

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



**Đồ án Thiết kế Luận lý
(CO2011)**

Vườn thông minh

"Trong canh tác cây trồng"

Advisor: ThS. Trần Thanh Bình
Students: Nguyễn Hữu Hiếu - MSSV: 2013149
Dư Văn An - MSSV: 2012545
Vũ Trung Hiếu - MSSV: 2012500

HO CHI MINH CITY, DECEMBER 2022



Contents

1 Thành viên & Nhiệm vụ	2
2 Mở đầu	3
3 Giới thiệu đề tài	3
3.1 IoT là gì?	3
3.2 Ý tưởng về vườn thông minh	10
3.2.1 Chăm sóc vườn cây thực phẩm	10
3.2.2 Ý tưởng về vườn thông minh	13
4 Cơ sở lý thuyết	14
4.1 Các nhân tố tác động tới cây trồng	14
4.2 Hệ thống IoT gồm những thành phần gì?	19
4.3 Node Sensor là gì?	19
4.4 Gateway là gì?	21
4.5 Server là gì?	23
4.6 MQTT và MQTT mosquitto	25
4.6.1 MQTT	25
4.6.2 MQTT mosquitto	27
5 Linh kiện	28
5.1 ESP-12S ESP8266 Ai-Thinker NodeMCU CH340C WIFI Module IOT	28
5.2 Module RF thu phát Wifi ESP8266 NodeMCU Lua CP2102	30
5.3 Cảm Biến Độ Âm Đất	31
5.4 DHT11 Mạch Cảm Biến Nhiệt Độ Và Độ Âm Đen	32
5.5 Bơm nước 365DC 12V	32
5.6 LED dây 5050 màu đỏ	33
6 Tiến hành mô hình demo	34
6.1 Sensor Node	35
6.2 GateWay	36
6.3 Server	39
6.3.1 MQTT Broker	39
6.3.2 Node-RED	41
6.3.3 Database	41
6.4 Ứng dụng điện thoại	43
6.4.1 Giới thiệu về MIT App Inventor	43
6.4.2 Ứng dụng Android Vườn thông minh	44
6.4.2.a Giao diện và hành vi của Ứng dụng	44
6.4.2.b Lập trình chức năng của các Box	45
6.4.2.c Hướng dẫn sử dụng	50
6.5 Thi công vườn	51
6.6 Tính năng và Hành vi của mô hình	53
7 Tổng kết	54



1 Thành viên & Nhiệm vụ

No.	Họ & Tên	MSSV	Tasks	Mức độ hoàn thành
1	Nguyễn Hữu Hiếu	2013149	<ul style="list-style-type: none">- Thiết kế hệ thống- Thiết kế và lập trình SensorNode- Thiết kế và lập trình GateWay- Server- Ứng dụng Điện thoại- Thi công vườn (Hệ thống thiết bị)- Report : 20%	100%
2	Dư Văn An	2012545	<ul style="list-style-type: none">- Cơ sở lý thuyết- Thi công vườn (Cơ khí)- Report : 80 %	100%
3	Vũ Trung Hiếu	2012500	<ul style="list-style-type: none">- Cơ sở lý thuyết- Thi công vườn (Cơ khí)- Report : 80 %	100%

2 Mở đầu

Hiện nay, việc phát triển của thế giới điện tử số phát triển một cách nhanh chóng và mạnh mẽ, điện tử số cụ thể là vi xử lý ngày càng trở nên đa dạng và các ứng dụng cũng gần gũi với chúng ta hơn. Cùng với sự phát triển đa dạng của ngành công nghiệp vi xử lý nên tài nguyên của vi xử lý cũng được nâng cao để đáp ứng các ứng dụng khác nhau trong thực tế. Mạng Internet ngày càng ứng dụng rộng rãi trong mọi lĩnh vực của đời sống xã hội. Công nghệ ngày càng phát triển đòi hỏi nhu cầu ứng dụng vào ngành công nghiệp nhằm giảm lao động, đảm bảo sức khỏe nhân công để không ảnh hưởng xấu đến chất lượng sản phẩm... đem lại hiệu quả cao cho nền công nghiệp ngày càng tiên tiến.

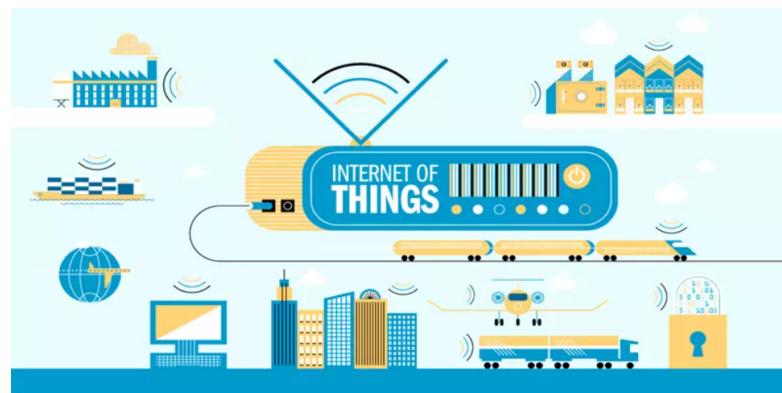
Ở đồ án tốt nghiệp này nhóm em thiết kế, thi công mô hình 1 trạm xử lý và 1 trạm trung tâm xử lý cho phép điều khiển và giám sát hệ thống. Trạm xử lý sẽ gửi tín hiệu, các thông số môi trường về cho trung tâm xử lý, các thông số này sẽ được giám sát tại trung tâm. Người sử dụng có thể điều khiển thiết bị điện ở khoảng cách xa, ở bất cứ nơi nào có Internet, Wifi, 3G, 4G trên ứng dụng được thiết kế, lệnh điều khiển sẽ được gửi về trạm xử lý để đóng/cắt mạch công suất. Các giá trị nhiệt độ, độ ẩm không khí, độ ẩm đất, khí CO₂, cường độ ánh sáng của mô hình cũng được cập nhật lên giao diện này.

3 Giới thiệu đề tài

3.1 IoT là gì?

IoT là tên viết tắt của **Internet of Things** có nghĩa là **Internet vạn vật**. Hay cụ thể hơn nữa là Mang lưỡi van vật kết nối Internet hoặc là Mang lưỡi thiết bị kết nối Internet. Đây là một hệ thống các thiết bị tính toán có liên quan với nhau, máy móc cơ khí, kỹ thuật số hoặc con người được cung cấp một mã nhận dạng duy nhất (UID) và khả năng truyền dữ liệu qua mạng mà không yêu cầu sự tương tác giữa con người với máy tính.

Các thiết bị Internet of Things có thể là đồ vật được gắn thêm cảm biến để thu thập dữ liệu về môi trường xung quanh, các máy tính hay bộ điều khiển tiếp nhận dữ liệu và ra lệnh cho các thiết bị khác, hoặc là các đồ vật được tích hợp cả hai tính năng trên.



Hình 1: IoT

Vậy tại sao IoT lại quan trọng?.

Trong vài năm qua, IoT đã trở thành một trong những công nghệ quan trọng. IoT giúp mọi người sống và làm việc thông minh hơn, cũng như chủ động hoàn toàn trong cuộc sống.

IoT cung cấp các thiết bị thông minh để tự động hóa ngôi nhà. Với IoT, người dùng có thể khởi động các thiết bị trong nhà như hệ thống chiếu sáng, điều hòa, bình nóng lạnh tự động... bằng điện thoại thông minh hay máy tính bảng. Các thao tác hẹn giờ, thay đổi nhiệt độ được thực hiện nhanh chóng và dễ dàng trên thiết bị thông minh, giúp tiết kiệm tối đa thời gian sử dụng và lượng điện tiêu thụ, tránh trường hợp quên tắt các thiết bị khi đi ra ngoài.

IoT là yếu tố cần thiết đối với hoạt động kinh doanh. IoT giúp cho doanh nghiệp nắm được cách thức mà doanh nghiệp của họ vận hành theo thời gian thực, cung cấp thông tin chi tiết về mọi thứ từ hiệu suất của máy móc đến chuỗi cung ứng và hoạt động hậu cần.

Ngày nay, các doanh nghiệp trong nhiều ngành khác nhau đang sử dụng Internet of Things (IoT) để hoạt động hiệu quả hơn, hiểu rõ hơn về khách hàng để cung cấp dịch vụ khách hàng nâng cao, cải thiện khả năng ra quyết định và gia tăng giá trị của doanh nghiệp.

Do đó, IoT là một trong những công nghệ quan trọng nhất của cuộc sống hàng ngày và nó sẽ tiếp tục tăng trưởng khi có nhiều doanh nghiệp nhận ra tiềm năng của các thiết bị được kết nối để giữ cho chúng cạnh tranh.

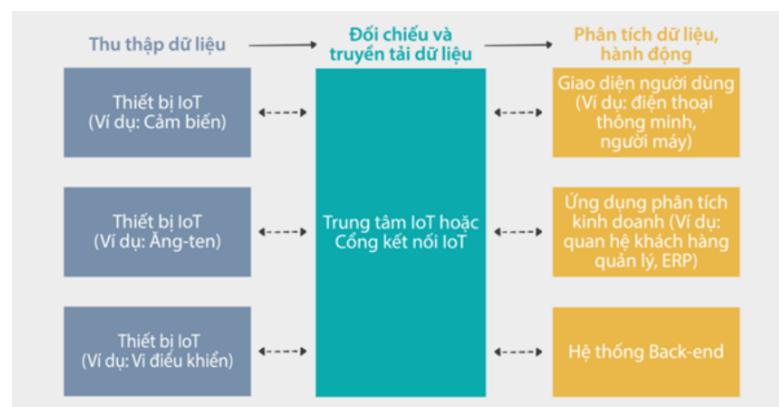
Cách thức IoT vận hành.

Một hệ sinh thái IoT bao gồm các thiết bị web-enabled (web kích hoạt) thông minh được nhúng vào trong hệ thống, chẳng hạn như bộ xử lý, cảm biến và phần cứng truyền thông, để thu thập, gửi và thực hiện hành động trên dữ liệu mà chúng thu thập được. Dữ liệu thu thập có thể đơn giản là nhiệt độ, độ ẩm... hoặc phức tạp hơn là video, hình ảnh...

Các thiết bị IoT chia sẻ dữ liệu cảm biến mà chúng thu thập được bằng cách kết nối với cổng IoT hoặc thiết bị cảm biến khác, nơi dữ liệu được gửi lên cloud (đám mây) để phân tích hoặc phân tích cục bộ. Đôi khi, các thiết bị này giao tiếp với các thiết bị liên quan khác và hoạt động dựa trên thông tin chúng nhận được từ nhau. Các thiết bị thực hiện hầu hết công việc mà không có sự can thiệp của con người, mặc dù mọi người có thể tương tác với các thiết bị.

Các giao thức kết nối, mạng và cách thức giao tiếp được sử dụng với các thiết bị web-enabled này phần lớn phụ thuộc vào các ứng dụng IoT cụ thể được triển khai.

IoT cũng có thể tận dụng trí tuệ nhân tạo (AI) và máy học để giúp cho quá trình thu thập dữ liệu trở nên dễ dàng và chủ động hơn.



Hình 2: Cách thức IoT vận hành

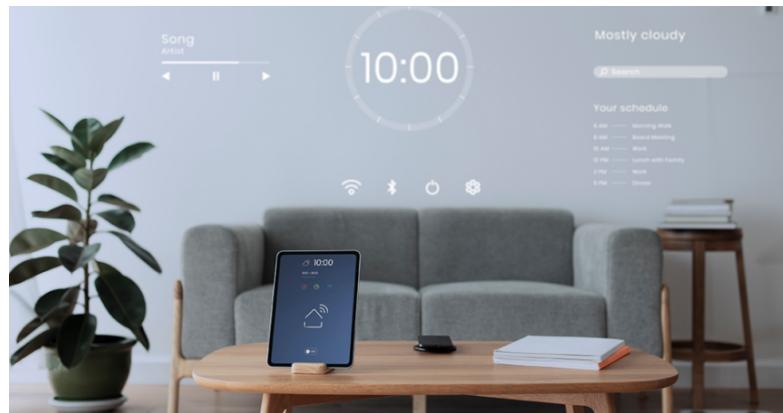


Ứng dụng IoT trong cuộc sống hiện đại.

• Nhà và văn phòng thông minh

Nhà thông minh là một ví dụ điển hình về IoT. Trong ngôi nhà thông minh các thiết bị điện tử dân dụng như đèn, quạt, máy lạnh... có thể được kết nối với nhau thông qua mạng internet. Sự kết nối này cho phép người dùng vận hành các thiết bị này từ xa. Một ngôi nhà thông minh có khả năng điều khiển ánh sáng, quản lý năng lượng, mở rộng và truy cập từ xa.

Hiện tại, ứng dụng IoT này không được sử dụng rộng rãi do chi phí lắp đặt quá cao, khó có khả năng chi trả.



Hình 3: Nhà thông minh

- **Thiết bị đeo được (Wearable)**

Đồng hồ thông minh là ví dụ tiêu biểu về các thiết bị đeo tay thông minh. Đồng hồ thông minh có khả năng đọc tin nhắn văn bản, hiển thị thông báo về các ứng dụng khác, theo dõi vị trí, theo dõi trạng thái tập luyện, nhắc nhở lịch trình và liên tục theo dõi tình trạng sức khỏe. Ngoài ra còn có một số thiết bị đeo được khác như: Kính thực tế ảo, vòng đeo thông minh, tai nghe không dây...



Hình 4: Đồng hồ thông minh

- Quản lý thiên tai

IoT giúp dự đoán và quản lý các thảm họa thiên nhiên. Lấy ví dụ về cháy rừng. Để tránh sự hỗn loạn và tàn phá do cháy rừng, nhiều cảm biến khác nhau có thể được lắp đặt xung quanh ranh giới của các khu rừng. Các cảm biến này liên tục theo dõi nhiệt độ và hàm lượng carbon trong khu vực. Báo cáo chi tiết sẽ được gửi đến trung tâm giám sát chung một cách thường xuyên. Trong trường hợp xảy ra cháy rừng, một cảnh báo được gửi đến phòng kiểm soát, dồn cảnh sát và đội cứu hỏa. Do đó, IoT giúp các cơ quan chức năng lập kế hoạch và phản ứng nhanh chóng trong trường hợp khẩn cấp.



Hình 5: Thiết bị cảnh báo cháy rừng

- Ô tô tự lái

Ô tô tự lái là một trong những dòng ô tô thông minh đã và đang phát triển với việc ứng dụng trí tuệ nhân tạo (AI) và công nghệ cảm biến thông minh trong IoT. Một trong những thành phần quan trọng của IoT trong ô tô là các cảm biến thông minh liên tục thu thập các thông tin về xe, tình trạng giao thông, các phương tiện khác và các đối tượng khác trên đường đi.

Hệ thống bao gồm các đơn vị camera, cảm biến khoảng cách, RADAR, mảng ăng-ten RF để thu thập thông tin và giúp xe đưa ra quyết định dựa trên những thay đổi đột ngột trên đường. Các phương tiện và đồ vật thông minh có thể chia sẻ thông tin với nhau bằng công nghệ RF.

Khi dữ liệu khổng lồ được thu thập, AI có thể dự đoán được các tình huống nhất định trên đường đi, cảnh báo tình trạng trên đường và phương tiện, hỗ trợ người lái xe an toàn, tránh va chạm.

Ô tô tự lái giúp hỗ trợ kiểm soát hành trình, quản lý nhiên liệu, thông báo có tai nạn trên tuyến đường, tình trạng giao thông đông đúc ở một tuyến đường cụ thể...



Hình 6: Ô tô tự lái

- **Phân tích dữ liệu lớn (Big Data Analytics)**

Một trong những thành phần cơ bản của phân tích dữ liệu lớn (Big data) chính là bản thân của dữ liệu đó; nhiều tổ chức coi dữ liệu là tài sản quý giá nhất để phát triển chiến lược kinh doanh của họ. Nguồn dữ liệu có thể từ bất kỳ đâu như máy móc, môi trường, thực vật, con người hoặc thậm chí động vật.

Internet of Things sử dụng hàng trăm loại cảm biến được thiết kế để thu thập dữ liệu từ nhiều loại ứng dụng. Lượng dữ liệu khổng lồ từ hàng triệu cảm biến thông minh sẽ giúp cho việc phân tích dữ liệu cần sử dụng trí tuệ nhân tạo và máy học cải thiện được các thuật toán ra quyết định.



Hình 7: Big data

- **Máy đọc mã vạch thông minh**

Máy đọc mã vạch IoT có thể giúp quản lý hàng tồn kho tốt hơn cho các nhà bán lẻ. Các đầu đọc hỗ trợ xử lý tín hiệu kỹ thuật số dựa trên AI. Những thiết bị này có thể tối ưu hóa hoạt động của nhiều lĩnh vực bao gồm bán lẻ, hậu cần, kho hàng...

Đầu đọc thẻ thanh toán dựa trên IoT có tính năng kết nối dữ liệu đám mây để kết nối với các hệ thống khác như phần mềm ERP có tích hợp QRCode, Barcode. Sử dụng đầu đọc mã vạch được kết nối sẽ giúp doanh nghiệp dễ dàng hơn trong quá trình quản lý hàng tồn kho.



Hình 8: Máy quét mã QR

- **Nông nghiệp thông minh**

Nếu như trước đây toàn bộ quá trình nông nghiệp đều phụ thuộc vào sức lao động con người thì giờ đây được đơn giản hóa nhờ sự xuất hiện của máy móc và công nghệ. Ứng dụng IoT trong ngành trồng trọt giúp nông dân có thể kiểm soát và nắm bắt những thông tin cần thiết như thời điểm tốt nhất thu hoạch, độ dinh dưỡng của đất, lượng phân bón phù hợp, độ ẩm của đất...

Những mô hình trang trại chăn nuôi thông minh cũng dần ra đời với những tiến bộ của IoT giúp cho người chủ kiểm soát và thu thập các dữ liệu cần thiết về nhiệt độ chuồng, độ ẩm không khí hoặc dữ liệu sức khỏe của vật nuôi... từ đó tiết kiệm sức lao động và nâng cao năng suất.



Hình 9: Hệ thống tưới nước tự động

• Công nghiệp sản xuất

Công nghiệp sản xuất là một trong những ngành sớm áp dụng IoT, nó đã thay đổi hoàn toàn một số giai đoạn của chu kỳ phát triển sản phẩm. IoT công nghiệp sẽ giúp tối ưu hóa các công đoạn sản xuất khác nhau của sản phẩm như:

- Giám sát chuỗi cung ứng và quản lý hàng tồn kho.
- Tối ưu hóa trong phát triển sản phẩm.
- Tự động hóa quy trình sản xuất hàng loạt.
- Kiểm tra chất lượng và cải tiến sản phẩm.
- Cải thiện việc đóng gói và quản lý.
- Tối ưu hóa quy trình bằng cách sử dụng dữ liệu được thu thập từ số lượng lớn các mạng cảm biến.
- Giải pháp hiệu quả về chi phí để quản lý tổng thể nhà máy.



Hình 10: Thiết bị sản xuất

3.2 Ý tưởng về vườn thông minh

3.2.1 Chăm sóc vườn cây thực phẩm

Trong những năm qua, việc trồng thực phẩm tại các khu vực thành thị đã trở thành một xu hướng toàn cầu, và nó tiếp tục phát triển. Nó đã trở nên hiện đại, tạo thành xu thế, nó đang được khẳng định là tương lai của thực phẩm, và những thương hiệu làm vườn thông minh mới xuất hiện nhanh hơn bao giờ hết. Những lợi ích mà vườn thông minh mang lại là không thể phủ nhận:

• Hoạt động hiệu quả hơn

- Với sự giúp đỡ của nông nghiệp thông minh, người ta có thể sản xuất thậm chí nhiều gấp 100 lần thực phẩm hơn so với canh tác truyền thống (mỗi mét vuông). Làm thế nào là nó có thể? Hầu hết các trang trại đô thị được thiết kế theo chiều dọc, cho phép dễ phát triển sản phẩm ở nhiều cấp độ.

- Thời tiết có tác động khá nhiều và làm giảm năng suất trong một năm nhất định. Nhưng trong hệ thống nông nghiệp thông minh, thời tiết là yếu tố thực sự rất nhỏ đối với sản xuất lương thực vì nó thường diễn ra trong nhà và dựa vào hệ thống nước được lắp đặt bao gồm ánh sáng nhân tạo và các chất dinh dưỡng đã được cung cấp trong đất thông qua các hệ thống đo dưỡng chất, độ PH, độ ẩm...



Hình 11: Tăng năng suất bằng cách trồng lên các nông trại thăng

- **Tình hình tài chính bền vững hơn**

- Hầu hết các hệ thống làm vườn thông minh đều tiết kiệm nước, năng lượng và không gian đáng kể. Nếu chúng ta nói về các hệ thống canh tác đô thị Click & Grow, họ sử dụng khoảng 90% nước ít hơn và 4 lần không gian hơn, nếu so sánh với canh tác truyền thống. Nhiều người chỉ ra rằng đầu tư một trang trại ở đô thị có thể tốn kém. Trên thực tế, một trang trại ở đô thị với \$ 500 tiền đầu tư sẽ "hoàn vốn" trong một năm.

- Bên cạnh đó, tiền đầu tư không lỗ phí, nó mang lại nhiều hiệu quả về mặt tài chính, giúp tài chính của chúng ta bền vững. Điều này được chứng minh bằng việc tiết kiệm nước, điện và không gian. Ngoài ra chúng ta còn góp phần bảo vệ môi trường sống chẳng hạn như làm giảm sự phát triển của hạn hán, xói mòn đất và các vấn đề tương tự.

- **Hạn chế sử dụng các chất hóa học đối với thực phẩm**

- Nông nghiệp thông minh buộc nông dân phải trồng cây trồng một cách có kiểm soát và có ý thức hơn, dẫn đến việc trồng cây đơn thuần mà không cần đầu tư thêm. Làm thế nào là nó có thể? Lý do là vì nông dân buộc phải sử dụng thuốc trừ sâu hóa học để đảm bảo sản lượng tốt khi các yếu tố môi trường tác động xấu đối với cây trồng. Như chúng ta đã đề cập ở trên, thời tiết, yếu tố môi trường được giảm xuống tối thiểu ở nông nghiệp thông minh, do đó không có nhu cầu thực sự sử dụng các chất điều chỉnh tăng trưởng hóa học.



Hình 12: Hệ thống canh tác truyền thống sử dụng nhiều chất hóa học

- **Cải thiện không gian sống, cung cấp lương thực sạch quanh năm và có cấu hình đơn giản**

- Nông nghiệp thông minh có thể áp dụng ở nhiều nơi, kể cả những nơi có diện tích hạn chế, có thể sử dụng ngay trong nhà. Việc áp dụng nông nghiệp thông minh trong nhà sẽ cho phép bạn có thể tự quản lý ngay chính “vườn thông minh” của chính mình, điều chỉnh hợp lý cho loại cây trồng mà chủ nhà áp dụng... Từ đó, việc sử dụng lương thực sạch cụ thể là rau, củ... sạch sẽ không còn là điều khó khăn do “vườn thông minh” của bạn có thể cung cấp lương rau củ sạch quanh năm. Bên cạnh đó, việc đầu tư và lắp đặt vô cùng đơn giản nhưng đem lại hiệu quả vô cùng to lớn.



Hình 13: Canh tác thực phẩm trong nhà

3.2.2 Ý tưởng về vườn thông minh

- **Tình hình nghiên cứu ngoài nước**

Có thể nói Israel, Singapore và Nhật Bản là những quốc gia đi tiên phong trong phát triển nông nghiệp cao với những công nghệ chăm sóc cây trồng vô cùng độc đáo như hệ thống tưới nhỏ giọt, hệ thống tưới phun sương, hệ thống tưới phun mưa, hệ thống tưới cảnh quan và ngay cả hệ thống thủy canh trồng cây không dùng đất. Và gần đây nhất hướng đến nhu cầu trồng cây trên những diện tích nhỏ hẹp nên có thêm giải pháp trồng rau xanh trên những nông trại thẳng đứng "Vertical farm" đang là xu hướng được quan tâm tại nhiều nơi hiện nay bởi vừa tận dụng tối đa diện tích canh tác, vừa cho rau sạch với sản lượng lớn, phục vụ nhu cầu cho nhiều hộ gia đình.



Hình 14: Trang trại thẳng đứng tại Nhật Bản

Tóm lại, hiện nay nước ngoài đã áp dụng hình thức "Vườn thông minh" từ rất lâu và cho rất nhiều nơi và ứng dụng này thậm chí còn được sử dụng tại ngay những hộ gia đình nhỏ. Chính vì sự tiện lợi mà nó đem lại nên nó dần trở thành ứng dụng khá phổ biến hiện nay trên toàn thế giới.

- **Tình hình nghiên cứu trong nước**

Nước ta hiện nay thuật ngữ nông nghiệp hiện đại không còn xa lạ gì với nhiều người nhưng do chi phí đầu tư hệ thống chăm sóc cây tự động khá cao do phải nhập chu yến từ nước ngoài, dẫn đến việc nhân rộng mô hình chưa phổ biến, và hạn chế những hệ thống chăm sóc cây tự động chủ yếu hướng tới đối tượng người dùng phải có diện tích trồng cây rộng lớn, chưa chú trọng đến thị trường người dùng có diện tích trồng eo hẹp.

Trong nước cũng có rất nhiều nghiên cứu như hệ thống tưới cây tự động của kỹ sư Vi Toàn Nghĩa 2013, sau đó là hàng loạt công trình nghiên cứu hệ thống trồng cây điều khiển từ xa, hệ thống trồng cây tự động hướng đến cả phân khúc người sử dụng có quỹ tích eo hẹp. Nhưng hạn chế những nghiên cứu chưa tận dụng hạ tầng internet hiện có để mở rộng người dùng với hệ thống trồng cây có giám sát tự động từ xa, hướng đến cả người dùng không có diện tích trồng eo hẹp, không có thời gian chăm sóc nhưng vẫn có vườn au an toàn cho riêng mình.



Hình 15: Trang trại thẳng đứng tại Nhật Bản

Từ những đánh giá trên, chúng em quyết định làm một hệ thống quản lý “Vườn thông minh” với tính năng đáp ứng được cơ bản nhu cầu của người sử dụng.

4 Cơ sở lý thuyết

4.1 Các nhân tố tác động tới cây trồng

• Ánh sáng

Ánh sáng giúp cho việc quang hợp của cây và thông qua quá trình đó mà tạo ra các chất hữu cơ, các hoạt chất trong dược liệu. Thiếu ánh sáng cây mọc chậm, yếu ớt, sinh trưởng không bình thường, lá mỏng, không ra hoa, hoặc ra hoa không đều, song nếu ánh sáng quá mạnh thì lá nhỏ, phiến lá dày, hoa cũng biến sắc.

Cường độ ánh sáng và nhu cầu ánh sáng của mỗi loài cây thuốc là khác nhau. Để tạo điều kiện thích hợp cho cây sinh trưởng và phát triển tốt phải căn cứ vào yêu cầu ánh sáng đối với cây, được chia theo các nhóm sau:



- Nhóm cây ưa sáng: Là đa số nhóm cây có tinh dầu như Bạc Hà, Hương Nhu, Ba gạc, Hoắc Đắng, Ngũ gia hương bì, Sinh Địa. Dối với những loại cây này, không nên trồng xen, mật độ thưa hợp lý, không trồng nơi có nhiều bóng cây.



- Nhóm cây ưa sáng trung bình: Hoắc hương, Gừng, Nghệ ưa râm mát, nếu trồng ở chỗ có ánh nắng trực tiếp thì lá vàng, không thơm, cây phát triển kém. Cây sinh địa ưa nắng, nếu trồng trong vườn rợp cây chết lui, củ nhỏ, những cây này có thể trồng xen hợp lý.



- Nhóm cây ưa ánh sáng yếu: Yêu cầu ánh sáng tán xạ với cường độ ánh sáng khác nhau. Tam thất, sâm Ngọc Linh phải trồng dưới tán rừng hoặc dưới dàn mái che. Ba Kích trồng trong tán rừng có độ che phủ 0,3 – 0,5; Thảo quả yêu cầu độ che phủ 0,5 – 0,6; Sa nhân có độ che phủ 0,4 – 0,5 (so với ánh sáng trực xạ), nên trồng xen dưới các cây ăn quả, rừng Luồng, Quế, Trúc, mổi sinh trưởng sẽ cho dược liệu tốt.

Lưu ý: Cùng một cây với thời gian sinh trưởng khác nhau thì yêu cầu ánh sáng cũng sáng cũng khác nhau. Ví dụ: Quế, Hồi, Chè,... dưới 4 năm tuổi cần ít ánh sáng (che bóng), trên 4 năm tuổi lại cần nhiều ánh sáng. Thông thường yêu cầu ánh sáng gắn liền với độ cao địa hình.

Thời gian chiếu sáng (quang chu kỳ) cũng là một yếu tố quan trọng dùng làm cơ sở để bố trí thời vụ thích hợp. Một số nhóm cây được phân loại theo thời gian chiếu sáng như:

- Nhóm cây phản ứng với ánh sáng ngày ngắn (ra hoa khi độ dài ngày chiếu sáng trong ngày nhỏ hơn 11,45 giờ) như: Cúc hoa, Dậu tương, Day,...

- Nhóm cây phản ứng với ánh sáng ngày dài (ra hoa khi độ dài chiếu sáng trong ngày dài hơn 11,45 giờ): Bạc Hà, Ích Mẫu,...

- Nhóm cây phản ứng trung tính với ánh sáng: là những cây ra hoa không phụ thuộc vào độ dài chiếu sáng trong ngày, chủ yếu theo thời gian sinh trưởng, phát triển của cây như: Địa Hoàng, Dương Quy, Bạch Chỉ, Quế, Thảo quả, Sâm Ngọc Linh,...

Đa số cây thuốc đều ưa ánh sáng cũng như ít mầm cảm với độ chiếu sáng dài ngày, song đó nhu cầu ánh sáng khác nhau nên cần chú ý xác định thời vụ, mật độ, với từng cây thuốc cho thích hợp mới đạt được năng suất, chất lượng dược liệu tốt.

• Nhiệt độ

Mỗi loài cây thuốc đều có nguồn gốc từ những vùng sinh thái khác nhau, có điều kiện sinh trưởng và phát triển trong một giới hạn nhất định về nhiệt độ (tổng nhiệt độ, tổng tích ôn, nhiệt độ thấp, nhiệt độ tối ưu). Trong quá trình sinh trưởng, phát triển nhiệt độ ảnh hưởng rõ rệt ở các thời kỳ sau:

- Nhiệt độ ảnh hưởng đến quá trình mọc mầm:

VD: Ngưu tất gieo cần nhiệt độ 20 – 25°C, hạt sẽ mọc mầm sau 7 – 10 ngày, nếu nhiệt độ thấp 10 – 13°C phải sau 10 – 15 ngày cây mới mọc. Hạt Sa sâm phải gieo vào mùa đông nhiệt



độ $<12^{\circ}\text{C}$) mới mọc. Dương quy mọc mầm thích hợp ở 20°C , nếu nhiệt độ lớn hơn 25°C tỷ lệ mọc mầm thấp ($<50\%$).

- Nhiệt độ ảnh hưởng đến sinh trưởng của cây:

Đối với những cây yêu cầu một tổng tích ôn nhất định để sinh trưởng, cùng một loại nếu trồng ở miền núi cây sẽ có thời gian sinh trưởng dài hơn ở đồng bằng.

- Dương quy trồng ở Hà Giang yêu cầu hơn 1 năm để cho thu hoạch (> 12 tháng). Tuy nhiên, nếu cây đem trồng ở đồng bằng thì chỉ sau 6 – 7 tháng đã cho thu hoạch. Cát Cánh trồng ở miền núi phải 2 năm mới cho thu hoạch, trong khi đó nếu trồng ở đồng bằng 1 năm đã cho thu hoạch.

- Cây ích mẫu trồng ở đồng bằng, nếu gieo muộn vào tháng 4 gặp nắng nóng, sau trồng 1 tháng (tháng 5) cây chỉ cao thêm 30cm đã ra hoa cần phải thu hoạch ngay, năng suất không đáng kể. Vì vậy cần chọn cho từng loại cây thời vụ trồng thích hợp và cần có biện pháp chống nắng như che vường ướm, phủ tro hoặc rơm rạ...

Nhìn chung mỗi loại cây thuốc chỉ có thể sinh trưởng và phát triển trong một giới hạn nhất định về nhiệt độ không khí và nhiệt độ đất. Da số cây thuốc ưa nhiệt độ ôn hòa, thích hợp trong khoảng nhiệt độ từ $18 - 28^{\circ}\text{C}$.

• Độ ẩm

Cần chú ý 2 loại độ ẩm: Độ ẩm đất và độ ẩm không khí, cả 2 loại này đều cần thiết cho sự sinh trưởng và phát triển của cây thuốc. Cần chú ý đến lượng mưa và sự phân bố mưa trong năm. Nếu thiếu ẩm mặt đất và cây trồng đều tăng cường thoát hơi nước dẫn đến cây bị khô héo cằn cỗi. Tuy nhiên ở từng thời kỳ sinh trưởng cây thuốc có những yêu cầu khác nhau về độ ẩm. Lúc mới gieo trồng cây còn non yếu phải có đủ ẩm thường xuyên, nhưng khi cây ra hoa kết hạt nếu độ ẩm quá cao sẽ làm cho hoa nở ít, hạt lép. Điều kiện thời tiết khô ráo sẽ cho hoa nở đều tập trung nhiều quả, chắc hạt rất có ý nghĩa tối chất lượng các hoạt chất trong cây thuốc.

- Da số cây thuốc đều ưa ẩm nhưng lại sợ úng, đặc biệt đối với cây lấy củ dễ thối. Nếu trời mưa liên tục độ ẩm không khí cao, sâu bệnh nhiều, củ, rễ, hoa quả bị thối.

- Úng nước thì cây chết như cây Thanh Hao, Bạc Hà, 2 – 3 ngày úng dẫn đến cây chết. Bạch chỉ, Sinh địa bị úng thì thối toàn bộ củ. Do vậy vào mùa mưa cần khơi rãnh thoát nước lên luồng cao (cho cây lấy củ). Căn cứ vào mức độ chịu hạn chia ra:

+ Cây chịu hạn tốt: như Sả, Bạch đàn, Sim, Mua. Trong cùng loài thì khả năng chịu hạn cũng khác nhau. Sả Srilabca chịu hạn hơn sả Java, nhưng cho năng suất tinh dầu thấp hơn.

+ Cây ưa ẩm: Mân tưới, Mã đề, Lô hội,...

+ Cây mọc ở ao nước sinh lầy: Trạch tả, Thủy xương bồ....

Nhìn chung nhiều loài cây thuốc có nhu cầu độ ẩm thích hợp không quá khô hoặc không quá ẩm. Cần một lượng vừa phải khoảng 1.500 – 2.000mm và phân bố đều trong năm.

• Đất

Chức năng của đất là làm nền cho cây mọc. Chúng giữ gìn, cung cấp chất dinh dưỡng cần thiết, nước và không khí cho cây, tạo điều kiện thuận lợi cho cây sinh trưởng và phát triển.

Đất tốt sẽ thực hiện được cả ba chức năng trên. Đất tốt có kết cấu tốt, có độ ẩm tối ưu, giàu chất dinh dưỡng và hoạt động sinh học cao.

• Nước

Hòa tan dinh dưỡng trong đất: Cây trồng sống chủ yếu nhờ các chất dinh dưỡng có trong đất. Tuy nhiên, rễ của cây không thể hấp thụ các muối khoáng, đậm tự nhiên trong đất nếu không có nước.

Vận chuyển dinh dưỡng đến từng bộ phận của cây: Nước trong thân cây đóng vai trò là dòng chảy đưa các chất dinh dưỡng này đi từ gốc cây lên tới các bộ phận của cây như lá, chồi, ngọn,

hoa quả.

Cấu thành các tế bào của cây: Gần 90% khối lượng của cây trồng là nước nên nước là thành phần chủ chốt trong các tế bào của cây. Thiếu nước, các tế bào này có thể bị co lại, không thể hoạt động và thậm chí hình dáng sẽ dị dạng, quăn queo và phát triển èo uột dẫn đến năng suất cây trồng giảm sút.

- **Chất khoáng**

Cây trồng cần 3 thành phần dinh dưỡng chủ yếu: các chất dinh dưỡng cơ bản (NPK), đa lượng và vi lượng.

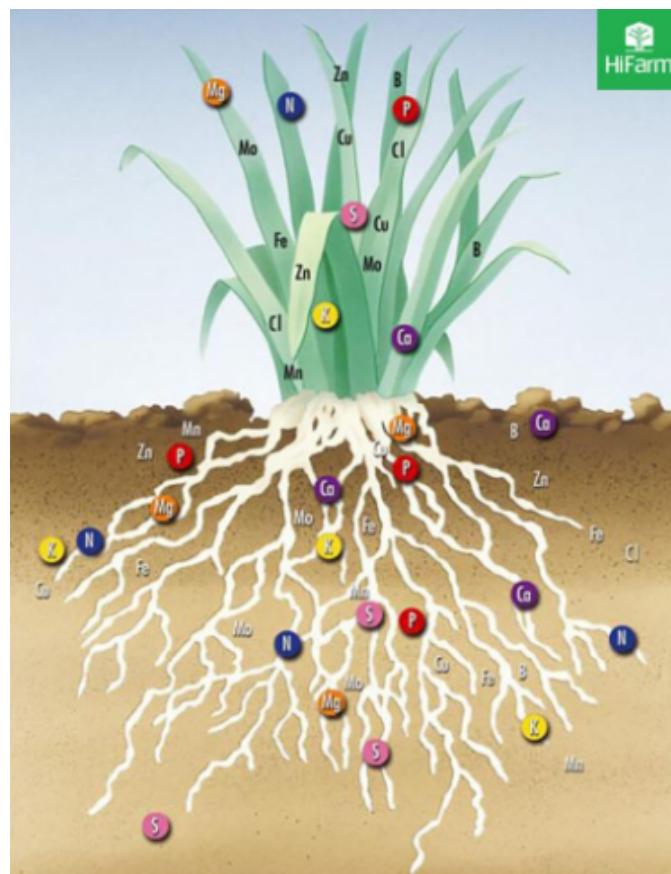
- Dinh dưỡng cơ bản cho cây trồng (NPK)**

- **Nito:** là chất dinh dưỡng quan trọng giúp cây phát triển khỏe mạnh, lá xanh đậm và quang hợp tốt.

- **Phốt pho:** Giúp rễ tăng trưởng và phát triển. Cây đủ phốt pho sẽ cho nhiều hoa hơn.

- **Kali:** Cải thiện sức khỏe tổng thể của cây, giúp cây chống lại bệnh tật.

- Khoáng chất đa lượng**



- **Canxi:** Thúc đẩy rễ và chồi non phát triển, giúp cây xây dựng các thành tế bào.

- **Magie:** Giúp điều chỉnh sự hấp thu các dinh dưỡng khác. Hỗ trợ trong việc hình thành hạt giống và chất diệp lục.

- **Lưu huỳnh:** Giúp duy trì màu xanh đậm, kích thích sự phát triển của cây trồng. Lưu

huỳnh cung cần thiết để sản xuất chất diệp lục.

Khoáng chất vi lượng

- **Boron:** Giúp phát triển tế bào và điều chỉnh sự trao đổi chất của thực vật. Cụ thể: giúp tổng hợp protein, phát triển thành tế bào, chuyển hóa carbohydrate. Chuyển vị đường, kiểm soát hormone, thụ tinh của hạt phấn hoa và tăng trưởng số lượng quả.
- **Clo:** Một chất cần thiết để trao đổi khí, quang hợp và bảo vệ cây chống lại bệnh tật.
- **Đồng:** Giúp hình thành chất diệp lục, kích hoạt các enzyme trong cây, giúp tổng hợp lignin.
- **Sắt:** Giúp cây sản xuất chất diệp lục và các quá trình sinh hóa khác.
- **Mangan:** Là chất cần thiết để sản xuất chất diệp lục.
- **Kẽm:** Giúp phát triển enzyme và hormone. Đồng thời giúp cây sản xuất chất diệp lục.

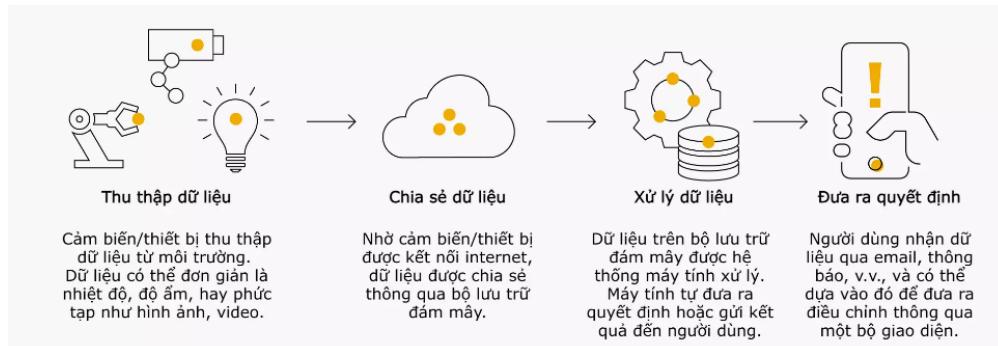
4.2 Hệ thống IoT gồm những thành phần gì?

Một hệ thống IoT sẽ bao gồm 4 thành phần chính đó là:

- **Things:** Thiết bị.
- **Gateway:** Trạm kết nối.
- **Network and Cloud:** Hạ tầng mạn.
- **Services-creation and Solution Layers:** Bộ phân tích và xử lý dữ liệu.

Hệ thống IoT bao gồm các thiết bị thông minh hỗ trợ web sử dụng hệ thống nhúng, như bộ xử lý, cảm biến và phần cứng truyền thông, để thu thập, gửi và xử lý trên dữ liệu mà chúng thu thập được. Các thiết bị IoT chia sẻ dữ liệu cảm biến thu thập được bằng cách kết nối với cổng IoT hoặc thiết bị biên khác, nơi dữ liệu được gửi đến đám mây để phân tích hoặc phân tích cục bộ.

Mỗi hệ thống IoT hoàn chỉnh đều có 4 bước: Thu thập, chia sẻ, xử lý dữ liệu và đưa ra quyết định.



Hình 16: Cách thức hoạt động cơ bản của IoT

Đôi khi, các thiết bị này giao tiếp với các thiết bị liên quan khác và hoạt động dựa trên thông tin chúng nhận được từ nhau. Các thiết bị thực hiện hầu hết công việc mà không có sự can thiệp của con người, mặc dù mọi người có thể tương tác với các thiết bị.

4.3 Node Sensor là gì?

Mạng WSNs gồm nhiều cảm biến phân bố phân tán phủ một vùng địa lý. Các node (sensor nodes hay còn gọi là WNs) có khả năng liên lạc vô tuyến với các node lân cận và các

chức năng cơ bản như xử lý tín hiệu, quản lý giao thức mạng và bắt tay với các node lân cận để truyền dữ liệu từ nguồn đến trung tâm. Chức năng cơ bản của các node trong mạng WSNs phụ thuộc vào ứng dụng của nó, một số chức năng chính:

- Xác định được giá trị các thông số tại nơi lắp đặt. Như có thể trả về nhiệt độ, áp suất, cường độ ánh sáng... tại nơi khảo sát.

- Phát hiện sự tồn tại của các sự kiện cần quan tâm và ước lượng các thông số của sự kiện đó.

Như mạng WSN dùng trong giám sát giao thông, cảm biến phải nhận biết được sự di chuyển của xe cộ, do được tốc độ và hướng di chuyển của các phương tiện đang lưu thông...

- Phân biệt các đối tượng. Ví dụ phương tiện lưu thông mà cảm biến nhận biết được là gì xe con, xe tải, hay xe buýt...

- Theo dõi các đối tượng. Ví dụ trong mạng WSN quân sự, mạng cảm biến phải cập nhật được vị trí các phương tiện của đối phương khi chúng di chuyển trung vùng bao phủ của mạng....

Liên quan đến thiết kế node trong mạng WSNs, các chức năng cần phải có: chức năng cơ bản của node; chức năng xử lý tín hiệu, gồm xử lý số tín hiệu, nén, phát hiện và sửa lỗi, điều khiển và thừa hành; phân nhóm và tính toán trong mạng; thông tin; tự kết hợp; định tuyến; và quản lý kết nối. Để có các chức năng này, phần cứng của node phải có cảm biến và bộ phận thực thi, bộ xử lý, nguồn, và các phần phục vụ cho chức năng khác. Vậy nên cấu tạo của node cảm biến thông thường gồm có 2 phần : **phần cứng** và **phần mềm**.

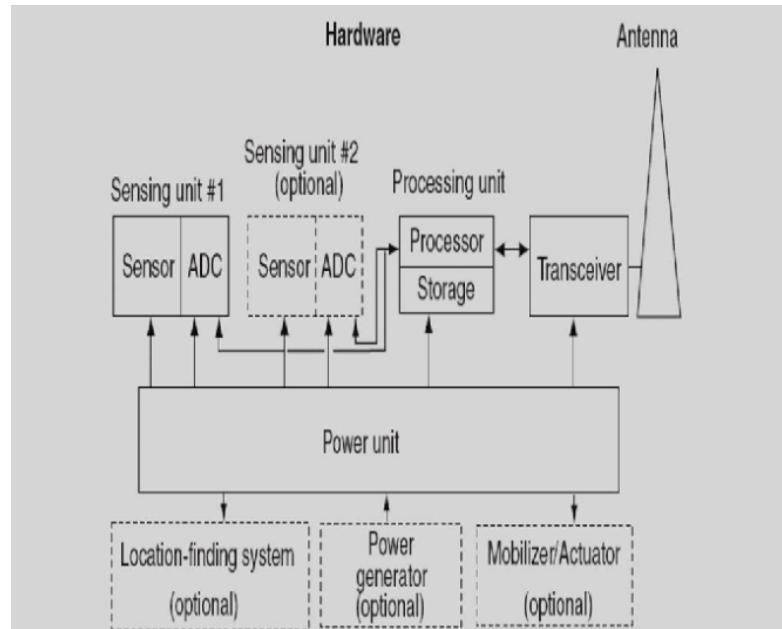
Phần cứng: 4 nhóm chính

+ **Nguồn cung cấp**: đảm bảo năng lượng cho node hoạt động trong vài giờ, vài tháng hay vài năm

+ **Lưu trữ và tính toán**: phục vụ cho các chức năng xử lý, điều chế số, định tuyến...

+ **Cảm biến**: biến đổi các thông số môi trường thành thông tin

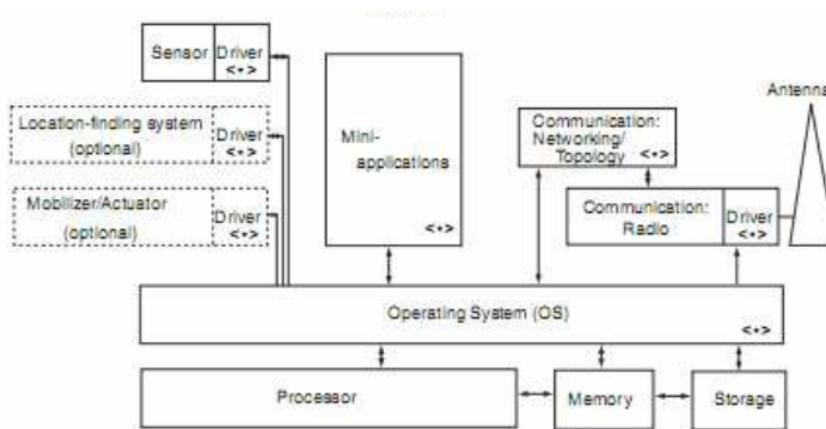
+ **Liên lạc**: trao đổi dữ liệu giữa các node với nhau và với trung tâm



Hình 17: Hardware

Phần mềm: 5 nhóm chính

- + **Hệ điều hành (OS) microcode** (còn gọi là middleware): liên kết phần mềm và chức năng bộ xử lý. Các nghiên cứu hướng đến thiết kế mã nguồn mở cho OS dành riêng cho mạng WSNs
- + **Sensor Drivers**: đây là những module quản lý chức năng cơ bản của phần tử cảm biến.
- + **Bộ xử lý thông tin**: quản lý chức năng thông tin, gồm định tuyến, chuyển các gói, duy trì giao thức, mã hóa, sửa lỗi,...
- + **Bộ phận xử lý dữ liệu**: xử lý tín hiệu đã lưu trữ, thường ở các node xử lý trong mạng



Hình 18: Software

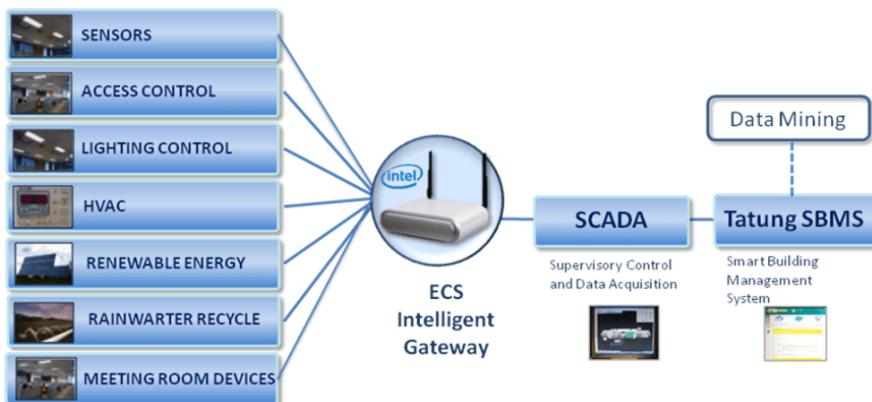
4.4 Gateway là gì?

Gateway là một nút mạng được sử dụng trong viễn thông nhằm kết nối hai mạng có giao thức truyền thông khác nhau có thể giao tiếp được với nhau. Gateway có vai trò xử lý đầu vào và ra của mạng vì tất cả dữ liệu phải đi qua hoặc giao tiếp với gateway trước khi được định tuyến. Trong hầu hết các mạng IP, lưu lượng duy nhất không đi qua gateway là lưu lượng truyền giữa các nút trên cùng một phân đoạn mạng cục bộ (LAN). Thuật ngữ default gateway hoặc network gateway cũng có thể được sử dụng để mô tả khái niệm trên.

Gateway được sử dụng chủ yếu trong các tình huống cá nhân hoặc doanh nghiệp muốn đơn giản hóa việc kết nối internet cho một thiết bị. Trong doanh nghiệp, một gateway cũng có thể hoạt động như một máy chủ proxy và tường lửa.

Tất cả các mạng đều có một ranh giới và giới hạn giao tiếp với các thiết bị được kết nối trực tiếp với nó. Do đó, nếu một mạng muốn giao tiếp với các thiết bị, nút hoặc mạng bên ngoài ranh giới đó, chúng yêu cầu chức năng của một gateway. Một gateway thường có những đặc trưng của sự kết hợp giữa router và modem.

Gateway hoạt động bên cạnh một mạng và quản lý tất cả dữ liệu được chuyển hướng nội bộ hoặc bên ngoài từ mạng đó. Khi một mạng muốn giao tiếp với mạng khác, gói dữ liệu được chuyển đến gateway và sau đó được chuyển đến đích thông qua đường truyền hiệu quả nhất.



Hình 19: Cách thức hoạt động của Gateway

Ngoài dữ liệu định tuyến, một gateway cũng sẽ lưu trữ thông tin về đường dẫn nội bộ của mạng chủ và đường dẫn của bất kỳ mạng bổ sung nào gặp phải.

Gateway về cơ bản là bộ chuyển đổi giao thức, tạo điều kiện tương thích giữa hai giao thức và hoạt động trên bất kỳ lớp nào của mô hình kết nối hệ thống mở (OSI).

Ví dụ: mạng của bạn sử dụng giao thức IP và mạng của ai đó sử dụng giao thức IPX, Novell, DECnet, SNA... hoặc một giao thức nào đó thì Gateway sẽ chuyển đổi từ loại giao thức này sang loại khác.

Các loại Gateway: Gateway có nhiều loại, nhiều định dạng và thực hiện nhiều nhiệm vụ khác nhau:

- **Web application firewalls (tường lửa web):** loại này lọc lưu lượng truy cập đến và từ máy chủ web và xem xét dữ liệu lớp ứng dụng.
- **Cloud storage gateways (lưu trữ đám mây):** loại này dịch các yêu cầu lưu trữ với các lệnh gọi API dịch vụ lưu trữ đám mây khác nhau. Nó cho phép các tổ chức tích hợp lưu trữ từ private cloud vào các ứng dụng mà không cần di chuyển vào public cloud.
- **API, SOA hay XML gateways:** loại này quản lý lưu lượng truy cập vào và ra khỏi dịch vụ, kiến trúc hướng dịch vụ vi mô hoặc dịch vụ web dựa trên nền tảng XML.
- **IoT gateways:** loại này tổng hợp dữ liệu từ các thiết bị cấp hiện trường (cảm biến,..) trong môi trường IoT, chuyển đổi giữa các giao thức cảm biến và xử lý dữ liệu cảm biến trước khi gửi đi.
- **Media gateways:** loại này chuyển đổi dữ liệu từ định dạng của một loại mạng sang định dạng cho một loại mạng khác.
- **Email security gateways (bảo mật email):** loại này ngăn chặn việc truyền các email vi phạm chính sách của công ty hoặc chuyển thông tin với mục đích xấu.
- **VoIP trunk gateways:** loại này tạo sự thuận lợi cho việc sử dụng các thiết bị dịch vụ điện thoại cũ đơn thuần, chẳng hạn như điện thoại cố định và máy fax, với mạng thoại qua IP (VoIP).
- v.v..

4.5 Server là gì?

Server hay còn gọi là máy chủ là một hệ thống (phần mềm và phần cứng máy tính phù hợp) đáp ứng yêu cầu trên một mạng máy tính để cung cấp, hoặc hỗ trợ cung cấp một dịch vụ mạng. Các server có thể chạy trên một máy tính chuyên dụng, mà cũng thường được gọi là "máy chủ", hoặc nhiều máy tính nối mạng có khả năng máy chủ lưu trữ. Trong nhiều trường hợp, một máy tính có thể cung cấp nhiều dịch vụ và dịch vụ chạy đa dạng.



Hình 20: Server

Các máy chủ thường hoạt động trong một mô hình client-server, server (máy chủ) là các chương trình máy tính đang chạy để phục vụ yêu cầu của các chương trình khác, các client (khách hàng). Do đó, các máy chủ thực hiện một số nhiệm vụ thay mặt cho khách hàng. Các khách hàng thường kết nối với máy chủ thông qua mạng nhưng có thể chạy trên cùng một máy tính. Trong hệ thống hạ tầng của mạng Internet Protocol (IP), một máy chủ là một chương trình hoạt động như một socket listener (giao thức nghe).

Các máy chủ thường cung cấp các dịch vụ thiết yếu qua mạng, hoặc là để người dùng cá nhân trong một tổ chức lớn hoặc cho người dùng nào thông qua Internet. Các máy chủ máy tính điển hình là máy chủ cơ sở dữ liệu (database server), máy chủ tập tin (file server), máy chủ mail (mail server), máy chủ in (print server), máy chủ web (web server), máy chủ game (game server), máy chủ ứng dụng (application server), hoặc một số loại khác của máy chủ.

Nhiều hệ thống sử dụng mô hình client/server mạng này bao gồm các trang web và các dịch vụ email. Một mô hình thay thế, mạng peer-to-peer cho phép tất cả các máy tính để hoạt động như một trong hai (client hoặc server) khi cần thiết.

Ví dụ: trong một văn phòng có 10 máy tính, nếu mỗi máy tính độc lập muốn in ấn một cái gì đấy thì mỗi máy đó phải tự kết nối đến máy in hay phải cài driver cho 10 máy, chưa kể còn phải rút ra rút vào dây cáp kết nối giữa máy in và máy tính nhiều lần. Nhưng sử dụng server thì chúng ta không cần phải làm thế, chỉ cần kết nối máy in với máy chủ rồi nối mạng tất cả các máy còn lại với máy chủ là các máy cá nhân kia không cần phải in độc lập nữa.

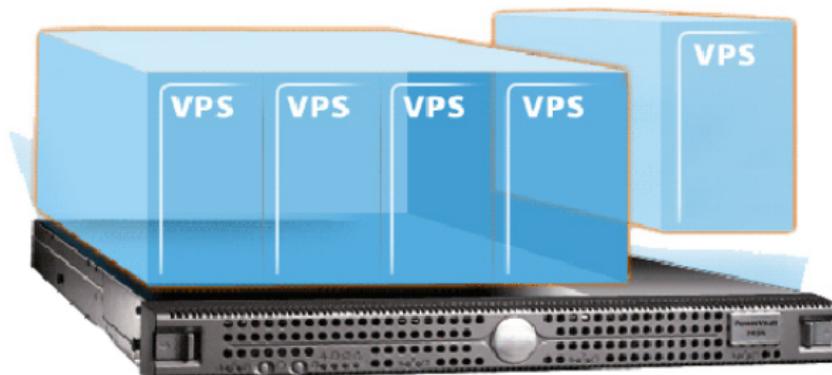
Cơ sở để phân loại các loại máy chủ là dựa theo phương pháp chế tạo ra máy chủ, ta có 3 loại máy chủ thường gặp sau:

- **Máy chủ riêng (Dedicated):** là máy chủ chạy trên phần cứng và các thiết bị hỗ trợ riêng biệt gồm: HDD, CPU, RAM, Card mạng,... hay còn gọi với cái tên khác là Máy chủ vật lý. Việc nâng cấp hoặc thay đổi cấu hình của máy chủ riêng đòi hỏi phải thay đổi phần cứng của máy chủ, việc này cần những người có kiến thức chuyên sâu về phần cứng và máy chủ mới có thể đảm bảo được những linh kiện tạo ra máy chủ.



Hình 21: Máy chủ riêng

- **Máy chủ ảo (Virtual Private Server – VPS):** là loại máy chủ được tách từ máy chủ vật lý kể trên ra bằng phương pháp sử dụng công nghệ ảo hóa. Từ một máy chủ riêng, có thể tách được thành nhiều máy chủ ảo khác nhau có chức năng như máy chủ vật lý và chia sẻ tài nguyên trên máy chủ vật lý gốc.



Hình 22: Máy chủ ảo

- **Máy chủ đám mây (Cloud Server):** là máy chủ được kết hợp từ nhiều máy chủ vật lý gốc khác nhau cùng với hệ thống lưu trữ SAN và máy chủ đám mây được xây dựng trên nền tảng công nghệ điện toán đám mây.



Hình 23: Máy chủ đám mây

4.6 MQTT và MQTT mosquitto

4.6.1 MQTT

MQTT (Message Queuing Telemetry Transport) là giao thức truyền thông điệp (message) theo mô hình publish/subscribe (cung cấp / thuê bao), được sử dụng cho các thiết bị IoT với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong mạng lưới không ổn định. Nó dựa trên một Broker (tạm dịch là “Máy chủ môi giới”) “nhẹ” (khá ít xử lý) và được thiết kế có tính mở (tức là không đặc trưng cho ứng dụng cụ thể nào), đơn giản và dễ cài đặt.

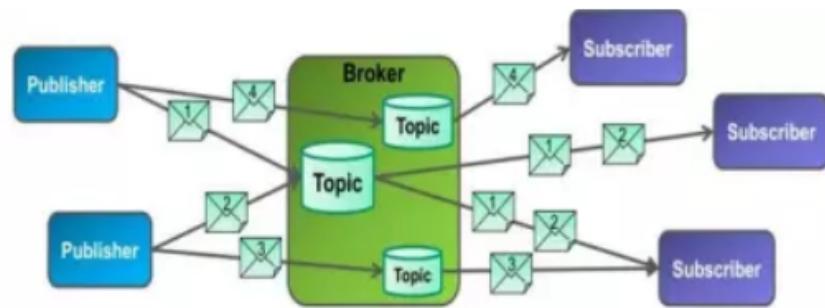
MQTT là lựa chọn lý tưởng trong các môi trường như:

- Những nơi mà giá mạng viễn thông đắt đỏ hoặc băng thông thấp hay thiếu tin cậy.
- Khi chạy trên thiết bị nhúng bị giới hạn về tài nguyên tốc độ và bộ nhớ.
- Bởi vì giao thức này sử dụng băng thông thấp trong môi trường có độ trễ cao nên nó là một giao thức lý tưởng cho các ứng dụng M2M (Machine to Machine).

Cơ chế MQTT:

MQTT hoạt động theo cơ chế client/server, nơi mà mỗi cảm biến là một khách hàng (client) và kết nối đến một máy chủ, có thể hiểu như một Máy chủ môi giới (broker), thông qua giao thức TCP (Transmission Control Protocol). Broker chịu trách nhiệm điều phối tất cả các thông điệp giữa phía gửi đến đúng phía nhận.

MQTT là giao thức định hướng bản tin. Mỗi bản tin là một đoạn rời rạc của tín hiệu và broker không thể nhìn thấy. Mỗi bản tin được publish một địa chỉ, có thể hiểu như một kênh (Topic). Client đăng ký vào một kênh để nhận/gửi dữ liệu, gọi là subscribe. Client có thể subscribe vào nhiều kênh. Mỗi client sẽ nhận được dữ liệu khi bất kỳ trạm nào khác gửi dữ liệu vào kênh đã đăng ký. Khi một client gửi một bản tin đến một kênh nào đó gọi là publish.



Kiến trúc thành phần:

Thành phần chính của MQTT là Client (Publisher/Subscriber), Server (Broker), Sessions (tạm dịch là Phiên làm việc), Subscriptions và Topics.

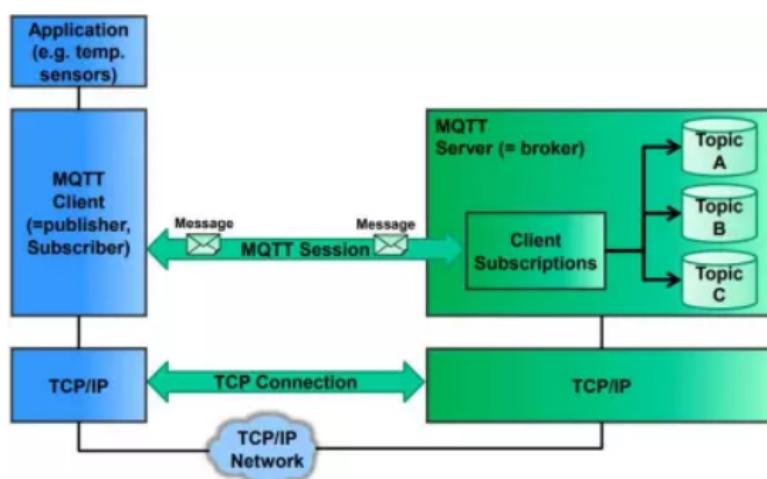
- **MQTT Client (Publisher/Subscriber)**: Clients sẽ subscribe một hoặc nhiều topics để gửi và nhận thông điệp từ những topic tương ứng.

- **MQTT Server (Broker)**: Broker nhận những thông tin subscribe (Subscriptions) từ client, nhận thông điệp, chuyển những thông điệp đến các Subscriber tương ứng dựa trên Subscriptions từ client.

- **Topic**: Có thể coi Topic là một hàng đợi các thông điệp, và có sẵn khuôn mẫu dành cho Subscriber hoặc Publisher. Một cách logic thì các topic cho phép Client trao đổi thông tin với những ngữ nghĩa đã được định nghĩa sẵn. Ví dụ: Dữ liệu cảm biến nhiệt độ của một tòa nhà.

- **Session**: Một session được định nghĩa là kết nối từ client đến server. Tất cả các giao tiếp giữa client và server đều là 1 phần của session.

- **Subscription**: Không giống như session, subscription về mặt logic là kết nối từ client đến topic. Khi đã subscribe một topic, Client có thể nhận/gửi thông điệp (message) với topic đó.





4.6.2 MQTT mosquitto

Mosquitto là một MQTT Broker mã nguồn mở cho phép thiết bị truyền nhận dữ liệu theo giao thức MQTT version 5.0, 3.1.1 và 3.1 – Một giao thức nhanh, nhẹ theo mô hình publish/-subscribe được sử dụng rất nhiều trong lĩnh vực Internet of Things. Mosquitto cung cấp một thư viện viết bằng ngôn ngữ C để triển khai các MQTT Client và có thể dễ dàng sử dụng bằng dòng lệnh: “mosquitto_pub” và “mosquitto_sub”. Ngoài ra, Mosquitto cũng là một phần của Eclipse Foundation, là dự án iot.eclipse.org và được tài trợ bởi cedalo.com

Ưu điểm:

- Ưu điểm nổi bật của Mosquitto là tốc độ truyền nhận và xử lý dữ liệu nhanh, độ ổn định cao, được sử dụng rộng rãi và phù hợp với những ứng dụng embedded.
 - Mosquitto rất nhẹ và phù hợp để sử dụng trên tất cả các thiết bị.
 - Ngoài ra, Mosquitto cũng được hỗ trợ các giao thức TLS/SSL (các giao thức nhằm xác thực server và client, mã hóa các message để bảo mật dữ liệu).

Nhược điểm:

- Một số nhược điểm của mosquitto là khó thiết kế khi làm những ứng dụng lớn và ít phương thức xác thực thiết bị nên khả năng bảo mật vẫn chưa tối ưu.

Cài đặt Mosquitto broker với Ubuntu và test:

Để cài đặt một MQTT server sử dụng Mosquitto broker với Ubuntu, trước hết chúng ta sẽ cần một server Ubuntu có quyền truy cập root và một cổng TCP:1883 được mở trên firewall.

- Với Ubuntu 16.04 trở lên, ta thực hiện theo các bước sau:

```
sudo apt-get update
```

```
sudo apt-get install mosquitto
```

- Với các phiên bản Ubuntu < 16.04, ta thực hiện:

```
sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppapppa
```

```
sudo apt-get update
```

```
sudo apt-get install mosquitto
```

Service Mosquitto sẽ được start ngay sau khi cài đặt và các message được gửi đến qua port 1883.

```
root@mqtt-broker:~# service mosquitto status
● mosquitto.service - LSB: mosquitto MQTT v3.1 message broker
   Loaded: loaded (/etc/init.d/mosquitto; bad; vendor preset: enabled)
   Active: active (running) since Thu 2020-07-16 08:05:51 UTC; 2 months 3 days ago
     Docs: man:systemd-sysv-generator(8)
     Tasks: 1
    Memory: 1.0M
      CPU: 57min 21.974s
     CGroup: /system.slice/mosquitto.service
             └─1023 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

Warning: Journal has been rotated since unit was started. Log output is incomplete or unavailable.
root@mqtt-broker:~# netstat -plntu | grep 1883
tcp        0      0 0.0.0.0:1883          0.0.0.0:*          LISTEN      1023/mosquitto
tcp6       0      0 ::1:1883            ::*           LISTEN      1023/mosquitto
```

- Cài đặt mosquitto client

```
sudo apt-get install mosquitto-clients
```

Mosquitto clients sẽ giúp chúng ta dễ dàng kiểm tra hoạt động của một MQTT broker Mosquito thông qua dòng lệnh. Chúng ta sẽ thực hiện test Publish và Subscribe đến Mosquitto broker vừa được cài đặt. Chúng ta sẽ sử dụng 2 cửa sổ dòng lệnh, một để thực hiện publish message đến broker, một để thực hiện subscribe message từ broker.

Một **topic** được broker sử dụng như là các nhãn để lọc message cho từng client kết nối đến broker. Một client subscribe một **topic** nào sẽ chỉ nhận được các message được các client khác



publish vào đúng topic đó.

- Ở một cửa sổ dòng lệnh, chúng ta sẽ thực hiện subscribe đến một topic `test` như sau:
`mosquitto_sub -t "test"`

Option `-t` sẽ theo sau là tên một topic.

```
mosquitto_pub -m "Hello! Viblo" -t "test"  
mosquitto_pub -m "message from mosquitto_pub client" -t "test"  
mosquitto_pub -m "Hihi" -t "test"
```

Option `-m` được thêm vào ở đây sẽ theo sau là 1 message được publish đến broker theo topic.

- Ở phía cửa sổ dòng lệnh thực hiện subscribe topic, chúng ta sẽ thấy được các message:

```
root@mqtt-broker:~# mosquitto_sub -t "test"  
Hello viblo  
message from mosquitto_pub client  
Hihi
```

Như vậy, nếu có kết quả tương tự như trên, chúng ta đã hoàn thành việc cài đặt và kiểm tra một MQTT broker Mosquitto rồi.

Mosquitto còn cung cấp một cơ chế đặt password để bảo mật cho broker server với `mosquitto_passwd`:

```
sudo mosquitto_passwd -c /etc/mosquitto/passwd mqtt_user_name  
Password: mqtt_password
```

Tiếp đó chúng ta sẽ tạo một file config cho mosquitto để trỏ đến file password mà chúng ta vừa tạo với nội dung như sau:

```
allow_anonymous false  
password_file /etc/mosquitto/passwd
```

Để hoàn thành đặt password, chúng ta thực hiện restart service mosquitto:

```
sudo systemctl restart mosquitto
```

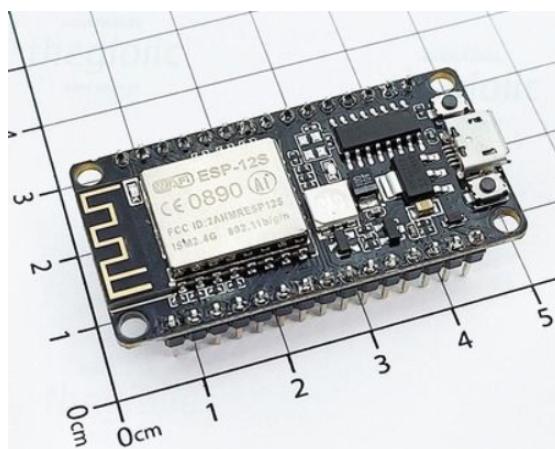
Sau khi đặt user, password, từ client muốn publish hoặc subscribe đến Mosquitto broker, chúng ta sẽ phải thông qua khai báo user và password như sau:

```
mosquitto_sub -t "test" -u "mqtt_user_name" -P "mqtt_password"  
mosquitto_pub -m "Hello! Viblo" -t "test" -u "mqtt_user_name" -P "mqtt_password"
```

5 Linh kiện

5.1 ESP-12S ESP8266 Ai-Thinker NodeMCU CH340C WIFI Module IOT

ESP-12S ESP8266 Ai-Thinker NodeMCU là một Kit thu phát WiFi dựa trên nền mô-đun ESP-12S rất dễ sử dụng để tạo các dự án với các ứng dụng WiFi và IoT (Internet of Things). Bo mạch này hỗ trợ các giao tiếp như: SPI, UART,...



Thông số kỹ thuật:



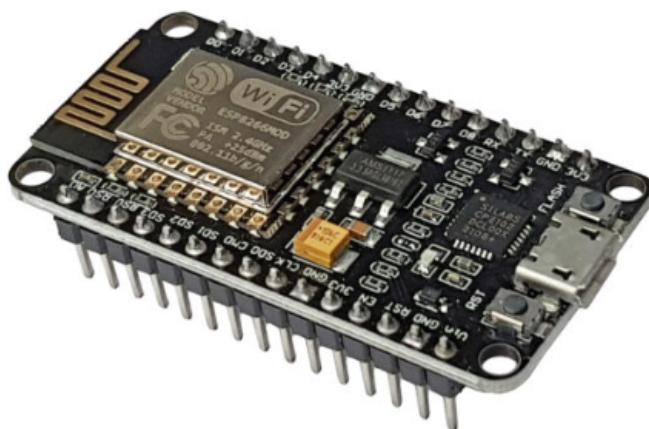
- NodeMCU có cổng USB-TTL với IC CH340C hoạt động ổn định với chuẩn công nghiệp.
- Điện áp giao tiếp: 3.3 VDC.
- Chuẩn 802.11 b / g / n không dây
- Nguồn vào: 5V nguồn từ cổng USB
- Tích hợp MCU 32-bit công suất thấp
- Tích hợp ADC 10-bit
- Tích hợp giao thức TCP / IP
- Tích hợp PLL, bộ điều chỉnh và đơn vị quản lý điện năng
- Bộ nhớ Flash 4MB
- Hỗ trợ đa dạng ăng-ten
- Wi-Fi 2.4 GHz, hỗ trợ WPA / WPA2
- Hỗ trợ các chế độ hoạt động STA / AP / STA + AP
- Hỗ trợ chức năng liên kết thông minh cho cả thiết bị Android và iOS
- Tích hợp công tắc TR, balun, LNA, bộ khuếch đại công suất và mạng kết hợp
- Dải tần số: 2.4GHz-2.5GHz (2400M-2483.5M)
- Bus ngoại vi: UART / HSPI / I2C / I2S / Ir Remote Control; GPIO / PW
- Dòng hoạt động trung bình: 80mA
- Phạm vi nhiệt độ hoạt động: -40 ° 125 °
- Chế độ Wi-Fi: trạm / softAP / SoftAP + trạm

5.2 Module RF thu phát Wifi ESP8266 NodeMCU Lua CP2102

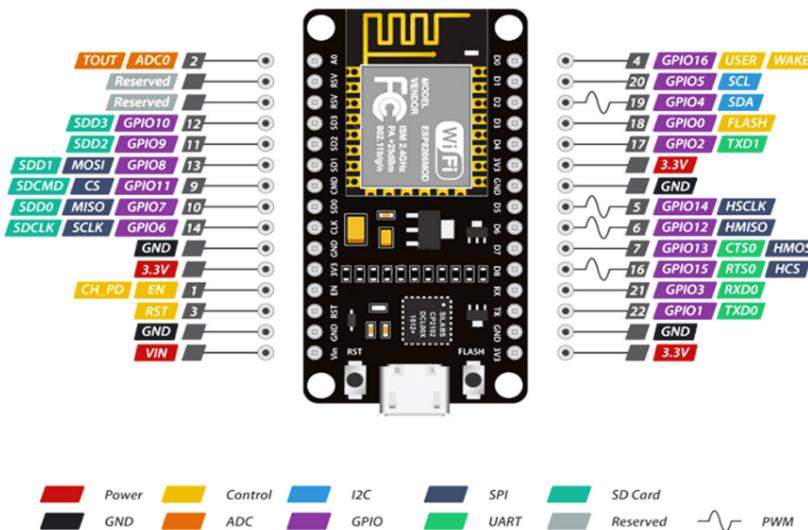
KIT thu phát WiFi NodeMcu Lua ESP8266 CP2102 - là KIT thu phát wifi dựa trên nền chip WiFi SoC ESP8266 và chip giao tiếp CP2102 mạnh mẽ. Được dùng cho các ứng dụng cần kết nối, thu thập dữ liệu và điều khiển qua sóng WiFi, đặc biệt là các ứng dụng liên quan đến IoT.

KIT thu phát WiFi NodeMcu Lua ESP8266 CP2102 Với thiết kế dễ dàng sử dụng trực tiếp trình biên dịch của Arduino IDE để lập trình và nạp code, điều này khiến việc sử dụng và lập trình các ứng dụng trên ESP8266 trở nên rất đơn giản.

KIT thu phát WiFi NodeMcu Lua ESP8266 CP2102 sử dụng chip nạp và giao tiếp UART mới và ổn định nhất là CP2102 có khả năng tự nhận Driver trên tất cả các hệ điều hành Window và Linux, đây là phiên bản nâng cấp từ các phiên bản sử dụng IC nạp giá rẻ CH340.



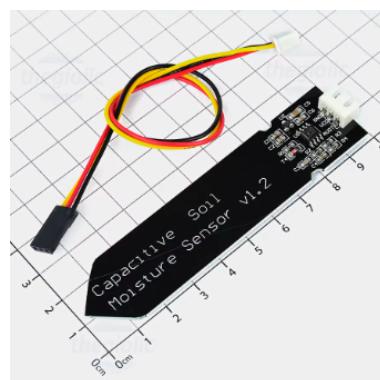
Thông số kỹ thuật:



- IC chính: ESP8266 Wifi SoC.
- Phiên bản Firmware: NodeMCU Lua.
- Chip nạp và giao tiếp UART: CP2102.
- GPIO tương thích hoàn toàn với firmware Node MCU.
- Cấp nguồn: 5VDC (qua MicroUSB hoặc Vin)
- GPIO giao tiếp mức: 3.3VDC
- Tích hợp Led báo trạng thái, nút Reset, Flash.
- Tương thích hoàn toàn với trình biên dịch: Arduino IDE
- Kích thước: 50mm x 25mm

5.3 Cảm Biến Độ Ẩm Đất

Cảm Biến Độ Ẩm Đất sử dụng cảm biến điện dung để phát hiện độ ẩm của đất. Nó được làm bằng vật liệu chống ăn mòn mang lại cho nó một tuổi thọ sử dụng tuyệt vời. Chip điều chỉnh điện áp tích hợp, hỗ trợ 3.3-5.5 VDC.

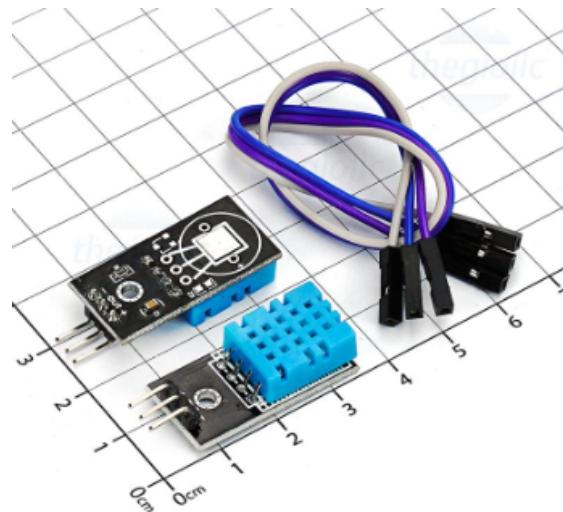


Thông số kỹ thuật:

- Điện áp hoạt động: 3.3 – 5.5 VDC
- Điện áp đầu ra: 0 – 3 VDC
- Kết nối: PH2.54-3P
- Kích thước: 98 x 23mm

5.4 DHT11 Mạch Cảm Biến Nhiệt Độ Và Độ Ẩm

DHT11 Mạch Cảm Biến Nhiệt Độ Và Độ Ẩm được hiệu chuẩn với công nghệ cảm biến điện trở kết hợp với điện trở NTC để đọc chính xác nhiệt độ và độ ẩm xung quanh. Mạch cảm biến này có cấu tạo vô cùng nhỏ gọn, giá thành rẻ, phản hồi nhanh, khả năng chống nhiễu lớn. Đầu ra của DHT11 ở dạng tín hiệu kỹ thuật số trên một chân dữ liệu duy nhất, tần số cập nhật cảm biến sẽ được đo ở mỗi 2 giây (0,5Hz).



Thông số kỹ thuật:

- Điện áp: 3.3 – 5VDC
- Dòng điện max: 5mA
- Thang độ ẩm: 20 → 80% độ chính xác ±5%
- Thang nhiệt độ: 0 → 50°C độ chính xác ± 2°C
- Kích thước: 15.5 x 12 x 5.5 mm

5.5 Bơm nước 365DC 12V

Máy bơm nước mini DC sử dụng động cơ 365DC- Motor 12V là máy bơm nước loại mini motor DC12V được sử dụng phổ biến trong ứng dụng gia dụng, Y tế, chế đồ DIY, bơm nước bình trà, thiết bị bể cá, hệ thống cấp và tuần hoàn nước,... và sử dụng động cơ 365DC - 12V có công suất tốt, hoạt động bền bỉ, liên tục.

Thông số kỹ thuật:

- Model: 365 bơm màng micro DC
- Điện áp làm việc: 12VDC
- Dòng điện không tải: 0.23

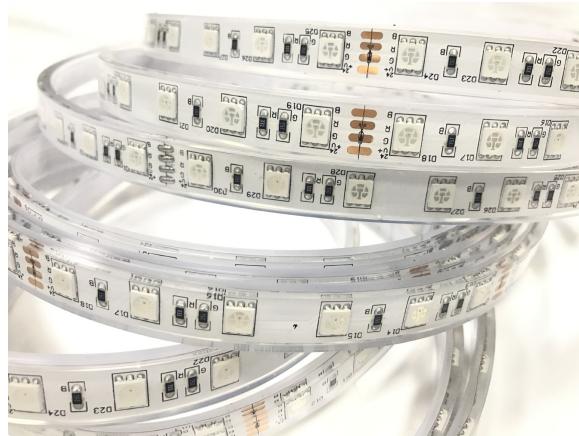


- Tải trọng: 450
- Lưu lượng tối đa: 2-3 lít / phút
- Áp suất tối đa: 1-2,5 kg
- lực nâng tối đa: 1-2,5 mét
- Giờ làm việc bình thường: 2-3 năm.
- Lực hút tối đa: 2 mét
- Đường kính đầu vào và đầu ra: 8 mm (đường kính ngoài)
- Chiều dài động cơ: 32MM
- Đường kính động cơ: 28MM
- Chiều dài bơm: 36MM
- Tổng chiều dài: 69MM
- Đường kính bơm: 40MM * 35MM
- Trọng lượng: 111g

5.6 LED dây 5050 màu đỏ

Nhóm chọn LED màu đỏ. Bởi lẽ, đối với LED chiếu sáng sẽ là dải nhiệt từ 2000 đến 4000K và bước sóng cao nhất là 680 nanomet. Màu sắc ánh sáng này đóng vai trò cực quan trọng trong quá trình thúc đẩy cây phát triển chiều cao và đơm hoa kết trái. Bạn sẽ thấy chúng được sử dụng nhiều cho cây dài ngày và ít cho cây ngắn ngày. **Thông số kỹ thuật:**

- Nguồn 12VDC
- Công suất 9.6W/m
- Nhiệt độ hoạt động $-25^{\circ}\text{C} \div 60^{\circ}\text{C}$
- Mật độ led: 60 chíp / 1 mét

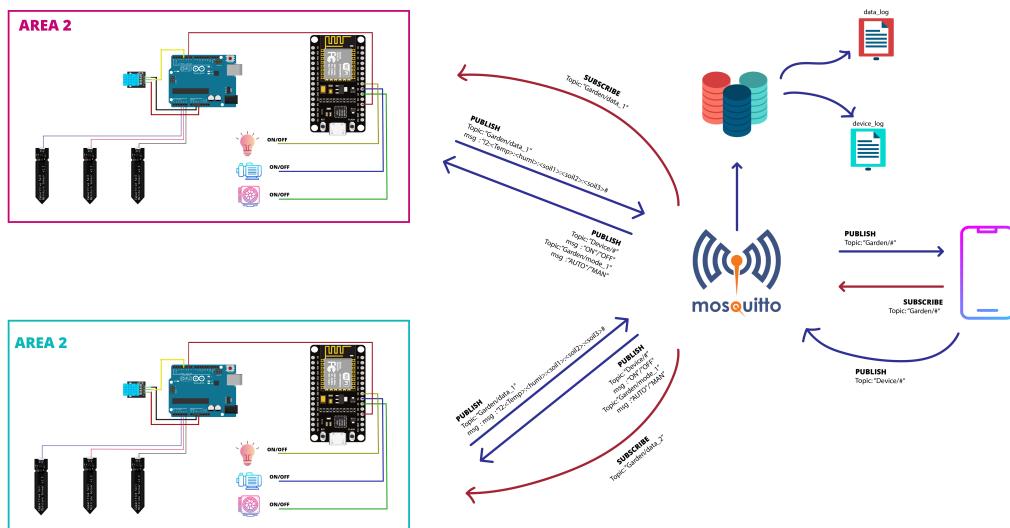


Hình 24: LED dây 5050

6 Tiến hành mô hình demo

Mẫu data gửi theo định dạng

`!ID :< Temperature >:< Humidity >:< Soil_1 >:< Soil_2 >:< Soil_3 > #`



Hình 25: Tổng quan về Hệ Thống

6.1 Sensor Node

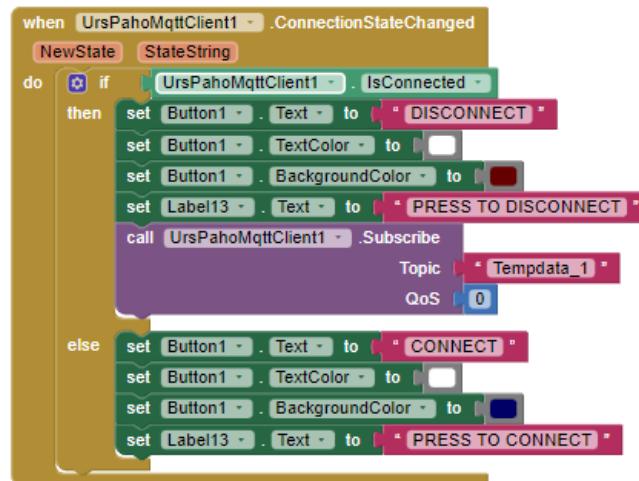
```

59 void read_dht() {
60     unsigned long currentMillis = millis(); //read current time
61     if (currentMillis - previousMillis >= interval) { //if current time - last time > 5 sec
62         previousMillis = currentMillis;
63         //read temp and humidity
64         sensors_event_t event;
65         dht.temperature().getEvent(&event);
66         if (!isnan(event.temperature)) {
67             // Serial.println(F("Error reading temperature!"));
68         }
69     } else {
70         // Serial.print(F("Temperature: "));
71         temp = event.temperature;
72         // Serial.print(temp);
73         // Serial.println(F("C"));
74     }
75     // Get humidity event and print its value.
76     dht.humidity().getEvent(&event);
77     if (!isnan(event.relative_humidity)) {
78         // Serial.println(F("Error reading humidity!"));
79     } else {
80         // Serial.print(F("Humidity: "));
81         hum = event.relative_humidity;
82         // Serial.print(hum);
83         // Serial.println(F("%"));
84     }
85     soil1 = analogRead(SOIL1);
86     soil2 = analogRead(SOIL2);
87     soil3 = analogRead(SOIL3);
88 }

int soil1_percent = map(soil1, 0, 1023, 0, 100);
int soil2_percent = map(soil2, 0, 1023, 0, 100);
int soil3_percent = map(soil3, 0, 1023, 0, 100);
msgStr = String(temp) + ":" + String(hum) + ":" + String(soil1_percent) + ":" + String(soil2_percent) + ":" + String(soil3_percent) ;
byte arrSize = msgStr.length() + 1;
char msgArr[arrSize];
msgStr.toCharArray(msg, arrSize);
espSerial.print(msg);
}
}

```

Hình 26: Đọc và gửi dữ liệu tới GateWay



Hình 27: Dữ liệu từ Sensor Node



```
void setup_wifi() {  
    delay(10);  
  
    WiFi.begin(ssid, password);  
  
    while (WiFi.status() != WL_CONNECTED) {  
        delay(500);  
        Serial.print(".");  
    }  
  
    Serial.println("");  
    Serial.println("WiFi connected");  
    Serial.println("IP address: ");  
    Serial.println(WiFi.localIP());  
}
```

Hình 28: Kết nối với Wifi

6.2 GateWay

```
63 void reconnect() {
64     while (!client.connected()) {
65         if (client.connect(clientID, mqttUserName, mqttPwd)) {
66             Serial.println("MQTT connected");
67             client.subscribe("Device/lights_1");
68             Serial.println("Topic lights 1 Subscribed");
69             client.subscribe("Device/pump_1");
70             Serial.println("Topic pump 1 Subscribed");
71             client.subscribe("Device/mist_1");
72             Serial.println("Topic quat 1 Subscribed");
73             client.subscribe("Device/time_1");
74             Serial.println("Topic time 1 Subscribed");
75             client.subscribe("Garden/mode_1");
76             Serial.println("Topic garden/mode_1 Subscribed");
77             status_mode = MAN ;
78         }
79     else {
80         Serial.print("failed, rc=");
81         Serial.print(client.state());
82         Serial.println(" try again in 5 seconds");
83         delay(5000); // wait 5sec and retry
84         status_mode = AUTO ;
85     }
86 }
87 }
```

Hình 29: Kết nối với MQTT Broker

```
if (Serial.available()) {
    String msgStr = "";
    msgStr = Serial.readString();
    msgStr = "!1:" + msgStr + "#";
    process_msg(msgStr);
    byte arrSize = msgStr.length() + 1;
    char msg[arrSize];
    Serial.print("PUBLISH DATA:");
    Serial.println(msgStr);
    msgStr.toCharArray(msg, arrSize);
    client.publish(topic, msg);
    delay(50);
}
```

Hình 30: Gửi dữ liệu lên MQTT Broker

```
if (status_mode == AUTO) {  
    if (tb_soil <= 40) {  
        if (digitalRead(PUMP) == LOW) {  
            digitalWrite(PUMP, HIGH) ;  
            client.publish("Device/pump_1", "ON_P");  
        }  
    }  
    if (tb_soil >= 60) {  
        if (digitalRead(PUMP) == HIGH) {  
            digitalWrite(PUMP, LOW);  
            client.publish("Device/pump_1", "OFF_P");  
        }  
    }  
    if (tb_humi >= 80 ) {  
        if (digitalRead(SUONG) == LOW) {  
            digitalWrite(SUONG, HIGH) ;  
            client.publish("Device/mist_1", "ON_S");  
        }  
    }  
    if (tb_humi >= 50 && tb_humi <= 80) {  
        if (digitalRead(SUONG) == HIGH) {}  
        digitalWrite(SUONG, LOW);  
        client.publish("Device/mist_1", "OFF_S");  
    }  
    if (tb_soil <= 40 && tb_humi <= 50 ) {  
        if (digitalRead(SUONG) == LOW) {  
            digitalWrite(SUONG, HIGH) ;  
            client.publish("Device/mist_1", "ON_S");  
        }  
        if (digitalRead(PUMP) == LOW) {  
            digitalWrite(PUMP, HIGH) ;  
            client.publish("Device/pump_1", "ON_P");  
        }  
    }  
}
```

Hình 31: Chế độ Automatic



```
.....  
WiFi connected  
IP address:  
192.168.137.230  
MQTT connected  
Topic lights 1 Subscribed  
Topic pump 1 Subscribed  
Topic quat 1 Subscribed  
Topic time 1 Subscribed  
Topic garden/mode_1 Subscribed
```

Hình 32: GateWay kết nối Wifi và MQTT Broker thành công

6.3 Server

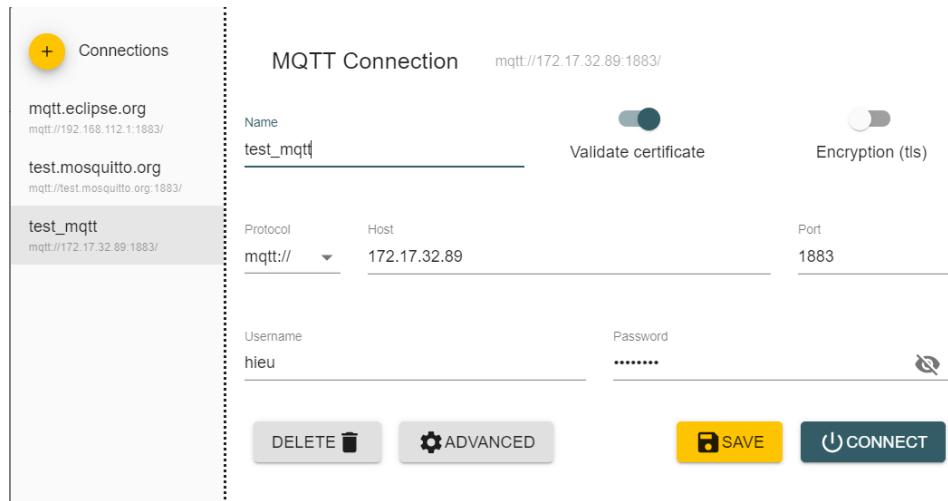
Trong phần này, nhóm sử dụng giao thức kết nối MQTT, và Node-Red để ghi dữ liệu lên MySQL Server.

6.3.1 MQTT Broker

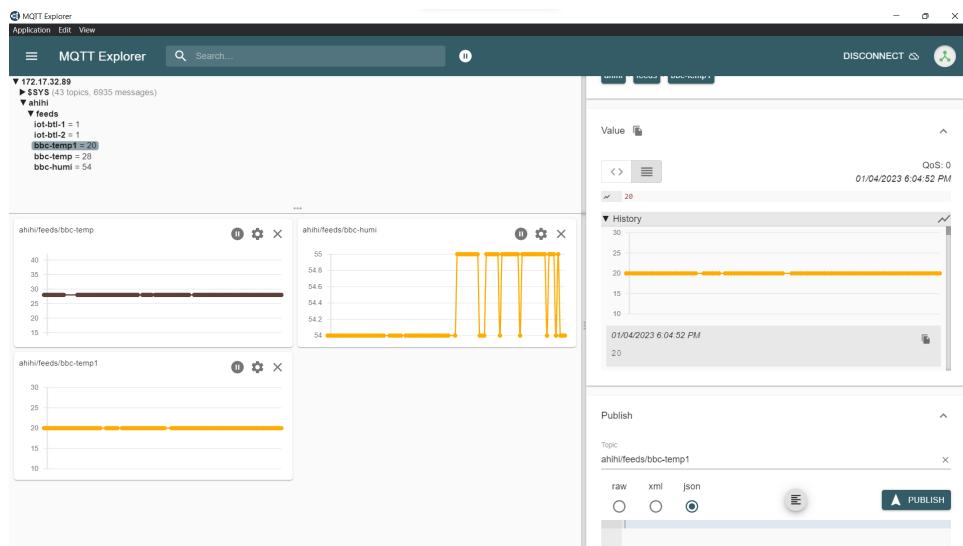
```
Administrator: Command Prompt  
Microsoft Windows [Version 10.0.22000.1335]  
(c) Microsoft Corporation. All rights reserved.  
C:\WINDOWS\system32>cd C:\Program Files\mosquitto  
C:\Program Files\mosquitto>mosquitto -v  
1673159386: mosquitto version 2.0.15 starting  
1673159386: Using default config.  
1673159386: Starting in local only mode. Connections will only be possible from clients running on this machine.  
1673159386: Create a configuration file which defines a listener to allow remote access.  
1673159386: For more details see https://mosquitto.org/documentation/authentication-methods/  
1673159386: Opening ipv4 listen socket on port 1883.
```

Hình 33: Khởi động MQTT Broker Service

Tiến hành kiểm tra kết nối của MQTT Broker bằng tool **MQTT Explorer**.

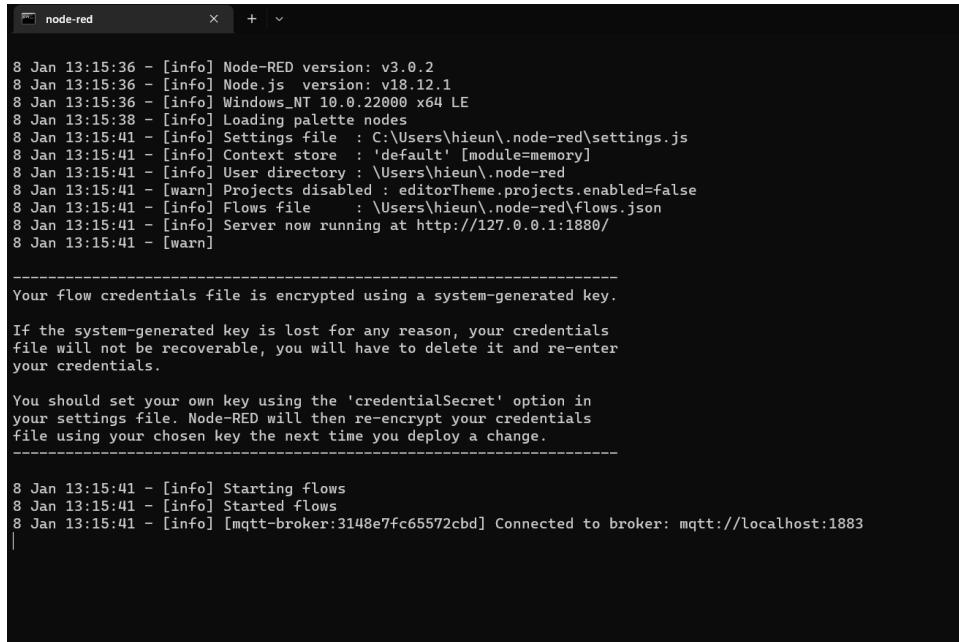


Hình 34: Kết nối với MQTT Broker



Hình 35: Giao diện khi MQTT đã khởi động thành công

6.3.2 Node-RED



```
node-red
```

```
8 Jan 13:15:36 - [info] Node-RED version: v3.0.2
8 Jan 13:15:36 - [info] Node.js version: v18.12.1
8 Jan 13:15:36 - [info] Windows_NT 10.0.22000 x64 LE
8 Jan 13:15:38 - [info] Loading palette nodes
8 Jan 13:15:41 - [info] Settings file : C:\Users\hieun\.node-red\settings.js
8 Jan 13:15:41 - [info] Context store : 'default' [module=memory]
8 Jan 13:15:41 - [info] User directory : \Users\hieun\.node-red
8 Jan 13:15:41 - [warn] Projects disabled : editorTheme.projects.enabled=false
8 Jan 13:15:41 - [info] Flows file : \Users\hieun\.node-red\flows.json
8 Jan 13:15:41 - [info] Server now running at http://127.0.0.1:1880/
8 Jan 13:15:41 - [warn]

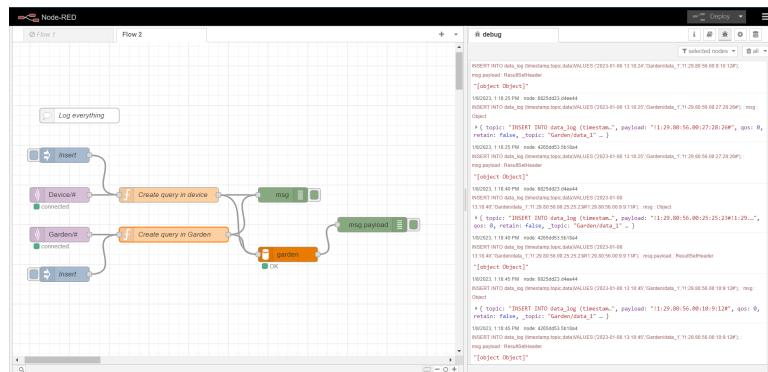
-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.

-----
8 Jan 13:15:41 - [info] Starting flows
8 Jan 13:15:41 - [info] Started flows
8 Jan 13:15:41 - [info] [mqtt-broker:3148e7fc65572cbd] Connected to broker: mqtt://localhost:1883
|
```

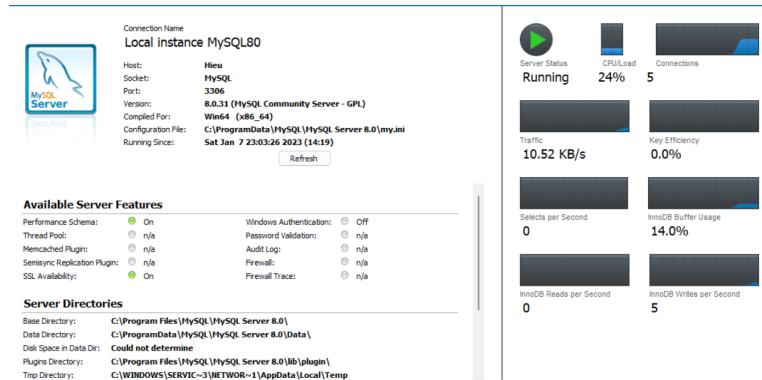
Hình 36: Khởi động Node-red bằng lệnh *node-red*



Hình 37: Lập trình kéo thả trên Node-Red

6.3.3 Database

Nhóm sử dụng MySQL server để làm database cho hệ thống.



Hình 38: Khởi động và kiểm tra trạng thái của server MySQL

ID	timestamp	topic	data
104	2023-01-07 18:11:23	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:30:30 27.60:58.00:30:30 #
105	2023-01-07 18:11:26	Garden/mode_1	AUTO
106	2023-01-07 18:11:30	Garden/mode_1	MAN
107	2023-01-07 18:11:41	Garden/data_1	!1:27.60:58.00:31:31:27.60:58.00:31:31:27.60:58.00:30:30 #
108	2023-01-07 18:11:47	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:30:30 27.60:58.00:31:30:31 #
109	2023-01-07 18:11:53	Garden/data_1	!1:27.60:58.00:31:31:27.60:58.00:31:31:27.60:58.00:31:31:31 #
110	2023-01-07 18:11:56	Garden/mode_1	AUTO
111	2023-01-07 18:11:59	Garden/data_1	!1:27.60:58.00:31:31:27.60:58.00:31:30:30 27.60:58.00:30:30:30 #
112	2023-01-07 18:12:05	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:30:30 27.60:58.00:30:30:30 #
113	2023-01-07 18:12:11	Garden/data_1	!1:27.60:58.00:31:30:31 27.60:58.00:31:31:31 27.60:58.00:31:31:31 #
114	2023-01-07 18:12:17	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:30:30 27.60:58.00:30:30:30 #
115	2023-01-07 18:12:23	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:31:31:31 27.60:58.00:31:31:31 #
116	2023-01-07 18:12:29	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:30:30 27.60:58.00:30:30:30 #
117	2023-01-07 18:12:35	Garden/data_1	!1:27.60:58.00:31:31:31 27.60:58.00:31:31:31 27.60:58.00:30:30:30 #
118	2023-01-07 18:12:41	Garden/data_1	!1:27.60:58.00:30:30 27.60:58.00:30:30 27.60:58.00:31:31:31 #
119	2023-01-07 18:12:47	Garden/data_1	!1:27.60:58.00:31:31:27.60:58.00:30:30 27.60:58.00:30:30:30 #
120	2023-01-07 18:12:53	Garden/data_1	!1:27.60:58.00:31:31:31 27.60:59.00:30:30 27.60:59.00:30:30:30 #
121	2023-01-07 18:12:59	Garden/data_1	!1:27.60:59.00:31:31:31 27.60:59.00:31:31:31 27.60:59.00:30:30:30 #

Hình 39: Dữ liệu về cảm biến

ID	timestamp	topic	data
1	2023-01-07 11:44:18	Device/pump_1	ON_P
2	2023-01-07 11:44:18	Device/pump_1	OFF_P
3	2023-01-07 11:44:28	Device/pump_1	ON_P
4	2023-01-07 11:45:37	Device/mist_1	ON_S
5	2023-01-07 11:45:37	Device/mist_1	OFF_S
6	2023-01-07 11:45:39	Device/pump_1	OFF_P
7	2023-01-07 11:45:39	Device/pump_1	ON_P
8	2023-01-07 11:45:40	Device/lights_1	OFF_L
9	2023-01-07 11:45:41	Device/lights_1	ON_L
10	2023-01-07 12:43:33	Device/pump_1	OFF_P
11	2023-01-07 12:43:33	Device/lights_1	OFF_L
12	2023-01-07 12:43:34	Device/lights_1	ON_L

Hình 40: Dữ liệu về thiết bị

6.4 Ứng dụng điện thoại

6.4.1 Giới thiệu về MIT App Inventor

MIT App Inventor là một môi trường phát triển phần mềm do Google tạo ra để tạo các ứng dụng cho hệ điều hành Android. Ngôn ngữ này miễn phí và có thể dễ dàng truy cập từ web. Các ứng dụng được tạo bằng AppInventor bị hạn chế bởi tính đơn giản của chúng, mặc dù chúng cho phép đáp ứng một số lượng lớn các nhu cầu cơ bản trên thiết bị di động.

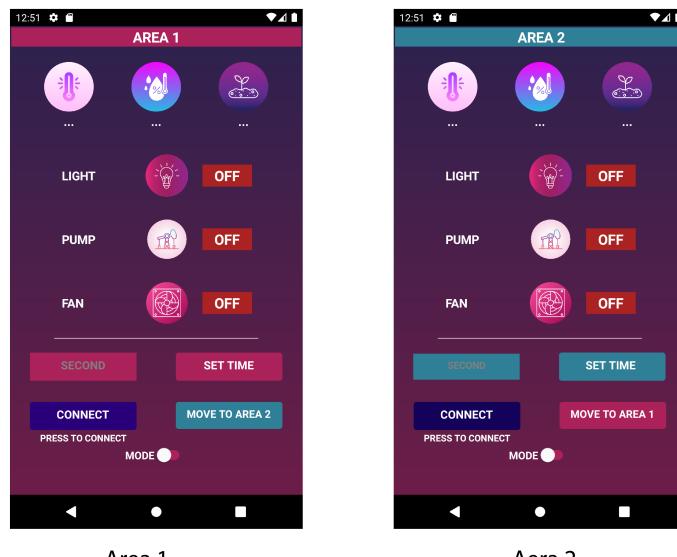


Hình 41: MIT App Inventor

Với MIT App Inventor, số lượng ứng dụng Android sẽ tăng đáng kể do hai yếu tố chính: tính đơn giản trong sử dụng, sẽ tạo điều kiện cho sự xuất hiện của một số lượng lớn các ứng dụng mới; và Google Play, trung tâm phân phối ứng dụng Android, nơi bất kỳ người dùng nào cũng có thể tự do phân phối sáng tạo của họ.

6.4.2 Ứng dụng Android Vườn thông minh

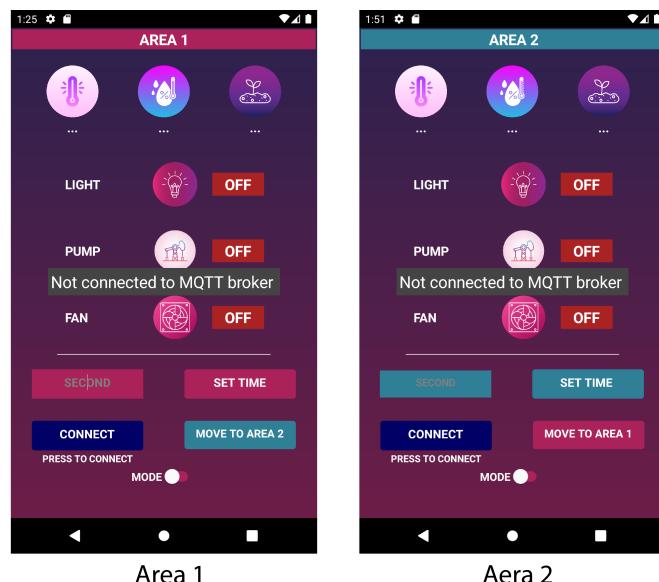
6.4.2.a Giao diện và hành vi của Ứng dụng



Area 1

Aera 2

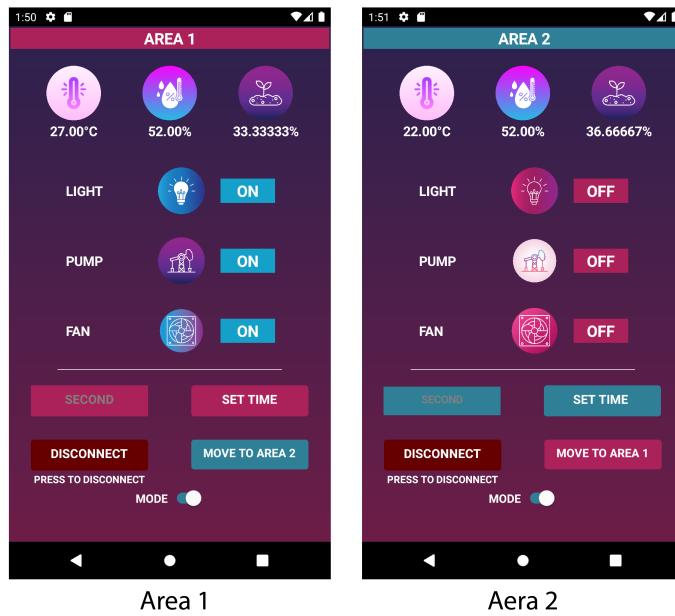
Hình 42: Giao diện Ứng dụng



Area 1

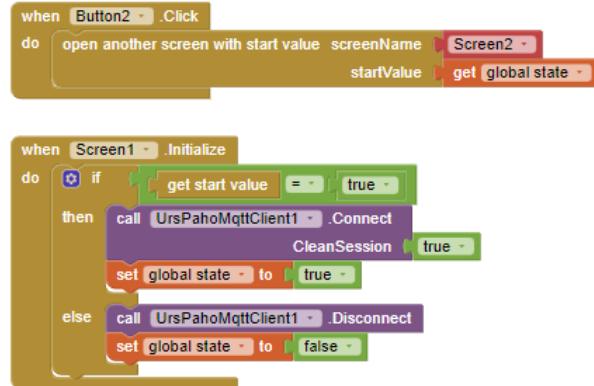
Aera 2

Hình 43: Thao tác mà chưa kết nối MQTT

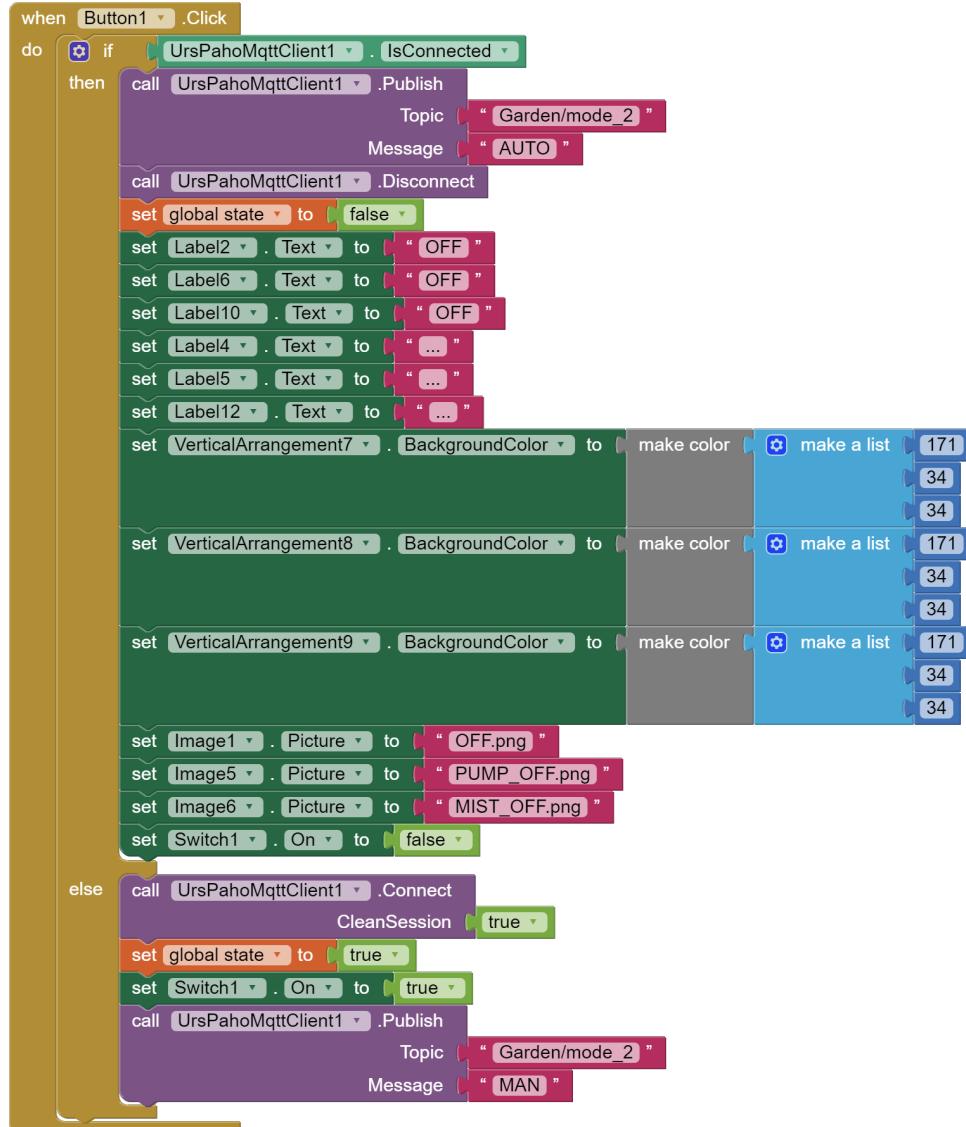


Hình 44: Giao diện sau khi kết nối thành công

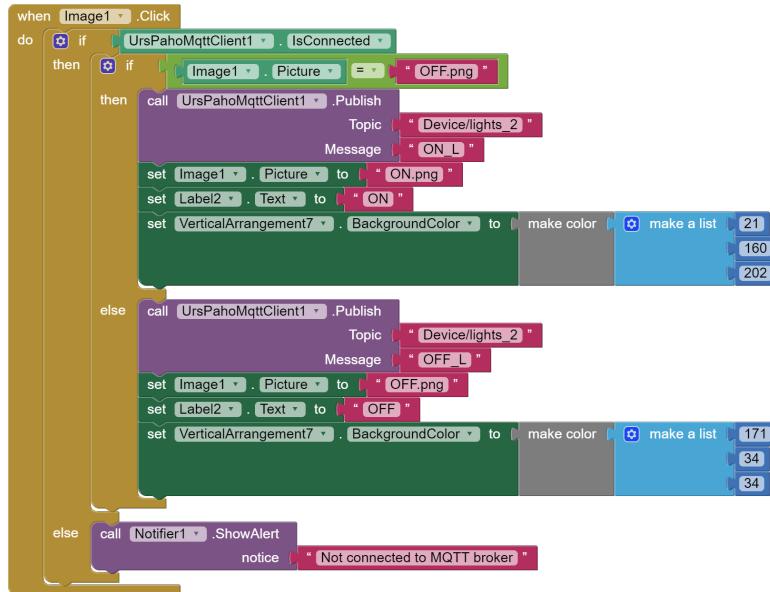
6.4.2.b Lập trình chức năng của các Box



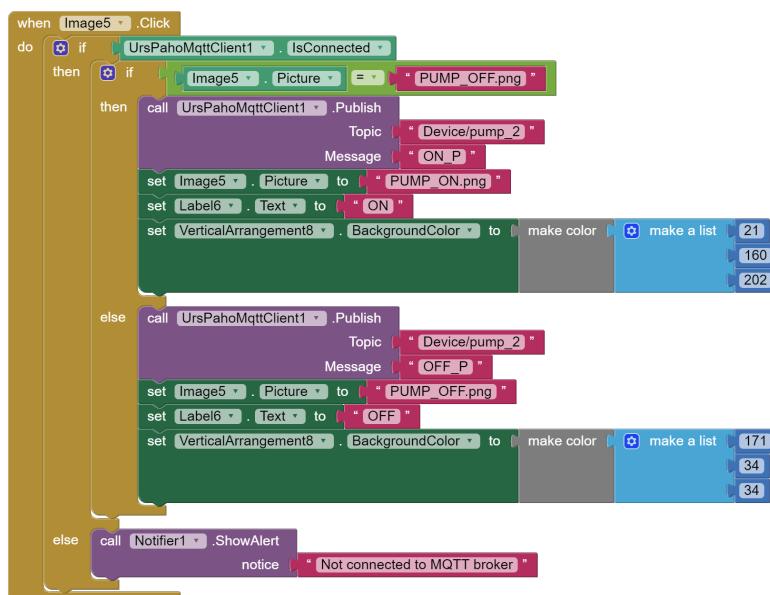
Hình 45: Intial Value for Screen - Switch Screen



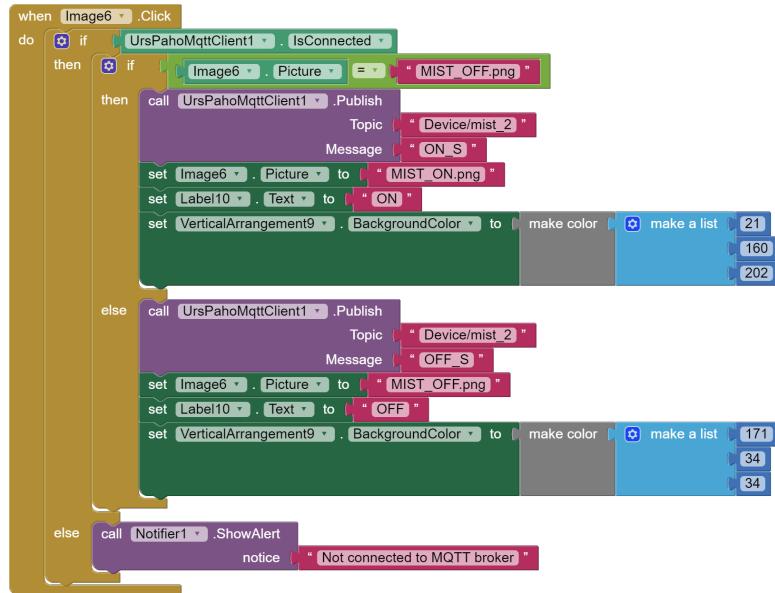
Hình 46: Connect to MQTT Broker



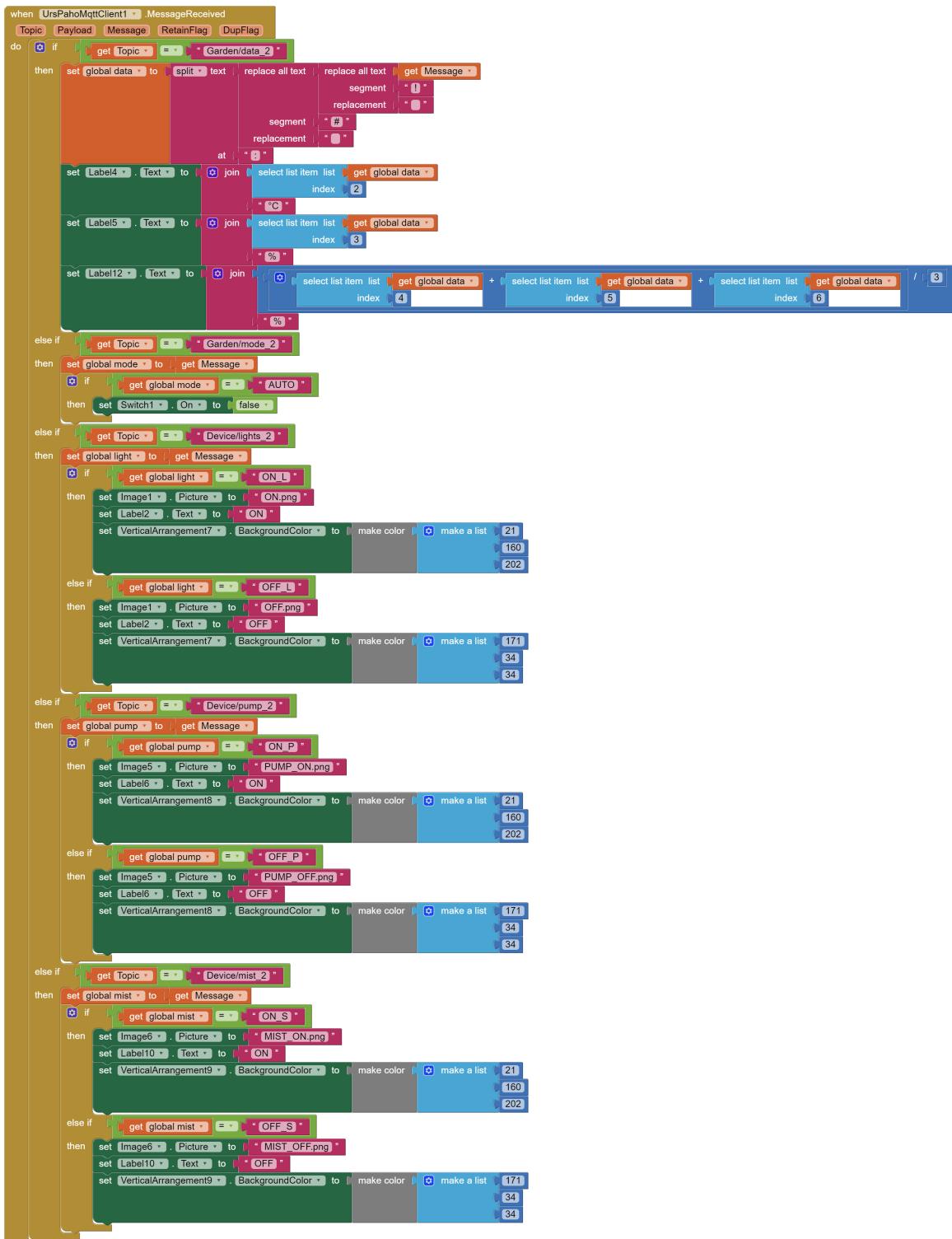
Hình 47: Public Request ON/OFF Light from App



Hình 48: Public Request ON/OFF Pump from App

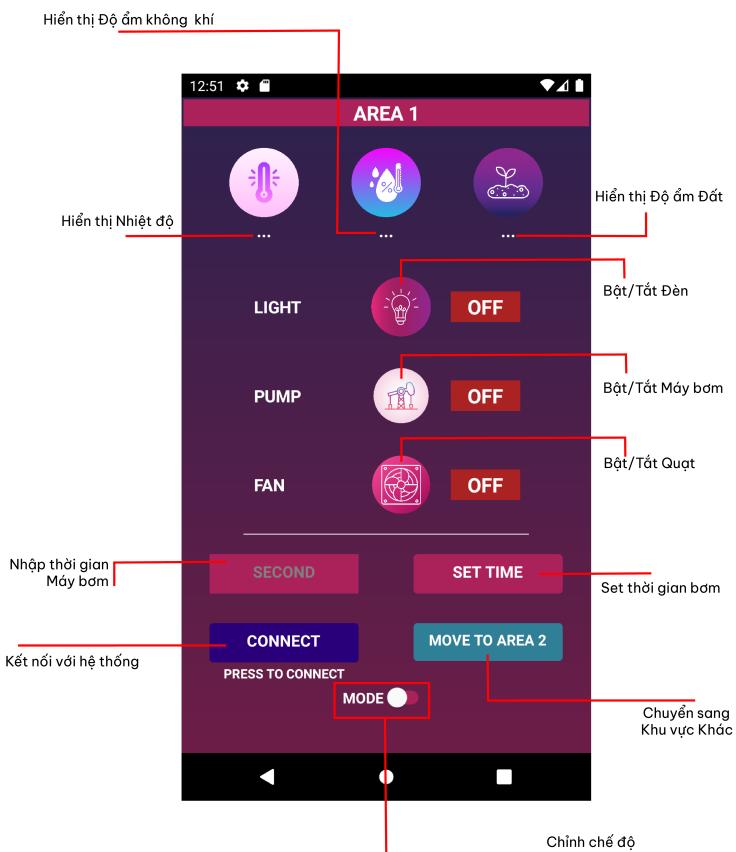


Hình 49: Public Request ON/OFF Fan from App



Hình 50: Recive Data from Sensor

6.4.2.c Hướng dẫn sử dụng



Hình 51: Chức năng trong Ứng dụng

1. "CONNECT": Kết nối với hệ thống.
2. "DISCONNECT": Ngắt kết nối với hệ thống.
3. Icon Bóng đèn: Bật/Tắt Đèn.
4. Icon Máy bơm: Bật/Tắt Bơm.
5. Textbox SECOND: Chọn thời gian cho máy bơm.
6. Button SET TIME: Set thời gian cho máy bơm.
7. Button MOVE TO AREA N: Chuyển sang khu vực N.
8. Switch MODE: Chuyển chế độ (MANUAL/AUTO).

6.5 Thi công vườn

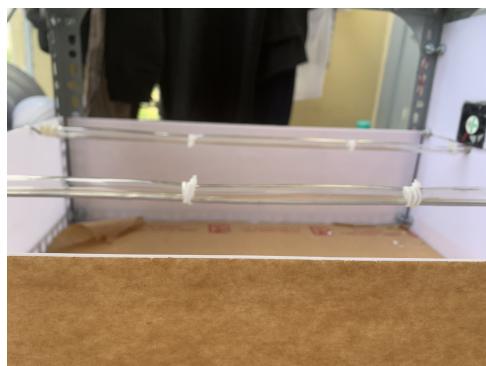


Hình 52: Hình ảnh vườn sau khi thi công

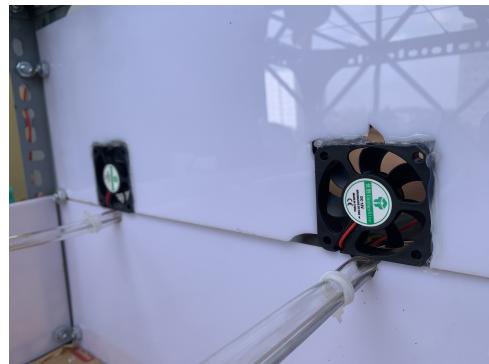
Nhóm xây dựng hệ thống vườn với ý tưởng 2 tầng, mỗi tầng tương ứng trung cho 1 khu vực vườn khác biệt. Nhưng do những lý do ngoài nên nhóm chỉ làm được một tầng trên.

Mô hình bao gồm:

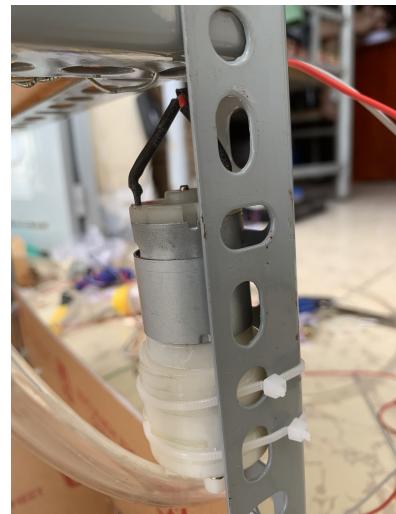
1. 2 Quạt.
2. 2 dải đèn LED.
3. 2 máy bơm đi kèm 2 hệ thống tưới nước.
4. 2 GateWay - 2 Sensor node. Tương ứng cho 2 khu vực



Hình 53: Hệ thống tưới vườn



Hình 54: Quạt



Hình 55: Máy bơm



6.6 Tính năng và Hành vi của mô hình

Đọc dữ liệu quan trắc:

- Độ ẩm Đất.(**Soil**)
- Độ ẩm Không khí (**Humi**)
- Nhiệt độ Không khí (**Temp**)
- Thời gian giữa 2 lần lấy mẫu: 15s.(Với hệ thống vườn tầng trong nhà, ít khi có sự thay đổi đột ngột của môi trường. Nên nhóm quyết định chọn 15s cho mỗi lần lấy mẫu, giảm phần nào về bài toán Big Data (so với tần số lấy mẫu tối đa 1Hz (1 giây / lần) của cảm biến DHT11)

Nhóm xây dựng demo với 2 chế độ. Chế độ người dùng. (**MANUAL**) và Chế độ tự động (**AUTOMATIC**).

Chế độ mặc định (khi người dùng không kết nối) : **AUTOMATIC**. Khi người dùng không kết nối với vườn qua ứng dụng, hệ thống sẽ luôn ở chế độ tự động.

Luân chuyển giữa 2 chế độ:

- Thủ công: Sử dụng switch trên ứng dụng điện thoại. (Màu xanh - **MANUAL**, Màu đỏ - **AUTOMATIC**)
- Tự động: Khi ở chế độ **MANUAL**, người dùng không thao tác trên ứng dụng trong vòng 30 giây. Vườn sẽ tự chuyển sang chế độ **AUTOMATIC**.

Tính năng:

1. Chế độ người dùng. (**MANUAL**) sử dụng qua ứng dụng điện thoại. Chức năng:

- Bật/Tắt Quạt
- Bật/Tắt Đèn
- Bật/Tắt Máy bơm
- Thiết lập thời gian cho máy bơm (theo giây)

2. Chế độ tự động (**AUTOMATIC**). Hành vi - chức năng:

- **Humi** $\geq 80\%$: Tự động bật quạt. Phân tán bớt độ ẩm của môi trường.
- **Soil** $\leq 40\%$: Tự động bật máy bơm. Cung cấp đủ ẩm cần thiết cho đất.
- **Soil** $\geq 60\%$: Tự động bật máy bơm. Độ ẩm đất 60% là độ ẩm thích hợp cho đa số cây, nên máy bơm sẽ tự tắt.
- $50\% \leq \text{Soil} \geq 80\%$ và **Humi** $\geq 50\%$. Tắt quạt. Do vườn đang ở độ ẩm thích hợp.
- **Soil** $< 40\%$ và **Humi** $< 50\%$. Tự động bật đồng thời cả quạt và máy bơm. Đất đang thiếu độ ẩm, và không khí độ ẩm thấp. Việc bật máy bơm sẽ cung cấp độ ẩm cho đất, đồng thời nhóm thiết kế bơm theo dạng phun từ trên xuống nên cung cấp phần nào cung cấp độ ẩm cho không khí. Quạt được bật để phân tán độ ẩm không khí đi toàn bộ vườn.

Lưu ý: Tất cả hành vi trong chế độ **AUTOMATIC** đều sẽ chỉ được thực hiện nếu các điều kiện thỏa mãn 3 lần lấy mẫu liên tiếp (30 giây).



7 Tổng kết

Với những mục tiêu đề ra, nhóm đã phần nào giải quyết được yêu cầu ban đầu là thiết kế hệ thống có thể điều khiển các thiết bị điện trong mô hình vườn, đưa được dữ liệu cảm biến lên ứng dụng di động, giám sát giá trị cảm biến theo thời gian thực, điều khiển được thiết bị giao diện ứng dụng. Ngoài ra nhóm cũng đã giải quyết được bài toán về mô hình cơ khí, mô hình chắc chắn, chuyên nghiệp và an toàn.

Trong quá trình làm đồ án, sinh viên đã rút ra được nhiều kinh nghiệm để tạo ra một sản phẩm hoàn thiện như: đầu tư thời gian, linh kiện trên thị trường, hiểu biết về các linh kiện, thiết kế board mạch, cơ khí, tính toán, đo đạc, thi công mô hình.

Bên cạnh đó, nhóm chưa hoàn thành được ổn định mô hình demo, chưa hiện thực được giao thức liên lạc không dây (wireless communication) giữa GateWay - Sensor Node và mạch in PCB. Nên đã lựa chọn UART, sử dụng breadBoard và các module để thực hiện demo.

Source code : [GitHub](#)

Khả năng mở rộng của đề tài:

1. Website điều khiển
2. Thay đổi màu sắc đèn LED phù hợp với từng thời kỳ tăng trưởng
3. Phương thức giao tiếp không dây giữa GateWay và SensorNode
4. Khả năng mở rộng của vườn (thêm vườn, thêm thiết bị) trên ứng dụng điện thoại mà không cần truy cập vào source code.

Cuối cùng, nhóm xin gửi lời cảm ơn tới giáo viên hướng dẫn ThS. Trần Thanh Bình đã đưa ra ý tưởng, giúp đỡ và đưa ra những hướng giải quyết cho nhóm.



References

- [1] Trần Thu Hà, "Điện tử cơ bản", NXB DH Quốc Gia Tp.HCM, 2013.
- [2] Nguyễn Thị Trọng Nghĩa, "Phương thức GET và POST", 2017.
<https://viblo.asia/p/phuong-thuc-get-va-post-aWj53VBYl6m>
- [3] MechaSolution, "UART và giao tiếp Serial", 2018.
<https://mechasolution.vn/Blog/uart-va-giao-tiep-serial>
- [4] TAPIT ARM, "STM32 CubeMX và Keil C lập trình STM32", 2019.
<https://tapit.vn/huong-dan-su-dung-stm32cubemx-va-keil-c-lap-trinh-stm32/>
- [5] Building a wifi Gateway using ESP8266.
https://www.mysensors.org/build/esp8266_gateway?fbclid=IwAR01Lrtzv6j4DPyS9R6C0QLHml9WkSZvbfNpe0QH81KNopsg1b5wSiHi-FI
- [6] Shop bán IC online.
<https://icdayroi.com/>