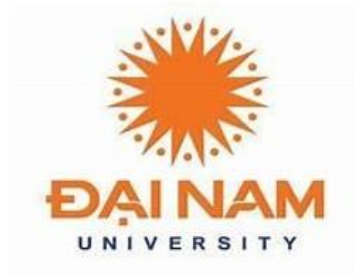


**TRƯỜNG ĐẠI HỌC ĐẠI NAM
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN TỐT NGHIỆP
XÂY DỰNG THIẾT BỊ GIÁM SÁT BẢO
ĐẢM AN TOÀN CHO GIA ĐÌNH DỰA
TRÊN KỸ THUẬT IOT

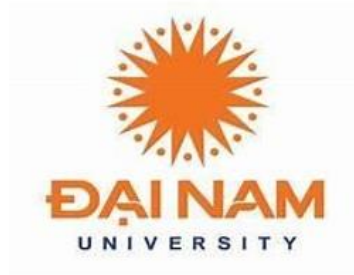
SINH VIÊN THỰC HIỆN : LÊ THỊ HỒNG TRANG

MÃ SINH VIÊN : 1357010065

KHOA : CÔNG NGHỆ THÔNG TIN

Hà Nội – 2024

**TRƯỜNG ĐẠI HỌC ĐẠI NAM
KHOA CÔNG NGHỆ THÔNG TIN**



LÊ THỊ HỒNG TRANG

**XÂY DỰNG THIẾT BỊ GIÁM SÁT BẢO
ĐẢM AN TOÀN CHO GIA ĐÌNH DỰA
TRÊN KỸ THUẬT IOT**

CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN

MÃ SỐ : 74.80.201

NGƯỜI HƯỚNG DẪN: TS. TRẦN ĐĂNG CÔNG

Hà Nội – 2024

NHẬN XÉT

(Của giảng viên hướng dẫn)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ký và ghi họ tên

LỜI CAM ĐOAN

Tôi xin cam đoan rằng báo cáo có tên: “Xây dựng thiết bị giám sát bảo đảm an toàn cho gia đình dựa trên kỹ thuật IoT” là kết quả của công việc nghiêm túc và tự chủ của bản thân. Trong quá trình nghiên cứu và phát triển dự án này, tôi đã tuân thủ các nguyên tắc và chuẩn mực đạo đức nghiêm ngặt.

Mọi thông tin, số liệu và kết quả được trình bày trong báo cáo đều tuân thủ nguyên tắc trung thực và minh bạch nhất có thể. Tôi đã sử dụng tài liệu tham khảo và nguồn thông tin từ các nguồn uy tín, và những phần này đã được dẫn chứng rõ ràng và nêu rõ trong phần tài liệu tham khảo.

Đồng thời, tôi cam kết rằng nếu có bất kỳ sai sót nào trong thông tin trình bày, tôi sẽ chịu trách nhiệm hoàn toàn và tuân theo mọi quy định và kỷ luật do bộ môn và nhà trường áp đặt.

Tôi cũng xác nhận rằng quy trình kiểm tra, xác minh thông tin và kiểm soát chất lượng đã được thực hiện để đảm bảo tính chính xác và đáng tin cậy của mọi thông tin được trình bày trong báo cáo này.

Hà Nội, ngày 24 tháng 4 năm 2024

LỜI CẢM ƠN

Tôi xin chân thành cảm ơn các thầy cô trong khoa CNTT cũng như các thầy cô giảng dạy trong trường đã giúp tôi có được những kiến thức tổng quan để hoàn thành bài báo cáo này. Đặc biệt, tôi muốn gửi lời cảm ơn sâu sắc nhất đến GVHD TS.Trần Đăng công và thầy Nguyễn Văn Nhân là những người đã đồng hành và hướng dẫn tôi suốt quãng thời làm đồ án tốt nghiệp của bản thân. Trong quá trình này, sự tận tâm và sự quan tâm từ các thầy đã là nguồn động viên lớn, giúp tôi tiếp cận kiến thức một cách sâu sắc và tự tin hơn khi tiếp cận dự án thực tế của mình.

Tôi rất mong nhận được sự đánh giá và góp ý chân thành từ thầy để bài báo cáo của tôi có thể được cải thiện và hoàn thiện hơn. Lời hướng dẫn và ý kiến từ thầy sẽ giúp tôi nắm bắt rõ hơn những khía cạnh cần điều chỉnh, từ đó phát triển kỹ năng và hiểu biết của mình một cách toàn diện hơn.

Cuối cùng, tôi xin gửi lời cảm ơn đến tất cả những người đã đóng góp ý kiến, phản hồi và ý tưởng quan trọng. Sự sáng tạo và đóng góp của các bạn đã làm phong phú và nâng cao chất lượng của dự án.

Bài báo cáo của tôi có thể chưa tránh khỏi những sai sót do những hạn chế trong quá trình thực hiện. Mong rằng thầy và quý độc giả có thể xem xét và góp ý để giúp tôi hoàn thiện bài báo cáo một cách tốt nhất. Sự chỉ dẫn và phản hồi từ thầy sẽ giúp tôi cải thiện kiến thức và kỹ năng một cách toàn diện hơn.

Tôi xin chân thành cảm ơn!

DANH MỤC KÝ HIỆU HOẶC CHỮ VIẾT TẮT

STT	Chữ viết tắt	Chữ viết đầy đủ (tiếng Anh)	Chữ viết đầy đủ (tiếng Việt)
1	ADC0	Analog to Digital Converter 0	Chuyển đổi từ tín hiệu analog sang digital
2	BMP180	Barometric Pressure Monitor	Máy đo áp suất khí quyển
3	CNTT		Công nghệ thông tin
4	DHT11	Digital Humidity and Temperature seri 11	Cảm biến kỹ thuật số đo độ ẩm và nhiệt độ mã số 11
5	GND	Ground	Điểm đất(chân âm)
6	GPIO	General Purpose Input/Output	Đầu vào hoặc đầu ra trong một vi điều khiển
7	GPS	Global Positioning System	Hệ thống định vị toàn cầu
8	GVHD		Giảng viên hướng dẫn
9	HTTP	HyperText Transfer Protocol	Giao thức truyền tải siêu văn bản
10	IDE	Integrated Development Environment	Môi trường phát triển tích hợp
11	IEEE	Institute of Electrical and Electronics Engineers	Viện kỹ sư điện và điện tử
12	IoT	Internet of thing	Vạn vật kết nối Internet
13	IP	Internet Protocol	Giao thức Internet
14	IPv6	Internet Protocol version 6	Giao thức Internet phiên bản 6
15	MQTT	Message Queuing Telemetry Transport	Giao thức vận chuyển telemetry thông qua hàng đợi tin nhắn
16	PCB	Printed Circuit Board	Bảng mạch in

17	PIC	Peripheral Interface Controller	Vi điều khiển
18	SCL	Serial Clock Data	Tín hiệu đồng hồ
19	SDA	Serial Data	Tín hiệu dữ liệu
20	SDK	Software Development Kit	Bộ kit phát triển phần mềm
21	TCP	Transmission Control Protocol	Giao thức kiểm soát truyền
22	UDP	User Datagram Protocol	Giao thức datagram người dùng
23	VCC	Voltage at the common collector	Điện áp tại bộ thu chung(chân dương)
24	WiFi	Wireless Fidelity	Kết nối không dây

DANH MỤC HÌNH ẢNH

Hình 2.1: Hình ảnh minh hoạ cảm biến lửa.....	10
Hình 2.2: Hình ảnh minh hoạ cảm biến nhiệt độ độ ẩm DHT11.....	11
Hình 2.3: Hình ảnh minh hoạ cảm biến áp suất BMP180.....	12
Hình 2.4: Hình ảnh minh hoạ cảm ESP8266 NodeMCU.....	13
Hình 2.5: Hình ảnh minh hoạ phần mềm Arduino IDE.....	14
Hình 2.6: Mối quan hệ giữa các dự án, ứng dụng và sản phẩm Firebase.....	16
Hình 2.7: Hình ảnh minh hoạ các tầng kiến trúc android.....	18
Hình 3.1.1: Mô hình tổng quan hệ thống cảm biến giám sát ngôi nhà	26
Hình 3.1.2: Sơ đồ lắp đặt kết nối các thiết bị	28
Hình 3.1.3: Hình ảnh chi tiết cấu hình Esp8266 NodeMCU.....	29
Hình 3.1.4: Hình ảnh thiết bị thực tế	29
Hình 3.1.5: Sơ đồ luồng hoạt động từ thiết bị đến ứng dụng	34
Hình 3.2.1: Hình ảnh giao diện đăng nhập ứng dụng.....	36
Hình 3.2.2: Hình ảnh giao diện màn hình chính ứng dụng	37
Hình 3.2.3: Hình ảnh giao diện màn hình thông báo ứng dụng	39
Hình 3.2.4: Hình ảnh đọc dữ liệu từ môi trường ra Serial Arduino IDE.....	42
Hình 3.2.5: Hình ảnh lưu trữ dữ liệu trên firebase	43

MỤC LỤC

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT.....	1
1.1. Giới thiệu về IoT	1
1.1.1. Khái niệm về IoT	1
1.1.2. Lịch sử phát triển của IoT	1
1.2. Nguyên lý hoạt động của IoT	2
1.3. Ứng dụng của IoT trong đời sống	3
1.4. Ưu điểm và thách thức trong việc ứng dụng IoT	4
1.4.1. Ưu điểm của IoT.....	4
1.4.2. Thách thức trong việc nghiên cứu và triển khai	5
1.5. Hệ thống giám sát đảm bảo an toàn cho gia đình.....	6
Tiểu kết chương 1:	7
CHƯƠNG 2: CÁC THÀNH PHẦN CHÍNH CỦA HỆ THỐNG CẢM BIẾN GIÁM SÁT NGÔI NHÀ.....	8
2.1. Thiết bị cảm biến	8
2.1.1. Định nghĩa	8
2.1.2. Vai trò của cảm biến trong hệ thống IoT	8
2.2. Các loại thiết bị cảm biến được sử dụng	9
2.2.1. Module flame sensor	9
2.2.2. Module DHT11	10
2.2.3. Module BMP180	11
2.3. Bộ xử lý và truyền thông	13
2.3.1. Vi điều khiển ESP8266 NodeMCU	13
2.3.2. Các thiết bị xử lý trung gian trong hệ thống IoT	14

2.3.3. Chuẩn giao tiếp không dây WiFi	15
2.4. Giới thiệu phần mềm nạp code và vẽ sơ đồ	15
2.4.1. Phần mềm Arduino IDE	15
2.4.2. Phần mềm Proteus	16
2.5. Công nghệ lưu trữ đám mây - Firebase	17
2.5.1. Tổng quan về Firebase.....	17
2.5.2. Mối quan hệ giữa các dự án, ứng dụng và Firebase	18
2.6. Nền tảng sử dụng cho hệ thống IoT	19
2.6.1. Giới thiệu nền tảng Android	19
2.6.2. Kiến trúc Android	21
2.6.3. Cơ chế giao tiếp giữa thiết bị IoT và ứng dụng	23
Tiểu kết chương 2:	25
CHƯƠNG 3: THIẾT KẾ VÀ XÂY DỰNG MÔ HÌNH	26
3.1. Xây dựng hệ thống cảm biến giám sát ngôi nhà	26
3.1.1. Yêu cầu bài toán	26
3.1.2. Lắp đặt phần cứng	27
3.1.3. Cách thức hoạt động của từng thành phần trong hệ thống	30
3.1.4. Luồng hoạt động từ cảm biến đến ứng dụng	34
3.2. Thiết kế phần mềm	36
3.2.1. Lập trình ứng dụng người dùng.....	36
3.2.2. Lập trình đọc dữ liệu từ cảm biến	39
3.3. Hướng dẫn sử dụng cho người dùng	43
Tiểu kết chương 3:	45
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ.....	46

4.1. Kết quả thực hiện.....	46
4.2. Đánh giá.....	47
Tiểu kết chương 4:	47
KẾT LUẬN	48
TÀI LIỆU THAM KHẢO	49

LỜI MỞ ĐẦU

1. Tính cấp thiết của đề tài:

Trong thời đại công nghệ ngày nay, việc xây dựng thiết bị giám sát bảo đảm an toàn cho gia đình dựa trên kỹ thuật IoT trở nên cực kỳ cấp thiết và phù hợp với nhu cầu của xã hội hiện đại. Sự gia tăng của cuộc sống nhanh chóng, nhu cầu về an ninh và sự bảo vệ gia đình cũng tăng lên theo đó. Với sự phát triển của Internet of Things (IoT), việc áp dụng công nghệ này vào việc giám sát và bảo vệ gia đình trở nên linh hoạt và hiệu quả hơn bao giờ hết. Điều này giúp cho việc quản lý và kiểm soát an ninh trong gia đình trở nên thuận tiện hơn, đồng thời cung cấp sự yên tâm tối đa cho người sử dụng.

Gia tăng về an ninh và an toàn: Trong một thế giới ngày càng phức tạp và không đoán trước được, nhu cầu về an ninh và an toàn cho gia đình trở nên ngày càng quan trọng. Việc sở hữu một hệ thống giám sát thông minh dựa trên IoT không chỉ giúp người dùng theo dõi và bảo vệ tài sản của mình mà còn giúp giảm thiểu rủi ro về an ninh và tội phạm.

Sự tiện ích và linh hoạt: Thiết bị giám sát IoT mang lại sự tiện ích và linh hoạt cho người dùng. Dựa trên kết nối internet, người dùng có thể theo dõi và kiểm soát từ xa qua điện thoại di động hoặc máy tính bảng mọi lúc mọi nơi mà không cần phải ở gần nhà.

Tính đa năng và đồng bộ: Khả năng tích hợp với các thiết bị khác trong gia đình như cảm biến khói, cảm biến CO, cảm biến nhiệt độ và các thiết bị thông minh khác giúp tăng tính đa năng và đồng bộ của hệ thống. Điều này cung cấp một góc nhìn toàn diện về an ninh và an toàn cho gia đình.

Tiết kiệm và hiệu quả: Mặc dù sở hữu một hệ thống giám sát IoT có thể đòi hỏi một khoản đầu tư ban đầu, nhưng nó có thể tiết kiệm chi phí và thời gian trong việc duy trì và quản lý so với các hệ thống truyền thống. Hơn nữa, việc sử dụng các thiết bị thông minh có thể giúp tiết kiệm năng lượng và tài nguyên.

Sự phát triển của công nghệ: Với sự phát triển không ngừng của công nghệ IoT, việc xây dựng các thiết bị giám sát thông minh trở nên dễ dàng hơn và có nhiều tính năng hơn. Sự tiến bộ trong cảm biến, mạng lưới và phần mềm đảm bảo rằng hệ thống giám sát IoT sẽ ngày càng trở nên thông minh và hiệu quả hơn theo thời gian.

Tổng quan, việc chọn chủ đề “Xây dựng thiết bị giám sát bảo đảm an toàn cho gia đình dựa trên kỹ thuật IoT” không chỉ đáp ứng nhu cầu an ninh và an toàn ngày càng tăng của cộng đồng mà còn thúc đẩy sự tiến bộ và phát triển trong lĩnh vực công nghệ thông tin và truyền thông.

2. Mục đích nghiên cứu:

Mục đích của nghiên cứu là phát triển và triển khai một hệ thống giám sát thông minh dựa trên IoT để bảo đảm an toàn cho gia đình. Hệ thống này sẽ kết hợp các cảm biến thông minh, thiết bị kết nối internet và phần mềm điều khiển để cung cấp các tính năng như giám sát an ninh, báo động, cảnh báo xâm nhập trái phép,...

Mục tiêu là tạo ra một giải pháp toàn diện và linh hoạt, giúp người dùng dễ dàng quản lý và kiểm soát an ninh và an toàn cho gia đình của mình.

3. Phạm vi nghiên cứu:

Phạm vi của nghiên cứu sẽ tập trung vào việc phát triển các thiết bị cảm biến thông minh, phần mềm điều khiển và hệ thống kết nối dựa trên IoT. Nghiên cứu cũng sẽ xem xét về tính tiện ích, hiệu suất và tính bảo mật của hệ thống. Ngoài ra, phạm vi cũng bao gồm việc thử nghiệm và tối ưu hóa hệ thống trước khi triển khai thực tế cho các gia đình.

4. Phương pháp nghiên cứu:

Trong phần này, tôi xác định mục tiêu nghiên cứu của mình là xây dựng và triển khai một thiết bị giám sát bảo đảm an toàn cho gia đình dựa trên kỹ thuật IoT, nhằm mục đích giám sát và cải thiện mức độ an toàn cho các thành viên trong gia đình.

Trong đề tài này, tôi đã sử dụng các phương pháp nghiên cứu:

- Phương pháp tham khảo tài liệu: Bằng cách thu thập thông tin từ sách, tạp chí về điện tử và truy cập từ mạng Internet.
- Phương pháp quan sát: Khảo sát một số mạch điện thực tế đang có trên thị trường và tham khảo thêm một số dạng mạch từ mạng Internet.
- Phương pháp thực nghiệm: Từ những ý tưởng kiên thức vốn có của mình kết hợp với sự hướng dẫn từ giảng viên, các trang điện tử trên Youtube, Internet, tôi đã lắp ráp thử nghiệm nhiều dạng mạch khác nhau để có sự chọn lọc.

Chương 1

CƠ SỞ LÝ THUYẾT

1.1. Giới thiệu về IoT

1.1.1. Khái niệm về IoT

Mạng lưới vạn vật kết nối Internet hay Internet of Things, viết tắt là IoT, là một kịch bản trên thế giới trong đó mỗi đồ vật và con người được cung cấp một định danh duy nhất của riêng mình và tất cả đều có danh tính riêng. Khả năng truyền tải và trao đổi thông tin, dữ liệu qua một mạng duy nhất mà không yêu cầu sự tương tác trực tiếp giữa con người với con người, hay con người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và internet.

Nó đơn giản là tập hợp các thiết bị có khả năng kết nối với nhau, với internet và với thế giới bên ngoài để thực hiện một công việc nhất định. Nói một cách đơn giản, IoT là tất cả các thiết bị có thể kết nối với nhau.

Việc kết nối có thể được thực hiện thông qua wifi, kết nối băng thông rộng (3G, 4G), Bluetooth, ZigBee, hồng ngoại... Thiết bị có thể là điện thoại thông minh, máy tính bảng, điều hòa, bóng đèn, máy giặt và nhiều thiết bị gia dụng khác. IoT sẽ là một mạng lưới khổng lồ kết nối mọi thứ trong đó có con người và sẽ tồn tại các mối quan hệ giữa người với người, người với thiết bị, thiết bị với thiết bị.

1.1.2. Lịch sử phát triển của IoT

Internet of Things (IoT) đã được phát triển qua nhiều giai đoạn khác nhau, từ sự ra đời của những thiết bị đầu tiên đối với tầm nhìn và triển vọng hiện tại của nó.

- Giai đoạn đầu tiên (tháng 12 năm 1980): Khái niệm IoT được đưa ra vào những năm 1980, khi máy tính bắt đầu xuất hiện. Một số thiết bị như máy tính cá nhân, điện thoại di động, máy tính bảng và máy tính nhúng đã được phát triển.

- Giai đoạn tiếp theo (thập kỷ 1990): Trong giai đoạn này, Internet đã trở nên phổ biến và các công ty bắt đầu cho ra mắt các sản phẩm IoT đầu tiên như máy chấm công tự động, mã vạch máy quét và các thiết đo lường.

- Giai đoạn phát triển nhanh chóng (thập kỷ 2000): Trong thời kỳ này, công nghệ kết nối không dây như Wi-Fi, Bluetooth, 3G, 4G được phát triển, mở ra cơ hội cho sự phát triển của các sản phẩm IoT. Các thiết bị thông minh đầu tiên như đèn LED thông minh và năng lượng tiêu thụ đã được ra đời.

- Giai đoạn phát triển mạnh mẽ (tháng 12 năm 2010): Trong giai đoạn này, IoT đã có phát triển mạnh mẽ với sự ra đời của các sản phẩm thông minh như điện thoại thông minh, thiết bị thông minh đồng hồ, tivi thông minh, ô tô thông minh và nhà thông minh.

Ứng dụng IoT trong các lĩnh vực y tế, nông nghiệp, công nghiệp và giao thông vận tải cũng được triển khai rộng rãi.

- Tương lai (hiện tại): IoT đang tiếp tục phát triển mạnh mẽ và mở ra cơ hội cho những ứng dụng mới và tiềm năng trong nhiều lĩnh vực, từ giải quyết các vấn đề môi trường đến tăng cường an toàn và an ninh. an ninh cho người dân. Nhiều công ty đang đầu tư vào IoT nghiên cứu và phát triển và các chuyên gia dự đoán rằng trong tương lai, số lượng thiết bị IoT sẽ tăng gấp đôi hoặc gấp ba.

1.2. Nguyên lý hoạt động của IoT

IoT là hệ thống kết nối các thiết bị và cảm biến thông qua internet. Với sự phát triển của công nghệ truyền thông, các thiết bị có thể giao tiếp với nhau và truyền dữ liệu qua Internet. Trong hệ thống IoT, các thiết bị và cảm biến được kết nối với nhau thông qua mạng các giao thức như Wi-Fi, Bluetooth, ZigBee, Z-Wave và LoRa.

Điểm quan trọng của IoT là các đối tượng phải có thể được nhận dạng và định dạng được (identifiable). Nếu mọi thứ đều được “đánh dấu” để phân biệt chính vật thể đó với những vật xung quanh thì chúng ta hoàn toàn có thể quản lý được thông qua máy tính. Ngoài các kỹ thuật trên, nếu nhìn từ thế giới web, chúng ta có thể sử dụng các địa chỉ độc nhất để nhận dạng từng thứ, chẳng hạn như địa chỉ IP. Mỗi thiết bị sẽ có một IP riêng biệt, không thể

nhằm lẫn. Sự xuất hiện của IPv6 với không gian địa chỉ cực lớn sẽ giúp mọi thứ dễ dàng kết nối Internet cũng như kết nối với nhau.

Các thiết bị IoT thường được trang bị các cảm biến để thu thập dữ liệu, các cảm biến này có thể cảm biến nhiệt độ, độ ẩm, ánh sáng, âm thanh, chuyển động, tốc độ và áp suất. Dữ liệu được thu thập bởi các cảm biến sẽ được truyền đến các thiết bị IoT khác hoặc được lưu trữ trên đám mây để phân tích và sử dụng.

1.3. Ứng dụng của IoT trong đời sống

IoT có nhiều ứng dụng trong nhiều lĩnh vực khác nhau như tự động hóa gia đình, mua sắm thông minh, quản lý thiết bị cá nhân, đồng hồ đo thông minh, phản hồi trong các tình huống khẩn cấp, quản lý môi trường.

Hiện nay, nhiều hãng, công ty, tổ chức trên thế giới đang nghiên cứu các nền tảng giúp xây dựng nhanh các ứng dụng cho IoT.

- Nhà thông minh (Smart Home): Các thiết bị như đèn, tủ lạnh, máy điều hòa, và camera an ninh có thể được điều khiển từ xa qua smartphone. Các hệ thống tự động hóa giúp tối ưu hóa việc sử dụng năng lượng, tăng cường an ninh và tiện ích cho người dùng.

- Y tế thông minh (Smart Healthcare): Các thiết bị y tế đeo tay (wearables) như máy theo dõi nhịp tim, máy đo đường huyết giúp theo dõi sức khỏe người dùng liên tục. Hệ thống quản lý bệnh viện tự động, cung cấp dữ liệu bệnh nhân một cách nhanh chóng và chính xác.

- Nông nghiệp thông minh (Smart Agriculture): Cảm biến IoT giúp nông dân theo dõi điều kiện đất đai, thời tiết và tình trạng cây trồng, từ đó tối ưu hóa việc tưới nước và sử dụng phân bón. Máy bay không người lái (drone) được sử dụng để giám sát và chăm sóc cây trồng từ trên cao.

- Công nghiệp thông minh (Smart Industry): Các nhà máy thông minh sử dụng IoT để quản lý dây chuyền sản xuất, kiểm soát chất lượng và bảo trì thiết bị. Hệ thống IoT giúp tối ưu hóa quy trình sản xuất và giảm chi phí vận hành.

- Quản lý năng lượng thông minh (Smart Energy Management): Hệ thống IoT giúp theo dõi và quản lý tiêu thụ năng lượng, từ đó cải thiện hiệu suất sử dụng năng lượng và

giảm thiểu lãng phí. Các lưới điện thông minh (smart grid) giúp tối ưu hóa việc phân phối và sử dụng điện.

IoT đang dần thay đổi cách chúng ta sống, làm việc và tương tác với thế giới xung quanh, mang lại nhiều lợi ích về hiệu quả, an toàn và tiện ích.

1.4. Ưu điểm và thách thức trong việc ứng dụng IoT

IoT là một công nghệ đang ngày càng phát triển, đem lại nhiều lợi ích cho đời sống của con người. Tuy nhiên, như bất kỳ công nghệ nào khác, IoT cũng có những ưu điểm và nhược điểm nhất định.

1.4.1. Ưu điểm của IoT

- Tăng cường hiệu quả: IoT giúp tự động hóa nhiều quy trình, giảm sự can thiệp của con người, từ đó tăng cường hiệu suất và giảm sai sót.

Ví dụ, trong sản xuất công nghiệp, IoT có thể giám sát và điều chỉnh quy trình để tối ưu hóa sản lượng và chất lượng sản phẩm.

- Tiết kiệm chi phí: IoT giúp giám sát và quản lý tài nguyên hiệu quả hơn, từ đó giảm lãng phí và tiết kiệm chi phí. Trong quản lý năng lượng, các hệ thống IoT có thể giảm tiêu thụ điện năng bằng cách tự động tắt các thiết bị không sử dụng.

- Cải thiện chất lượng cuộc sống: Các thiết bị nhà thông minh giúp cuộc sống hàng ngày trở nên tiện nghi hơn, từ việc điều khiển ánh sáng, nhiệt độ, đến an ninh gia đình. Trong y tế, các thiết bị đeo thông minh giúp theo dõi sức khỏe liên tục và cảnh báo sớm các vấn đề sức khỏe.

- Dữ liệu và phân tích: IoT thu thập một lượng lớn dữ liệu từ các thiết bị kết nối, giúp các tổ chức và doanh nghiệp phân tích và ra quyết định dựa trên dữ liệu thực tế.

Ví dụ, các cảm biến IoT trong nông nghiệp có thể cung cấp dữ liệu về độ ẩm đất, từ đó giúp nông dân điều chỉnh tưới nước hợp lý.

- Tăng cường an ninh và giám sát: IoT cho phép giám sát từ xa các hệ thống và tài sản, từ đó cải thiện an ninh và quản lý rủi ro. Các camera an ninh thông minh và hệ thống báo động giúp bảo vệ gia đình và doanh nghiệp khỏi các mối đe dọa.

1.4.2. Thách thức trong việc nghiên cứu và triển khai

IoT vẫn còn một số trở ngại như chưa có ngôn ngữ chung. Ở cấp độ cơ bản nhất, Internet là một mạng được sử dụng để kết nối thiết bị này với thiết bị khác. Chỉ riêng khả năng kết nối không đảm bảo rằng các thiết bị biết cách giao tiếp với nhau. Để các thiết bị có thể giao tiếp với nhau, chúng sẽ cần một hoặc nhiều giao thức(protocols), có thể coi đây là ngôn ngữ chuyên biệt để giải quyết một nhiệm vụ nào đó. Chắc hẳn chúng ta đã ít nhiều sử dụng một trong những giao thức phổ biến nhất trên thế giới là Giao thức truyền siêu văn bản (HTTP) để tải web.

Ngoài ra chúng ta còn có SMTP, POP, IMAP cho email, FTP trao đổi file... Các giao thức như thế này hoạt động tốt trên bờ các máy chủ web, mail và FTP thường không phải nói với nhau nhiều, khi cần, một phần mềm dịch thuật đơn giản sẽ đóng vai trò trung gian để hai bên hiểu nhau. Còn với các thiết bị IoT, chúng phải làm rất nhiều việc, phải nói chuyện với nhiều loại máy móc, thiết bị khác nhau. Đáng tiếc, hiện tại không có nhiều sự đồng thuận về các giao thức để IoT trao đổi dữ liệu.

- Bảo mật và riêng tư: Sự kết nối và truyền dữ liệu liên tục giữa các thiết bị IoT tạo ra nhiều điểm yếu bảo mật. Nếu không được bảo vệ tốt, hệ thống IoT dễ bị tấn công và dữ liệu cá nhân có thể bị xâm phạm. Các thiết bị IoT có thể thu thập rất nhiều thông tin về người dùng, gây lo ngại về quyền riêng tư.

- Chi phí đầu tư ban đầu cao: Việc triển khai các hệ thống IoT đòi hỏi chi phí ban đầu cao cho thiết bị, cơ sở hạ tầng mạng và phần mềm. Doanh nghiệp cần đầu tư vào các công nghệ mới và đào tạo nhân viên để quản lý và vận hành hệ thống IoT.

- Phức tạp hóa hệ thống: Sự kết nối và tích hợp giữa nhiều thiết bị IoT có thể làm cho hệ thống trở nên phức tạp và khó quản lý. Việc duy trì và cập nhật hệ thống IoT yêu cầu kiến thức kỹ thuật và kỹ năng chuyên môn.

- Phụ thuộc vào kết nối internet: Hệ thống IoT phụ thuộc nhiều vào kết nối internet ổn định. Mất kết nối internet có thể làm gián đoạn các dịch vụ và chức năng của các thiết bị IoT. Ở những khu vực có kết nối internet kém, việc triển khai IoT có thể gặp nhiều khó khăn.

- Vấn đề tương thích: Các thiết bị IoT đến từ nhiều nhà sản xuất khác nhau có thể không tương thích với nhau, gây khó khăn trong việc tích hợp và vận hành. Thiếu các tiêu chuẩn chung cho IoT cũng có thể gây ra sự không đồng bộ và hiệu suất kém.

1.5. Hệ thống giám sát đảm bảo an toàn cho gia đình

Trước đây, khi nói đến bảo vệ an ninh và an toàn cho gia đình, chúng ta nghĩ đến một điều: thuê nhân viên bảo vệ hoặc sử dụng bảo vệ thú cưng... Ngày nay, với sự phát triển của công nghiệp cũng như công nghệ điện tử và tự động hóa, đặc biệt là công nghệ điện tử số, con người đã tạo ra những phát minh mới trong lĩnh vực chống đột nhập. Về nguyên tắc, hệ thống giám sát đảm bảo an toàn cho gia đình gồm 3 phần chính: Cảm biến, bộ xử lý trung tâm và các thiết bị cảnh báo. Các cảm biến chính là các cảm biến thu thập tín hiệu sau đó gửi về bộ xử lý trung tâm (có nhiều loại cảm biến như cảm biến khói, cảm biến từ, cảm biến nhiệt, cảm biến hồng ngoại, cảm biến quang học, cảm biến cơ học, cảm biến áp suất, cảm biến âm thanh, cảm biến điện...).

Bộ xử lý trung tâm là bộ phận gửi thông tin từ cảm biến rồi xử lý. Tùy thuộc vào người lập trình mà nó có thể đưa ra những phản ứng khác nhau khi nhận được tín hiệu. Hầu hết các phản hồi của bộ điều khiển trung tâm đều được gửi đến các thiết bị thông báo để thông báo tình hình cho người dùng. Ở đây bộ xử lý trung tâm được sử dụng là module ESP8266Wifi. Các thiết bị thông báo là loa, còi, điện thoại, đèn báo...

Hiện nay, các Hệ thống giám sát đảm bảo an toàn cho gia đình hiện đại được tích hợp thêm nhiều chức năng như nguồn điện dự phòng, mật khẩu điều khiển, bật/tắt từ xa qua điện thoại, bật/tắt hệ thống điện và kết nối với các hệ thống thông minh khác. Một tính năng quan trọng khác là khả năng lưu lại dữ liệu lịch sử biến động. Các hệ thống này có thể ghi lại các sự kiện và thông tin từ cảm biến, tạo ra một cơ sở dữ liệu lịch sử về hoạt động trong và xung quanh ngôi nhà. Từ dữ liệu lịch sử này, hệ thống có thể phát hiện ra các bất thường hoặc mẫu hình đáng ngờ, chẳng hạn như các hoạt động bất thường vào những thời điểm không

hợp lý hoặc những biến động lặp đi lặp lại mà không có lý do rõ ràng. Việc phân tích dữ liệu này giúp tăng cường khả năng dự đoán và ngăn chặn các mối đe dọa tiềm ẩn, đảm bảo an toàn và an ninh tốt hơn cho gia đình."

Tiểu kết chương 1:

Chương này giới thiệu khái niệm, lịch sử phát triển và nguyên lý hoạt động của Internet of Things (IoT), nêu bật các ứng dụng trong đời sống như nhà thông minh, y tế, nông nghiệp và công nghiệp. Nó cũng phân tích những ưu điểm như tăng cường hiệu quả, tiết kiệm chi phí, và những thách thức như bảo mật, chi phí đầu tư cao. Cuối cùng, chương trình bày hệ thống giám sát an ninh gia đình hiện đại sử dụng IoT, với các thành phần cảm biến, bộ xử lý trung tâm và thiết bị cảnh báo.

Chương 2

CÁC THÀNH PHẦN CHÍNH CỦA HỆ THỐNG

CẢM BIẾN GIÁM SÁT NGÔI NHÀ

2.1. Thiết bị cảm biến

2.1.1. Định nghĩa

Cảm biến (sensor) là thiết bị hoặc module có khả năng phát hiện các thay đổi trong môi trường vật lý hoặc hóa học và chuyển đổi chúng thành các tín hiệu điện hoặc dữ liệu số để xử lý. Trong hệ thống Internet of Things (IoT), cảm biến đóng vai trò là "giác quan" của hệ thống, thu thập thông tin từ môi trường xung quanh và gửi dữ liệu này đến các thiết bị khác để xử lý và hành động.

2.1.2. Vai trò của cảm biến trong hệ thống IoT

- Thu thập dữ liệu: Cảm biến thu thập dữ liệu về các thông số vật lý như nhiệt độ, độ ẩm, ánh sáng, chuyển động, vị trí, áp suất, và nhiều thông số khác.

Ví dụ, cảm biến nhiệt độ có thể theo dõi nhiệt độ môi trường trong một nhà kính, cảm biến chuyển động có thể phát hiện sự di chuyển trong nhà để báo động an ninh.

- Giám sát và điều khiển từ xa: Các cảm biến giúp giám sát từ xa các điều kiện và trạng thái của môi trường hoặc thiết bị, từ đó điều khiển các hệ thống khác một cách tự động hoặc bán tự động.

Ví dụ, hệ thống HVAC (Heating, Ventilation, and Air Conditioning) sử dụng cảm biến nhiệt độ để duy trì nhiệt độ mong muốn trong tòa nhà.

- Tăng hiệu quả và tiết kiệm năng lượng: Cảm biến giúp tối ưu hóa việc sử dụng năng lượng bằng cách cung cấp thông tin chính xác về điều kiện môi trường và trạng thái của thiết bị.

Ví dụ, cảm biến ánh sáng có thể điều chỉnh độ sáng của đèn dựa trên ánh sáng tự nhiên, tiết kiệm năng lượng điện.

- Phát hiện và cảnh báo sự cố: Cảm biến có thể phát hiện các điều kiện bất thường hoặc nguy hiểm, từ đó gửi cảnh báo để người dùng hoặc hệ thống xử lý kịp thời.

Ví dụ, cảm biến khói phát hiện khói trong nhà và kích hoạt báo động cháy, giúp ngăn chặn thiệt hại và bảo vệ an toàn cho cư dân.

- Tích hợp với các hệ thống khác: Cảm biến IoT có thể tích hợp với các hệ thống thông minh khác để tạo ra một môi trường tự động và liên kết.

Ví dụ, trong một ngôi nhà thông minh, cảm biến chuyển động có thể kích hoạt camera an ninh, đèn chiếu sáng, và hệ thống báo động khi phát hiện sự di chuyển.

- Phân tích và dự báo: Dữ liệu từ cảm biến được thu thập và phân tích để phát hiện các xu hướng và mẫu hình, giúp dự đoán và tối ưu hóa hoạt động.

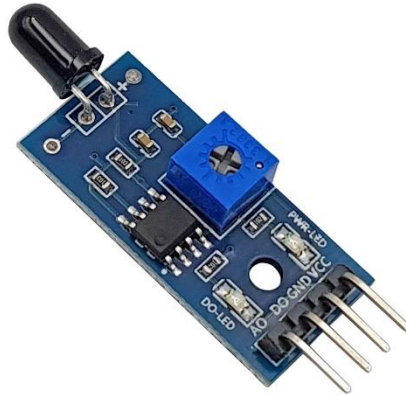
Ví dụ, trong nông nghiệp thông minh, dữ liệu từ cảm biến độ ẩm đất có thể được sử dụng để dự đoán nhu cầu tưới nước và cải thiện năng suất cây trồng.

2.2. Các loại thiết bị cảm biến được sử dụng

2.2.1. *Module flame sensor*

Module cảm biến lửa là một thiết bị được sử dụng để phát hiện sự xuất hiện của lửa trong môi trường xung quanh. Nó thường được sử dụng để phát hiện và cảnh báo về nguy cơ cháy trong các ứng dụng an ninh, bảo vệ cháy, và nhà thông minh.

Cảm biến lửa thường sử dụng nguyên lý của sự phát ra của các bức xạ từ hồng ngoại được tạo ra bởi lửa. Khi lửa xuất hiện, năng lượng hồng ngoại sẽ được cảm biến và chuyển đổi thành tín hiệu điện. Module cảm biến lửa sẽ nhận dạng tín hiệu này và kích hoạt các biện pháp bảo vệ như cảnh báo âm thanh, kích hoạt hệ thống phun nước hoặc báo động, và gửi thông báo cảnh báo đến người dùng thông qua các hệ thống thông tin liên lạc.



Hình 2.1: Hình ảnh minh họa cảm biến lửa (nguồn: electropeak.vn)

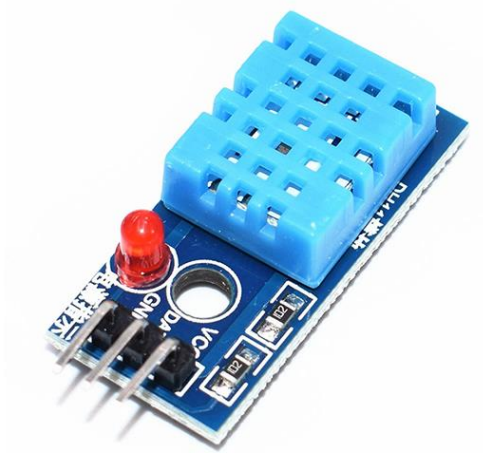
Một số tính năng phổ biến của module cảm biến lửa bao gồm:

- Độ nhạy: Có thể điều chỉnh độ nhạy của cảm biến để phù hợp với mức độ nguy cơ cháy cụ thể trong môi trường.
- Khoảng cách hoạt động: Xác định khoảng cách mà cảm biến có thể phát hiện được lửa, đảm bảo phủ sóng hiệu quả cho toàn bộ khu vực cần giám sát.
- Tích hợp: Có thể tích hợp với các hệ thống bảo mật và an ninh thông minh khác, cho phép quản lý và kiểm soát từ xa.
- Tin nhắn cảnh báo: Có khả năng gửi tin nhắn cảnh báo đến người dùng thông qua các phương tiện như điện thoại di động hoặc email.
- Tiêu thụ năng lượng: Hoạt động với mức tiêu thụ năng lượng thấp để tiết kiệm điện năng và tăng tính ổn định của hệ thống.

2.2.2. Module DHT11

DHT11 là một cảm biến kỹ thuật số giá rẻ để cảm nhận nhiệt độ và độ ẩm. Cảm biến này có thể dễ dàng giao tiếp với bất kỳ bộ vi điều khiển vi nào như Arduino, Raspberry Pi, ... để đo độ ẩm và nhiệt độ ngay lập tức.

DHT11 là một cảm biến độ ẩm tương đối. Để đo không khí xung quanh, cảm biến này sử dụng một điện trở nhiệt và một cảm biến độ ẩm điện dung.



Hình 2.2: Hình ảnh minh họa cảm biến nhiệt độ độ ẩm DHT11 (nguồn: nabatechshop.com)

DHT11 hoạt động dựa trên nguyên lý của hiệu ứng cảm biến nhiệt độ và độ ẩm. Cảm biến này có một cặp điện cực dẫn điện dựa trên polymer, có khả năng thay đổi điện trở khi nhiệt độ và độ ẩm thay đổi.

Đặc điểm kỹ thuật:

- Nhiệt độ đo: Từ 0°C đến 50°C với độ chính xác $\pm 2^{\circ}\text{C}$.
- Độ ẩm đo: Từ 20% đến 90% với độ chính xác $\pm 5\%$.
- Điện áp hoạt động: 3.5V đến 5.5V.
- Giao tiếp: Sử dụng giao tiếp 1 dây để truyền dữ liệu số từ cảm biến đến vi điều khiển.

Ứng dụng:

Giám sát môi trường trong nhà: DHT11 thường được sử dụng trong các dự án giám sát nhiệt độ và độ ẩm trong nhà, ví dụ như giám sát môi trường trong nhà thông qua Arduino hoặc Raspberry Pi.

Thiết bị điều khiển tự động: Cảm biến này cũng có thể được sử dụng trong các hệ thống tự động hóa để điều khiển các thiết bị dựa trên điều kiện môi trường.

2.2.3. Module BMP180

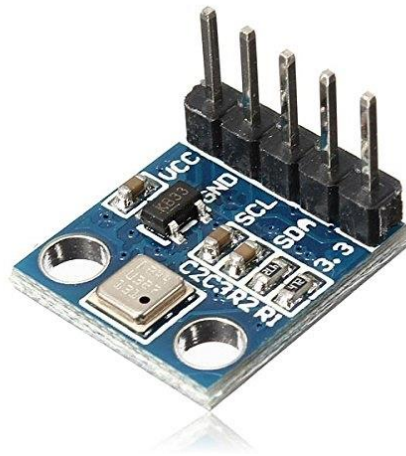
Cảm biến áp suất khí quyển BMP180 là loại cảm biến được sử dụng chủ yếu để đo áp suất khí hoặc sinh trắc học áp suất. Nếu nói về áp suất, thì áp suất là một lực tác dụng lên trên một đơn vị diện tích.

Đơn vị áp suất trong hệ SI là newton trên mét vuông tương ứng với một pascal (Pa). Có rất nhiều trường hợp đo các loại áp suất khác nhau nhưng ở đây chúng ta chỉ quan tâm đến áp suất khí quyển.

Áp suất khí quyển về cơ bản là lực tác dụng lên tất cả các vật xung quanh. Có nghĩa là, trọng lượng các chất khí trong khí quyển tạo ra áp suất khí quyển.

Để đo áp suất của các khí này, cảm biến áp suất khí quyển được sử dụng. Đây là giải pháp cảm biến có chi phí thấp với độ chính xác cao và được thiết kế đặc biệt cho các ứng dụng như dự báo thời tiết, thiết bị thể thao, định vị GPS, thiết bị ngoại vi máy tính, điều hướng thông khí trong nhà, các dự án báo vận tốc góc, v.v.

Có thể sử dụng để đo độ cao vì áp suất sẽ thay đổi theo độ cao. Rất dễ sử dụng và dễ hàn trên bảng mạch in (PCB) với kích thước nhỏ và tiêu thụ ít điện năng. Nếu sử dụng để cảm biến nhiệt độ thì BMP180 là sự lựa chọn hoàn hảo. Có sẵn trên thị trường hoặc cửa hàng trực tuyến. Dưới đây là hình cảm biến áp suất khí quyển BMP180



Hình 2.3: Hình ảnh minh họa cảm biến áp suất BMP180(nguồn: makerelectronics.com)

Đặc điểm kỹ thuật:

- Nhiệt độ đo: Từ 0°C đến 50°C với độ chính xác $\pm 2^\circ\text{C}$.
- Độ ẩm đo: Từ 20% đến 90% với độ chính xác $\pm 5\%$.
- Điện áp hoạt động: 3.5V đến 5.5V.
- Giao tiếp: Sử dụng giao tiếp 1 dây để truyền dữ liệu số từ cảm biến đến vi điều khiển.

Ứng dụng:

- Giám sát môi trường trong nhà: DHT11 thường được sử dụng trong các dự án giám sát nhiệt độ và độ ẩm trong nhà, ví dụ như giám sát môi trường trong nhà thông qua Arduino hoặc Raspberry Pi.

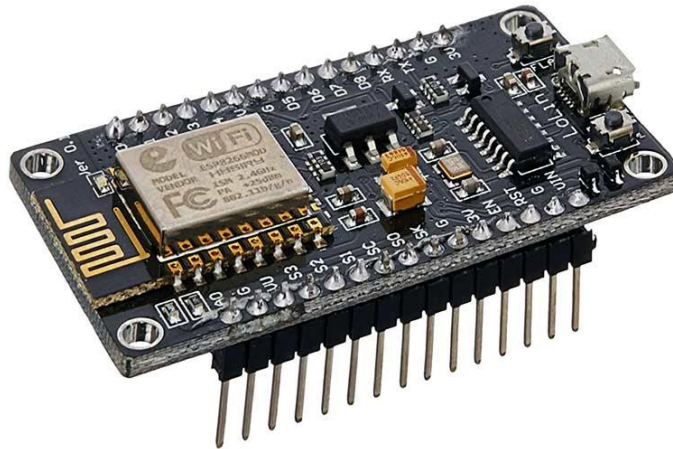
- Thiết bị điều khiển tự động: Cảm biến này cũng có thể được sử dụng trong các hệ thống tự động hóa để điều khiển các thiết bị dựa trên điều kiện môi trường.

2.3. Bộ xử lý và truyền thông

Trong xây dựng hệ thống cảm biến giám sát ngôi nhà, tôi sử dụng vi điều khiển Module ESP8266 NodeMCU và kết nối WiFi để xử lý dữ liệu và kết nối với các thiết bị khác nhau. Vì khả năng kết nối mạng mạnh mẽ, giá thành rẻ, và dễ dàng lập trình.

2.3.1. Vi điều khiển ESP8266 NodeMCU

ESP8266 NodeMCU là một board phổ biến trong cộng đồng IoT và nhà phát triển điện tử. Nó kết hợp vi điều khiển ESP8266 và một mô-đun wifi vào một board duy nhất, cung cấp một giải pháp tiện lợi và linh hoạt cho việc phát triển các ứng dụng IoT.



Hình 2.4: Hình ảnh minh họa ESP8266 NodeMCU (nguồn: www.okystar.com)

Các tính năng chính của ESP8266 NodeMCU bao gồm vi điều khiển ESP8266 với tốc độ xung nhịp lên đến 80MHz hoặc 160MHz, bộ nhớ lưu trữ flash 4MB hoặc 16MB, module wifi tích hợp, và các chân kết nối GPIO, ADC, UART, và SPI. Điều này cho phép

người dùng kết nối và điều khiển các thiết bị và cảm biến khác nhau thông qua wifi và giao thức TCP/IP.

Một trong những điểm mạnh của ESP8266 NodeMCU là khả năng lập trình mạnh mẽ và linh hoạt. Nó hỗ trợ việc lập trình bằng ngôn ngữ Lua hoặc C++ thông qua môi trường phát triển tích hợp (IDE) như Arduino IDE hoặc PlatformIO. Ngoài ra, có rất nhiều thư viện và ví dụ sẵn có cho ESP8266, giúp người dùng dễ dàng phát triển các ứng dụng từ các dự án đơn giản đến các dự án phức tạp hơn.

ESP8266 NodeMCU cũng có các chân kết nối chuẩn, cho phép người dùng kết nối với các linh kiện ngoại vi và mạch mở rộng một cách dễ dàng. Điều này tạo ra một môi trường linh hoạt cho việc phát triển các ứng dụng IoT và nhúng. Với tính năng wifi tích hợp và khả năng lập trình đa dạng, nó là một công cụ lý tưởng để bắt đầu trong thế giới của Internet of Things.

2.3.2. Các thiết bị xử lý trung gian trong hệ thống IoT

- Gateway (Cổng kết nối):

- + Chức năng: Gateway hoạt động như một trung gian giữa các thiết bị IoT và mạng internet. Nó thu thập dữ liệu từ các cảm biến và gửi đến đám mây hoặc các hệ thống phân tích.
- + Vai trò: Đảm bảo kết nối liên tục, bảo mật và quản lý dữ liệu từ các thiết bị IoT.

- Router:

- + Chức năng: Router kết nối các thiết bị IoT trong mạng nội bộ với internet, cung cấp địa chỉ IP và quản lý lưu lượng mạng.
- + Vai trò: Đảm bảo kết nối internet ổn định và quản lý mạng nội bộ của các thiết bị IoT.

- Bộ lưu trữ đám mây (Firebase - Realtime):

- + Chức Năng: Lưu trữ và quản lý dữ liệu thu thập từ các thiết bị IoT, cung cấp khả năng truy cập từ xa và phân tích dữ liệu.
- + Vai Trò: Cung cấp không gian lưu trữ lớn, bảo mật và khả năng xử lý dữ liệu.

- Máy Chủ (Server):

- + Chức Năng: Xử lý và lưu trữ dữ liệu, chạy các ứng dụng và dịch vụ liên quan đến IoT.
- + Vai Trò: Cung cấp sức mạnh tính toán và quản lý dữ liệu cho hệ thống IoT.

2.3.3. Chuẩn giao tiếp không dây WiFi

WiFi (Wireless Fidelity) là công nghệ kết nối không dây sử dụng sóng radio để truyền tải dữ liệu giữa các thiết bị như máy tính, điện thoại di động, máy tính bảng và các thiết bị IoT. WiFi tuân theo các tiêu chuẩn do IEEE (Institute of Electrical and Electronics Engineers) định nghĩa, cụ thể là các tiêu chuẩn thuộc họ IEEE 802.11.

Trong hệ thống này sử dụng chuẩn giao tiếp không dây WiFi 4 (IEEE 802.11n) có băng tần 2.4 GHz và 5GHz, với tốc độ tối đa 600Mbps. Chuẩn giao tiếp không dây này thường được sử dụng rộng rãi trong các hộ gia đình, văn phòng và các doanh nghiệp nhỏ.

2.4. Giới thiệu phần mềm nạp code và vẽ sơ đồ

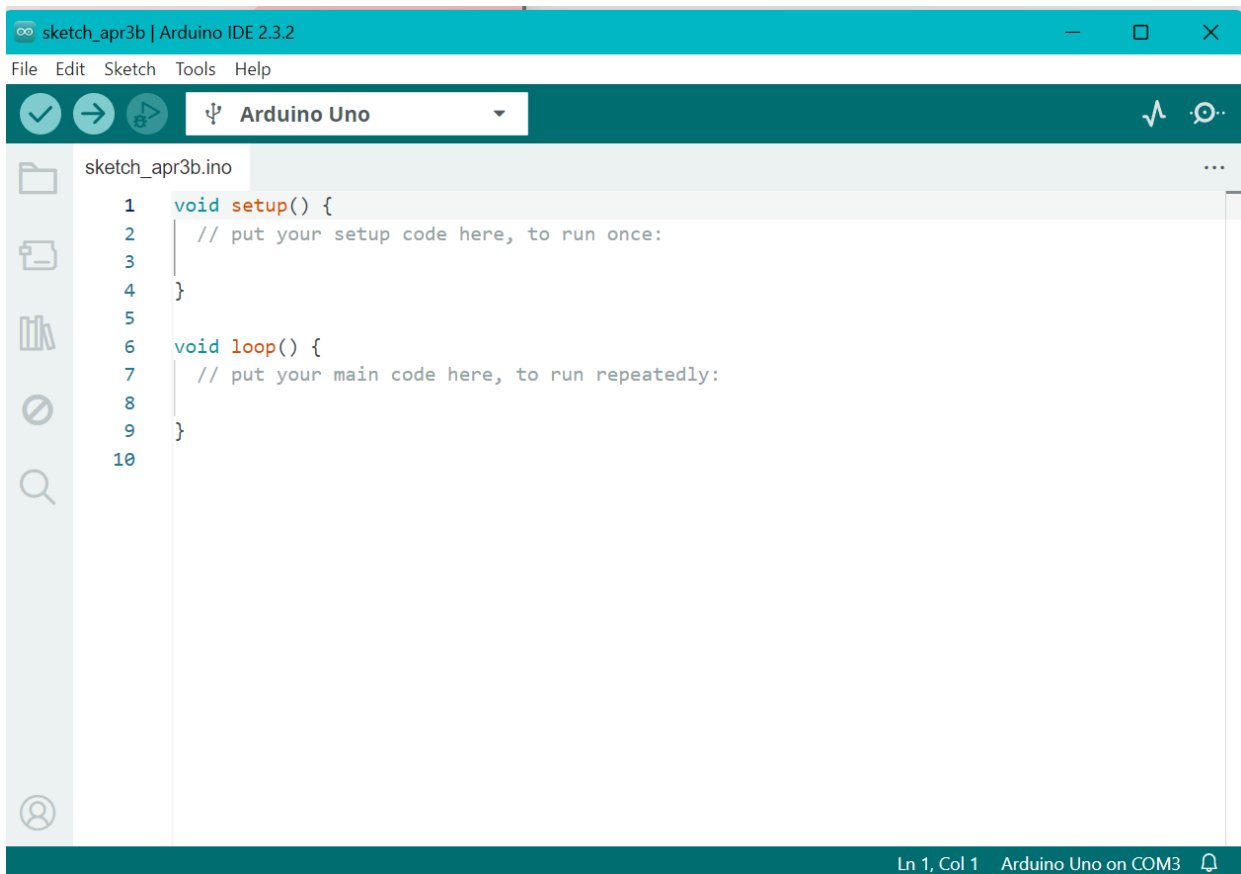
2.4.1. Phần mềm Arduino IDE

Môi trường phát triển tích hợp Arduino IDE là một ứng dụng đa nền tảng được viết bằng Java và được trích dẫn từ IDE cho các ngôn ngữ lập trình quy trình xử lý và các dự án lắp ráp. Nó bao gồm một trình soạn thảo mã với các tính năng như tô sáng cú pháp, khớp dấu ngoặc khối chương trình, thực thi tự động và cũng có khả năng biên dịch và tải chương trình vào board mạch chỉ bằng một cú nhấp chuột duy nhất. Một chương trình hoặc mã viết cho Arduino được gọi là "sketch".

Chương trình Arduino được viết bằng C hoặc C++. Arduino IDE đi kèm với một thư viện phần mềm được gọi là “Wiring” từ dự án lắp ráp ban đầu, cho hoạt động đầu vào/đầu ra phổ biến trở lên dễ dàng hơn nhiều. Người sử dụng chỉ cần định nghĩa hai hàm để thực hiện một chương trình điều hành theo chu kỳ.

Khi chúng ta bật mạch Arduino, reset hay nạp chương trình mới, hàm setup() sẽ được gọi đến đầu tiên. Sau khi xử lý xong hàm setup(), Arduino sẽ nhảy đến hàm loop() và lặp vô hạn hàm này cho đến khi tắt điện board mạch Arduino.

Dưới đây là giao diện của Arduino IDE



Hình 2.5: Hình ảnh minh họa phần mềm Arduino IDE

2.4.2. Phần mềm Proteus

Phần mềm Proteus là một ứng dụng mạnh mẽ được sử dụng rộng rãi trong lĩnh vực thiết kế và mô phỏng mạch điện tử. Nó cung cấp một môi trường tích hợp để thiết kế, kiểm tra và mô phỏng các mạch điện tử và hệ thống nhúng.

Với Proteus, người dùng có thể thiết kế và mô phỏng mạch điện tử bằng cách sử dụng các phần tử điện tử có sẵn trong thư viện hoặc tự tạo các phần tử tùy chỉnh. Phần mềm cung cấp một giao diện đồ họa thân thiện và dễ sử dụng, giúp người dùng dễ dàng kéo và thả các linh kiện để tạo ra mạch điện tử theo ý muốn. Proteus cũng cung cấp các tính năng mô phỏng mạnh mẽ, cho phép người dùng kiểm tra và xem xét hoạt động của mạch trước khi triển khai vào thực tế. Người dùng có thể mô phỏng các tín hiệu đầu vào và ra, điều khiển các linh kiện và thiết bị, và kiểm tra hiệu suất và hoạt động của mạch trong các điều kiện khác nhau.

Ngoài ra, Proteus còn cung cấp tính năng kết hợp với các mô-đun vi điều khiển như Arduino và PIC, cho phép người dùng thiết kế và mô phỏng các hệ thống nhúng phức tạp. Tóm lại, Proteus là một công cụ đa năng và mạnh mẽ cho việc thiết kế và mô phỏng mạch điện tử và hệ thống nhúng, đồng thời cung cấp một cách tiếp cận linh hoạt và hiệu quả cho các kỹ sư và nhà thiết kế điện tử.

2.5. Công nghệ lưu trữ đám mây - Firebase

2.5.1. Tổng quan về Firebase

Firebase là một nền tảng phát triển ứng dụng di động và web được cung cấp bởi Google. Nó cung cấp một loạt các dịch vụ đám mây, từ việc lưu trữ và cơ sở dữ liệu đến xác thực người dùng và phân tích, giúp cho việc phát triển và triển khai các ứng dụng trở nên dễ dàng và hiệu quả. Dưới đây là một số điểm nổi bật của Firebase:

- Realtime database: Firebase cung cấp một cơ sở dữ liệu thời gian thực cho phép đồng bộ dữ liệu giữa các thiết bị một cách tức thì. Dữ liệu được cập nhật ngay lập tức khi có thay đổi, giúp cho việc phát triển các ứng dụng đa nền tảng trở nên dễ dàng hơn.
- Authentication: Firebase Authentication cung cấp các phương thức xác thực người dùng an toàn và dễ dàng. Người dùng có thể đăng nhập bằng email, số điện thoại, hoặc tài khoản mạng xã hội như Google, Facebook, và Twitter.
- Cloud firestore: Firestore là một cơ sở dữ liệu linh hoạt cho phép lưu trữ và truy vấn dữ liệu từ bất kỳ thiết bị nào. Nó hỗ trợ cấu trúc dữ liệu linh hoạt và cho phép truy vấn dữ liệu một cách hiệu quả.
- Cloud functions: Firebase Cloud Functions cho phép bạn viết và triển khai các hàm backend mà không cần quản lý các máy chủ. Điều này giúp cho việc xử lý logic phía server trở nên linh hoạt và dễ dàng hơn.
- Hosting: Firebase Hosting cung cấp một cách dễ dàng và nhanh chóng để triển khai các ứng dụng web và tĩnh. Bạn có thể triển khai các trang web chỉ trong vài phút mà không cần phải quản lý cơ sở hạ tầng.

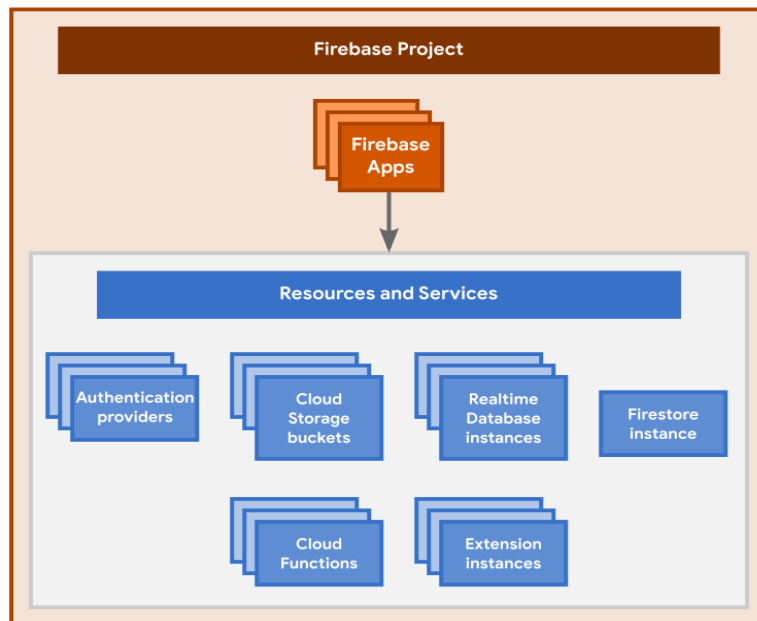
- Analytics: Firebase Analytics cung cấp các công cụ phân tích mạnh mẽ giúp bạn hiểu rõ hơn về cách người dùng tương tác với ứng dụng của bạn. Bạn có thể theo dõi các sự kiện, người dùng và thu thập dữ liệu phân tích để cải thiện trải nghiệm của người dùng.

Firebase là một nền tảng đáng tin cậy và mạnh mẽ cho việc phát triển ứng dụng di động và web. Nó cung cấp một loạt các dịch vụ cần thiết để xây dựng, triển khai và quản lý các ứng dụng của bạn một cách dễ dàng và hiệu quả.

2.5.2. *Mối quan hệ giữa các dự án, ứng dụng và Firebase*

Dự án Firebase là thực thể cấp cao nhất của Firebase. Trong một dự án, bạn có thể đăng ký các ứng dụng Apple, Android hoặc web của mình. Sau khi đăng ký ứng dụng của mình với Firebase, bạn có thể thêm SDK Firebase cho bất kỳ số lượng sản phẩm Firebase nào, như Analytics, Cloud Firestore, Giám sát hiệu suất hoặc Cấu hình từ xa.

Hệ thống phân cấp của các dự án Firebase:



Hình 2.6: *Mối quan hệ giữa các dự án, ứng dụng và sản phẩm Firebase*

(nguồn: firebase.google.com)

Sơ đồ hiển thị thứ bậc cơ bản của dự án Firebase, bao gồm dự án, các ứng dụng đã đăng ký cũng như các tài nguyên và dịch vụ được cung cấp. Sơ đồ này hiển thị hệ thống phân cấp cơ bản của dự án Firebase. Dưới đây là các mối quan hệ chính:

Dự án Firebase giống như một nơi chứa tất cả ứng dụng của bạn cũng như mọi tài nguyên và dịch vụ được cung cấp cho dự án.

Một dự án Firebase có thể có một hoặc nhiều ứng dụng Firebase được đăng ký với dự án đó (ví dụ: cả phiên bản iOS và Android của một ứng dụng hoặc cả phiên bản miễn phí và trả phí của một ứng dụng).

Tất cả ứng dụng Firebase đã đăng ký cùng chia sẻ dự án Firebase và có quyền truy cập vào tất cả các tài nguyên cũng như dịch vụ giống nhau được cung cấp cho dự án. Dưới đây là một số ví dụ:

- Tất cả các ứng dụng Firebase được đăng ký cho cùng một dự án Firebase đều có chung phần phụ trợ, như Dịch vụ lưu trữ Firebase, Xác thực, Cơ sở dữ liệu thời gian thực, Cloud Firestore, Cloud Storage và Cloud Functions.
- Tất cả ứng dụng Firebase được đăng ký cho cùng một dự án Firebase đều được liên kết với cùng một thuộc tính Google Analytics, trong đó mỗi Ứng dụng Firebase là một luồng dữ liệu riêng biệt trong thuộc tính đó.

2.6. Nền tảng sự dụng cho hệ thống IoT

Android là hệ điều hành di động phổ biến nhất thế giới, được phát triển bởi Google và sử dụng rộng rãi trên các thiết bị di động như điện thoại thông minh, máy tính bảng, đồng hồ thông minh, TV và các thiết bị IoT.

2.6.1. Giới thiệu nền tảng Android

Android được Google tự phát triển riêng cho đến khi những thay đổi và cập nhật đã hoàn thiện, khi đó mã nguồn mới được công khai. Sau đó, mã nguồn sẽ được cung cấp cho dự án nguồn mở Android (Android Open Source Project) (AOSP), một sáng kiến nguồn mở do Google dẫn đầu. Mã nguồn này, nếu không sửa đổi, chỉ chạy trên một số thiết bị, thường là thiết bị thuộc dòng Nexus. Có nhiều thiết bị có chứa những thành phần được giữ bản quyền do nhà sản xuất đặt vào thiết bị Android của họ.

- Giao diện: Google đã phát triển giao diện Android phiên bản 1.0 với sự hỗ trợ từ TAT, viết tắt từ The Astonishing Tribe, một công ty thiết kế tương tác của Thụy Điển. Dấu ấn rõ ràng nhất mà TAT để lại trên phiên bản Android từ phiên bản 1.0 cho đến 2.2 chính là widget đồng hồ kim nằm ngoài Home Screen tuy đơn giản nhưng rất đẹp mắt. Công ty này

sau đó ngừng hợp tác với Google và bị RIM mua lại để tập trung phát triển sản phẩm Blackberry cũng như nền tảng BBX sau này.

- Kho ứng dụng Android: Thật khó có thể tưởng tượng một chiếc smartphone mà không hề có kho ứng dụng, nhưng vào thời điểm Android mới ra mắt, gần như không có bất kì điện thoại nào có kho ứng dụng nào được tích hợp và chính Android đã mở đầu cuộc cách mạng ứng dụng di động này. Android Market trên G1 thời bấy giờ có rất ít ứng dụng và giao diện cực kỳ đơn giản, hơn nữa tính năng mua ứng dụng trên phiên bản này vẫn chưa được xuất hiện mãi cho đến năm sau - những vấn đề này dễ hiểu vì thời điểm này Android chỉ mới được khai sinh nên mọi thứ còn khá thô sơ.

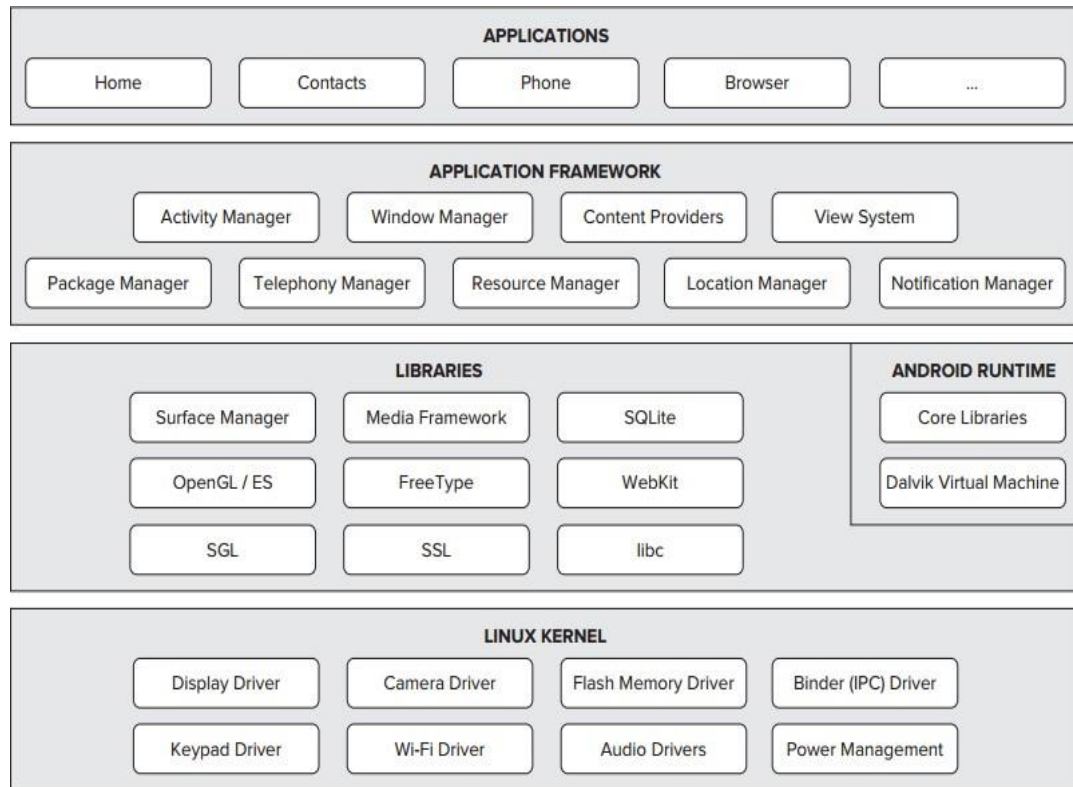
- Màn hình chính (Home Screen) và các widget: Một điểm khác biệt giữa Android so với các hệ điều hành khác là phần màn hình chính của mình. Bên cạnh việc thay đổi được hình nền, Android còn cho phép người dùng tùy biến màn hình chính của mình với nhiều widgets kèm theo, chẳng hạn như đồng hồ, lịch, trình nghe nhạc, đưa các icon ứng dụng ra ngoài hoặc thậm chí có thể can thiệp sâu hơn để thay đổi toàn bộ giao diện màn hình Home Screen.

- Đồng bộ và tích hợp chặt chẽ với Gmail: Vào thời điểm điện thoại G1 được bán ra, Gmail đã hỗ trợ giao thức POP và IMAP để tích hợp với các trình email trên di động. Tuy nhiên, lúc bấy giờ không có bất kì sản phẩm nào có thể hỗ trợ được hoàn toàn những tính năng ưu việt này của Gmail. Mãi cho đến khi Android 1.0 xuất hiện, vấn đề này đã được khắc phục và G1 trở thành chiếc điện thoại mang lại trải nghiệm Gmail tốt nhất trên thị trường lúc bấy giờ. Giao diện gmail của phiên bản Android thời kỳ đầu

- Thanh thông báo vuốt từ trên xuống (Notification bar): Ngay từ những ngày đầu tiên của Android, thanh thông báo này đã đánh dấu một bước quan trọng mà trước đây chưa hề có hệ điều hành nào làm được - đưa tất cả thông tin tin nhắn, tin thoại hoặc các cuộc gọi nhớ chỉ với thao tác vuốt xuống.

2.6.2. Kiến trúc Android

Android gồm 5 phần chính sau được chứa trong 4 lớp:



Hình 2.7: Hình ảnh minh họa các tầng kiến trúc android(nguồn: vncoder.vn)

- Tầng nhân linux (linux kernel):

Android dựa trên Linux phiên bản 2.6 cho hệ thống dịch vụ cốt lõi như security, memory management, process management, network stack, and driver model. Kernel Linux hoạt động như một lớp trừu tượng hóa giữa phần cứng và phần còn lại của phần mềm stack.

- Tầng Libraries và Android Runtime:

Thư viện Android bao gồm một tập hợp các thư viện C/C++ được sử dụng bởi nhiều thành phần khác nhau trong hệ thống Android. Điều này được thể hiện thông qua nền tảng ứng dụng Android. Một số các thư viện cơ bản được liệt kê dưới đây:

- + Hệ thống thư viện C: một BSD có nguồn gốc từ hệ thống thư viện tiêu chuẩn C (libc), điều chỉnh để nhúng vào các thiết bị dựa trên Linux.

- + Thư viện Media - dựa trên PacketVideo's OpenCORE; các thư viện hỗ trợ phát lại và ghi âm của âm thanh phổ biến và các định dạng video, cũng như các tập tin hình ảnh tĩnh, bao gồm cả MPEG4, H.264, MP3, AAC, AMR, JPG, PNG.
- + Thư viện 3D: một thực hiện dựa vào OpenGL ES 1.0 APIs, các thư viện sử dụng phần cứng tăng tốc 3D (nếu có), tối ưu hóa cao rasterizer phần mềm 3D.
- + FreeType: vẽ phông chữ bitmap và vector.

Android cũng bao gồm một bộ thư viện cung cấp hầu hết các chức năng của ngôn ngữ lập trình Java, bao gồm một số tính năng của ngôn ngữ Java 8 mà khung API Java sử dụng.

- Tầng Application Framework

- + Tầng này xây dựng bộ công cụ các phần tử ở mức cao để các lập trình viên có thể nhanh chóng xây dựng ứng dụng. Nó được viết bằng Java, có khả năng sử dụng chung để tiết kiệm tài nguyên.
- + Đây là một nền mở, có ưu điểm:
 - Với các hãng sản xuất điện thoại: có thể tùy biến để phù hợp với cấu hình điện thoại mà họ sản xuất như để có nhiều mẫu mã, style hợp thị hiếu người dùng. Vì thế nên tuy cùng chung nền tảng Android mà điện thoại của Google có thể khác hẳn với Motorola, HTC, T-Mobile, Samsung ...
 - Với lập trình viên: cho phép lập trình viên có thể sử dụng các API ở tầng trên mà không cần phải hiểu rõ cấu trúc bên dưới, tạo điều kiện cho lập trình viên tự do sáng tạo bởi vì chỉ cần quan tâm đến nội dung mà ứng dụng mà học làm việc. Một tập hợp API rất hữu ích được xây dựng sẵn như hệ thống định vị, các dịch vụ chạy nền, liên lạc giữa các ứng dụng, các thành phần giao diện cấp cao...
- + Giới thiệu một số thành phần của phần này:
 - *Activity Manager*: Quản lý các chu kỳ sống của một ứng dụng cũng như cung cấp công cụ điều khiển các Activity.
 - *Telephony Manager*: Cung cấp công cụ để thực hiện liên lạc như gọi điện thoại.

- *XMPP Service*: Cung cấp công cụ để liên lạc trong thời gian thực.
- *Location Manager*: Cho phép xác định vị trí điện thoại dựa vào hệ thống định vị toàn cầu GPS và Google Maps.
- *Window Manager*: Quản lý việc xây dựng và hiển thị các giao diện người dùng cũng như tổ chức quản lý các giao diện giữa các ứng dụng.
- *Resource Manager*: Quản lý tài nguyên tĩnh của các ứng dụng bao gồm các file hình ảnh, âm thanh, layout, string...
- *Notification Manager*: Quản lý việc hiển thị các thông báo (như báo có tin nhắn, có email mới...)

- Tầng Application

- + Tầng ứng dụng là tầng giao tiếp với người dùng, ví dụ: trên thiết bị Android như danh bạ, trình duyệt... mọi ứng dụng viết đều nằm trên tầng này. Giao diện hệ điều hành Android.
- + Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp, sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo giãn và thu lại để xử lý các đối tượng trên màn hình. Sự phản ứng với tác động của người dùng diễn ra gần như ngay lập tức, nhằm tạo ra giao diện cảm ứng mượt mà, thường dùng tính năng rung của thiết bị để tạo phản hồi rung cho người dùng.
- + Những thiết bị phần cứng bên trong như gia tốc kế, con quay hồi chuyển và cảm biến khoảng cách được một số ứng dụng sử dụng để phản hồi một số hành động khác của người dùng, ví dụ như điều chỉnh màn hình từ chế độ hiển thị dọc sang chế độ hiển thị ngang tùy theo vị trí của thiết bị, hoặc cho phép người dùng lái xe đua bằng xoay thiết bị, giống như đang điều khiển vô-lăng.

2.6.3. Cơ chế giao tiếp giữa thiết bị IoT và ứng dụng

Trong hệ thống này để giao tiếp và xử lý dữ liệu tôi sử dụng phần mềm lưu trữ Firebase Realtime Database làm trung gian giao tiếp giữa thiết bị IoT và ứng dụng Android. Dưới đây là quá trình giao tiếp:

- a. Thiết lập cơ sở dữ liệu thời gian thực Firebase:

Bắt đầu bằng việc tạo một dự án trên Firebase và thiết lập Realtime Database. Cơ sở dữ liệu này sẽ lưu trữ dữ liệu từ thiết bị IoT và ứng dụng Android có thể đọc và ghi dữ liệu vào đó.

b. Gửi dữ liệu từ thiết bị IoT lên Firebase:

Thiết bị IoT được lập trình để gửi dữ liệu của nó lên Firebase Realtime Database. Điều này có thể được thực hiện thông qua một giao thức truyền thông như WiFi, HTTP hoặc WebSocket.

Ví dụ: Tôi thiết lập một kết nối Internet mạng WiFi trong nhà có thiết bị IoT sử dụng vi điều khiển là ESP8266 NodeMCU, sau đó sử dụng thư viện Firebase để gửi dữ liệu (ví dụ: các giá trị cảm biến như nhiệt độ, độ ẩm) lên Realtime Database của Firebase.

c. Đọc dữ liệu trên ứng dụng Android:

Ứng dụng Android cần được lập trình để đọc dữ liệu từ Firebase Realtime Database và hiển thị nó cho người dùng trong thời gian thực. Điều này có thể được thực hiện bằng cách lắng nghe sự thay đổi của dữ liệu trên Firebase.

Ví dụ: Sử dụng thư viện Firebase SDK trong ứng dụng Android để đăng ký lắng nghe sự thay đổi của dữ liệu trên Realtime Database. Khi dữ liệu thay đổi, cập nhật giao diện người dùng để hiển thị thông tin mới nhất từ thiết bị IoT.

d. Xử lý sự kiện và tương tác:

Ứng dụng Android có thể phản ứng với dữ liệu từ thiết bị IoT thông qua Firebase Realtime Database. Điều này có thể bao gồm cả thông báo cho người dùng và thực hiện hành động tương ứng.

Ví dụ: Nếu dữ liệu từ thiết bị IoT vượt quá một ngưỡng được đặt trước (ví dụ: nhiệt độ quá cao), ứng dụng Android có thể thông báo cảnh báo cho người dùng hoặc thực hiện các hành động khẩn cấp, chẳng hạn như tắt thiết bị.

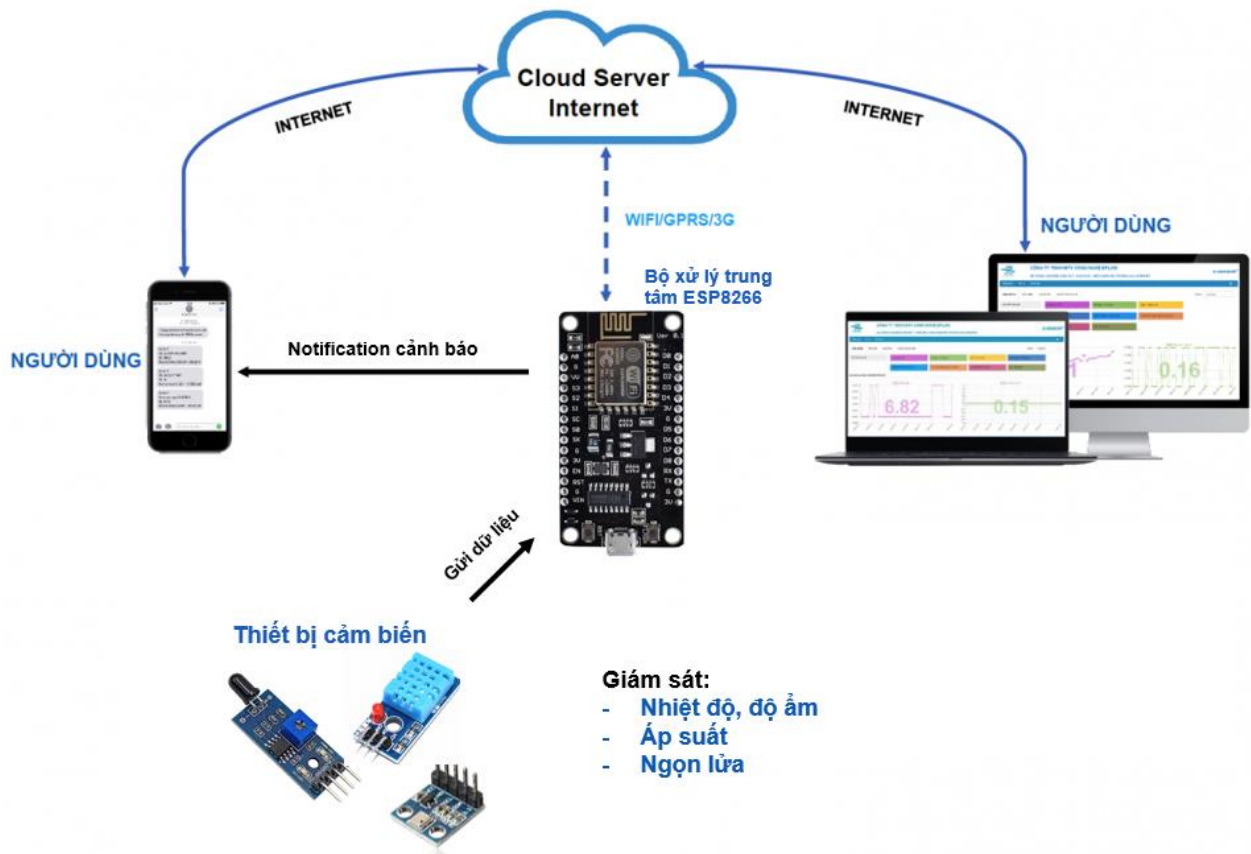
Tiểu kết chương 2:

Trong chương 2, chúng ta đã khám phá vai trò quan trọng của các thiết bị cảm biến trong hệ thống IoT, từ việc thu thập dữ liệu đến giám sát và điều khiển từ xa. Các thiết bị xử lý trung gian như vi điều khiển ESP8266 NodeMCU và các chuẩn giao tiếp không dây WiFi đã được phân tích kỹ lưỡng. Chúng ta cũng đã xem xét công nghệ lưu trữ đám mây Firebase và cơ chế giao tiếp giữa thiết bị IoT và ứng dụng Android. Những công nghệ và thiết bị này đóng góp đáng kể vào việc tạo ra các hệ thống IoT hiệu quả và thông minh.

Chương 3

THIẾT KẾ VÀ XÂY DỰNG MÔ HÌNH

3.1. Xây dựng hệ thống cảm biến giám sát ngôi nhà



Hình 3.1.1: Mô hình tổng quan hệ thống cảm biến giám sát ngôi nhà

3.1.1. Yêu cầu bài toán

Mục tiêu chính:

Xây dựng một hệ thống cảm biến giám sát để đảm bảo an toàn và tiện ích cho ngôi nhà. Theo dõi và cảnh báo về các sự kiện bất thường như cháy, sự cố về nhiệt độ, hoặc môi trường không khí không an toàn,...

Các chức năng cơ bản:

- Cảm biến cháy: Theo dõi mức độ khói hoặc nhiệt độ trong nhà để phát hiện sự cố cháy. Gửi cảnh báo ngay lập tức đến chủ nhà hoặc cơ quan cứu hỏa.

- Cảm biến môi trường: Đo lường các thông số môi trường như nhiệt độ, độ ẩm, ánh sáng, và chất lượng không khí. Cảnh báo về các điều kiện môi trường không an toàn hoặc không lành mạnh.

- Tính linh hoạt và tùy chỉnh: Hệ thống cần có khả năng mở rộng để thêm các cảm biến mới hoặc tính năng bổ sung. Người dùng có thể tùy chỉnh cài đặt và cảnh báo theo nhu cầu cụ thể của mỗi ngôi nhà.

- Tính tiện lợi và dễ sử dụng: Giao diện người dùng trực quan và dễ sử dụng trên ứng dụng di động hoặc trang web. Các cảnh báo và thông tin cần thiết được hiển thị một cách rõ ràng và dễ hiểu.

- Tích hợp với hệ thống điện thông minh: Có khả năng kết nối và tương tác với các hệ thống điều khiển nhà thông minh khác như đèn, quạt, và thiết bị điện gia đình khác để tăng cường tiện ích và tiết kiệm năng lượng.

- Chi phí phù hợp: Phải đảm bảo rằng chi phí triển khai và vận hành hệ thống không quá cao và phù hợp với người dùng cá nhân hoặc gia đình.

3.1.2. Lắp đặt phần cứng

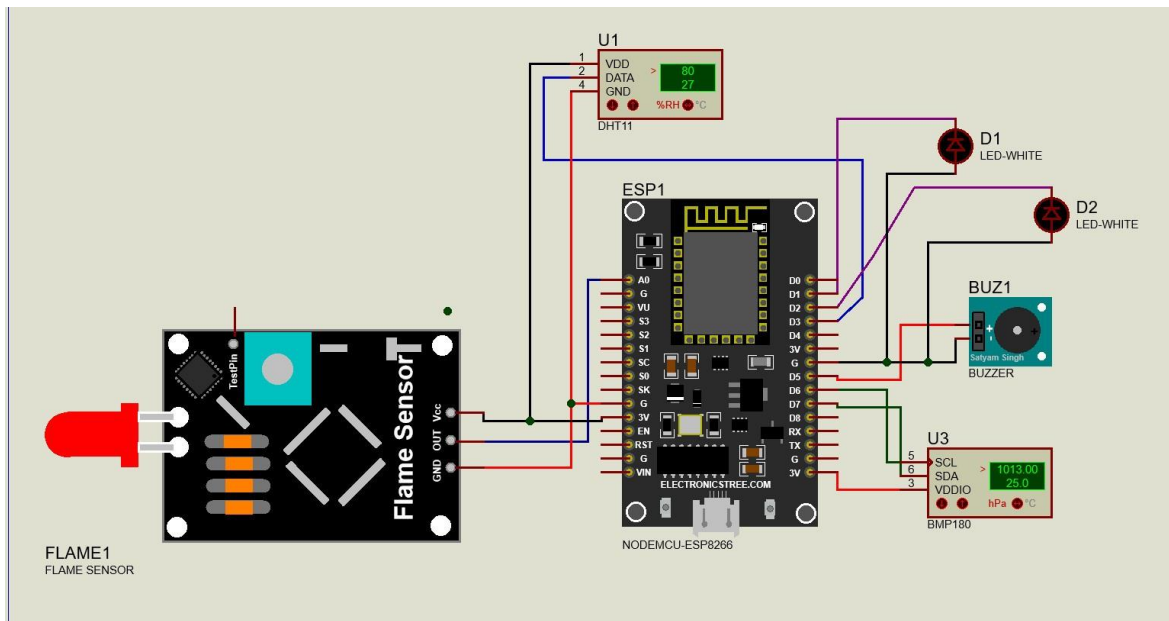
3.1.2.1. Thiết kế mạch điều khiển

Chuẩn bị các thiết bị cần thiết: 1 mạch kết nối Esp8166 NodeMCU, các module cảm biến: flame sensor, DHT11, BMP180, 2 đèn led, còi 5V.

ESP8266 NodeMCU:

- ESP8266 NodeMCU là một bo mạch phát triển dựa trên module ESP8266 Wi-Fi SoC.
- Nó tích hợp một vi điều khiển microcontroller ESP8266 và một module Wi-Fi, cho phép bạn kết nối với mạng Wi-Fi và giao tiếp với các thiết bị khác trên mạng.
- ESP8266 NodeMCU có thể được lập trình bằng Arduino IDE và hỗ trợ nhiều giao thức như TCP/IP, UDP, MQTT, HTTP, ...

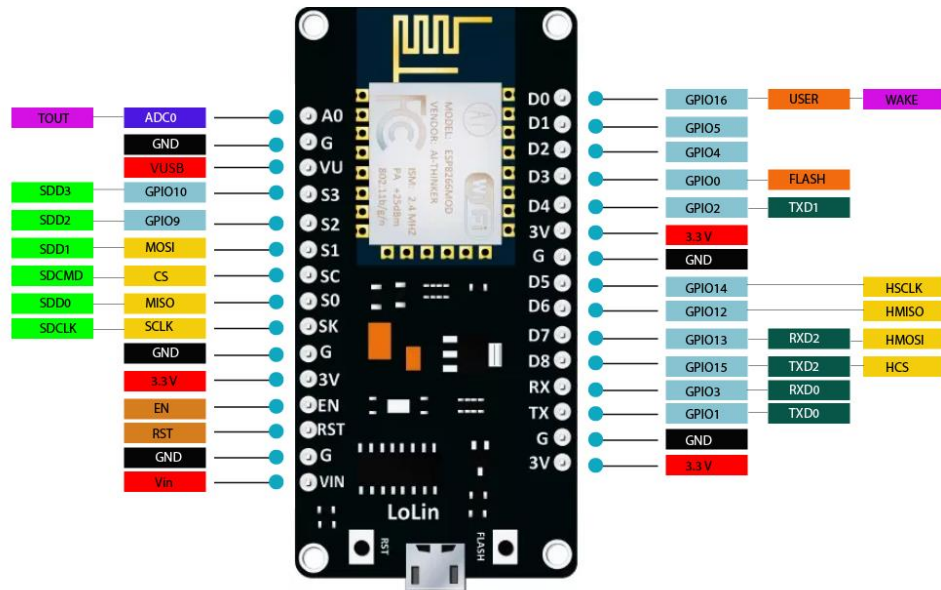
Các module cảm biến: Đọc các giá trị đo được từ môi trường .



Hình 3.1.2: Sơ đồ lắp đặt kết nối các thiết bị

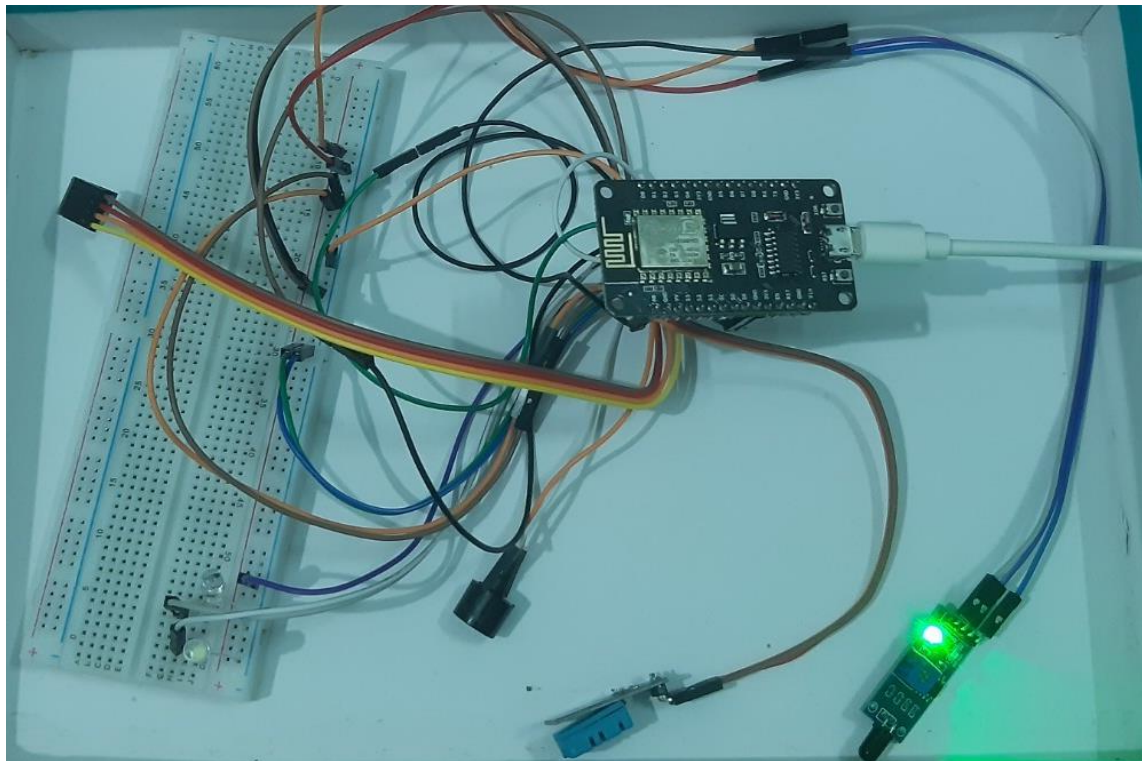
Sau khi có đủ các thiết bị cảm biến cần thiết và mạch kết nối, ta tiến hành kết nối các chân của thiết bị cảm biến vào bo mạch esp8266 như mô tả ở hình ảnh 3.1. Đảm bảo kết nối đúng để quá trình đọc, ghi, truyền nhận dữ liệu từ thiết bị lên phần mềm quản lý đúng đắn.

3.1.2.2. Lắp đặt thiết bị



Hình 3.1.3: Hình ảnh chi tiết cấu hình NodeMCU ESP8266 (nguồn: hub-plus.net)

Minh họa thực tế:



Hình 3.1.4: Hình ảnh thiết bị thực tế

3.1.3. Cách thức hoạt động của từng thành phần trong hệ thống

3.1.3.a. Cảm biến cháy (flame sensor)

Mô tả

- Cảm biến chuyên dùng để phát hiện lửa, thường dùng trong các hệ thống báo cháy. Tầm hoạt động trong khoảng 80cm với góc quét 60°.

- Cảm biến phát hiện lửa sử dụng một diode hồng ngoại thu để phát hiện ngọn lửa và đưa tín hiệu ra về bộ điều khiển.

- Cảm biến nhận biết được lửa tốt nhất với bước sóng 760nm – 1100nm. Mạch còn được tích hợp IC LM393 để so sánh tạo mức tín hiệu và có thể chỉnh được độ nhạy bằng biến trở.

Thông tin kỹ thuật cần lưu ý

- Điện áp hoạt động: 3.3 ~ 5.3 VDC
- Bước sóng phát hiện được: 760 ~ 1100 nm
- Góc quét: 0 – 60°
- Tín hiệu ra: có xuất ra 2 dạng tín hiệu: Analog (AOUT) và tín hiệu số (DOUT)
- Khoảng cách an toàn cho module là 80 cm và lớn hơn nếu ngọn lửa to hơn.

Nguyên lý hoạt động

Như đã biết mọi vật có nhiệt độ lớn hơn 0 độ K đều phát ra tia hồng ngoại nhưng ở các bước sóng khác nhau. Và ngọn lửa cũng không ngoại lệ, nó phát ra tia hồng ngoại ở dải 760nm-1100nm. Dựa vào điều này, module cảm biến phát hiện lửa dùng một diode hồng ngoại thu tín hiệu hồng ngoại ngọn lửa phát ra. Thông qua mạch tích hợp IC LM393 so sánh và đưa ra tín hiệu đầu ra.

Sơ đồ chân kết nối với ESP8266:

Module cảm biến phát hiện lửa	ESP8266
VCC (Nguồn dương)	5V
GND (Nguồn âm)	GND
OUT(Analog output)	A0

3.1.3.b. Cảm biến nhiệt độ và độ ẩm (DHT11)

Mô tả

- Cảm biến DHT11 bao gồm một phần tử cảm biến độ ẩm điện dung và một điện trở nhiệt để cảm nhận nhiệt độ. Tụ điện cảm biến độ ẩm có hai điện cực với chất nền giữ ẩm làm chất điện môi giữa chúng. Thay đổi giá trị điện dung xảy ra với sự thay đổi của các mức độ ẩm. IC đo, xử lý các giá trị điện trở đã thay đổi này và chuyển chúng thành dạng kỹ thuật số.

- Để đo nhiệt độ, cảm biến này sử dụng một nhiệt điện trở có hệ số nhiệt độ âm, làm giảm giá trị điện trở của nó khi nhiệt độ tăng. Để có được giá trị điện trở lớn hơn ngay cả đối với sự thay đổi nhỏ nhất của nhiệt độ, cảm biến này thường được làm bằng gốm bán dẫn hoặc polymer.

Thông tin kỹ thuật cần lưu ý

- Điện áp hoạt động: 3V - 5V DC
- Dòng điện tiêu thụ: 2.5mA
- Phạm vi cảm biến độ ẩm: 20% - 90% RH, sai số $\pm 5\%RH$
- Phạm vi cảm biến nhiệt độ: $0^{\circ}C \sim 50^{\circ}C$, sai số $\pm 2^{\circ}C$
- Tần số lấy mẫu tối đa: 1Hz (1 giây 1 lần)
- Kích thước: 23 * 12 * 5 mm

Nguyên lý hoạt động

Nguyên lý đo độ ẩm: Cảm biến DHT11 đo độ ẩm dựa trên sự thay đổi điện dung của một chất polyme đặc biệt nằm giữa hai tấm dẫn điện. Khi độ ẩm môi trường thay đổi, lượng nước hấp thụ vào chất polyme thay đổi, làm thay đổi điện dung, từ đó cảm biến sẽ đọc và tính toán giá trị độ ẩm.

Nguyên lý đo nhiệt độ: DHT11 sử dụng một điện trở nhiệt (thermistor) để đo nhiệt độ. Điện trở của thermistor thay đổi theo nhiệt độ môi trường, và cảm biến sẽ đọc sự thay đổi này để tính toán giá trị nhiệt độ.

Sơ đồ chân kết nối với ESP8266:

Module cảm biến phát hiện lửa	ESP8266
VCC (Nguồn dương)	5V
GND (Nguồn âm)	GND
DATA(Analog output)	D3

3.1.3.c. Cảm biến đo áp suất không khí và nhiệt độ (BMP180)

Mô tả

BMP180 là một trong những cảm biến của dòng BMP XXX được thiết kế để đo áp suất khí quyển. BMP180 là cảm biến có độ chính xác cao được thiết kế cho các ứng dụng tiêu dùng. Áp suất khí quyển là gì ngoài trọng lượng của không khí tác dụng lên mọi thứ. Không khí có trọng lượng và bất cứ nơi nào có không khí đều cảm nhận được áp suất của nó. Cảm biến BMP180 cảm nhận được áp suất đó và cung cấp thông tin đó ở đầu ra kỹ thuật số. Ngoài ra, nhiệt độ ảnh hưởng đến áp suất, vì vậy chúng ta cần đọc áp suất bù nhiệt độ. Bù lại, BM180 cũng có cảm biến nhiệt độ tốt.

Thông tin kỹ thuật cần lưu ý

- Phạm vi đo (Range):
 - + Áp suất: 300 hPa đến 1100 hPa (tương đương độ cao từ -500 m đến 9000 m).
 - + Nhiệt độ: -40°C đến +85°C.
- Độ chính xác (Accuracy):
 - + Áp suất: ± 1 hPa.
 - + Nhiệt độ: ± 1 °C.
- Độ phân giải (Resolution):
 - + Áp suất: 0.01 hPa (1 Pascal).
 - + Nhiệt độ: 0.1°C.
- Điện áp hoạt động (Operating Voltage): 1.8V đến 3.6V DC.
- Giao diện truyền thông (Communication Interface): Giao tiếp I2C hoặc SPI (tùy cấu hình).

Nguyên lý hoạt động

- Nguyên lý hoạt động của cảm biến áp suất khí quyển rất đơn giản, hoạt động dựa trên trọng lượng không khí. Bởi vì không khí xung quanh luôn có một trọng lượng nhất định và trọng lượng này có áp suất riêng.

- Áp suất này được cảm nhận bởi cảm biến áp suất khí quyển BMP 180. Nó có bốn thành phần cơ bản, thứ nhất là cảm biến điện trở piezo, thứ hai là bộ chuyển đổi analog sang digital, thứ ba là khối điều khiển với bộ nhớ EPROM và thứ tư là giao diện serial I2C.

- Khi trọng lượng hoặc áp suất không khí bị thay đổi thì điện trở của cảm biến sẽ thay đổi. Vì giá trị thay đổi này là giá trị analog do đó được chuyển đổi thành giá trị digital thông qua bộ chuyển đổi analog sang digital.

- Sau đó giá trị này được điều khiển bằng bộ điều khiển và cuối cùng được gửi tới bộ điều khiển bất kỳ với giao thức serial I2C.

Sơ đồ chân kết nối với ESP8266:

Module cảm biến phát hiện lửa	ESP8266
VCC (Nguồn dương)	5V
GND (Nguồn âm)	GND
SDA (Serial Data)	D6
SCL (Serial Clock)	D5

3.1.3.d. Hệ thống quản lý (ESP8266 NodeMCU)

Thông số kỹ thuật cần lưu ý

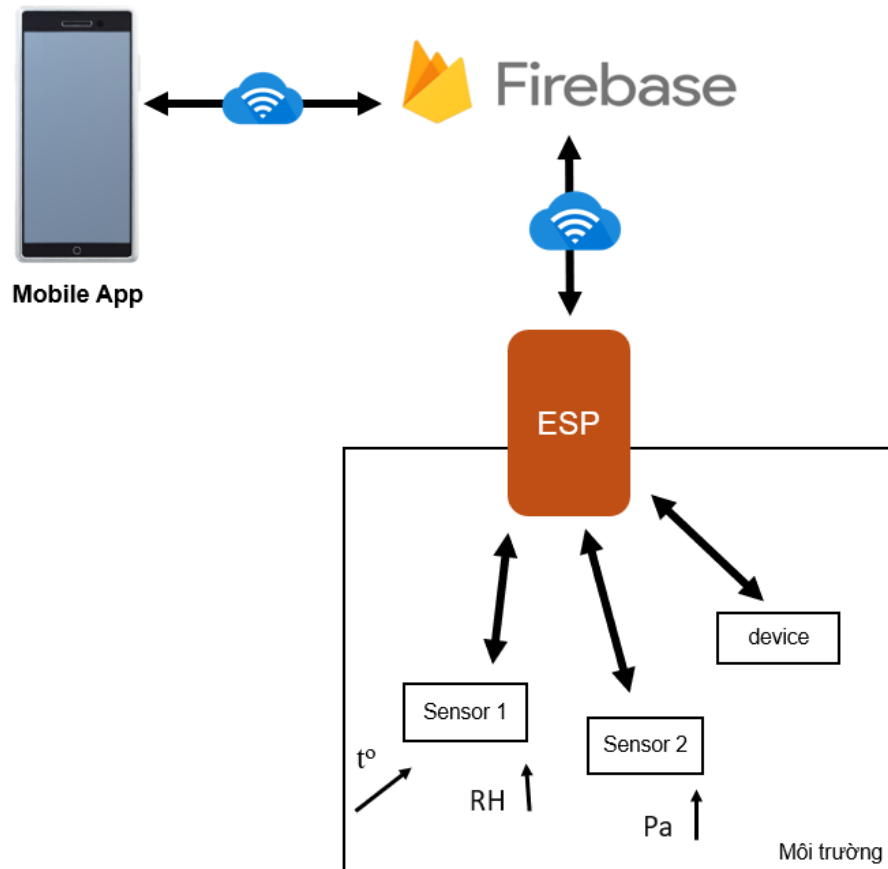
- Điện áp hoạt động: 3.3V
- Giao diện: UART, SPI, I2C
- Tốc độ truyền dữ liệu: Lên đến 1 Mbps
- Phạm vi phủ sóng Wi-Fi: 150 m (không có vật cản)

Kết nối chân ESP8266 với thiết bị

- Chân GPIO5 nối với chân dương của đèn led1.
- Chân GPIO4 nối với chân dương của đèn led2.

- Chân GPIO0 nối với chân out của cảm biến DHT.
- Chân GPIO14 nối với chân dương của Buzzer.
- Chân ADC0 nối với chân A0 của Flame sensor.
- Chân GPIO14 nối với chân SCL của cảm biến BMP180.
- Chân GPIO12 nối với chân SDA của cảm biến BMP180.
- Chân 3.3V nối với chân VCC của Flame sensor.
- Chân 3.3V nối với chân dương của DHT.
- Chân GND nối với chân âm của các thiết bị.

3.1.4. Luồng hoạt động từ cảm biến đến ứng dụng



Hình 3.1.5: Sơ đồ luồng hoạt động từ thiết bị đến ứng dụng

Sau khi kết nối các cảm biến với vi điều khiển ESP8266 như đã mô tả ở phần trước, hệ thống hoạt động như sau:

1. Kết nối và thu thập dữ liệu:

- Cảm biến DHT11 (đo nhiệt độ và độ ẩm), Flame Sensor (dò lửa), và BMP180 (đo áp suất không khí) được kết nối với ESP8266 qua các chân GPIO.
- ESP8266 đọc dữ liệu từ các cảm biến này thông qua giao thức I2C, GPIO, hoặc các giao thức phù hợp khác tùy theo loại cảm biến.

2. Xử lý và gửi dữ liệu:

- ESP8266 thu thập và xử lý dữ liệu cảm biến (bao gồm nhiệt độ, độ ẩm, trạng thái lửa, và áp suất không khí).
- Sử dụng thư viện Firebase, ESP8266 gửi dữ liệu này lên Firebase Realtime Database. Dữ liệu được chuẩn hóa và chuyển đổi thành định dạng phù hợp trước khi gửi qua kết nối Wi-Fi.

3. Lưu trữ và truy xuất dữ liệu:

- Firebase Realtime Database lưu trữ dữ liệu cảm biến một cách có cấu trúc và có tổ chức, cho phép truy xuất và đồng bộ hóa dữ liệu một cách hiệu quả.

4. Hiển thị và tương tác:

- Ứng dụng di động kết nối với Firebase thông qua API của Firebase, nhận dữ liệu thời gian thực từ cơ sở dữ liệu.
- Ứng dụng hiển thị các thông số từ cảm biến (như nhiệt độ, độ ẩm, trạng thái lửa, và áp suất) cho người dùng.
- Nếu người dùng thực hiện các thao tác trên ứng dụng (như thay đổi khoảng nhiệt độ, điều khiển thiết bị), yêu cầu này được gửi lên Firebase Realtime Database.

5. Tác động ngược lại thiết bị:

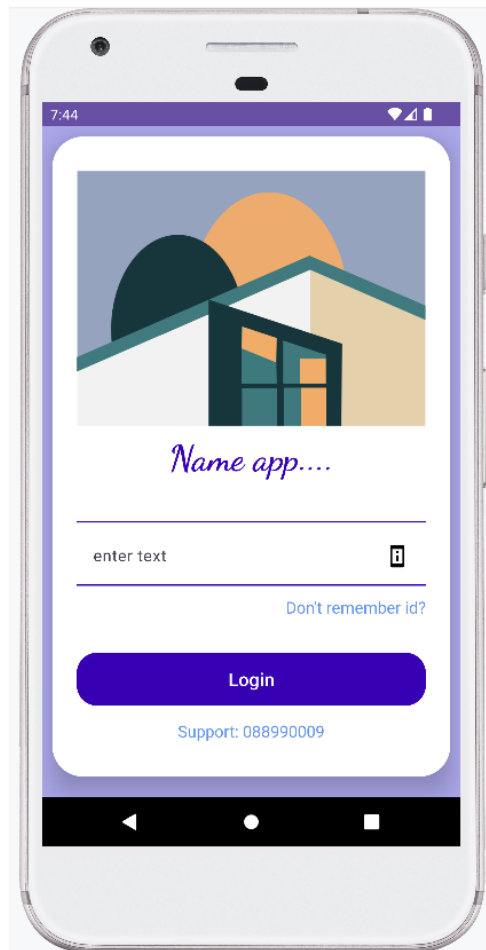
- ESP8266 liên tục theo dõi Firebase Realtime Database để nhận lệnh mới từ người dùng.
- Khi nhận được lệnh, ESP8266 điều chỉnh trạng thái của thiết bị hoặc thực hiện hành động tương ứng (ví dụ: bật/tắt thiết bị, thay đổi khoảng nhiệt độ).

Hệ thống này tạo ra một vòng lặp liên tục và tự động giữa việc thu thập dữ liệu cảm biến, xử lý và gửi dữ liệu lên đám mây, hiển thị thông tin cho người dùng, và thực hiện các lệnh điều khiển từ ứng dụng di động. Điều này đảm bảo rằng ngôi nhà được giám sát và quản lý hiệu quả, tăng cường sự an toàn và tiện ích cho người dùng.

3.2. Thiết kế phần mềm

3.2.1. Lập trình ứng dụng người dùng

Giao diện đăng nhập



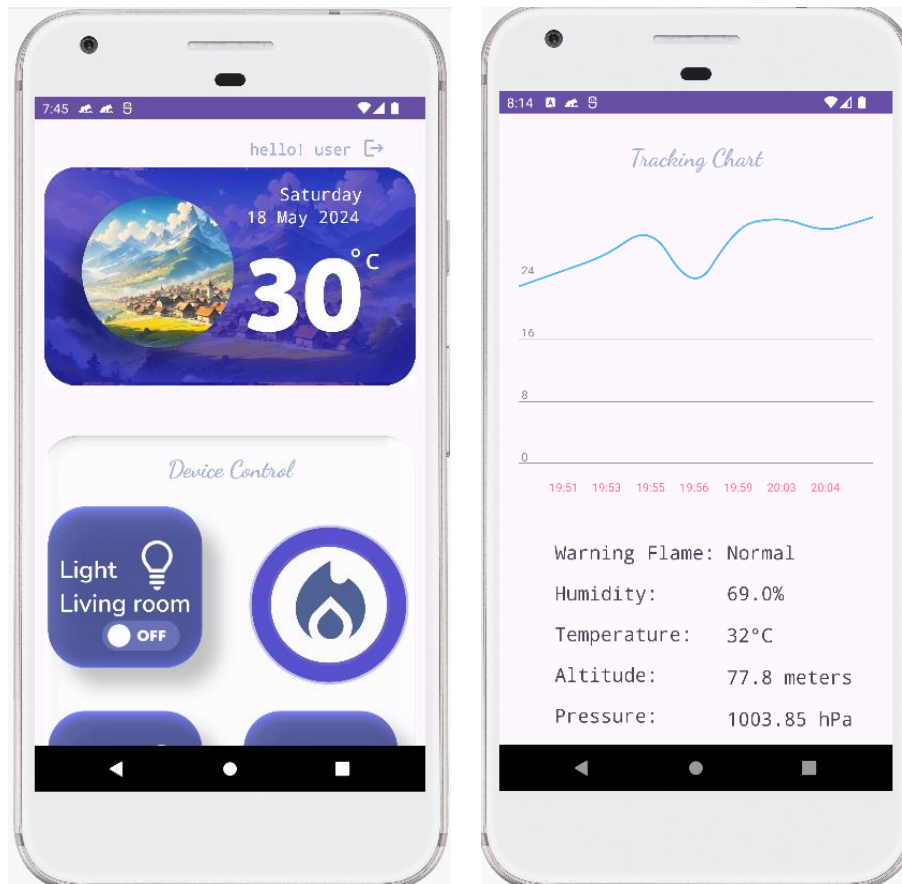
Hình 3.2.1: Hình ảnh giao diện đăng nhập ứng dụng

- Sau khi lắp đặt thiết bị giám sát cảm biến biến động cho ngôi nhà và tải ứng dụng của nhà phát triển. Khách hàng được cấp một mã ID riêng của thiết bị đọc cảm biến.

- Người dùng đăng nhập ứng dụng bằng việc nhập mã ID được cấp bởi nhân viên lắp đặt.
- Nếu mã đúng ứng dụng thay đổi vào giao diện màn hình chính hiển thị thông báo thành công. Nếu sai hiển thị thông báo mã ID không tồn tại.
- Tiếp đó khi đăng nhập thành công thiết bị sẽ lưu thông tin đăng nhập. Khi người dùng tắt ứng dụng và mở lại, hệ thống vẫn hiển thị màn hình chính và không phải đăng nhập.

Giao diện màn hình chính

- Đăng nhập thành công, khách hàng sẽ vào màn hình chính có giao diện như hình ảnh bên dưới. Màn hình này sẽ hiển thị các chỉ số đọc được từ môi trường ngôi nhà của bạn thông qua các thiết bị cảm biến nhân viên đã lắp đặt .



Hình 3.2.2: Hình ảnh giao diện màn hình chính ứng dụng

- Màn hình chính ứng dụng thực hiện hiển thị chỉ số đo được từ môi trường của ngôi nhà người sử dụng. Từ đó, giúp người sử dụng có thể theo dõi và ghi nhận các thay đổi nhiệt độ một cách liên tục và chính xác, cung cấp dữ liệu cần thiết để điều chỉnh và kiểm soát các quy trình nhiệt độ. Biết biến động từ môi trường xung quanh như chỉ số nhiệt độ, độ ẩm, áp suất,... đưa ra cảnh báo nhắc nhở cần thiết, kịp thời.

- Ngoài ghi nhận sự thay đổi của các chỉ số, ứng dụng có thể kiểm soát thiết bị trong ngôi nhà của khách hàng. Hiện tại có thể thực hiện kiểm soát cơ bản như điều khiển đèn. Và hiển thị lịch sử thay đổi của nhiệt độ qua các khoảng thời gian.

- Trong biểu đồ lịch sử biến động nhiệt độ, hệ tọa độ trục số được sắp xếp như sau:

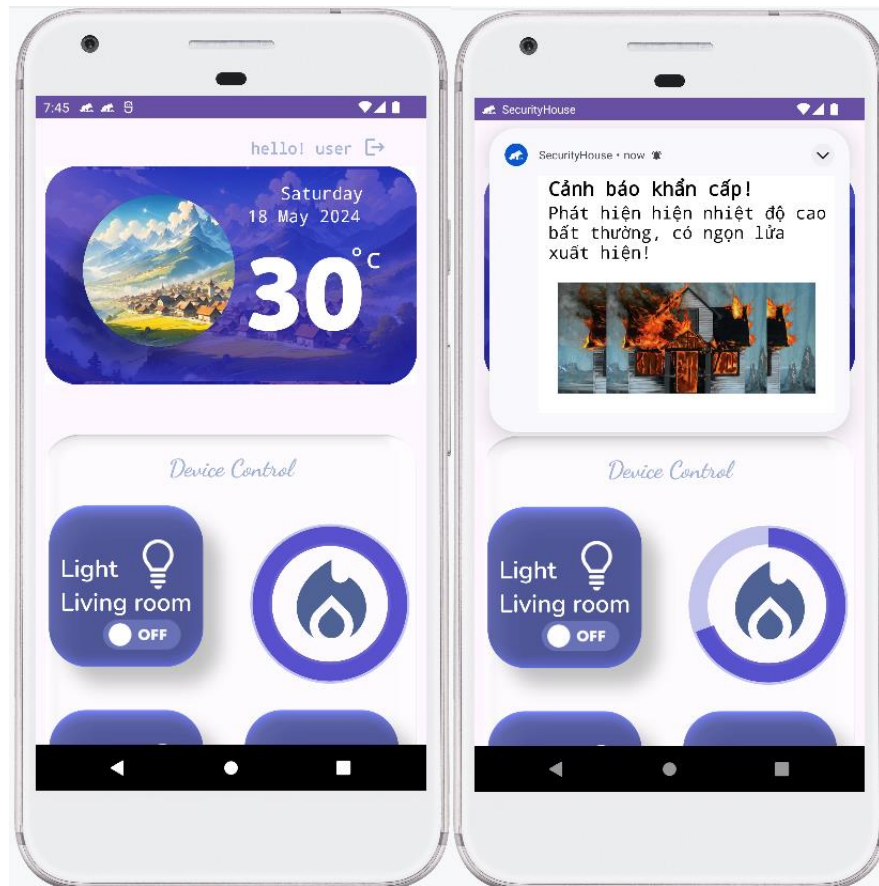
- + Trục tung (y), bên trái của đồ thị, các điểm được đánh số thể hiện các mức nhiệt độ, từ 0 đến 24 độ Celsius hoặc hơn, tượng trưng cho các điểm nhiệt độ từ môi trường.
- + Trục hoành (x), nằm dưới đồ thị, phản ánh thời gian, từng khoảng khắc khi nhiệt độ biến đổi. Khi hai trục này kết hợp, mỗi điểm giao nhau tạo ra một hình ảnh sống động về sự biến đổi nhiệt độ theo thời gian, cung cấp cái nhìn toàn diện và dễ hiểu về xu hướng và biến động của nhiệt độ qua thời gian.

Gửi thông báo

- Nếu thiết bị đo được khoảng nhiệt độ biến động bất thường quá cao hoặc phát hiện có xuất hiện ngọn lửa, ứng dụng sẽ đưa ra cảnh báo đến người dùng bằng việc gửi một thông báo cho người dùng biết.

- Hình 3.7 bên trái hiển thị trạng thái bình thường của ứng dụng, sau khi cảm biến (thiết bị cảm biến ở đây là cảm biến cháy) nhận được sự thay đổi bất thường từ môi trường và gửi thông điệp tới điện thoại qua kết nối internet.

- Khi nhận được dữ liệu từ thiết bị cảm biến căn cứ vào sự thay đổi dữ liệu của thiết bị trong khoảng cho phép ở đây cảnh báo chạy sẽ hoạt động nếu chỉ số xuống dưới 700nm và ứng dụng hiển thị lên một thông báo cảnh báo đến người dùng ở hình 3.7 bên phải.



Hình 3.2.3: Hình ảnh giao diện màn hình thông báo ứng dụng

3.2.2. Lập trình đọc dữ liệu từ cảm biến

Để có thể ghi, nhận được chỉ số của các thiết bị cảm biến thông qua module MCU ESP8266 và gửi chỉ số lên cơ sở lưu trữ dữ liệu firebase, ta cần phải thiết kế code để nạp vào trong module MCU ESP8266 được kết nối với cảm biến theo sơ đồ nối dây ở phần 3.1 bên trên. Ở đây, chúng ta sử dụng phần mềm Arduino IDE để có thể thiết kế, cũng như nạp code vào trong module MCU ESP8266 để thực hiện chương trình.

Đầu tiên, ta khai báo các thư viện cần dùng.

```
#include <ArduinoJson.h>
#include <ESP8266Firebase.h>
#include <ESP8266WiFi.h>
#include <Adafruit_BMP085.h>
#include <DHT.h>
#include <Wire.h>
```

Thư viện	Mô Tả
ArduinoJson.h	Thư viện này hỗ trợ việc phân tích cú pháp (parsing), tạo và sửa đổi các đối tượng JSON. Thư viện này rất hữu ích trong việc xử lý dữ liệu từ các API web hoặc từ các thiết bị IoT khác.
ESP8266Firebase.h	Thư viện này giúp bạn kết nối và tương tác với Firebase Realtime Database từ vi điều khiển ESP8266. Thư viện này cho phép gửi và nhận dữ liệu từ Firebase, hỗ trợ các ứng dụng IoT cần lưu trữ hoặc đồng bộ dữ liệu với đám mây
ESP8266WiFi.h	Thư viện này cung cấp khả năng kết nối Wi-Fi cho các board ESP8266. Đây là thư viện cơ bản và cần thiết để ESP8266 có thể giao tiếp với Internet.
Adafruit_BMP085.h	Thư viện này hỗ trợ cảm biến áp suất và nhiệt độ BMP085 của Adafruit. BMP085 là một cảm biến áp suất khí quyển chính xác, có khả năng đo cả nhiệt độ môi trường.
DHT.h	Thư viện này được sử dụng để giao tiếp với các cảm biến DHT (DHT11, DHT22, AM2302), vốn là các cảm biến đo nhiệt độ và độ ẩm.
Wire.h	Thư viện Wire là thư viện chuẩn của Arduino cho giao tiếp I2C (Inter-Integrated Circuit).

Tiếp theo khai báo các chân cắm theo sơ đồ vẽ, mỗi chân điều khiển một thiết bị cảm biến riêng.

```
#define device1 5 // D1 - Led stairs
#define device2 4 // D2 - Led living room
#define device3 12 // D6 - SCL(Bmp180)
#define device4 13 // D7 - SDA(Bmp180)
#define dht11 0 // D3 - Temperature & Humidity
#define Buzzer 14 // D5 - Bip
#define FlameSensor A0 // A0 - Fire warning
```

Trong hàm setup(), ta khởi chạy baudrate (tốc độ truyền) là 115200 với mục đích gỡ lỗi và kết nối với database. Và khởi tạo cảm biến DHT, BMP180, Flame sensor, WiFi bằng phương thức begin() và firebase bằng phương thức json().

```
void setup() {
  Serial.begin(115200);
  pinMode(device1, OUTPUT);
  pinMode(device2, OUTPUT);
  pinMode(device3, INPUT);
  pinMode(device4, INPUT);
  pinMode(dht11, INPUT);
  pinMode(Buzzer, OUTPUT);
  pinMode(FlameSensor, INPUT);
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, LOW);
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(1000);

  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(1000);
  // Connect to WiFi
  Serial.print("Connecting to: " _SSID);
  WiFi.begin(_SSID, _PASSWORD);
  > while (WiFi.status() != WL_CONNECTED) { ...
  }
  Serial.println("");
  Serial.println("WiFi Connected");

  firebase.json(true); // Make sure to add t
  dht.begin();
  Wire.begin(13, 12); // SDA pin D1, SCL pin
  > if (!bmp.begin()) { ...
  ,
```

Việc đọc các chỉ số cảm biến, tạo các hàm và cho vào hàm loop()

```
void loop() {
    Controlled();
    FireWarning();
    Temp_Humi();
    BMP180();
    //delay(100);
}
```

Ví dụ code đọc cảm biến Flame sensor:

```
void FireWarning(){
    uint32_t chipId = ESP.getChipId();

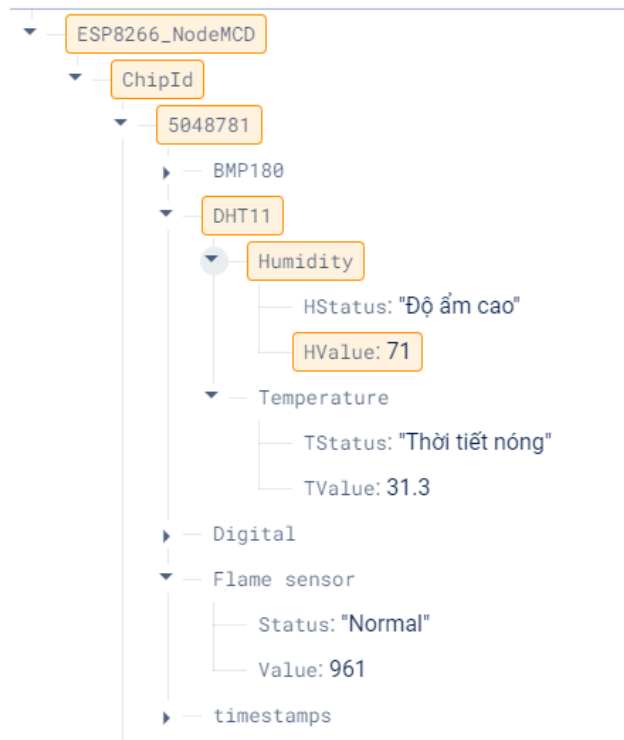
    // Đọc trạng thái của cảm biến cháy
    int flameSensorValue = analogRead(FlameSensor);
    Serial.print("Giá trị cảm biến cháy: ");
    Serial.println(flameSensorValue);
    firebase.setInt("ESP8266_NodeMCD/ChipId/" + String(chipId) + "/Flame sensor/Value", flameSensorValue);
}
> if(flameSensorValue <= NORMAL_THRESHOLD){...
}
else{
    Serial.println("Bình thường");
    firebase.setString("ESP8266_NodeMCD/ChipId/" + String(chipId) + "/Flame sensor/Status", "Normal");
}
}
```

Sau khi chạy chương trình, nạp code vào modul MCU ESP8266, ta thu được kết quả trên màn hình Serial Monitor.

```
23:45:22.490 -> Connecting to: anhtruong.....
23:45:26.714 -> WiFi Connected
23:45:26.714 -> IP Address: http://192.168.1.105/
23:45:33.618 -> Giá trị cảm biến cháy: 1021
23:45:35.316 -> Bình thường
23:45:43.797 -> Humidity: 70.00 %
23:45:43.797 -> Humidity status: Độ ẩm cao
23:45:43.836 -> Temperature: 31.30 *C
23:45:43.836 -> Temperature status: Thời tiết nóng
23:45:50.840 ->
23:45:50.840 -> Temperature: 30.50 C
23:45:50.840 -> Absolute Pressure: 1003.96 hPa
23:45:50.840 -> Altitude: 77.89 meters
23:45:50.840 -> Sea Level Pressure: 1003.97 hPa
23:45:50.840 ->
```

Hình 3.2.4: Hình ảnh đọc dữ liệu từ môi trường ra Serial Arduino IDE

Và từ MCU ESP8266 đã gửi thành công lên cơ sở dữ liệu Firebase.



Hình 3.2.5: Hình ảnh lưu trữ dữ liệu trên firebase

3.3. Hướng dẫn sử dụng cho người dùng

Hướng dẫn sử dụng đến người dùng cuối

Bước 1: Đăng nhập và truy cập vào ứng dụng

Đăng nhập vào ứng dụng bằng mã được cung cấp. Mở ứng dụng hoặc truy cập vào trang web được cung cấp để theo dõi và quản lý hệ thống giám sát.

Bước 2: Xem dữ liệu từ các cảm biến

Theo dõi nhiệt độ và độ ẩm Trên giao diện chính của ứng dụng/trang web, bạn sẽ thấy các giá trị hiện tại của nhiệt độ và độ ẩm được đo bởi cảm biến DHT11. Các giá trị này sẽ cập nhật thời gian thực và cho phép bạn biết được điều kiện môi trường trong gia đình.

Kiểm tra Áp suất không khí Nếu có cảm biến BMP180, bạn cũng có thể xem giá trị áp suất không khí hiện tại. Điều này sẽ giúp bạn theo dõi các thay đổi về thời tiết hoặc áp suất trong không khí. Cảnh báo từ Cảm biến Ngọn lửa (Flame Sensor) - Nếu hệ thống được trang bị cảm biến ngọn lửa, bạn sẽ nhận được cảnh báo ngay lập tức nếu có phát hiện ngọn lửa. Đảm bảo rằng bạn luôn kiểm tra thông báo này để có thể phản ứng kịp thời trong trường hợp khẩn cấp.

Bước 3: Quản lý và Cài Đặt

Cài đặt cảnh báo: Trong phần cài đặt của ứng dụng, bạn có thể thiết lập các cảnh báo cho nhiệt độ, độ ẩm và áp suất không khí. Chọn ngưỡng mà bạn muốn được cảnh báo khi các giá trị vượt quá hoặc thấp hơn.

Theo dõi lịch sử và thống kê: Ngoài việc xem dữ liệu hiện tại, bạn có thể truy cập vào lịch sử và thống kê để xem các biến động của môi trường trong gia đình qua thời gian. Điều này giúp bạn đánh giá được xu hướng và điều chỉnh các điều kiện môi trường để đảm bảo an toàn và tiện nghi.

Bước 4: Bảo Trì và Cập Nhật

Kiểm tra và Bảo trì thiết bị :Thường xuyên kiểm tra hoạt động của thiết bị và các cảm biến để đảm bảo chúng hoạt động đúng cách. Đảm bảo rằng thiết bị và cảm biến không bị che khuất hay bị hư hỏng.

Cập Nhật Firmware: Đảm bảo rằng thiết bị có phiên bản firmware mới nhất để bảo đảm tính ổn định và bảo mật của hệ thống. Kiểm tra và cập nhật firmware theo hướng dẫn từ nhà sản xuất hoặc từ ứng dụng/trang web.

Bước 5: Hỗ trợ và Liên hệ

Hỗ trợ Kỹ thuật: Nếu bạn gặp vấn đề hoặc cần hỗ trợ kỹ thuật, hãy liên hệ với nhà cung cấp thiết bị hoặc nhóm hỗ trợ kỹ thuật. Đảm bảo rằng bạn có thông tin liên lạc cần thiết để có thể giải quyết các vấn đề kịp thời.

Liên hệ Nhà cung cấp: Nếu có câu hỏi về sử dụng, bảo trì hoặc cập nhật của thiết bị, liên hệ với nhà cung cấp để được hỗ trợ chi tiết hơn.

Lưu ý quan trọng

- Luôn tuân thủ các hướng dẫn sử dụng và an toàn từ nhà sản xuất.
- Đảm bảo rằng bạn đã hiểu rõ các chức năng và tính năng của thiết bị để sử dụng một cách hiệu quả và an toàn.
- Nếu có sự cố hoặc vấn đề không thể giải quyết được, ngay lập tức liên hệ với nhà cung cấp hoặc kỹ thuật viên có kinh nghiệm để giải quyết.
- Việc thực hiện các bước và hướng dẫn trên sẽ giúp người dùng cuối có thể sử dụng và quản lý hệ thống giám sát an toàn gia đình dựa trên công nghệ IoT một cách dễ dàng và hiệu quả.

Tiểu kết chương 3:

Trong chương này, chúng ta đã thảo luận về quá trình thiết kế và xây dựng hệ thống cảm biến giám sát ngôi nhà. Bắt đầu từ việc xác định yêu cầu bài toán, chúng ta đã tiến hành lắp đặt phần cứng cần thiết và phân tích cách thức hoạt động của từng thành phần trong hệ thống. Về mặt phần mềm, chúng ta đã thiết kế và lập trình ứng dụng người dùng cũng như lập trình để đọc dữ liệu từ các cảm biến. Những bước này đảm bảo rằng hệ thống hoạt động một cách hiệu quả, cung cấp giải pháp giám sát nhà thông minh toàn diện.

Chương 4:**KẾT QUẢ VÀ ĐÁNH GIÁ****4.1. Kết quả thực hiện**

Sau 2 tháng thực hiện đề tài, tôi đã hoàn thành việc thiết kế và thi công mô hình giám sát bảo đảm an toàn cho gia đình dựa trên kỹ thuật IoT được điều khiển và kiểm soát từ xa bằng điện thoại qua thiết bị smartphone, hệ thống đã được kiểm tra thử và chạy ổn định.

Ưu điểm:

- + Phần cứng được thiết kế đơn giản, sử dụng số linh kiện tối thiểu, kết nối chân ra đáp ứng nhu cầu của phát triển sau này của đồ án.
- + Hệ thống sử dụng ứng dụng điện thoại để quản lý dễ dàng theo dõi và gửi cảnh báo đến người dùng
- + Hệ thống khá dễ thi công và lắp đặt.
- + Thiết bị rẻ và dễ thay thế nếu hỏng.

Nhược điểm:

- + Hệ thống chỉ dùng được ở những nơi có phủ sóng mạng internet.
- + Hệ thống có phạm vi hoạt động không lớn.
- + Các cảm biến biến động DHT và BMP180 có khả năng sai số, chưa thực sự chính xác.
- + Việc nhận và truyền dữ liệu từ thiết bị có độ trễ nhất định.
- + Thiết bị được cấp nguồn điện trực tiếp không sử dụng nguồn điện dự phòng.
- + Thiết bị lắp chưa gọn.

4.2. Đánh giá

Hệ thống hoạt động tốt, đúng, đủ yêu cầu đặt ra. Về kỹ thuật và ứng dụng, mô hình hoạt động chính xác. Song còn một số điểm hạn chế trong việc đọc ghi truyền dữ liệu lên cơ sở dữ liệu.

Tiểu kết chương 4:

Chương 4 nhận xét kết quả hệ thống đạt được. Hệ thống đã được thiết kế và triển khai thành công, cung cấp khả năng giám sát và điều khiển từ xa cho người dùng thông qua ứng dụng điện thoại. Tuy nhiên, còn một số hạn chế và điểm cần cải thiện, như khả năng hoạt động chỉ trong phạm vi mạng internet và sự không chính xác của một số cảm biến. Điều này cho thấy còn tiềm năng để phát triển và hoàn thiện hơn trong tương lai.

KẾT LUẬN

Đồ án “Xây dựng thiết bị giám sát bảo đảm an toàn cho gia đình dựa trên kỹ thuật IoT” đã xây dựng được một mô hình khá chi tiết và đảm bảo phương thức vận hành an toàn trong tình trạng làm việc bình thường cũng như xảy ra sự cố. Quá trình thực hiện đồ án đã giúp tôi hiểu rõ hơn về những vấn đề đã được học đặc biệt quá trình tính toán, tổng hợp kết quả để lựa chọn thiết bị phù hợp nhất cho toàn hệ thống. Những kiến thức đó chắc chắn sẽ giúp tôi rất nhiều trong quá trình làm việc thực tế sau này.

Kết quả đạt được:

Sau thời gian nghiên cứu và tìm tòi làm đồ án, hệ thống cảm biến giám sát môi trường ngôi nhà đáp ứng được những yêu cầu mà người dùng. Về tiêu chí, hoạt động được, dễ sử dụng, các chức năng cần thiết như đăng nhập thiết bị, hiển thị các chỉ số cảm biến, biểu đồ biến động nhiệt độ, đưa ra các thông báo cảnh báo, nhắc nhở đến người dùng. Ứng dụng xây dựng và thiết kế trang chủ bắt mắt dễ nhìn, giao diện trang quản lý thân thiện, dễ sử dụng, trực quan cho người dùng. Mỗi sản phẩm lắp đặt được thiết lập một ID riêng vì vậy khi triển khai hàng loạt sản phẩm ra thị trường không lo sợ vấn đề nhận sai dữ liệu cảm biến từ một sản phẩm khác.

Hướng phát triển của đề tài: Mặc dù dự án hiện tại đã đáp ứng được các mục tiêu ban đầu, vẫn còn nhiều hướng phát triển và cải tiến có thể thực hiện trong tương lai:

- Tối ưu hóa hiệu suất: Tối ưu hóa mã nguồn và cơ sở dữ liệu để cải thiện tốc độ và hiệu suất của ứng dụng. Áp dụng các kỹ thuật cache để giảm tải cho server và tăng tốc độ phản hồi.

- Mở rộng chức năng: Nâng cấp thêm các tính năng mới như phân tích nhiệt độ độ ẩm căn phòng điều chỉnh điều hoà đến nhiệt độ phù hợp, cảm biến chuyển động lắp đặt tại cầu thang hoặc cổng, sân khi có vật thể xuất hiện đèn sáng,...

Một lần nữa tôi xin cảm ơn chân thành tới giáo viên hướng dẫn thầy TS. Trần Đăng Công và thầy Nguyễn Văn Nhân cùng các bạn trong khoa đã giúp đỡ và góp ý cho đồ án của tôi được hoàn thành tốt đẹp.

TÀI LIỆU THAM KHẢO

Tiếng Việt:

- [1] Phạm Quang Huy, Nguyễn Tất Bảo Thiện (2018). *Lập Trình IOT Với ARDUINO, ESP8266 Và XBEE*. NXB Thanh Niên.

Tiếng Anh:

- [1] Bahga, A., & Madiseti, V. (2014). *Internet of Things: A Hands-On Approach*. VPT.
- [2] Höller, J., Tsiatsis, V., Mulligan, C., Karnouskos, S., Avesand, S., & Boyle, D. (2014). *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*. Academic Press.
- [3] Banzi, M., & Shiloh, M. (2014). *Getting Started with Arduino (3rd ed.)*. O'Reilly Media.
- [4] Jason Ostrander. (2012). *Android Development for Gifted Primates*. Addison-Wesley Professional.

Danh mục các Website tham khảo:

- [1] MurtadhaAlobaidi (2022), *Dự án kết nối Firebase-Arduino-ESP8266 github*. Truy cập ngày 20/04/2024 tại: <https://github.com/MurtadhaAlobaidi/Firebase-Arduino-ESP8266>
- [2] Google. *Tài liệu dành cho nhà phát triển Firebase*. Truy cập ngày 20/04/2024 tại: <https://firebase.google.com/docs/database/android/start>
- [3] Google. *Tài liệu tạo thông báo trên điện thoại*. Truy cập ngày 24/04/2024 tại: <https://developer.android.com/develop/ui/views/notifications/build-notification>
- [4] EazeGraph (2014), *Dự án tạo đồ thị trong lập trình android github*. Truy cập ngày 27/04/2024 tại: <https://github.com/paulroehr/EazeGraph?tab=readme-ov-file>