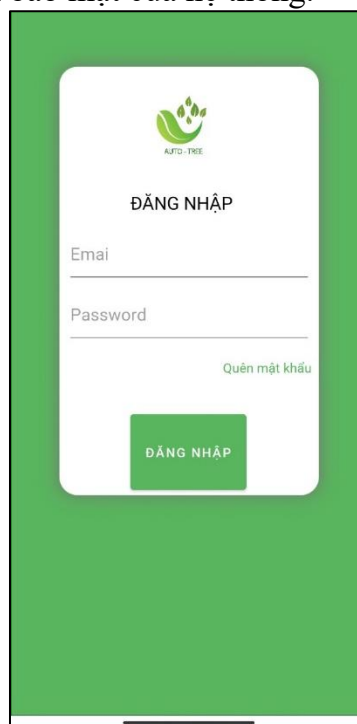
#### **4.3 HƯỚNG DẪN SỬ DỤNG**

Mở app lên sẽ mở trang AutoTree( là một activity bắt đầu) là màn hình hiển thị logo. Đếm 2s thì tắt. Nhằm mục đích để nhấn mạnh người dùng và ghi nhớ Logo của App.



*Hình 4.22: Màn hình giới thiệu.*

Login ( một activity ) là một màn hình đăng nhập vào hệ thống điều khiển. Dùng để xác thực người dùng, đảm bảo tính bảo mật của hệ thống.

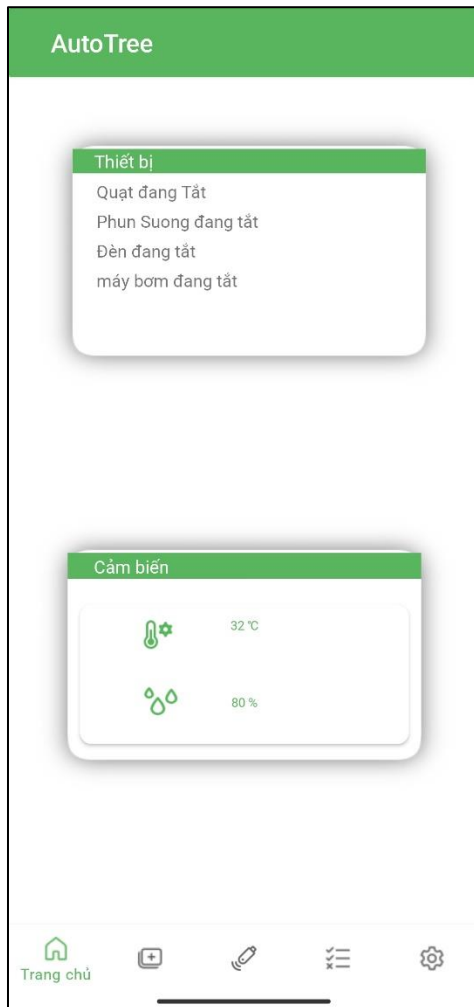


*Hình 4.23: Màn hình đăng nhập.*

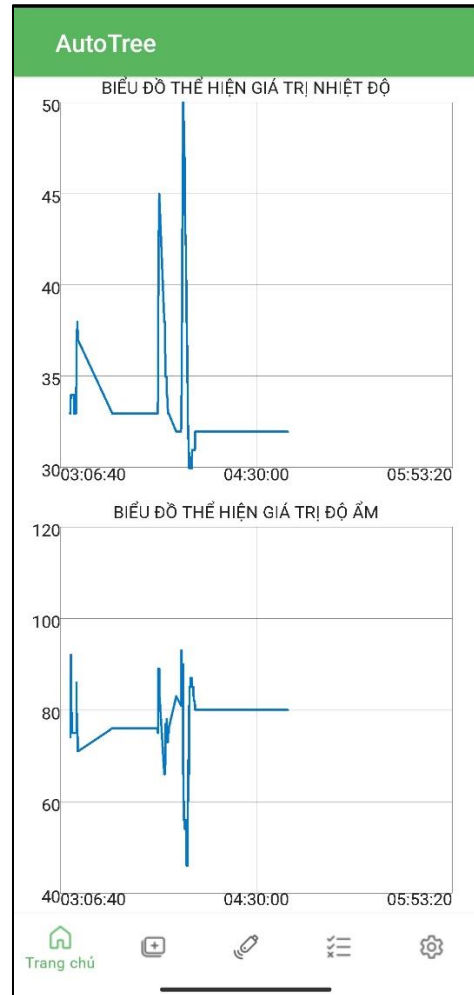


MainActivity (FirstFragment, SecondFragment, ThirtFragment, FourFragment, FiveFragment) nó hiện thị dưới dạng các tab tương tác với người dùng.

Mở đầu là tab Home (FirstFragment.class) là màn hình con của MainActivity. Nó sẽ hiển thị các giá trị nhiệt độ và độ ẩm mà DHT11 đo được đã được gửi lên Firebase và trạng thái của các thiết bị. Sau khi bấm vào giá trị đo nhiệt độ và độ ẩm hệ thống sẽ chuyển sang màn hình hiển thị đồ thị của nhiệt độ với thời gian thực và độ ẩm với thời gian thực.



Hình 4.24: Màn hình trang chủ.



Hình 4.25: Đồ thị nhiệt độ và độ ẩm.

Màn hình tab vườn (SecconFragment.class) là màn hình con của MainActivity.class. Nó sẽ hiển thị thông tin vườn của người dùng trên màn hình này dưới dạng một list. Nếu người dùng muốn xóa chỉ cần trượt ngang đối tượng. Khi bạn nhấn vào nút dấu cộng nó sẽ chuyển sang màn hình thông tin vườn nó bao gồm nhập tên vườn, mô tả về vườn, ảnh vườn, vị trí vườn. Trong trang này nếu nhấn vào thoát bạn sẽ quay lại tab vườn và nếu nhấn vào nút thêm bạn sẽ lưu dữ liệu đã ghi và quay lại tab vườn.



Hình 4.26: Màn hình vườn của người dùng.

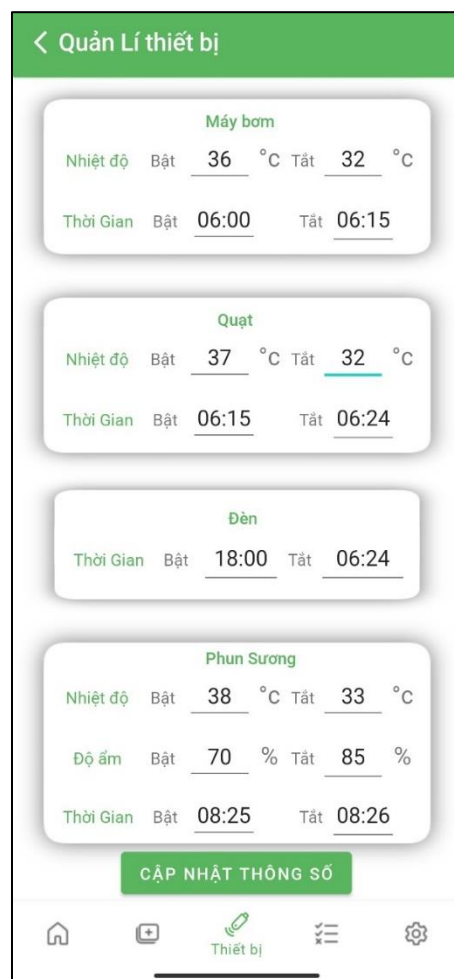


Hình 4.27: Màn hình thêm vườn.

Màn hình tab thiết bị là một phần của MainActivity.class. Ở tab này sẽ có những nút tương tác với thiết bị nút màu đen là lúc tắt đèn, còn nếu nút chuyển qua màu xanh thì thiết bị bật. Đồng thời có những lưu ý cho người dùng khi sử dụng thiết bị. Có nút chuyển sang màn hình quản lý. Màn hình này giúp người dùng đặt ngưỡng và thời gian cho thiết bị.



Hình 4.28: Màn hình thiết bị.



Hình 4.29: Màn hình quản lý thiết bị

Màn hình tab thống kê (ThirstFragment) Là màn hình ghi chép các vấn đề liên quan đến ghi chép thống kê vườn trong một vườn giữa các mùa. Đồng thời sẽ hiển thị thống kê trên biểu đồ để người dùng dễ thay đổi.



Hình 4.30: Màn hình thống kê

**Thống kê**

Loại rau:

rau cải

Ngày trồng:

11-7-2021

Ngày thu hoạch:

18-7-2021

tháng thu hoạch

tháng phân tích số liệu

Số kg/vụ:

100

Ghi chú:

sâu nhiều

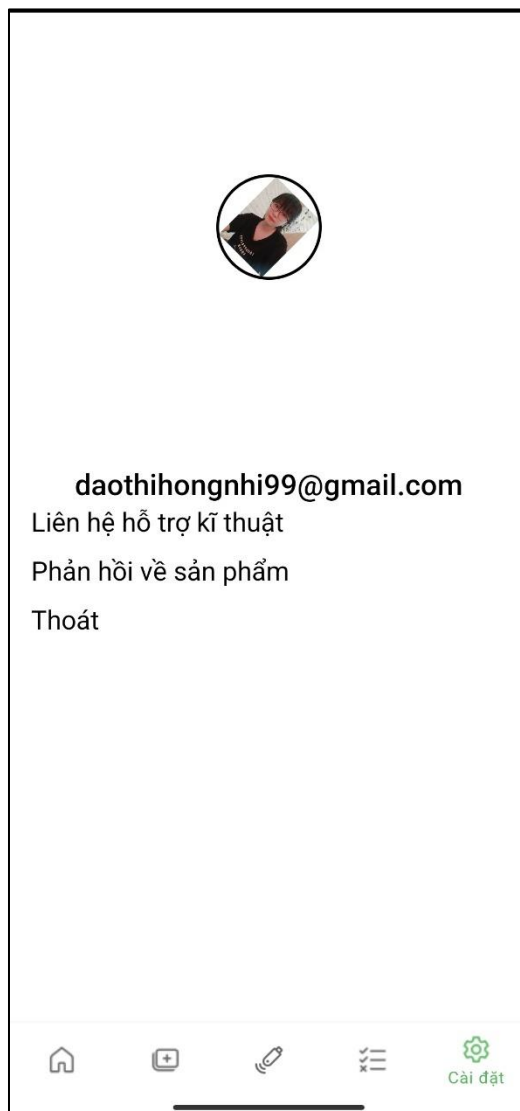
THOÁT

THÊM

Quản lý

Hình 4.31: Màn hình thêm thống kê.

Màn hình tab thông tin khách là màn hình hiển thị ảnh thông tin khách và cách liên lạc với người làm app, dùng để phản hồi về sản phẩm.



Hình 4.32: Màn hình thông tin khách hàng.

## CHƯƠNG 5: KẾT QUẢ\_NHẬN XÉT\_ĐÁNH GIÁ

### 5.1 KẾT QUẢ ĐẠT ĐƯỢC

#### Tổng quan kết quả đạt được

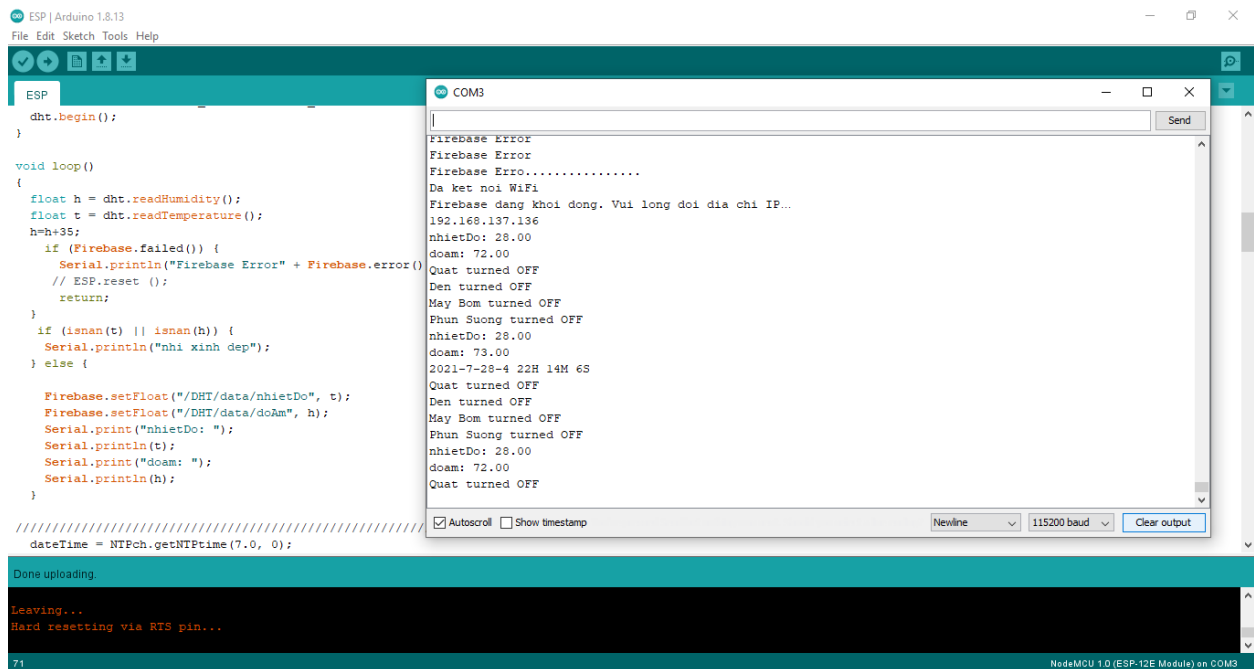
Đồ án “*quản lí hệ thống tự động hóa quản lí và chăm sóc cây trồng*” như đã trình bày là một đề tài có tính ứng dụng cao, người dùng sử dụng dễ dàng. Việc xây dựng hệ thống dựa trên nhiều mảng kiến thức, từ những kiến thức của mảng nhúng cho tới kiến thức về Android đồng thời có cả kiến thức làm mạch.

**Dựa vào kiến thức đã học cùng sự cố gắng em đã thực hiện được những vấn đề sau:**

- Hiểu rõ về hệ điều hành Android.
- Hiểu được kiến thức giao tiếp Wi-Fi.
- Thực hiện viết ứng dụng trên Mobile Android.
- Thực hiện việc nhúng lên module.
- Làm mạch in tại nhà.

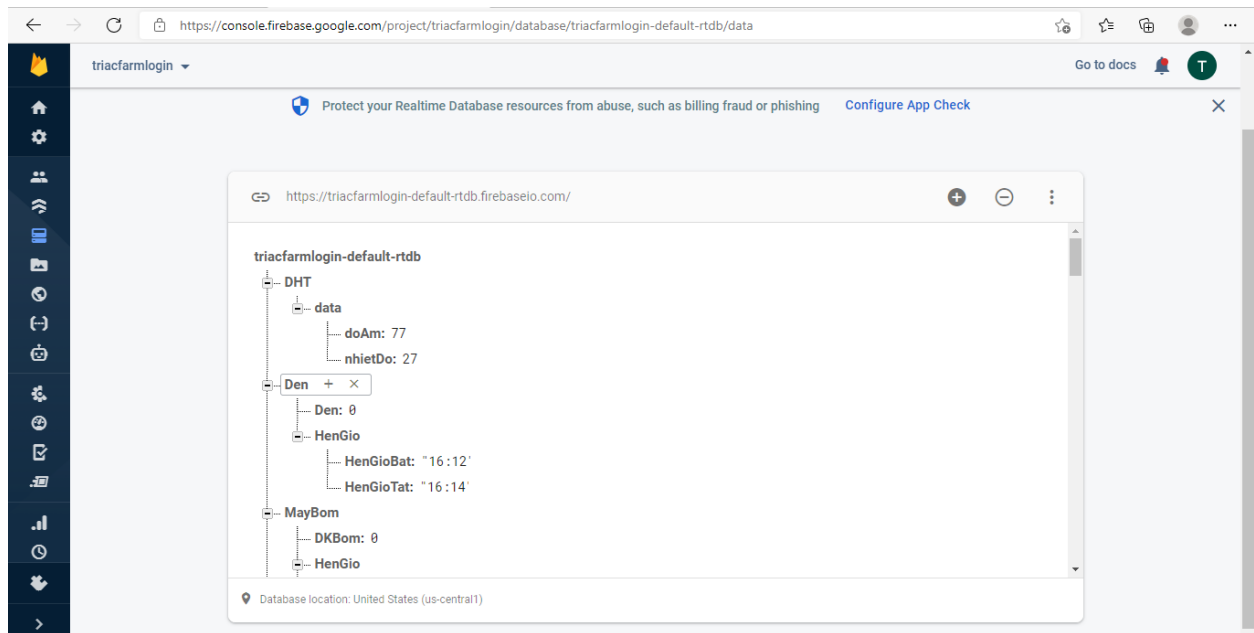
#### Kết quả thực nghiệm:

- Kết quả nạp code vào ESP8266:



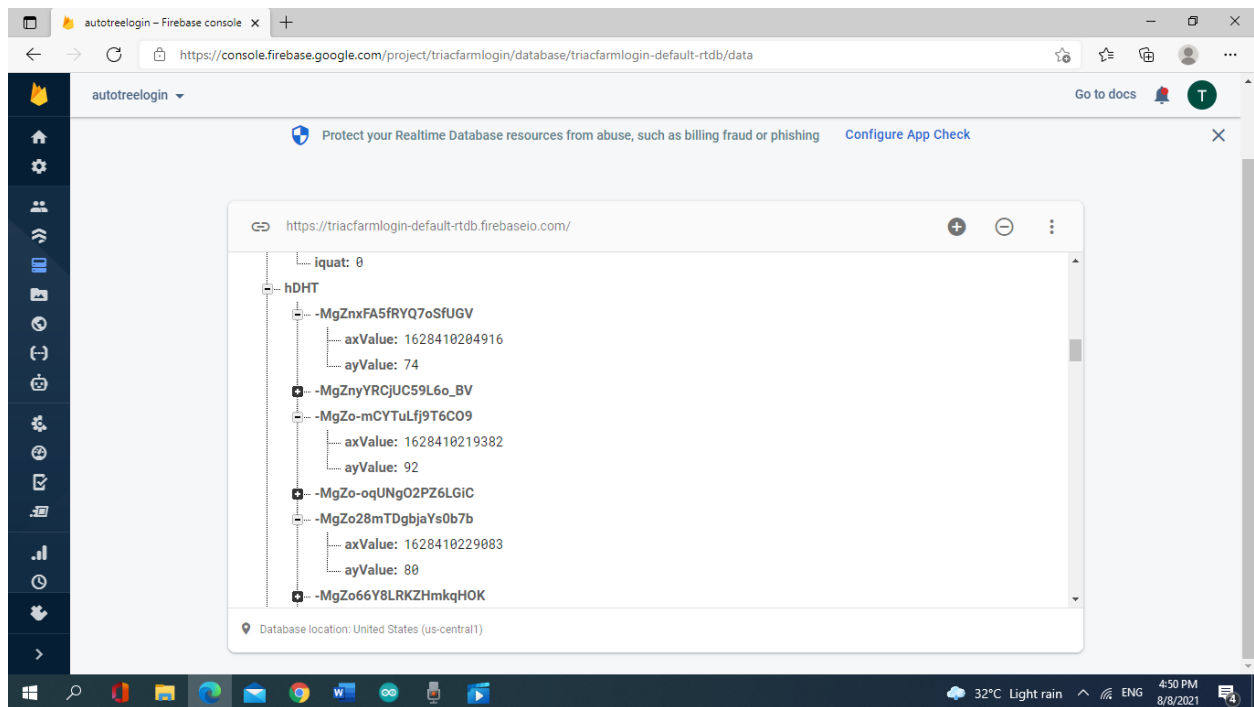
Hình 5.1: Kết quả nạp code vào ESP8266.

- Giá trị của cảm biến được cập nhật và gửi lên Firebase.



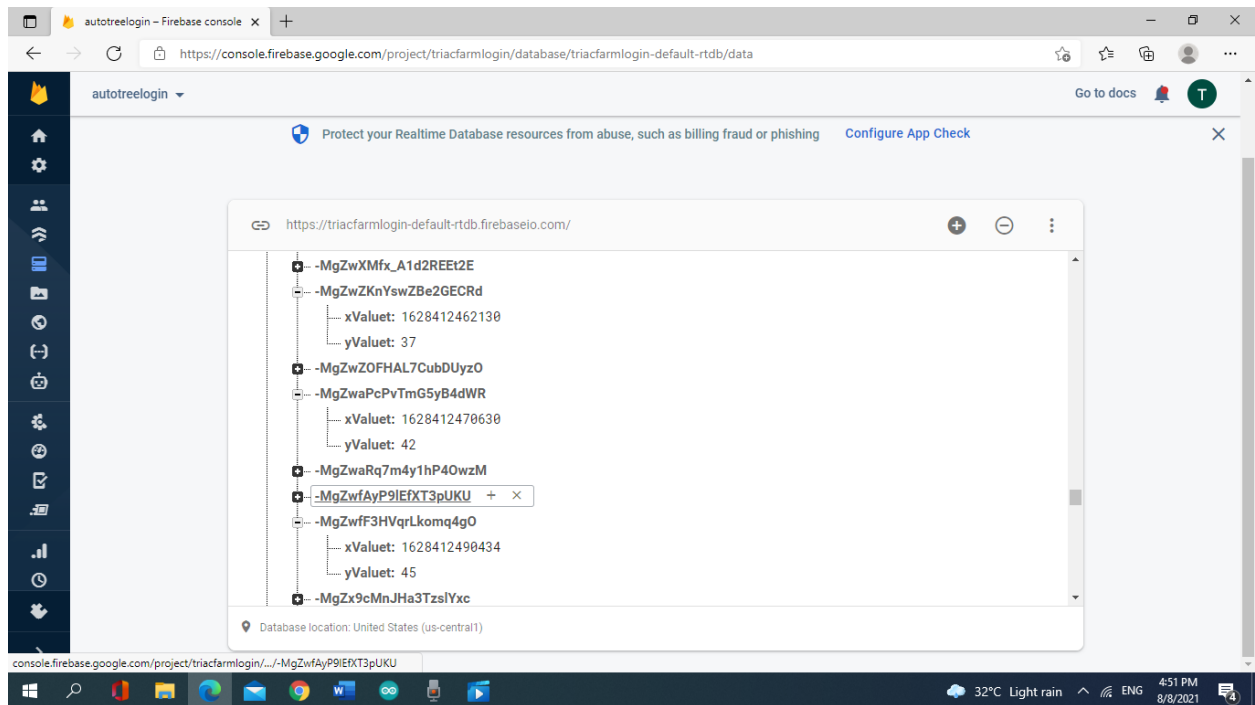
Hình 5.2: Giá trị cảm biến trên Firebase.

- Giá trị độ ẩm được lưu lại ở một dạng khác:



Hình 5.3: Giá trị độ ẩm theo thời gian thực.

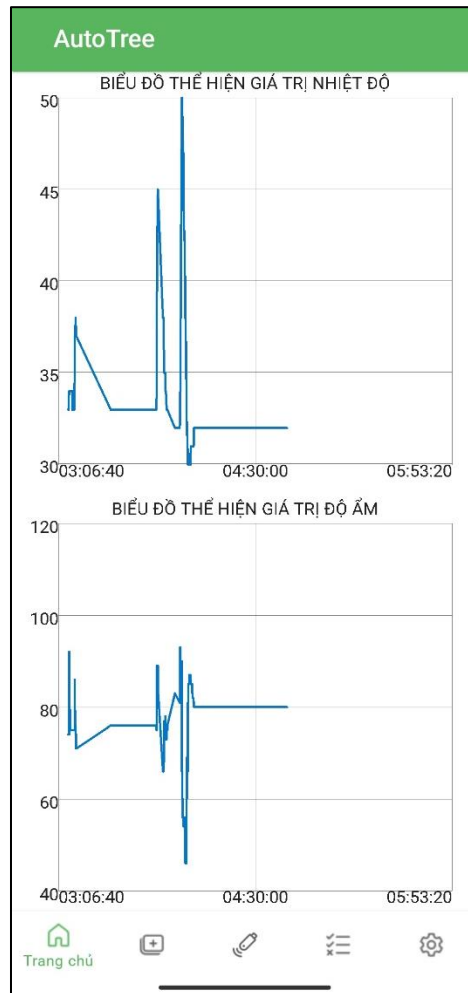
- Giá trị nhiệt độ được lưu lại ở một dạng khác:



Hình 5.3: Giá trị nhiệt độ theo thời gian thực.

- Biểu đồ thể hiện giá trị theo thời gian thực:





Hình 5.4: Biểu đồ nhiệt độ và độ ẩm theo thời gian thực.

- Trạng thái của thiết bị được hiển thị dưới dạng chữ:



Hình 5.5: Trạng thái của thiết bị.

### Thống kê kiểm thử hệ thống

Trong quá trình vận hành hệ thống, em ghi nhận được kết quả:

Chức năng	Số lần thao tác	Số lần đúng	Thời gian đáp ứng	Đánh giá
Giám sát cảm biến	50	50	2 giây	Đạt
Điều khiển thiết bị qua nút ấn trên app	50	50	3-4 giây	Đạt
Đặt ngưỡng cho thiết bị	50	47	3-4 giây	Đạt
Hẹn giờ cho thiết bị	50	48	10- 15 giây	Đạt
Đánh giá chung				Đạt

*Bảng 5.1: Thống kê kết quả thực tế.*

#### 5.2 Nhận xét \_ Đánh giá

Nhận xét:

- Hệ thống hoạt động ổn định qua nhiều lần thử nghiệm.
- Giám sát nhiệt độ và độ ẩm liên tục.
- Tốc độ điều khiển bằng tay thuận tiện.
- Dễ dàng sử dụng, lắp đặt và bảo dưỡng.
- Hệ thống phụ thuộc vào tốc độ mạng Wi-Fi, sự ổn định của Firebase.

Đánh giá:

Đồ án thực hiện đạt được khoảng 80% mục tiêu đặt ra.

## **CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **6.1 KẾT LUẬN**

Sau thời gian nghiên cứu, thi công thì đề tài “hệ thống thu thập thông tin nhiệt độ và độ ẩm trong nông nghiệp” đã hoàn thiện, đáp ứng được những yêu cầu ban đầu đặt ra. Mô hình hoạt động tương đối ổn định, có thể làm việc liên tục, đạt 90% yêu cầu đề ra ban đầu. Bên cạnh đó hệ thống mở rộng thống mở rộng thêm các chức năng giám sát nhiệt độ, độ ẩm trên biểu đồ.

#### **6.1.1 Ưu điểm**

- Mạch điều khiển nhỏ gọn, hoạt động khá ổn định.
- Giao diện điều khiển và giám sát dễ sử dụng, thân thiện với người dùng.
- Mô hình hệ thống có độ chính xác, tính an toàn.

#### **6.1.2 Nhược điểm**

Tuy nhiên, do sự hạn chế về kiến thức và thời gian thực hiện, tình hình dịch bệnh căng thẳng nên không tránh khỏi sai sót và còn một số hạn chế

- Giao diện app chưa hoàn chỉnh.
- Chưa thể để người dùng tự thay đổi Wi-Fi.
- Chưa để người dùng chủ động đăng kí tài khoản.
- Tác động điều khiển còn chậm phụ thuộc vào Wi-Fi và Firebase.
- Hoạt động chủ yếu trong môi trường có sóng Wi-Fi.

### **6.2 HƯỚNG PHÁT TRIỂN**

Đồ án đã đạt được một số kết quả nhất định, song để đáp ứng cho một bài toán thực tế còn chưa thật sự tối ưu. Sau đây là hướng phát triển của đồ án mà em dự định làm:

- Hoàn thiện trong việc đăng kí tạo vườn riêng cho từng người dùng.
- Thiết lập thiết bị được nhập từ người dùng.
- Thiết lập Wi-Fi từ người dùng.
- ứng dụng đề tài vào hệ thống thực tế.

### **Tài liệu tham khảo**

- [1] Lenka, R. K., Rath, A. K., & Sharma, S. (2019). Building Reliable Routing Infrastructure for Green IoT Network. *IEEE Access*, 7.
- [2] Sensor, *Temperature Sensor*.
- [3] Bulut F. Ersavas, Montrose C (US); Semih Pekol, Istanbul (TR); Atakan Bodur I (TR). Methods and systems for irrigation control. *United States Pat.* 2014;2(12).
- [5] Website: [www.developer.android.com](http://www.developer.android.com)
- [6] Lập trình Android cơ bản – NXB Nhất Nghệ.