

ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KỸ THUẬT MÁY TÍNH
MÔN ĐỒ ÁN 1

-----o0o-----



UIT
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

BÁO CÁO CUỐI KÌ

**HỆ THỐNG ĐO VÀ HIỂN THỊ NHIỆT ĐỘ LÊN
WEBSITE**

GVHD: TS. PHẠM QUỐC HÙNG

SVTH: TRẦN HOÀNG THIÊN PHÚ MSSV:18521250

HOÀNG MINH NGHĨA MSSV:18521141

TP. HỒ CHÍ MINH, 10 THÁNG 7 NĂM 2021

Contents

0. Tóm tắt	5
1. Giới thiệu	5
1.1 Tổng quan đề tài	5
1.2 Mục tiêu nghiên cứu	5
2. Cơ sở lý thuyết	7
2.1 Đo nhiệt độ bằng cảm biến nhiệt độ hồng ngoại	7
2.2 Ứng dụng của IOT trong cuộc sống	9
2.3 Wifi	13
2.4 Cơ sở dữ liệu GOOLE FIREBASE	14
3. Thiết kế và thực hiện phần cứng	17
3.1 Yêu cầu thiết kế	17
3.1.1 Phân tích yêu cầu:	17
3.1.2 Phân tích yêu cầu:	20
4. Thiết kế và thực hiện phần mềm	24
4.1 Yêu cầu đặt ra cho phần mềm	24
4.2 Phân tích yêu cầu	24
4.3 Lưu đồ thuật toán:	24
4.4 Phần mềm lập trình cho hệ thống	25
4.4.1 Visual Studio Code và PlatformIO	25
4.4.2 Cách thiết lập một Project	25
4.5 Lưu đồ giải thuật chi tiết	28
4.5.1 Lưu đồ của một chương trình trong PlatformIO IDE	28
4.5.2 Lưu đồ của chương trình vi điều khiển kết nối với wifi trong PlatformIO IDE	30
4.5.3 Lưu đồ thiết lập dữ liệu theo thời gian thực	33
4.5.4 Lưu đồ của chương trình vi điều khiển kết nối với MLX90614 PlatformIO IDE	36
4.5.5 Lưu đồ của Website kết nối với database Firebase	38

5. Kết quả thực hiện.....	40
5.1 Cách thứ đo đạc, thử nghiệm.....	40
5.2 Thu thập số liệu.....	41
5.3 Giải thích và phân tích về kết quả thu được	47
6. Kết luận và hướng phát triển.....	48
6.1 Các kết quả đã thực hiện:	48
6.2 So sánh với mục tiêu đặt ra trong chương 1	49
6.3 Hướng phát triển của đề tài:.....	49
7. Tài liệu tham khảo	49
8. Phụ lục.....	Error! Bookmark not defined.

Hình 1 Cấu tạo của một cảm biến hồng ngoại.....	8
Hình 2 Mô hình IOT Smarthome.....	10
Hình 3 Mô hình IOT Smart Health	11
Hình 4 Mô hình mạng Wifi.....	14
Hình 5 Cơ sở dữ liệu Google Firebase.....	15
Hình 6 Các dịch vụ của Google Firebase	16
Hình 7 Cảm biến hồng ngoại MLX90614	17
Hình 8 Sơ đồ nguyên lý của cảm biến hồng ngoại MLX90614	18
Hình 9 Sơ đồ chân của cảm biến MLX90614.....	19
Hình 10 Module Esp8266	21
Hình 11 Sơ đồ nối chân ESP8266-MLX90614	23
Hình 12 Lưu đồ thuật toán	24
Hình 13 Cài đặt PlatformIO trên VScode	26
Hình 14 Tạo Project với PlatformIO trên VScode	27
Hình 15 Giao diện chọn Board	27
Hình 16 Chương trình tổng khởi tạo	28
Hình 17 Lưu đồ của một chương trình trong PlatformIO IDE	29
Hình 18 Lưu đồ của chương trình vi điều khiển kết nối với wifi trong PlatformIO IDE	32
Hình 19 Lưu đồ thiết lập dữ liệu theo thời gian thực	35
Hình 20 Lưu đồ của chương trình vi điều khiển kết nối với MLX90614 PlatformIO IDE	37
Hình 21 Cấu trúc cơ bản của một Website	38
Hình 22 Lưu đồ của Website kết nối với database Firebase	40
Hình 23 Biểu đồ đường số liệu đo được ở khoảng cách 2 cm.....	42
Hình 24 Lưu trữ nhiệt độ 2cm trên Firebase.....	42
Hình 25 Biểu đồ đường số liệu đo được ở khoảng cách 4 cm.....	43
Hình 26 Lưu trữ nhiệt độ 4cm trên Firebase.....	44
Hình 27 Biểu đồ đường số liệu đo được ở khoảng cách 6 cm.....	45
Hình 28 Lưu trữ nhiệt độ 6m trên Firebase	46

0. Tóm tắt

Việc triển khai và sử dụng Internet vạn vật (IoT) đã biến đổi các ngành công nghiệp theo cách chúng vận hành, giao tiếp và sử dụng dữ liệu. Trong lĩnh vực sản xuất, những thay đổi đó diễn ra với tốc độ nhanh chóng, và một ngành công nghiệp từng chậm phát triển giờ đây đang số hóa với tốc độ cực nhanh. Những lợi ích của IOT là điều có thể dễ nhận thấy như tăng tính hiệu quả, độ chính xác, giảm lỗi trong quá trình sản xuất, nâng cao năng suất công việc. IOT hầu như có thể xuất hiện trong bất kỳ lĩnh vực nào như Smart Consumer and User, Smart Enterprise, Smart Data, Connected & Autonomous Things. Nhận thấy được tính hữu ích của IOT trong lĩnh vực quản lý, giám sát, cụ thể ở đây là đo đạt nhiệt độ con người, môi trường và quản lý dữ liệu thông qua mạng kết nối internet. Nhóm mình đã quyết định thực hiện đề án này.

1. Giới thiệu

1.1 Tổng quan đề tài

Hiện nay dịch bệnh covid-19 đang hoành hành khắp mọi nơi thế giới, tuy nhiên hoạt động của con người không thể dừng lại buộc ta phải vừa làm việc vừa phòng chống covid-19 bằng cách đo nhiệt độ để kiểm soát con người tại các cửa ra vào của các tòa nhà, trường học, rạp chiếu phim ...

Để thực hiện đo nhiệt độ, ban quản lý phải thành lập các tổ đội trực tiếp đo nhiệt độ của con người bằng các thiết bị đo cầm tay. Điều này khá bất tiện và có thể mang lại những rủi ro nhiễm bệnh đáng kể với người đo và nguy cơ cao hơn với cộng đồng. Ngoài ra việc đo nhiệt độ một cách thủ công như vậy tốn khá nhiều thời gian với một lần đo cho mỗi người.

Trước những nhược điểm đó, mô hình đo nhiệt độ mới này được nghiên cứu để giải quyết và mang lại hiệu quả hơn

Một số nghiên cứu đi trước về đề tài đo nhiệt độ:

- Mô hình đo nhiệt độ và hiển thị trên màn hình SSD1306
- Mô hình đo nhiệt độ người ứng dụng camera AI nhận diện khuôn mặt và cảnh báo nếu nhiệt độ cao
- Mô hình đo nhiệt độ sử dụng board Aduirno.

1.2 Mục tiêu nghiên cứu

Mục đích của hệ thống đo nhiệt độ và hiển thị thông số lên website là cho phép người quản lý có thể giám sát nhiệt độ của con người ra vào ở một khoảng cách xa

thông qua kết nối internet mà không cần con người trực tiếp đo đạt nhiệt độ thủ công nữa.

Sản phẩm của nghiên cứu:

Một thiết bị phần cứng để đo đạt nhiệt độ cơ thể của con người, của môi trường với độ trễ thấp

Một website: 1 trang web có giao diện để tiếp nhận dữ liệu đo đạt từ phần cứng và hiển thị thông số lên trang web. Người quản lý có thể truy cập và xem xét lại các dữ liệu đã ghi được thông qua internet.

Công việc cần làm :

Tuần	Nội Dung	Xác nhận GVHD
Tuần 1	GVHD hướng dẫn yêu cầu làm đồ án.	
Tuần 2	Đề cương tóm tắt nội dung đồ án	
Tuần 3	Tìm hiểu về Module ESP8266, Cảm biến nhiệt độ hồng ngoại MLX90614	
Tuần 4	Tìm hiểu về CSS, HTML, JS	
Tuần 5	Mô phỏng hệ thống phần cứng	
Tuần 6	Lập trình hệ thống phần cứng	
Tuần 7	Lập trình hệ thống phần cứng	
Tuần 8	Xây dựng giao diện trang web	
Tuần 9	Mô phỏng hệ thống, tính toán, cân chỉnh, kiểm tra	
Tuần 10	Tiến hành thi công, thiết kế phần cứng	
Tuần 11	Nghiệm thu hệ thống	
Tuần 12	Hoàn thiện báo cáo và gửi GVHD xem trước	

Tuần 13	Báo cáo cuối kì	
------------	-----------------	--

2. Cơ sở lý thuyết

2.1 Đo nhiệt độ bằng cảm biến nhiệt độ hồng ngoại

Theo định luật bức xạ Planck, tất cả các vật thể khi ở nền nhiệt độ trên 0 độ K đều mang nhiệt năng thì đều phá ra bức xạ điện từ, những bức xạ này là loại bức xạ có bước sóng nằm ngoài vùng ánh sáng khả kiến, vì thế mắt con người không thể nhìn thấy được các bức xạ này. Dựa vào đặc tính này của vật thể người ta nghĩ ra một cách để đo được nhiệt độ của vật thể đó đó là sử dụng cảm biến nhiệt độ hồng ngoại.

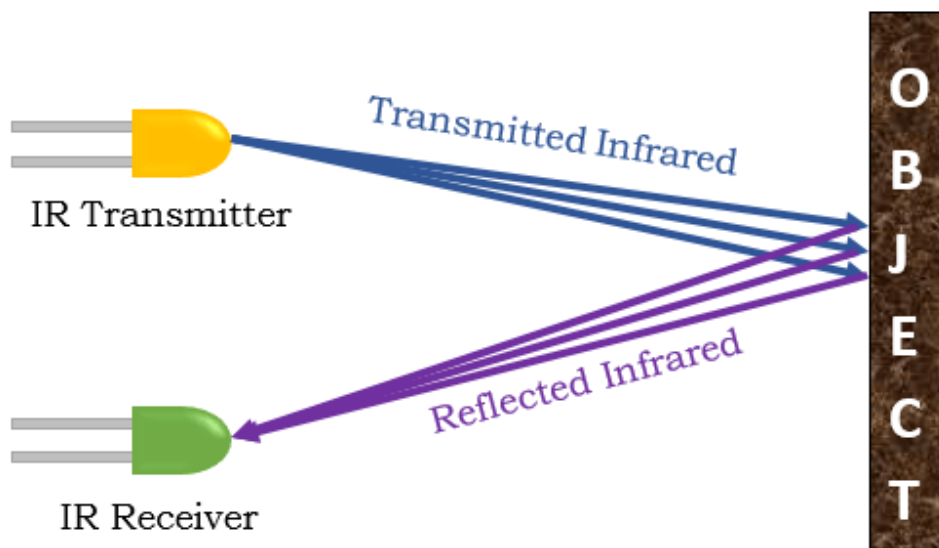
Cảm biến hồng ngoại là một thiết bị điện tử đơn giản phát ra và phát hiện bức xạ hồng ngoại để tìm ra các đối tượng / chướng ngại vật nhất định trong phạm vi của nó.

Cảm biến hồng ngoại hoạt động dựa trên ba định luật Vật lý cơ bản:

- **Định luật bức xạ Planck** : Bất kỳ vật thể nào có nhiệt độ không bằng 0 tuyệt đối (0 Kelvin) đều phát ra bức xạ.
- **Định luật Stephan Boltzmann** : Tổng năng lượng do vật đen phát ra ở tất cả các bước sóng có liên quan đến nhiệt độ tuyệt đối.
- **Định luật Dịch chuyển của Wein**: Các vật thể có nhiệt độ khác nhau phát ra quang phổ đạt cực đại ở các bước sóng khác nhau tỷ lệ nghịch với Nhiệt độ.

Cấu tạo của một cảm biến hồng ngoại:

- Máy phát hồng ngoại: Trong hầu hết các trường hợp, bộ tản nhiệt thân đen, đèn vonfram, cacbua silicon, tia laser hồng ngoại, đèn LED có bước sóng hồng ngoại được sử dụng làm nguồn.
- Phương tiện truyền dẫn: Chân không, khí quyển và sợi quang học được sử dụng làm môi trường.
- Máy thu hồng ngoại: bộ thu IR là diode quang và bóng bán dẫn quang. Chúng có khả năng phát hiện bức xạ hồng ngoại. Do đó bộ thu IR còn được gọi là bộ dò hồng ngoại.



Hình 1 Cấu tạo của một cảm biến hồng ngoại

2.2 Ứng dụng của IOT trong cuộc sống

Internet of Things. Mạng IoT là một trong những tính năng chính cần được hỗ trợ bởi IoT. Nó kết nối mọi thứ, con người, ứng dụng và dữ liệu, và để kết nối, mạng là điều cần thiết. Nó kết nối thông qua Internet, vì vậy đó là lý do tại sao chúng ta có Internet of Things, IoT, là tên gọi tổng thể cho công nghệ mới nổi này. Nó về cơ bản nhằm mục tiêu để kết nối mọi thứ có thể kiểm soát được và cần được giám sát. Nó cho phép điều khiển từ xa, quản lý, các dịch vụ tích hợp tương tác. Thang đo mạng IoT. Số lượng thiết bị di động sẽ vượt quá số người trên trái đất. Số người là khoảng 7,6 tỷ, và số lượng thiết bị di động dự kiến sẽ vượt quá con số này dựa trên thực tế là một số người sẽ có nhiều hơn một thiết bị thông minh, nhiều hơn một thiết bị di động hỗ trợ nhu cầu của họ.

Xét rằng, mong đợi số lượng các thiết bị di động, đặc biệt là các thiết bị thông minh vượt quá số lượng người, đó là rất rõ ràng. Sau đó, dự đoán sẽ có 50 tỷ thứ kết nối với Internet vào năm 2020, đó là một con số to lớn. Ngoài ra, bạn cần phải xem xét thực tế là nhìn vào tổng dân số 7,6 tỷ so với 50 tỷ.

Một vài ứng dụng trong cuộc sống áp dụng IOT:

❖ Smarthome:

Smarthome là ứng dụng IOT phổ biến trên toàn thế giới. Vậy, một ngôi nhà thông minh là một ngôi nhà như thế nào? Gần như mọi khía cạnh của cuộc sống, công nghệ IOT có thể xâm nhập vào không gian gia đình:



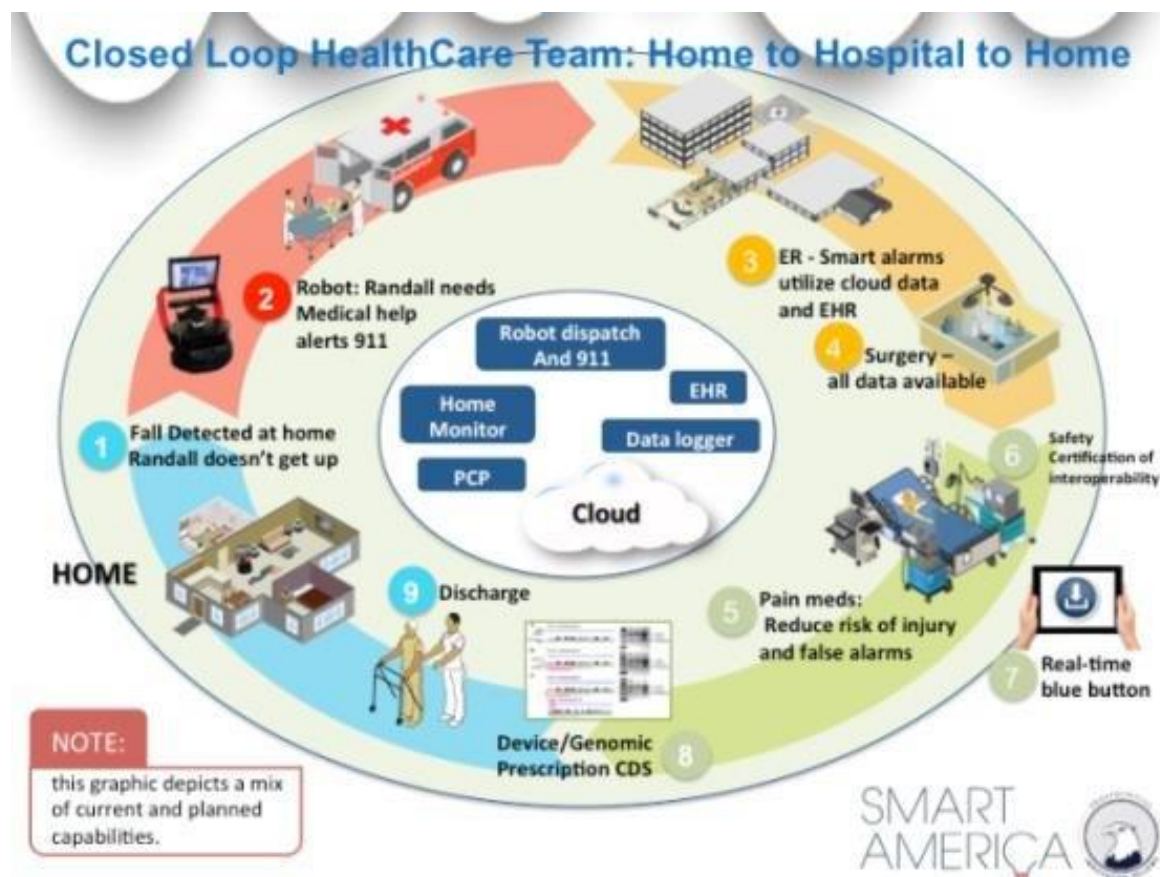
Hình 2 Mô hình IOT Smarthome

- TV thông minh kết nối internet để truy cập nội dung thông qua các ứng dụng, chẳng hạn như video và nhạc theo yêu cầu. Một số TV thông minh cũng bao gồm nhận dạng giọng nói hoặc cử chỉ.
- Ngoài việc có thể được điều khiển từ xa và tùy chỉnh, các hệ thống chiếu sáng thông minh, chẳng hạn như Hue của Philips Lighting Holding BV, có thể phát hiện khi nào có người ở trong phòng và điều chỉnh ánh sáng khi cần thiết. Bóng đèn thông minh cũng có thể tự điều chỉnh dựa trên mức độ sẵn có của ánh sáng ban ngày.
- Sử dụng khóa thông minh và bộ mở cửa nhà để xe, người dùng có thể cấp hoặc từ chối quyền truy cập cho khách. Ổ khóa thông minh cũng có thể phát hiện khi cư dân ở gần và mở khóa cửa cho họ.
- Với camera an ninh thông minh, cư dân có thể giám sát ngôi nhà của mình khi họ đi vắng hoặc đi nghỉ mát. Cảm biến chuyển động thông minh cũng có thể xác định sự khác biệt giữa cư dân, du khách, vật nuôi và kẻ trộm, đồng thời có thể thông báo cho cơ quan chức năng nếu phát hiện hành vi đáng ngờ.

❖ Smart Health:

Các thiết bị hỗ trợ Internet of Things (IoT) đã giúp khả năng giám sát từ xa trong lĩnh vực chăm sóc sức khỏe, giải phóng tiềm năng giữ cho bệnh nhân an toàn và khỏe mạnh, đồng thời trao quyền cho các bác sĩ trong việc cung cấp dịch vụ chăm sóc tuyệt vời nhất. Nó cũng làm tăng sự tương tác và sự hài lòng của bệnh nhân khi tương tác với bác sĩ trở nên dễ dàng và hiệu quả hơn. Hơn nữa, theo dõi từ xa sức khỏe của bệnh nhân giúp giảm thời gian nằm viện và ngăn ngừa tái nhập viện. IoT cũng có tác động lớn đến việc giảm đáng kể chi phí chăm sóc sức khỏe và cải thiện kết quả điều trị.

Không nghi ngờ gì nữa, IoT đang chuyển đổi ngành công nghiệp chăm sóc sức khỏe bằng cách xác định lại không gian của thiết bị và sự tương tác giữa con người trong việc cung cấp các giải pháp chăm sóc sức khỏe. IoT có các ứng dụng trong chăm sóc sức khỏe mang lại lợi ích cho bệnh nhân, gia đình, bác sĩ, bệnh viện và các công ty bảo hiểm.



Hình 3 Mô hình IOT Smart Health

IoT cho bệnh nhân - Các thiết bị ở dạng thiết bị đeo được như dây đeo thể dục và các thiết bị được kết nối không dây khác như vòng bút theo dõi huyết áp và nhịp tim, máy đo đường huyết, v.v. cho phép bệnh nhân tiếp cận với sự chú ý được cá nhân hóa. Các thiết bị này có thể được điều chỉnh để nhắc nhở đếm calo, kiểm tra tập thể dục, các cuộc hẹn, các biến thể huyết áp và hơn thế nữa.

IoT đã thay đổi cuộc sống của mọi người, đặc biệt là bệnh nhân cao tuổi, bằng cách cho phép theo dõi liên tục các tình trạng sức khỏe. Điều này có ảnh hưởng lớn đến những người sống một mình và gia đình của họ. Khi có bất kỳ xáo trộn hoặc thay đổi nào trong các hoạt động thường ngày của một người, cơ chế cảnh báo sẽ gửi tín hiệu đến các thành viên trong gia đình và các nhà cung cấp dịch vụ y tế có liên quan.

IoT cho bác sĩ - Bằng cách sử dụng thiết bị đeo được và thiết bị giám sát tại nhà khác được nhúng với IoT, bác sĩ có thể theo dõi sức khỏe của bệnh nhân hiệu quả hơn. Họ có thể theo dõi sự tuân thủ của bệnh nhân đối với kế hoạch điều trị hoặc bất kỳ nhu cầu chăm sóc y tế nào ngay lập tức. IoT cho phép các chuyên gia chăm sóc sức khỏe quan sát hơn và chủ động kết nối với bệnh nhân. Dữ liệu được thu thập từ các thiết bị IoT có thể giúp các bác sĩ xác định quy trình điều trị tốt nhất cho bệnh nhân và đạt được kết quả mong đợi.

IoT cho bệnh viện - Ngoài việc theo dõi sức khỏe của bệnh nhân, có nhiều lĩnh vực khác mà các thiết bị IoT rất hữu ích trong bệnh viện. Các thiết bị IoT được gắn thẻ cảm biến được sử dụng để theo dõi vị trí thời gian thực của thiết bị y tế như xe lăn, máy khử rung tim, máy phun sương, máy bơm oxy và các thiết bị giám sát khác. Việc triển khai của nhân viên y tế tại các địa điểm khác nhau cũng có thể được phân tích theo thời gian thực.

Sự lây lan của các bệnh nhiễm trùng là một mối quan tâm lớn đối với bệnh nhân trong bệnh viện. Các thiết bị giám sát vệ sinh hỗ trợ IoT giúp ngăn ngừa bệnh nhân bị lây nhiễm. Các thiết bị IoT cũng giúp quản lý tài sản như kiểm soát hàng tồn kho hiệu thuốc và giám sát môi trường, chẳng hạn như kiểm tra nhiệt độ tủ lạnh, kiểm soát độ ẩm và nhiệt độ.

IoT cho các công ty bảo hiểm sức khỏe - Có rất nhiều cơ hội cho các công ty bảo hiểm sức khỏe với các thiết bị thông minh được kết nối IoT. Các công ty bảo hiểm có thể tận dụng dữ liệu thu thập được thông qua các thiết bị theo dõi sức khỏe cho các hoạt động bảo lãnh phát hành và yêu cầu bồi thường của họ. Dữ liệu này sẽ cho phép họ phát hiện các khiếu nại gian lận và xác định các triển vọng đề bảo lãnh phát hành. Các thiết bị IoT mang lại sự minh bạch giữa các công ty bảo hiểm và khách hàng trong các quy trình bảo lãnh phát hành, định giá, xử lý khiếu nại và đánh giá rủi ro. Dưới ánh sáng của các quyết định dựa trên dữ liệu được IoT nắm

bắt trong tất cả các quy trình hoạt động, khách hàng sẽ có tầm nhìn đầy đủ về những suy nghĩ tiềm ẩn đằng sau mọi quyết định được đưa ra và kết quả của quá trình.

Các công ty bảo hiểm có thể khuyến khích khách hàng của họ sử dụng và chia sẻ dữ liệu sức khỏe do các thiết bị IoT tạo ra. Họ có thể thưởng cho khách hàng sử dụng thiết bị IoT để theo dõi các hoạt động thường ngày của họ và tuân thủ các kế hoạch điều trị và các biện pháp sức khỏe phòng ngừa. Điều này sẽ giúp doanh nghiệp bảo hiểm giảm thiểu các khoản bồi thường một cách đáng kể. Các thiết bị IoT cũng có thể cho phép các công ty bảo hiểm xác nhận các yêu cầu thông qua dữ liệu được thu thập bởi các thiết bị này.

Ngoài ra IOT còn có thể áp dụng rất nhiều trong những lĩnh vực khác nhau như là:

- **Smart Consumer and User:** Facilitative Reality, Connected homes, Connected cars, Smart health, Share Economy
- **Smart Enterprise:** Transportation, Retail, Building & Construction, Manufacturing, Oil & gas / Energy, Healthcare
- **Smart Data:** Big data, Data security, AI & Machinelearning
- **Smart Cloud:** Cloud life cycle, Event & Cloud integration, Data center
- **Connected & Autonomous Things:** Drones, Robots, Machine
- **Smart Network:** VPN/Network Security, Ethernet Wired, Platforms, Satellites, Wifi

2.3 Wifi

Mạng WiFi (802.11) chỉ đơn giản là một kết nối internet được chia sẻ với nhiều thiết bị trong gia đình hoặc cơ sở kinh doanh thông qua bộ định tuyến không dây. Bộ định tuyến được kết nối trực tiếp với modem internet của bạn và hoạt động như một trung tâm để phát tín hiệu internet đến tất cả các thiết bị hỗ trợ Wi-Fi của bạn. Điều này giúp bạn linh hoạt trong việc kết nối Internet miễn là bạn đang ở trong vùng phủ sóng của mạng.

Hệ thống này rất phổ biến ngày nay, ta có thể thấy hầu hết mọi nơi đều phủ sóng Wifi, từ gia đình, cửa hàng, nhà hàng, quán cafe cho đến trường học, bệnh viện, trung tâm mua sắm



Hình 4 Mô hình mạng Wifi

Các chuẩn giao tiếp của Wifi và dải tần số hoạt động

- **802.11b:** Đây là chuẩn wifi phiên bản đầu tiên trên thị trường, nó có thể phát ra tín hiệu ở dải tần số 2.4 GHz và xử lý dữ liệu ở tốc độ 1 đến 11 Mb/s. Chuẩn wifi này có tốc độ thấp và giá thành rẻ, không còn phổ biến ngày nay.
- **802.11g:** Chuẩn wifi này có thể phát ra tín hiệu ở dải tần số 2.4 GHz đến 54 Mb/s. Chuẩn wifi này có tốc độ cao hơn chuẩn 802.11b vì sử dụng công nghệ OFDM, một công nghệ mã hóa hiệu quả hơn
- **802.11a:** Chuẩn wifi này có thể phát ra tần số 5GHz và tốc độ xử lý dữ liệu đạt đến 54 Mb/s. Nó cũng được sử dụng công nghệ mã hóa OFDM.
- **802.11n:** Chuẩn wifi này phát ra tần số 2.4 GHz nhưng nhanh hơn nhiều so với chuẩn 802.11a. Tốc độ của chuẩn này đạt đến 300 Mb/s
- **802.11ac:** Chuẩn wifi này phát ra tần số 2.4 GHz và đạt tốc độ xử lý lên đến 6933 Mb/s

2.4 Cơ sở dữ liệu GOOLE FIREBASE

Về lịch sử, Firebase (tiền thân của Evolve) trước đây là một start up được thành lập vào năm 2011 bởi Andrew Lee và James Tamplin. Ban đầu, Evolve chỉ cung cấp cơ sở dữ liệu để các lập trình viên thiết kế ứng dụng chat. Tuy nhiên họ nhanh chóng nhận ra tiềm năng sản phẩm của mình khi nhận thấy các khách hàng không sử dụng CSDL để làm ứng dụng chat, mà thay vào đó để lưu các thông tin như là

game progress. Bộ đôi Lee và Tamplin quyết định tách mảng realtime ra để thành lập một công ty độc lập – chính là Firebase – vào tháng 4 năm 2012. Sau nhiều lần huy động vốn và gặt hái được những thành công nổi bật. Firebase được Google đề ý. Vào tháng 10 năm 2014, Firebase gia nhập gia đình Google. Fire, theo hướng đi của Google, chính thức hỗ trợ Android, IOS, Web.



Hình 5 Cơ sở dữ liệu Google Firebase

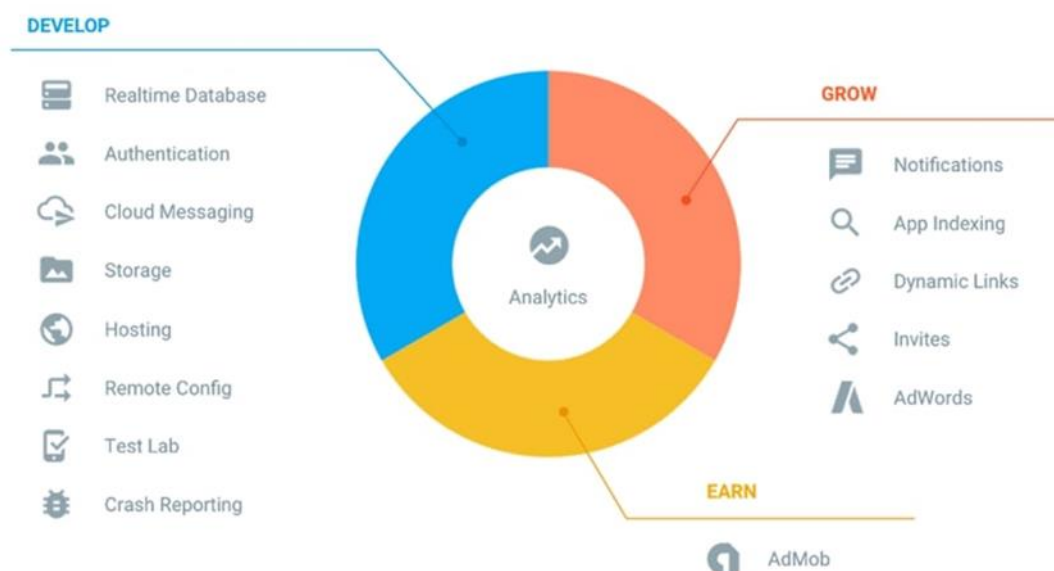
Về thư viện, Firebase hỗ trợ chính thức:

- GeoFire
- GularFire
- BerFire
- ReactFire
- Ionic

Google Firebase bao gồm:

- Cloud Firestore

- ML Kit
- Cloud Functions
- Authentication
- Hosting
- Cloud Storage
- Realtime Database



Hình 6 Các dịch vụ của Google Firebase

Trong đó Realtime Database (cơ sở dữ liệu thời gian thực) :

- Hỗ trợ : IOS, Android, Web, C++. Unity
- Lưu trữ và đồng bộ dữ liệu theo thời gian thực. Dịch vụ này được lưu trữ trực tiếp trên đám mây NoSQL. Dữ liệu được đồng bộ hóa trên tất cả các ứng dụng khách trong thời gian thực và vẫn khả dụng khi ứng dụng của bạn ngoại tuyến. Cơ sở dữ liệu thời gian thực Firebase là cơ sở dữ liệu được lưu trên đám mây. Dữ liệu được lưu trữ dưới dạng JSON và được đồng bộ hóa theo thời gian thực cho mọi máy khác được kết nối. Khi bạn tạo ứng dụng đa nền tảng với SDK IOS, Android và JavaScript, tất cả máy của bạn sẽ chia sẻ một phiên bản cơ sở dữ liệu thời gian thực và tự động nhận các bản cập nhật với dữ liệu mới nhất.

3. Thiết kế và thực hiện phần cứng

3.1 Yêu cầu thiết kế

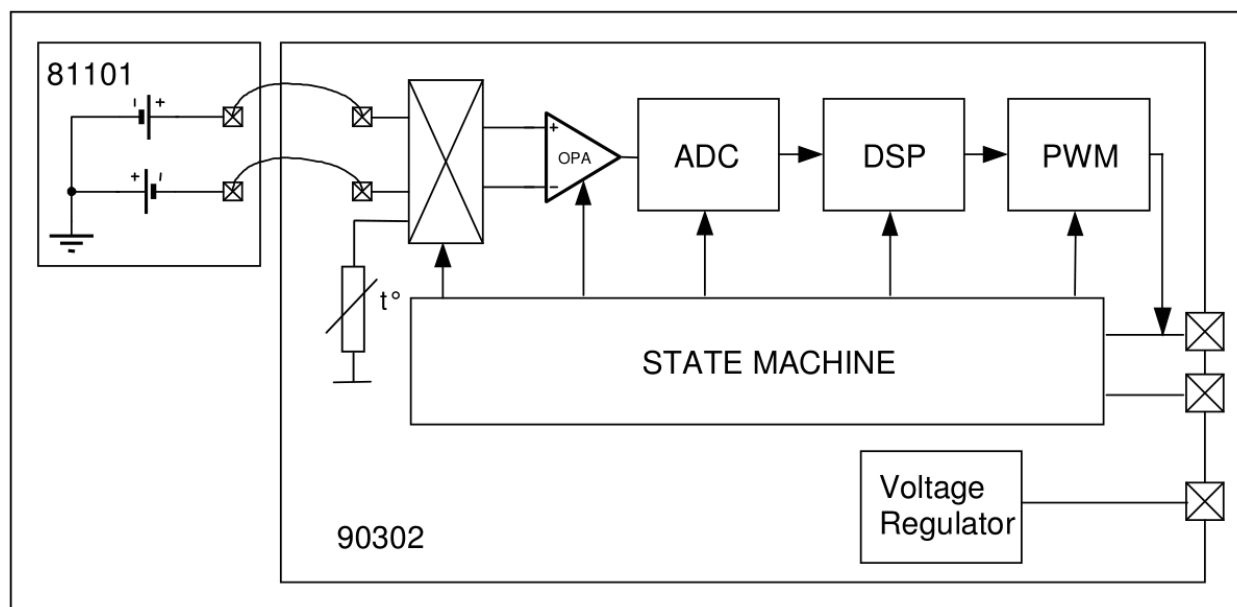
- Thiết kế nhỏ, gọn.
- Đo được nhiệt độ vật thể.
- Đo được nhiệt độ môi trường.
- Tiếp nhận dữ liệu đo được, xử lý và đưa thông tin lên máy chủ thông qua mạng internet.

3.1.1 Phân tích yêu cầu:

Để đo được nhiệt độ, cần một cảm biến đo nhiệt độ, trên thị trường có nhiều cảm biến nhiệt độ như là MLX90614, ZTP115, TPIS 1S, DHT11. Tuy nhiên để đáp ứng đầy đủ về mặt yêu cầu để đo được nhiệt độ từ xa, nguồn cung cấp, giao tiếp I2C để tránh việc dùng nhiều dây dẫn dẫn đến sụt áp và tích hợp bộ chuyển đổi analog to digital, ngoài ra còn đáp ứng về mặt giá cả phải chăng thì ta dùng MLX90614

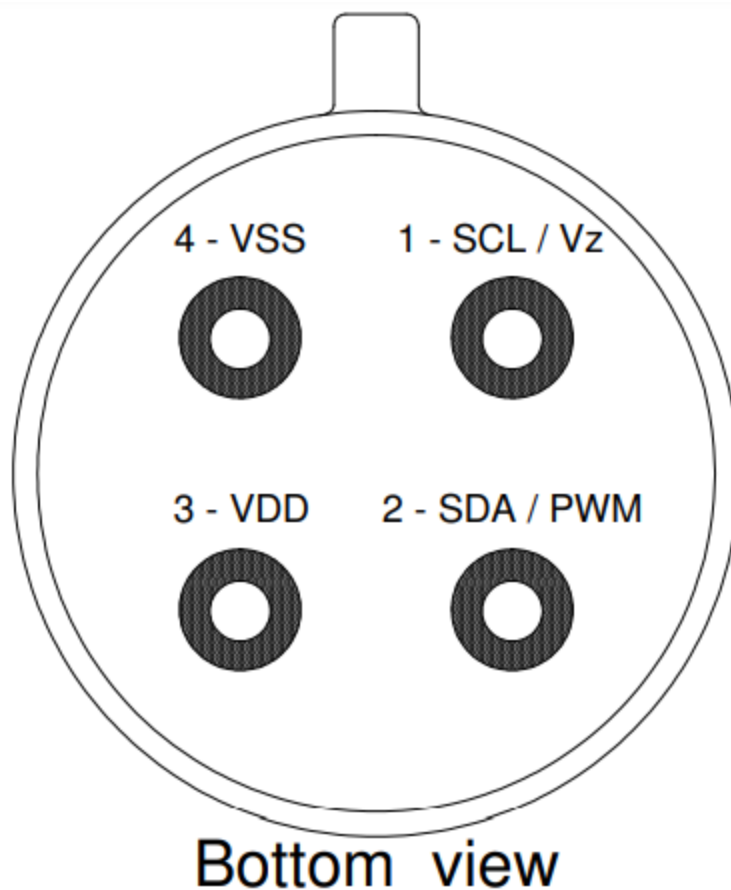


Hình 7 Cảm biến hồng ngoại MLX90614



Hình 8 Sơ đồ nguyên lý của cảm biến hồng ngoại MLX90614

Sơ đồ chân của MLX90614:



Hình 9 Sơ đồ chân của cảm biến MLX90614

Pin name	Chức năng
SCL / Vz	Serial clock input cho giao thức truyền thông 2 dây. Zener 5.7V có sẵn ở chân này để kết nối bóng bán dẫn lưỡng cực bên ngoài với MLX90614Axx để cung cấp thiết bị từ nguồn 8... 16V bên ngoài
SDA / PWM	Đầu vào / đầu ra kỹ thuật số. Ở chế độ bình thường, nhiệt độ đối tượng đo được có sẵn ở chân này được Điều chế Độ rộng Xung. Trong chế độ tương thích với SMBus, chân được tự động định cấu hình là NMOS cổng mở.
VDD	Điện áp cung cấp bên ngoài.

VSS	Ground
-----	--------

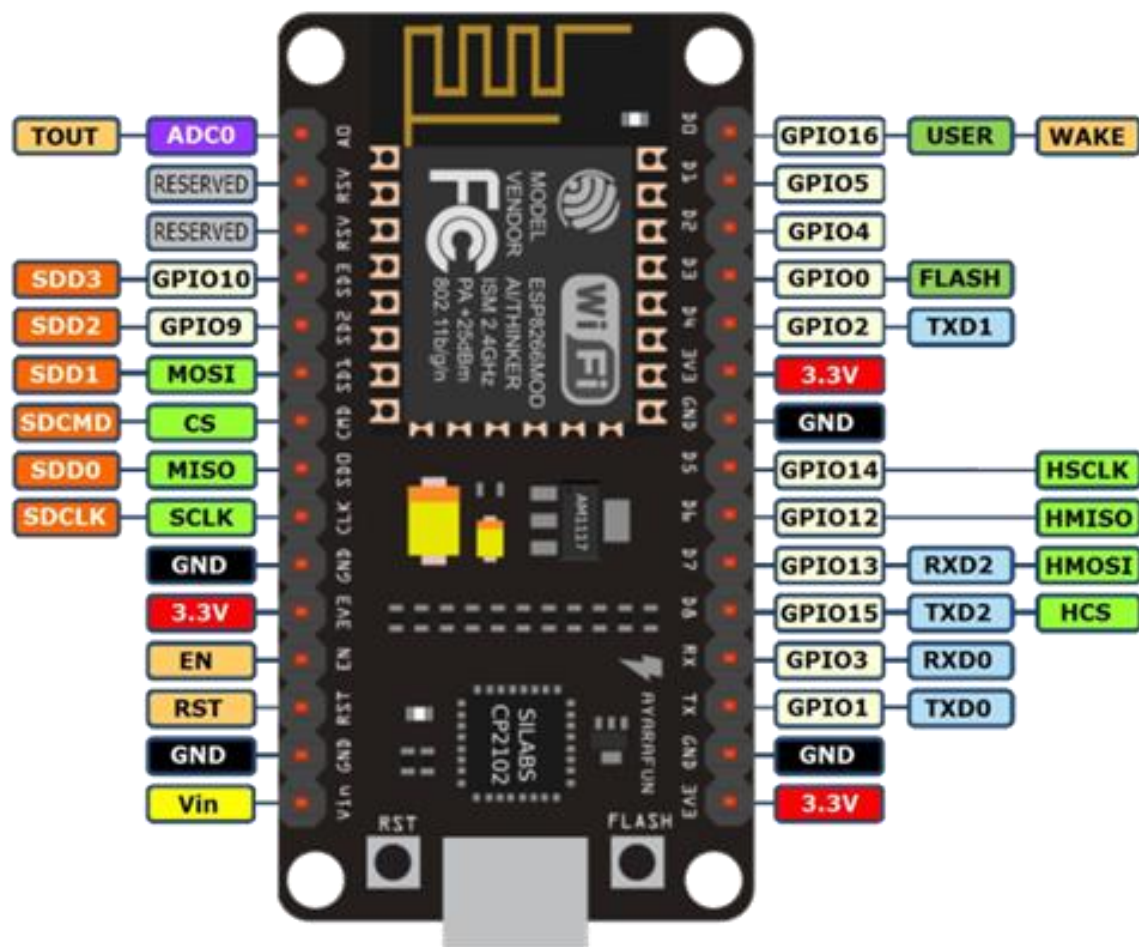
Tính năng và lợi ích:

- Kích thước nhỏ, chi phí thấp
- Dễ dàng tích hợp
- Phạm vi đo nhiệt độ rộng:
 - $-40^{\circ}\text{C} \dots +125^{\circ}\text{C}$ cho nhiệt độ cảm biến và
 - $-70^{\circ}\text{C} \dots +380^{\circ}\text{C}$ cho nhiệt độ vật thể.
- Độ chính xác cao $0,5^{\circ}\text{C}$ trong phạm vi rộng phạm vi nhiệt độ ($0^{\circ}\text{C} \dots +50^{\circ}\text{C}$ cho cả T_a và T_o)
- Hiệu chuẩn độ chính xác cao (y tế)
- Độ phân giải đo $0,02^{\circ}\text{C}$
- Phiên bản vùng đơn và vùng kép
- Giao diện kỹ thuật số tương thích SMBus
- Đầu ra PWM có thể tùy chỉnh để liên tục đọc hiệu
- Có sẵn trong các phiên bản 3V và 5V
- Thích ứng đơn giản cho các ứng dụng $8\text{V} \dots 16\text{V}$
- Chế độ ngủ để giảm năng lượng tiêu dùng
- Các tùy chọn gói khác nhau cho các ứng dụng và tính linh hoạt của phép đo

3.1.2 Phân tích yêu cầu:

Vì hệ thống này được thiết kế để lắp đặt trước một tòa nhà, và để hệ thống có thể tiếp nhận dữ liệu và truyền dữ liệu lên server thông qua bộ định tuyến được lắp đặt trong tòa nhà thì ta cần một Module kết nối Wifi, ngoài ra còn hỗ trợ chuẩn giao tiếp I2C phù hợp với Module cảm biến nhiệt độ MLX 90614. Ngoài ra còn tính đến chi phí giá cả phải chăng thì phương án tốt nhất đưa ra là sử dụng Node MCU esp8266.

Đây là một module nhiều tính năng, giá cả rẻ so với những tính năng mà nó mang lại, có chuẩn kết nối 802.11 với tần số 2.4 GHz phù hợp với băng tần Wifi mà các bộ định tuyến sử dụng tại các tòa nhà.



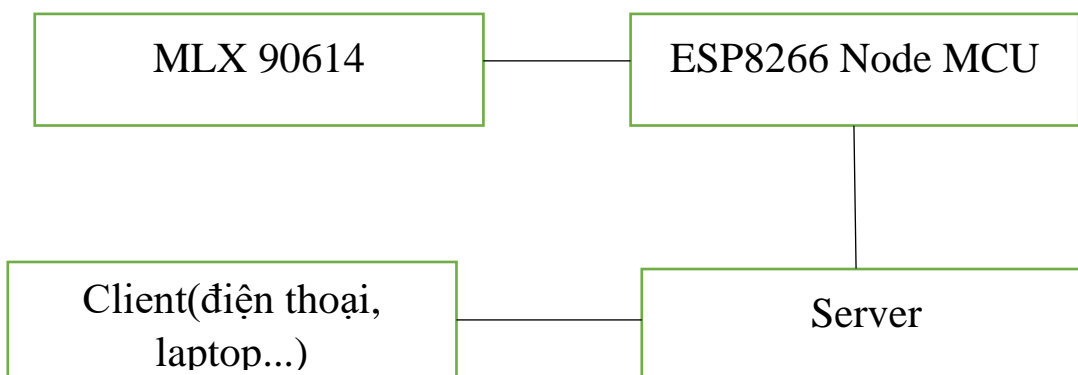
Hình 10 Module Esp8266

Thông số kỹ thuật:

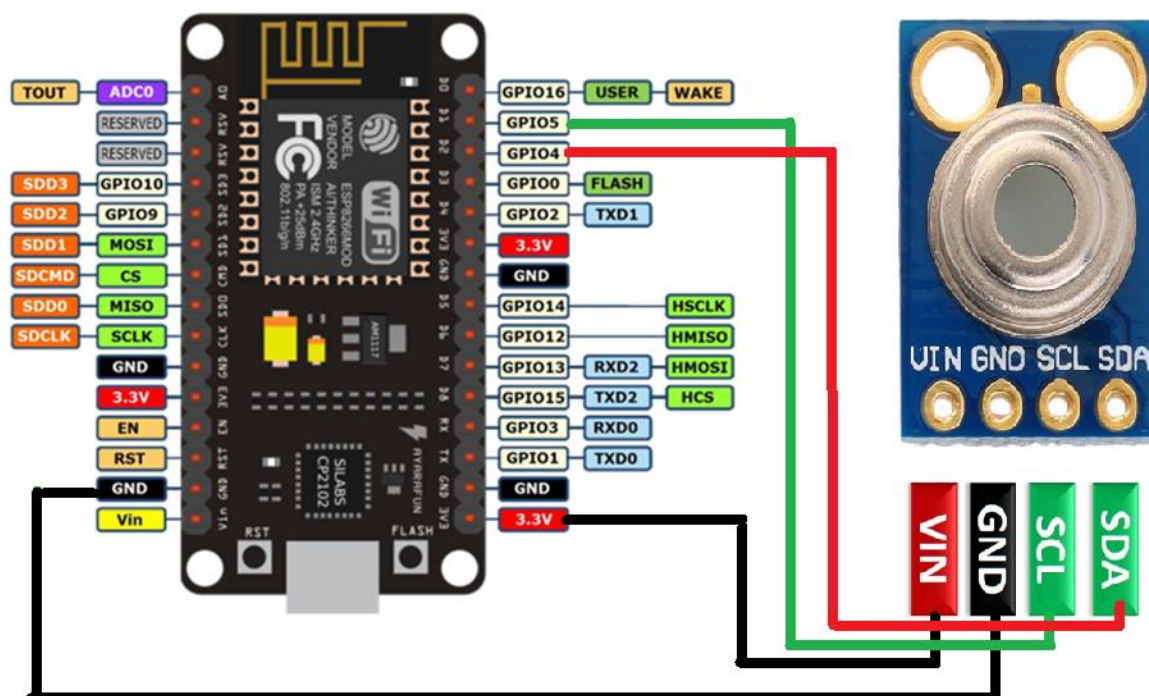
Categories	Items	Parameters
Wifi	Certification	Wifi Alliance
	Protocols	802.11 b/g/n (HT20)
	Tần số	2.4 GHz ~ 2.5GHz (2400 MHz ~ 2483.5 MHz)
	TX Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
		802.11 b: -91 dbm (11 Mbps)
		802.11 g: -75 dbm (54 Mbps)

		802.11 n: –72 dbm (MCS7)
	Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip
Hardware	CPU	Tensilica L106 32-bit processor
	Ngoại vi	UART/SDIO/SPI/I2C/I2S/IR Remote Control
		UART/SDIO/SPI/I2C/I2S/IR Remote Control GPIO/ADC/PWM/LED Light & Button
	Nguồn cấp	2.5 V ~ 3.6 V
	Dòng	Average value: 80 mA
	Phạm vi nhiệt độ	–40 °C ~ 125 °C
	Package Size	QFN32-pin (5 mm x 5 mm)
	External Interface	-
Software	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Bảo mật	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / Firmware and SDK for fast on-chip programming
	Giao thức mạng	IPv4, TCP/UDP/HTTP
	Cấu hình User	AT Instruction Set, Cloud Server, Android/iOS App

Sơ đồ khối kết nối các module:



Sơ đồ mạch tiết:



Hình 11 Sơ đồ nối chân ESP8266-MLX90614

MLX 90614 trực tiếp thu thập tín hiệu phát xạ từ vật thể, chuyển thành tín hiệu analog, khuếch đại tín hiệu, chuyển thành tín hiệu digital, xử lý rồi gửi nó đến ESP8266. ESP8266 nhận tín hiệu và đưa dữ liệu lên Firebase thông qua kết nối Wifi. Người dùng thiết bị có kết nối internet truy cập vào server để xem thông tin dữ liệu đã thu được.

4. Thiết kế và thực hiện phần mềm

4.1 Yêu cầu đặt ra cho phần mềm

- Xây dựng được hệ thống mô phỏng hệ thống đo nhiệt độ vật thể từ xa
- Xây dựng được website hiển thị nhiệt độ

4.2 Phân tích yêu cầu

Phần mềm hiển thị các giá trị nhiệt độ theo thời gian

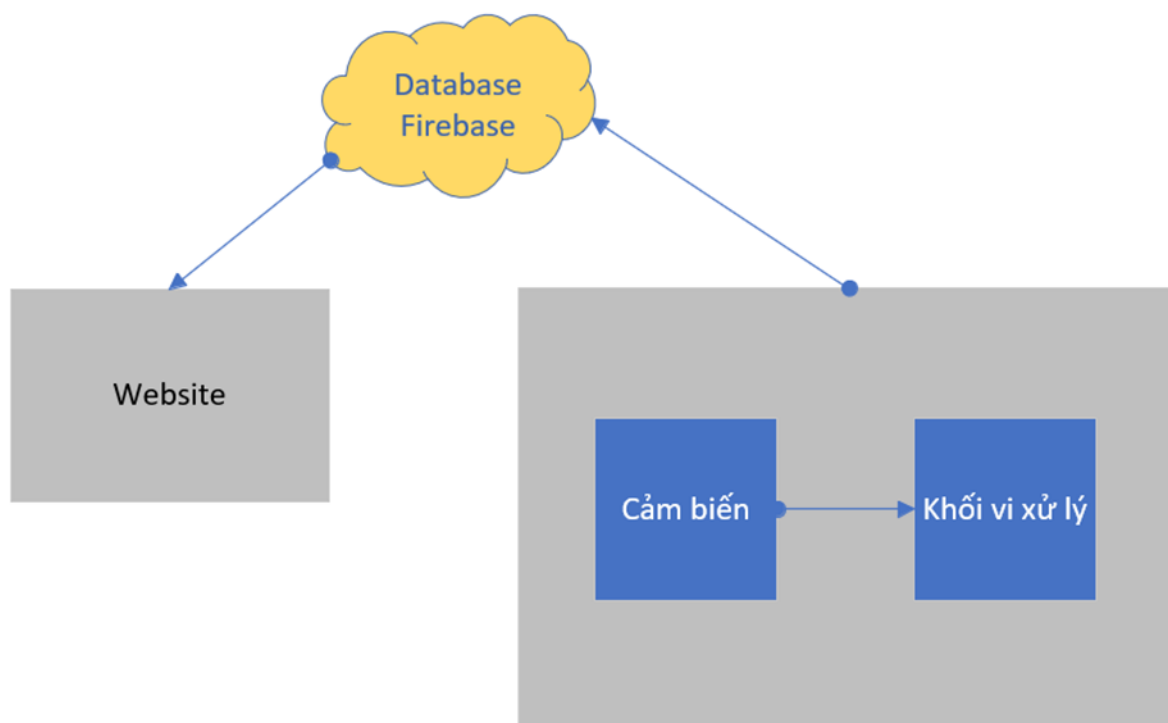
- Giao Diện :

+ Trang hiển thị các điểm nhiệt độ tại các thời gian thực

- Chức năng:

+ Cập nhập dữ liệu mỗi giây một lần

4.3 Lưu đồ thuật toán:



Hình 12 Lưu đồ thuật toán

Chức năng từng khối:

- Khối Websie : Người dùng truy cập để theo dõi nhiệt độ

- Khối Database Firebase : Lưu trữ dữ liệu trên database (Firebase)
- Khối vi xử lý : tạo lệnh điều khiển hoạt động của hệ thống. Nhận tín hiệu từ khối Cảm biến gửi về xử lý và gửi dữ liệu đến khối Database Firebase
- Khối cảm biến : đo đặc thông số rồi gửi đến Khối vi xử lý

4.4 Phần mềm lập trình cho hệ thống

4.4.1 Visual Studio Code và PlatformIO

- Visual Studio code

Visual Studio Code (VS Code hay VSC) là một trong những trình soạn thảo mã nguồn phổ biến nhất được sử dụng bởi các lập trình viên. Nhanh, nhẹ, hỗ trợ đa nền tảng, nhiều tính năng và là mã nguồn mở chính là những ưu điểm vượt trội khiến VS Code ngày càng được ứng dụng rộng rãi. Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor.

Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.

- PlatformIO

PlatformIO là hệ sinh thái mã nguồn mở, cung cấp thư viện để lập trình cho nhiều dòng vi điều khiển. PlatformIO có tích hợp thư viện Arduino, cho phép người dùng PlatformIO có thể dễ dàng lập trình bằng thư viện Arduino cho các dòng vi điều khiển được hỗ trợ – ESP8266 là một trong số đó.

PlatformIO là một package mà chúng ta có thể cài đặt trên Visual Studio Code. Chúng ta sẽ thấy dễ dàng và chuyên nghiệp hơn, khi sử dụng PlatformIO trên Visual Studio Code để lập trình cho ESP8266.

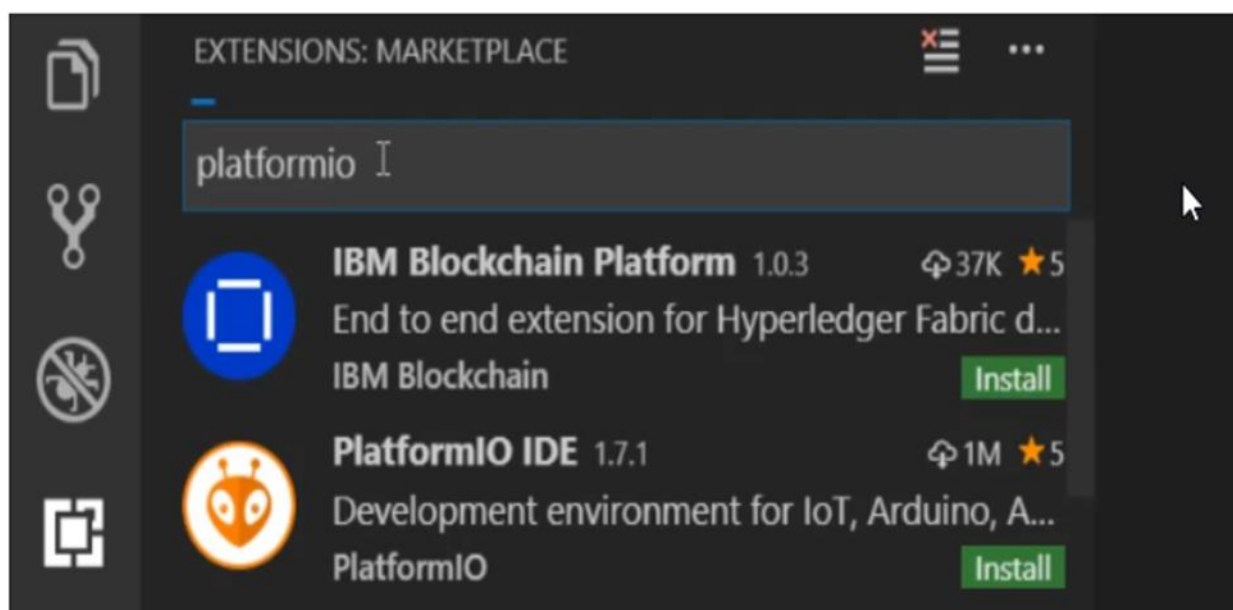
4.4.2 Cách thiết lập một Project

- Cài đặt Visual Studio Code

Truy cập trang web : <https://code.visualstudio.com/download> để tải về chương trình phù hợp với hệ điều hành hiện tại của máy tính đang sử dụng.

- Cài đặt PlatformIO

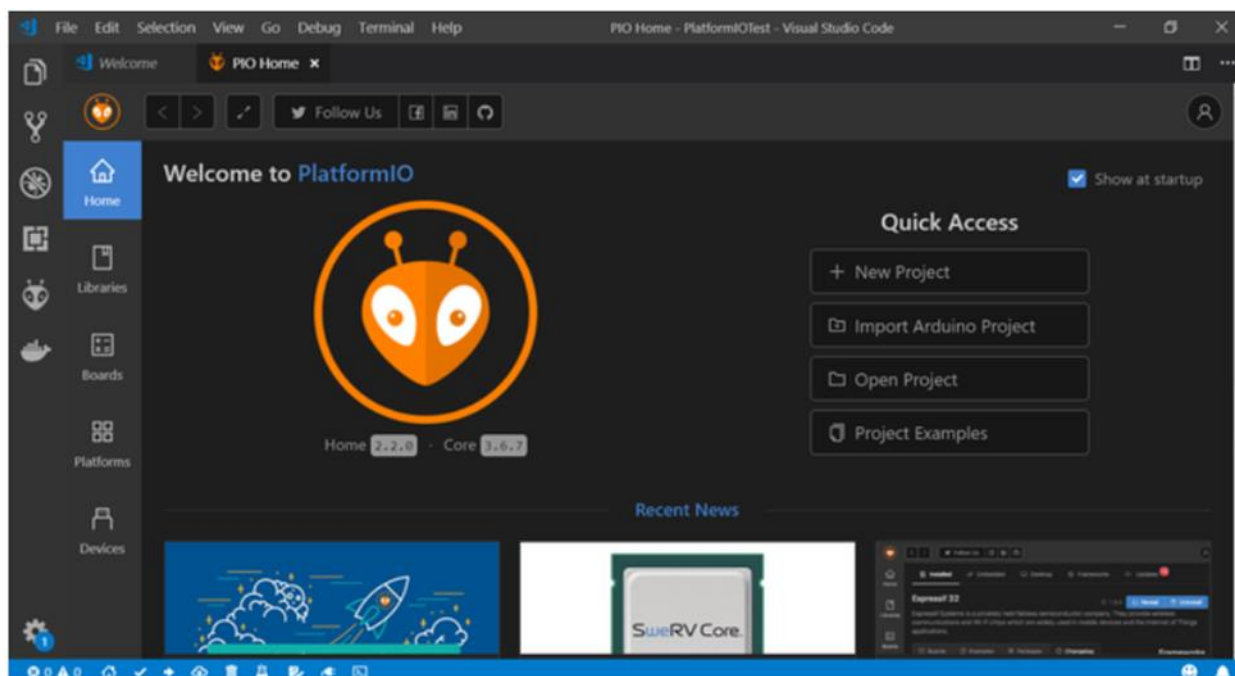
Mở EXTENSIONS từ điều khiển bên trái và tìm kiếm PlatformIO IDE. Nhấp vào cài đặt.



Hình 13 Cài đặt PlatformIO trên VScode

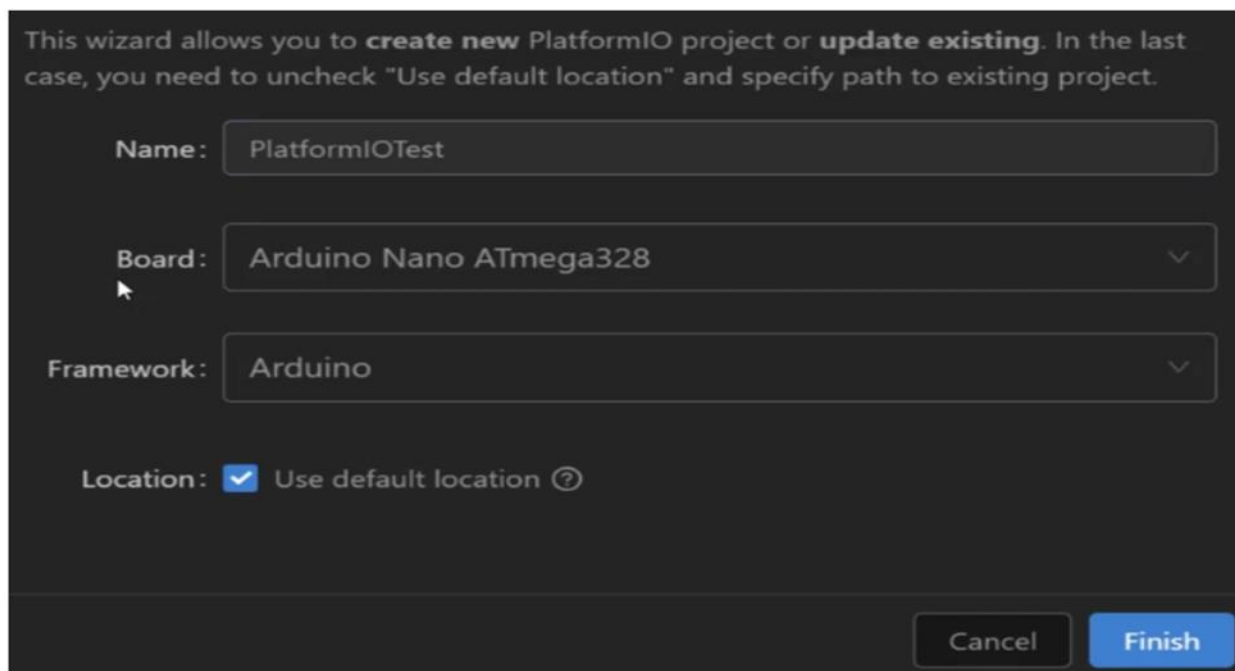
- Tạo Project

Sau khi cài đặt thành công sẽ hiện thị giao diện, nhấn New Project



Hình 14 Tạo Project với PlatformIO trên VScode

Lựa chọn Board, Framework và tên cho Project



Hình 15 Giao diện chọn Board

- Chương trình trống vừa khởi tạo



```

1  #include <Arduino.h>
2
3  void setup() {
4      // put your setup code here, to run once:
5  }
6
7  void loop() {
8      // put your main code here, to run repeatedly:
9  }

```

Hình 16 Chương trình trống khởi tạo



Các biểu tượng mới trên thanh dưới cùng của Visual Studio Code gồm :

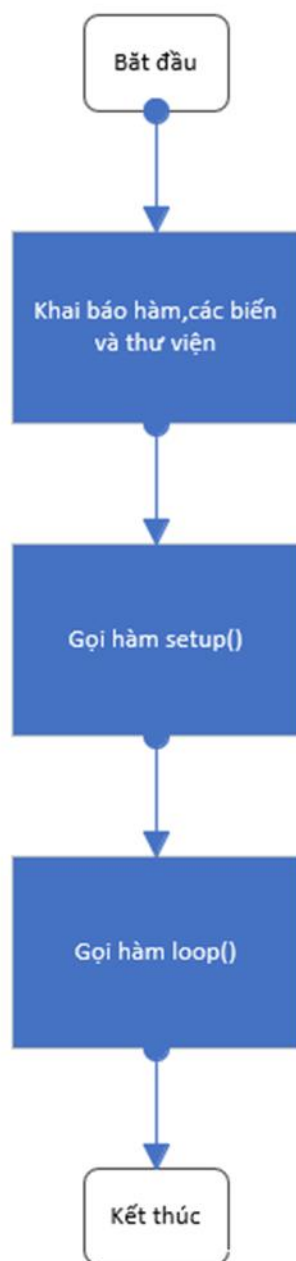
- Verify: kiểm tra code có lỗi hay không
- Upload : nạp code đang soạn thảo vào board
- Clear : clean enironment
- Serial monitor : đọc các thông báo cài đặt trong code

4.5 Lưu đồ giải thuật chi tiết

4.5.1 Lưu đồ của một chương trình trong PlatformIO IDE

Chương trình khởi tạo sẽ gồm :

- `Setup()` : hàm này chạy mỗi khi khởi động chương trình, dùng để thiết lập, đặt các thông số từ đầu.
- `Loop()` : Hàm này được hiểu là vòng lặp cho đến khi không sử dụng nữa hay ngắt nguồn board điều khiển



Hình 17 Lưu đồ của một chương trình trong PlatformIO IDE

4.5.2 Lưu đồ của chương trình vi điều khiển kết nối với wifi trong PlatformIO IDE

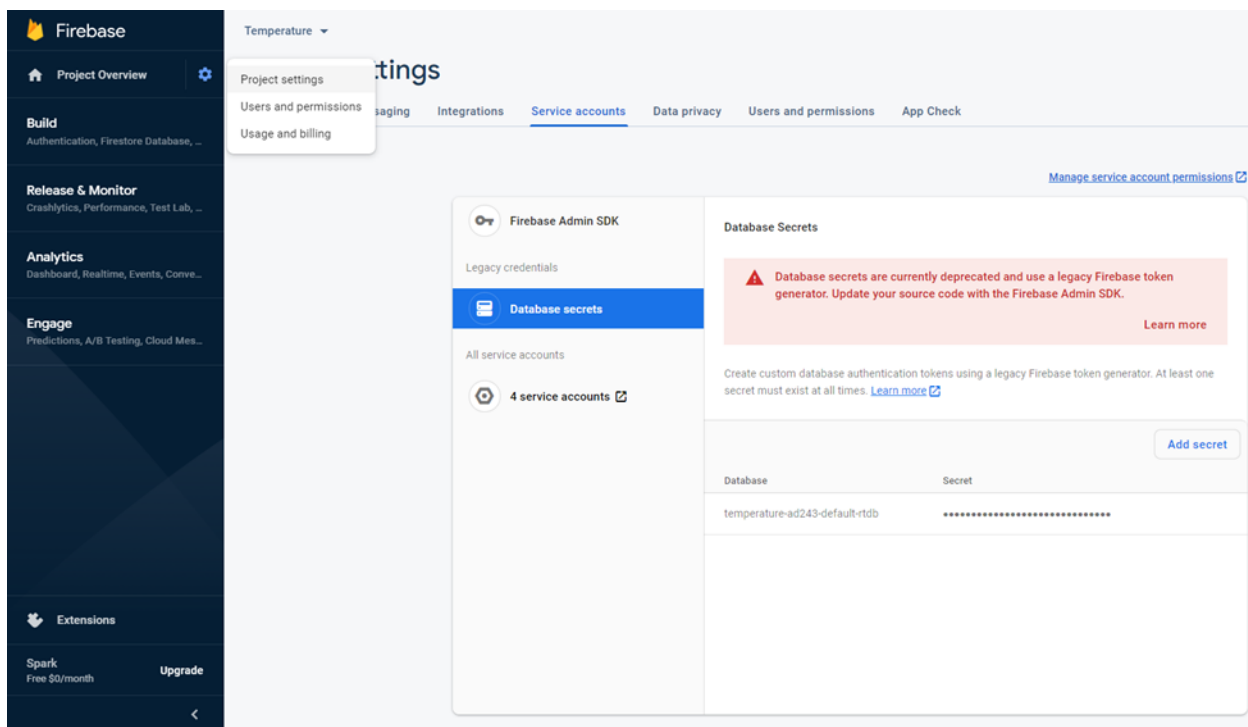
- Sử dụng vi điều khiển NodeMCU ESP8266 để gửi dữ liệu đến firebase thông qua mạng wifi nên cần phải khai báo thư viện liên quan

```
#include <ESP8266WiFi.h>
#include "FirebaseESP8266.h"
```

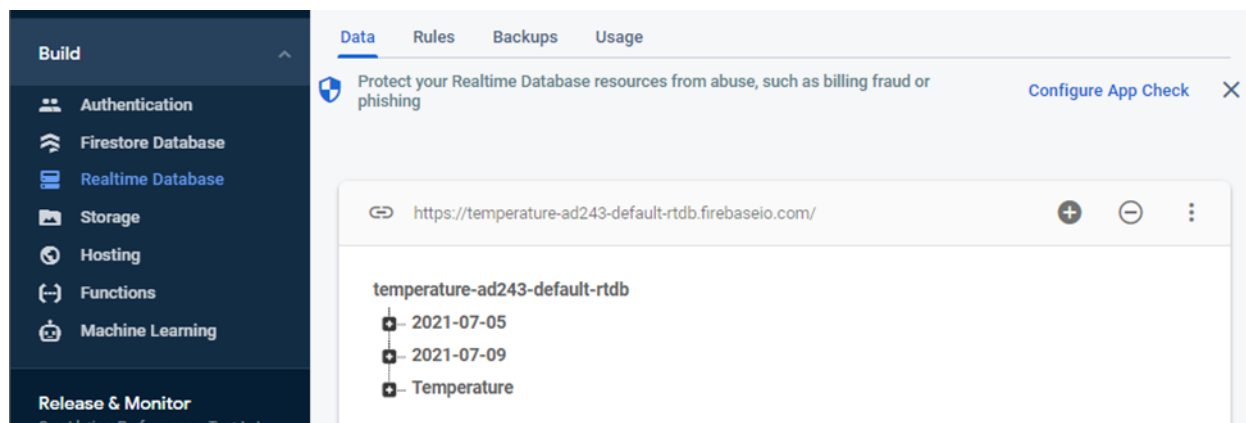
Khai báo URL, database secret của firebase, khai báo SSID và Password của mạng wifi đang dùng.

```
#define FIREBASE_HOST ""
#define FIREBASE_AUTH ""
#define WIFI_SSID ""
#define WIFI_PASSWORD ""
```

- Để lấy FIREBASE_AUTH cho chương trình ESP8266, chúng ta vào Project Setting trong firebase. Kế tiếp ta chọn Service Account -> Database secrets.



- Để lấy URL firebase chúng ta vào Database -> Realtime database



- Thiết lập kết nối wifi và kết nối firebase

```

Serial.begin(9600);
pinMode(LED_BUILTIN, OUTPUT);
digitalWrite(LED_BUILTIN, LOW);

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true);

WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED)
{
  Serial.print(".");
  delay(300);
}

Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();

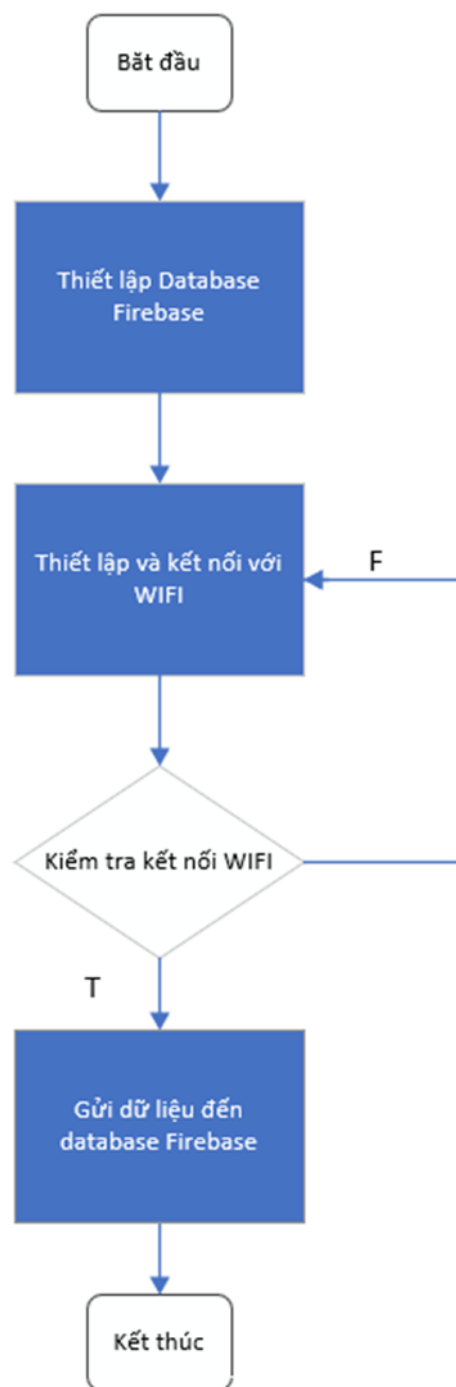
```

- Lưu đồ giải thuật

Giải thích lưu đồ :

+ Bắt đầu, hệ thống thiết lập và kết nối đến wifi chỉ định, thiết lập database firebase.

+ Tiếp theo kiểm tra kết nối Wifi, nếu có kết nối thì sẽ gửi dữ liệu lên database Firebase và ngược lại kết nối Wifi thất bại sẽ liên tục kiểm tra đến khi có thể kết nối đến Wifi.



Hình 18 Lưu đồ của chương trình vi điều khiển kết nối với wifi trong PlatformIO IDE

4.5.3 Lưu đồ thiết lập dữ liệu theo thời gian thực

- Sử dụng vi điều khiển NodeMCU ESP8266 để có thể lấy thời gian thực từ Internet thông qua NTP Sever vì vậy ta cần sử dụng thư viện NTP Client

```
#include<NTPClient.h>
#include<WiFiUdp.h>
```

- Khai báo các biến lưu trữ thời gian

```
String formattedDate;
String formattedTime;
String dayStamp;
String timeStamp;
```

- Khai báo kết nối UDP và NTP Client kết nối với NTP Server

```
const long utcOffsetInSeconds = 3600;
```

```
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", utcOffsetInSeconds);
```

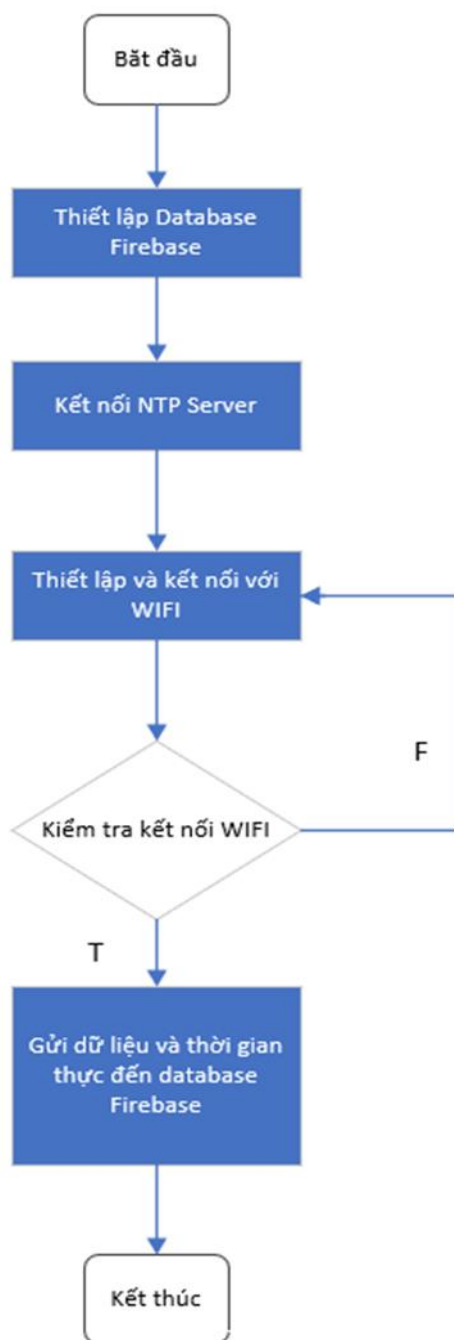
- Thiết lập kết nối

```
Serial.begin(9600);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED)
{
    Serial.print(".");
    delay(300);
}
timeClient.begin();
timeClient.setTimeOffset(+7*60*60);
Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();
}
```

Vì Timezone tại Việt Nam là +7 nên ta có

```
timeClient.setTimeOffset(+7*60*60);
```

- Lưu đồ giải thuật



Hình 19 Lưu đồ thiết lập dữ liệu theo thời gian thực

Giải thích lưu đồ :

+ Bắt đầu, hệ thống thiết lập và kết nối đến wifi chỉ định, thiết lập database firebase và kết nối đến NTP Server.

+ Tiếp theo kiểm tra kết nối Wifi, nếu có kết nối thì sẽ gửi dữ liệu lên database với thời gian thực từ NTP Server và ngược lại kết nối Wifi thất bại sẽ liên tục kiểm tra đến khi có thể kết nối đến Wifi.

4.5.4 Lưu đồ của chương trình vi điều khiển kết nối với MLX90614 PlatformIO IDE

- Sử dụng vi điều khiển NodeMCU ESP8266 kết nối với cảm biến MLX90614 vì vậy chúng ta cần sử dụng thư viện :

```
#include "Adafruit_MLX90614.h"
#include "Wire.h"
```

- Khai báo biến để lưu trữ giá trị được đo từ cảm biến

```
float Object;
float Ambient;
```

- Thiết lập kết nối

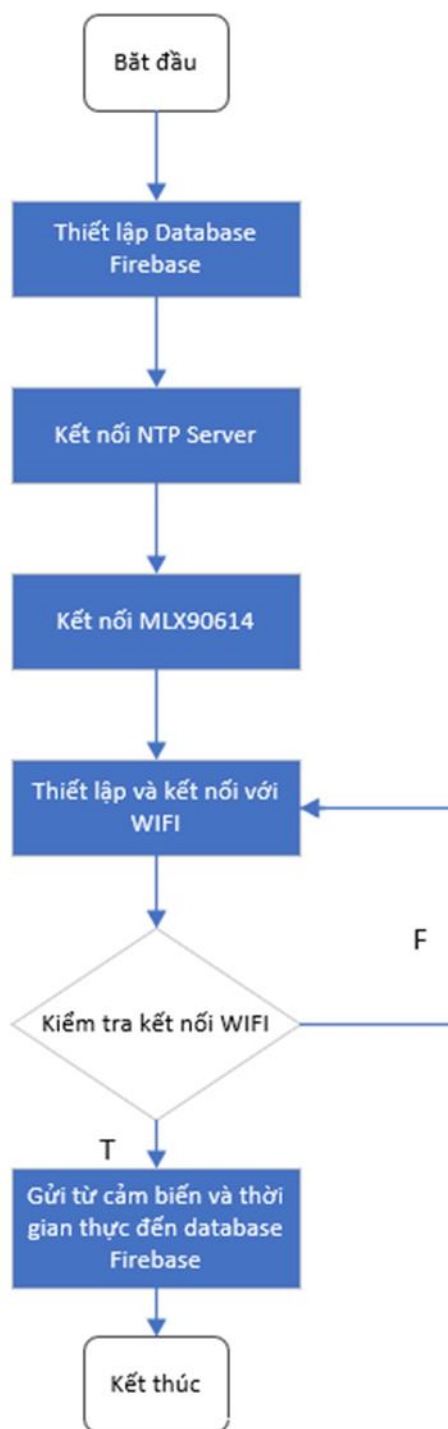
```
Adafruit_MLX90614 mlx = Adafruit_MLX90614();

Serial.begin(9600);
mlx.begin();
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED)
{
    Serial.print(".");
    delay(300);
}
timeClient.begin();
timeClient.setTimeOffset(+7*60*60);

Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true);
```

- Lưu đồ giải thuật



Hình 20 Lưu đồ của chương trình vi điều khiển kết nối với MLX90614 PlatformIO IDE

Giải thích lưu đồ :

- +Bắt đầu, hệ thống thiết lập kết nối đến wifi chỉ định, kết nối đến cảm biến MLX90616, thiết lập database firebase và kết nối đến NTP Server.
- + Tiếp theo kiểm tra kết nối Wifi, nếu có kết nối thì sẽ gửi dữ liệu đo được từ cảm biến lên database với thời gian thực từ NTP Server và ngược lại kết nối Wifi thất bại sẽ liên tục kiểm tra đến khi có thể kết nối đến Wifi.

4.5.5 Lưu đồ của Website kết nối với database Firebase

- Cấu trúc cơ bản của một Website



Hình 21 Cấu trúc cơ bản của một Website

Trong đó :

- + HTML (tiếng Anh viết tắt cho HyperText Markup Language, hay là “Ngôn ngữ đánh dấu văn bản”) là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web với mẫu thông tin được trình bày trên World Wide Web.
- + CSS là chữ viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để tìm và định dạng các phần tử được tạo ra bởi các đoạn văn bản, các tiêu đề, bảng, ... thì CSS sẽ giúp chúng ta có thể “trang trí” vào các phần tử HTML đó như đổi màu sắc trang, đổi màu chữ, thay đổi cấu trúc, ... rất nhiều.

+ JavaScript là một ngôn ngữ lập trình kịch bản dựa vào đối tượng phát triển có sẵn hoặc tự định nghĩa ra, javascript được sử dụng rộng rãi trong các ứng dụng Website, ... thậm chí các trình duyệt trên di động cũng có hỗ trợ này

- Chúng ta khai báo thư viện database Firebase để Website có kết nối

```
<script src="https://www.gstatic.com/firebasejs/5.4.0/firebase.js"></script>
```

- Thiết lập kết nối Website với database Firebase

```
var fb = {
  apiKey: "AIzaSyAE67jiSi8HUegkT_xld_f3Hzm9ulj3koU",
  authDomain: "projectId.firebaseio.com",
  databaseURL: "https://temperature-ad243-default-rtdb.firebaseio.com/",
  projectId: "temperature-ad243",
  storageBucket: "bucket.appspot.com",
  messagingSenderId: "messagingSenderId"
};

firebase.initializeApp(fb);
```

- Lưu đồ giải thuật



Hình 22 Lưu đồ của Website kết nối với database Firebase

Giải thích lưu đồ:

- + Bắt đầu, chúng ta khai báo thư viện, thiết lập thông số để Website để có thể kết nối đến database
- + Đọc dữ liệu từ database, hiện thị các dữ liệu ra giao diện người dùng

5. Kết quả thực hiện

5.1 Cách thử đo đạc, thử nghiệm

- Thiết bị sử dụng và sơ đồ kết nối trong thử nghiệm

Thiết bị sử dụng : Node MCU esp8266, MLX90614, Củ sạc 5v

- Phần mềm sử dụng trong việc viết và thực thi chương trình

Tất cả các phần mềm sử dụng trong việc viết và thực thi chương trình đã được nêu ở mục 4.4

- Các bước tiến hành thực nghiệm

- o Thực hiện phần cứng

Kết nối chân theo sơ đồ mạch lần lượt :

+ VIN -> 3.3v

+ GND -> GND

+ SCL -> D1

+ SDA -> D2

- o Thực hiện phần mềm :

Nạp code đã thực hiện vào Node MCU esp8266 bằng cổng nạp USB MICRO B

- o Thực hiện đo đạc

+ Đầu tiên nhóm thực hiện đo nhiệt độ với từng khoảng cách đến vật thể là 2 cm, 4 cm, 6 cm

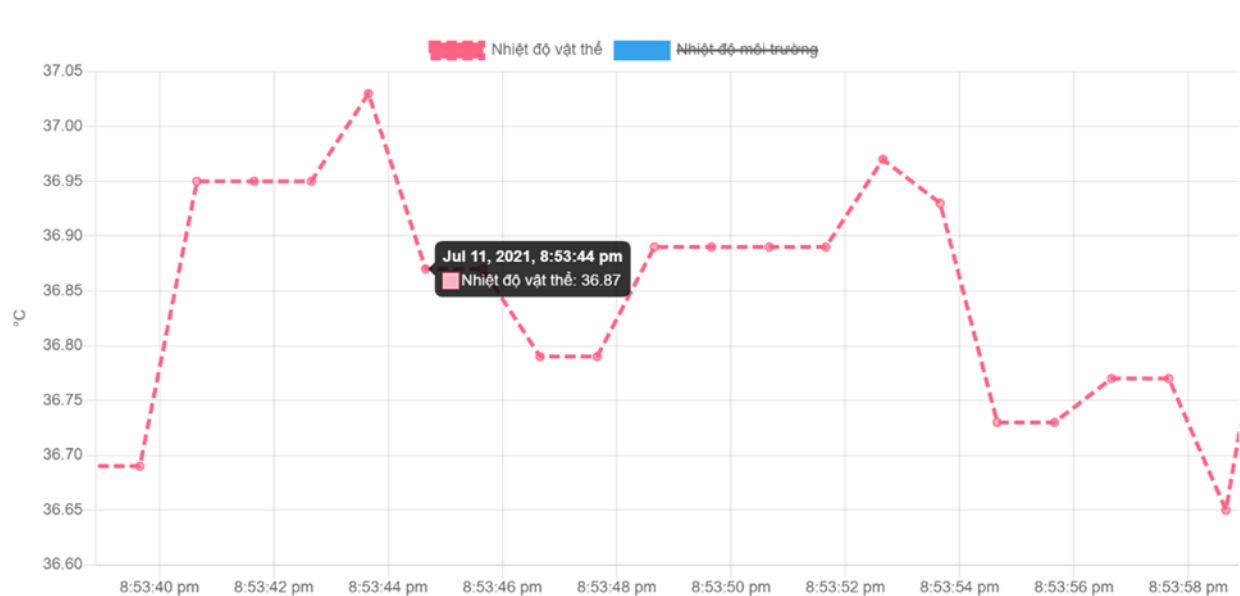
+ Đo liên tục trong vòng 10s

5.2 Thu thập số liệu

- Các trường hợp thực nghiệm

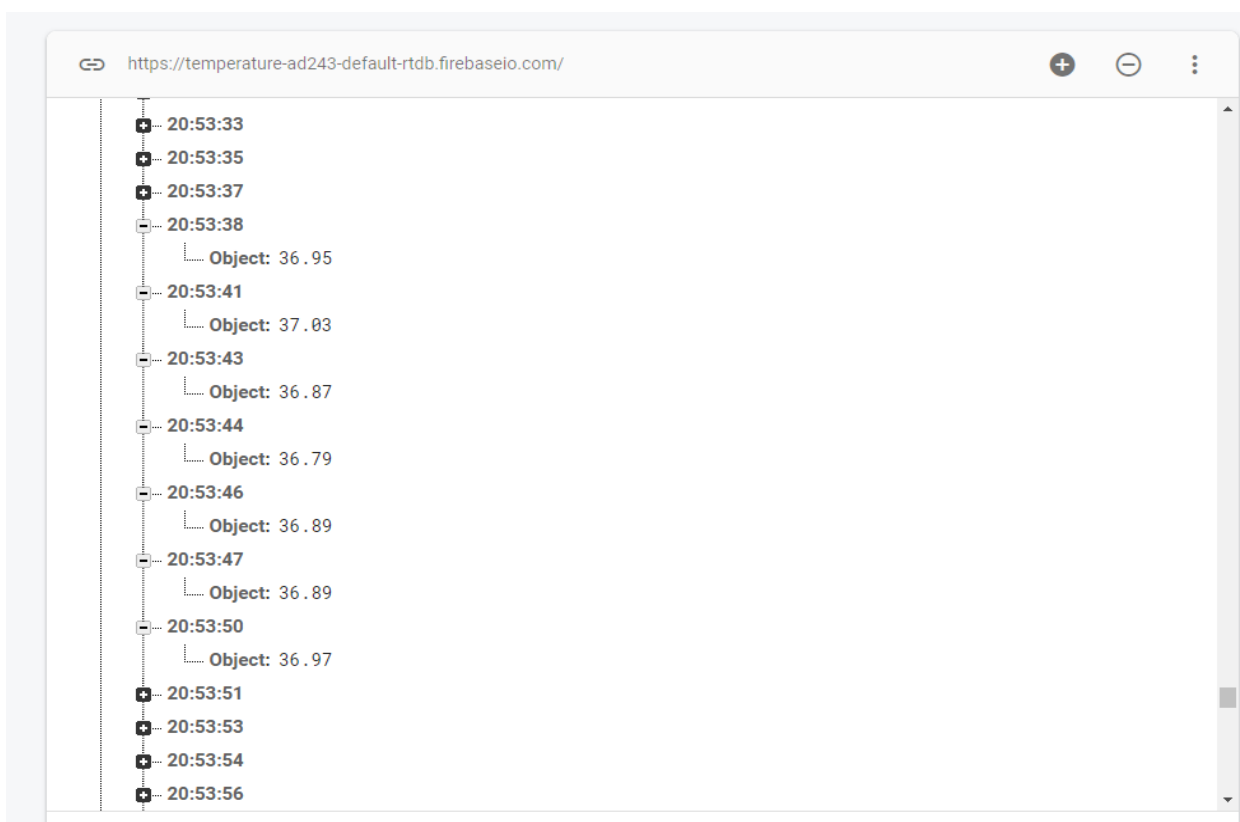
- o Với khoảng cách đến vật thể 2 cm

Biểu đồ đường:



Hình 23 Biểu đồ đường số liệu đo được ở khoảng cách 2 cm

Các điểm dữ liệu trên database thời điểm từ 20:53:38 đến 20:53:50

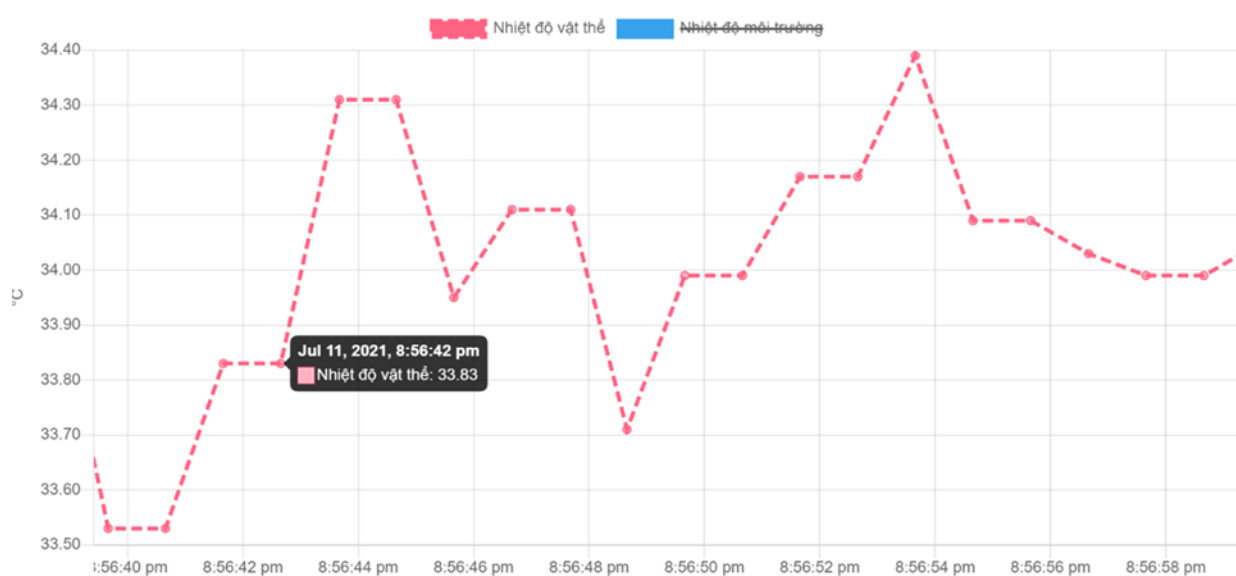


Hình 24 Lưu trữ nhiệt độ 2cm trên Firebase

Thời điểm	Nhiệt độ
20:53:38	36.95 °C
20:53:41	37.03 °C
20:53:43	36.87 °C
20:53:44	36.79 °C
20:53:46	36.89 °C
20:53:47	36.89 °C
20:53:50	36.97 °C

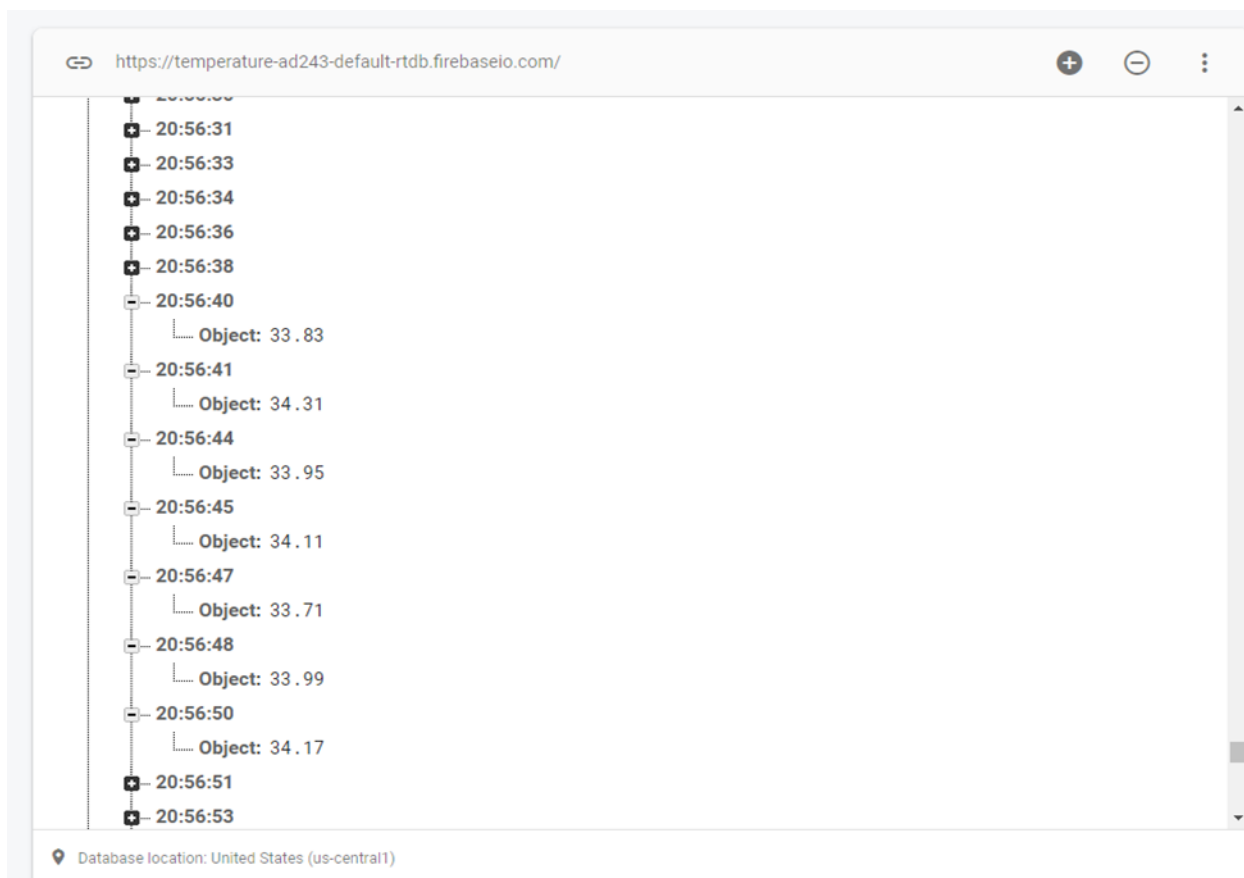
o Với khoảng cách đến vật thể 4 cm

Biểu đồ đường:



Hình 25 Biểu đồ đường số liệu đo được ở khoảng cách 4 cm

Các điểm dữ liệu trên database thời điểm từ 20:56:40 đến 20:56:50

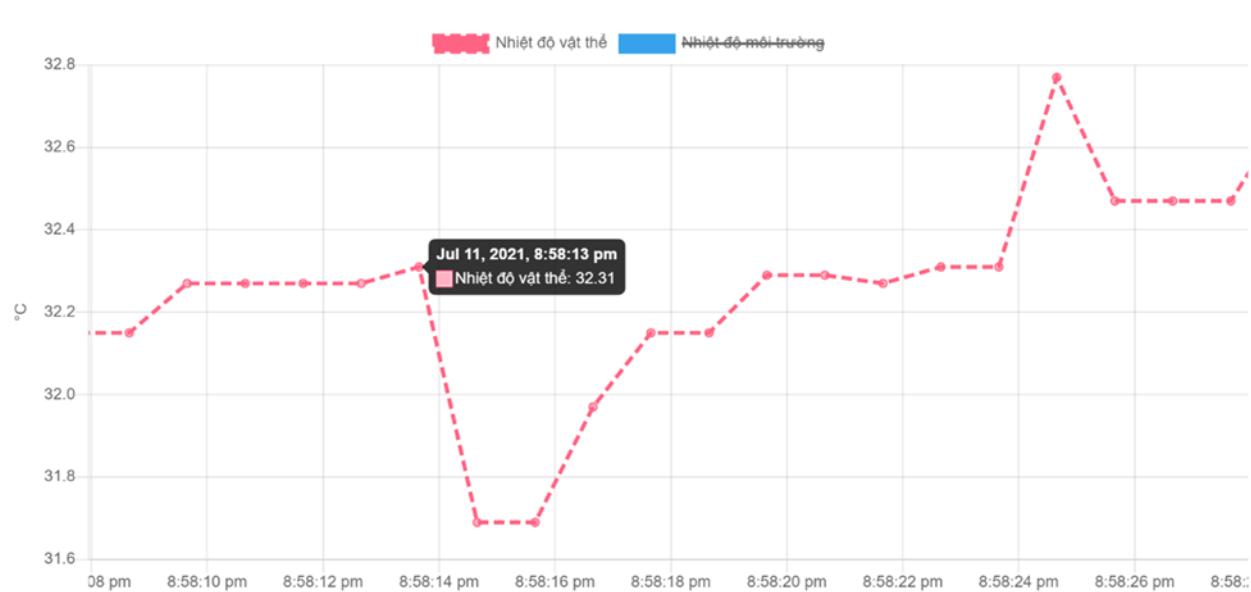


Hình 26 Lưu trữ nhiệt độ 4cm trên Firebase

Thời điểm	Nhiệt độ
20:56:40	33.83 °C
20:56:41	34.31 °C
20:56:44	33.95 °C
20:56:45	34.11 °C
20:56:47	33.71 °C
20:56:48	33.99 °C
20:56:50	34.17 °C

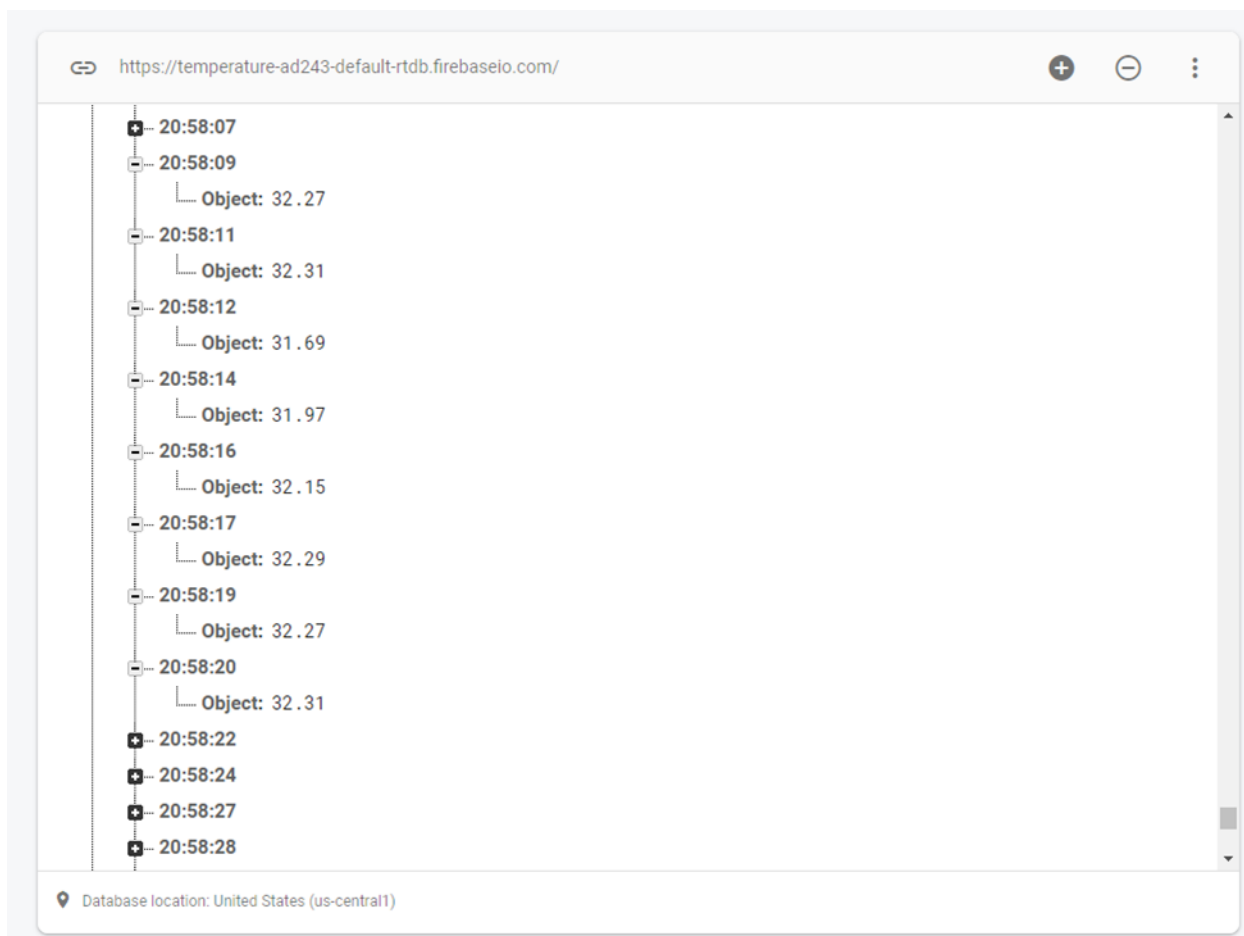
o Với khoảng cách đến vật thể là 6 cm

Biểu đồ đường:



Hình 27 Biểu đồ đường số liệu đo được ở khoảng cách 6 cm

Các điểm dữ liệu trên database thời điểm từ 20:58:09 đến 20:58:20



Hình 28 Lưu trữ nhiệt độ 6m trên Firebase

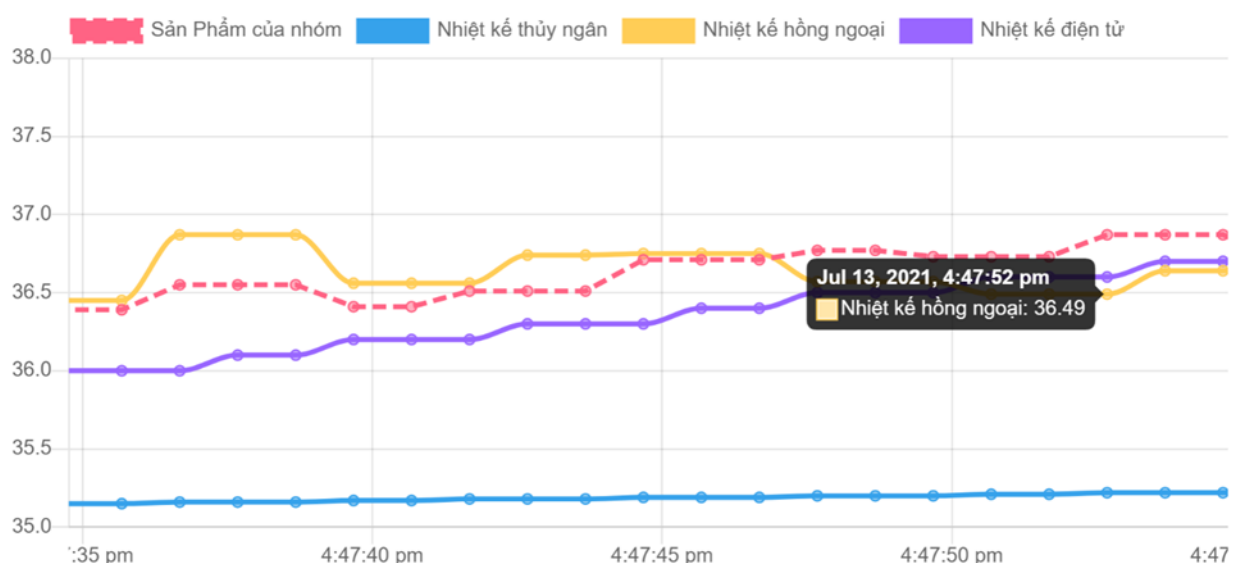
Thời điểm	Nhiệt độ
20:58:09	32.27 °C
20:58:11	32.31 °C
20:58:12	31.69 °C
20:58:14	31.97 °C
20:58:16	32.15 °C
20:58:17	32.29 °C
20:58:19	32.27 °C
22:58:20	32.31 °C

5.3 Giải thích và phân tích về kết quả thu được

+ Kết quả thu được khi đo ở khoảng cách là 2 cm có độ chính xác cao nhất, khi khoảng cách đo càng tăng thì giá trị sai số càng cao khiến cho kết quả thu được không còn chính xác

+ Vì vậy khoảng cách 2 cm là tối ưu để thực hiện việc đo đạc.

So sánh kết quả với các công cụ đo truyền thống khác:



Hình 29 Biểu đồ nhiệt độ đo được của các phương pháp đo khác nhau tại thời điểm xác định

Biểu đồ so sánh kết quả

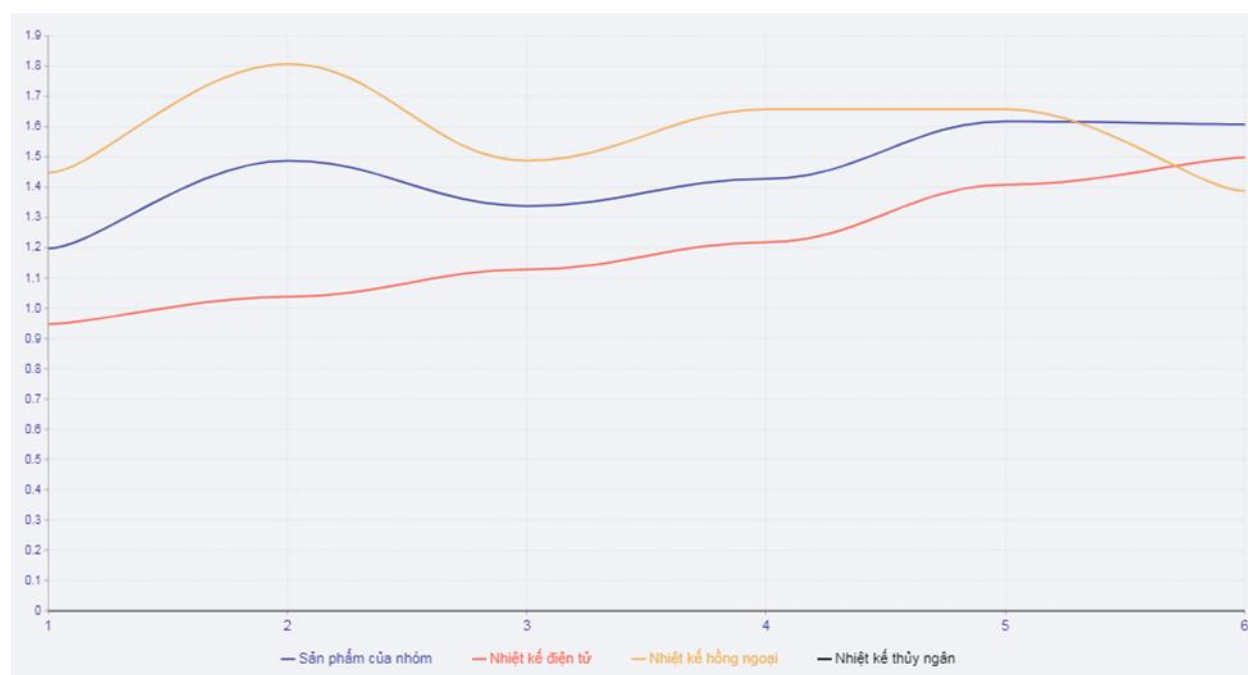
+ Ta thấy được giá trị của sản phẩm và nhiệt kế hồng ngoại khá là tương đồng, còn nhiệt kế điện tử thì tăng dần theo thời gian, nhiệt kế thủy ngân gần như không thay đổi

Thời điểm	Sản phẩm của nhóm	Nhiệt kế thủy ngân	Nhiệt kế hồng ngoại	Nhiệt kế điện tử
1	36.25	35.05	36.95	36
2	36.55	35.06	36.87	36.1
3	36.41	35.07	36.56	36.2
4	36.51	35.08	36.74	36.3
5	36.71	35.09	36.75	36.5
6	36.71	35.10	36.49	36.6

Xét sai số của nhiệt kế thủy ngân = 0, ta có biểu đồ ta có bảng sai số:

Thời điểm	Sản phẩm của nhóm	Nhiệt kế hồng ngoại	Nhiệt kế điện tử
1	1.20	1.45	0.95
2	1.49	1.81	1.04
3	1.34	1.49	1.13
4	1.43	1.66	1.22
5	1.62	1.66	1.41
6	1.61	1.39	1.5

Biểu đồ đường sai số của phương pháp đo của mô hình này và các công cụ đo khác:



Hình 30 Biểu đồ sai số của các phương pháp đo khi lấy phương pháp đo bằng thủy ngân làm chuẩn

6. Kết luận và hướng phát triển

6.1 Các kết quả đã thực hiện:

Về tổng quát, nhóm đã hoàn thiện mô hình đo đạt nhiệt độ bên ngoài tòa nhà và truyền tải dữ liệu lên Server để con người có thể xem xét và quản lý thông qua kết nối mạng.

Nhóm đã thực hiện phép đo nhiều lần và nhiều đối tượng, trường hợp.

6.2 So sánh với mục tiêu đặt ra trong chương 1

Kết luận: Hệ thống thực thi được cơ bản các chức năng mà nhóm đã đặt ra lúc đầu. Đo đạt được cơ bản các đối tượng, môi trường, truyền tốt được dữ liệu thông qua mạng wifi lên server với độ trễ thấp, thời gian đáp ứng đo nhiệt độ liên tục mỗi lần 2 giây. Người quản lý có thể truy cập vào Server và Firebase để tra cứu thông tin về dữ liệu ở hiện tại và quá khứ, có thể phân tích dữ liệu đo được để vạch ra những hướng đi đúng đắn cho công việc.

So sánh với đo nhiệt độ truyền thống:

Phép đo nhiệt độ trong mô hình này tuy có sai số so với các phương pháp đo nhiệt độ truyền thống như là thủy ngân, nhiệt kế điện tử, tuy nhiên ưu điểm của phương pháp này là đo được nhiệt độ đối tượng nhanh hơn các phương pháp đó. Điều này cũng đã đáp ứng một phần trong yêu cầu về độ chính xác này.

6.3 Hướng phát triển của đề tài:

Nhìn chung về mô hình mà nhóm đã thực hiện thì về cơ bản nhóm đã bước đầu hoàn thiện được một mô hình IOT đơn giản nhưng khá là hữu dụng trong cuộc sống.

Với sự kết hợp đa dạng của các khía cạnh về công nghệ đối với một hệ thống IOT thì với mô hình hiện tại, chúng ta có thể phát triển đề tài này thêm nhiều bước tiến mới hơn nữa.

- Ta có thể ứng dụng mô hình này vào một mô hình IOT rộng hơn, bao quát hơn như là Smart home, Smart City và cả Smart Health để phục vụ cho việc kiểm soát nhiệt độ tài nhà, tại thành phố và các trung tâm y tế, bệnh viện
- Để có thể tiết kiệm điện năng, thông minh hơn, ta có thể ứng dụng Machine Learning vào công nghệ nhận diện người, ứng dụng hoạt động, thực hiện việc đo đạt với mục tiêu cụ thể nhất định ví dụ như con người chẳng hạn. Điều này sẽ giúp cho dữ liệu thu vào nhất quán về đối tượng, dễ dàng quản lý hơn rất nhiều.

7. Tài liệu tham khảo

- [1] "Firebase," [Online]. Available: <https://firebase.google.com/docs>. [Accessed 12 05 2021].

- [2] TheElectromania, 2007. [Online]. Available:
<https://www.instructables.com/Programming-ESP8266-ESP-12E-NodeMCU-Using-Arduino-/>.
- [3] M. R. Thakur, NodeMCU ESP8266 Communication Methods and Protocols :
Programming with Arduino IDE, Amazon Media EU S.à r.l, 2018.