

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM  
KHOA ĐIỆN- ĐIỆN TỬ  
**BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG**



KHÓA LUẬN TỐT NGHIỆP

**XÂY DỰNG HỆ THỐNG QUẢN LÝ NÔNG NGHIỆP  
KẾT HỢP CÔNG NGHỆ MR**

**NGHÀNH HỆ THỐNG NHÚNG VÀ IOT**

Sinh viên: **ĐINH THANH TÙNG**  
20139096  
**TRẦN TOÀN THẮNG**  
20139015  
Hướng dẫn: **TS. HUỲNH THẾ THIỆN**

TP.HỒ CHÍ MINH - 5/2024

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM  
KHOA ĐIỆN - ĐIỆN TỬ  
**BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG**



KHÓA LUẬN TỐT NGHIỆP

**XÂY DỰNG HỆ THỐNG QUẢN LÝ NÔNG NGHIỆP  
KẾT HỢP CÔNG NGHỆ MR**

**NGHÀNH HỆ THỐNG NHÚNG VÀ IOT**

Sinh viên: **ĐINH THANH TÙNG**  
20139096  
**TRẦN TOÀN THẮNG**  
20139015  
Hướng dẫn: **TS. HUỲNH THẾ THIỆN**

TP.HỒ CHÍ MINH - 5/2024

# BẢN NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(Dành cho giảng viên hướng dẫn)

Đề tài: HỆ THỐNG QUẢN LÝ NÔNG NGHIỆP KẾT HỢP CÔNG NGHỆ MR

Sinh viên: + Trần Toàn Thắng

MSSV: 20139015

+ Đinh Thanh Tùng

MSSV: 20139096

Hướng dẫn: TS. Huỳnh Thế Thiện

1. Tính hợp lý trong cách đặt vấn đề và giải quyết vấn đề; ý nghĩa khoa học và thực tiễn:

*Dặt vấn đề rõ ràng, mục tiêu cụ thể; đề tài có tính mới, cấp thiết; đề tài có khả năng ứng dụng, tính sáng tạo.*

- Đề tài có tính ứng dụng và tính sáng tạo cao, kết hợp IoT và VR

2. Phương pháp thực hiện/ phân tích/ thiết kế:

*Phương pháp hợp lý và tin cậy dựa trên cơ sở lý thuyết; có phân tích và đánh giá phù hợp; có tính mới và tính sáng tạo.*

- Có phân tích đánh giá phù hợp và đáng tin cậy

3. Kết quả thực hiện/ phân tích và đánh giá kết quả/ kiểm định thiết kế:

*Phù hợp với mục tiêu đề tài; phân tích và đánh giá/ kiểm thử thiết kế hợp lý; có tính sáng tạo/ kiểm định chặt chẽ và đảm bảo độ tin cậy.*

- Kết quả thực hiện phù hợp với mục tiêu đề tài, có đánh giá và kiểm thử kết quả.

4. Kết luận và đề xuất:

*Kết luận phù hợp với cách đặt vấn đề, đề xuất mang tính cải tiến và thực tiễn; kết luận có đóng góp mới mẻ, đề xuất sáng tạo và thuyết phục.*

- Kết luận phù hợp với đề xuất.

5. Hình thức trình bày và bối cảnh báo cáo:

*Văn phong nhất quán, bối cảnh hợp lý, cấu trúc rõ ràng, đúng định dạng mẫu; có tính hấp dẫn, thể hiện năng lực tốt, văn bản trau chuốt.*

- Đúng định dạng mặc dù còn vài lỗi nhỏ.

6. Kỹ năng chuyên nghiệp và tính sáng tạo:

*Thể hiện các kỹ năng giao tiếp, kỹ năng làm việc nhóm, và các kỹ năng chuyên nghiệp khác trong việc thực hiện đề tài.*

- Sinh viên có giao tiếp và kỹ năng làm việc độc lập/nhóm.

7. Tài liệu trích dẫn

*Tinh trung thực trong việc trích dẫn tài liệu tham khảo; tính phù hợp của các tài liệu trích dẫn; trích dẫn theo đúng chỉ dẫn APA.*

- Tài liệu trích dẫn ở mức cơ bản

8. Đánh giá về sự trùng lặp của đề tài

*Cần khẳng định đề tài có trùng lặp hay không? Nếu có, đề nghị ghi rõ mức độ, tên đề tài, nơi công bố, năm công bố của đề tài đã công bố.*

- Độ trùng lặp trong mức cho phép.

9. Những nhược điểm và thiếu sót, những điểm cần được bổ sung và chỉnh sửa\*

- Nội dung và trình bày của luận văn cần được cải thiện

- Đánh giá kết quả cần thảo luận kỹ hơn.

10. Nhận xét tinh thần, thái độ học tập, nghiên cứu của sinh viên

- Sinh viên có thái độ nghiêm túc.

**Đề nghị của giảng viên hướng dẫn**

*Ghi rõ: "Báo cáo đạt/ không đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư, và được phép/ không được phép bảo vệ khóa luận tốt nghiệp"*

**Báo cáo đạt — Cho phép bảo vệ KLTN**

Tp. HCM, ngày 12 tháng 06 năm 2014

Người nhận xét  
(Ký và ghi rõ họ tên)

TS. Huỳnh Thế Thiện

\* Giảng viên hướng dẫn có thể ghi thêm vào mặt sau

## LỜI CẢM ƠN

Nhóm thực hiện khóa luận tốt nghiệp xin gửi lời cảm ơn sâu sắc đến TS. Huỳnh Thế Thiện và các giảng viên tại trường Đại học Sư Phạm Kỹ Thuật TP. Hồ Chí Minh, đặc biệt là các thầy cô trong ngành Hệ thống nhúng và IoT, khoa Điện - Điện Tử. Nhờ vào sự chỉ dẫn và hỗ trợ tận tình của Thầy Thiện, nhóm đã vượt qua những khó khăn trong quá trình nghiên cứu và hoàn thành đồ án tốt nghiệp về Xây dựng hệ thống quản lý nông nghiệp ứng dụng IoT và công nghệ MR. Thầy Thiện không chỉ đưa ra những lời khuyên quý báu mà còn hướng dẫn chi tiết từ đầu đến cuối dự án. Sự tận tâm và kiến thức sâu rộng của Thầy đã góp phần quan trọng vào thành công của đề tài này. Nhóm xin chân thành cảm ơn!

## Nhóm thực hiện đồ án tốt nghiệp (Ký và ghi rõ họ tên)

Đinh Thanh Tùng Trần Toàn Thắng

## LỜI CAM ĐOAN

Nhóm thực hiện khóa luận tốt nghiệp xin cam đoan rằng đề tài này được nghiên cứu và triển khai dựa trên nỗ lực và kết quả tự thân của nhóm, có tham khảo một số tài liệu hiện có nhưng không sao chép nội dung hay kết quả từ các đồ án hay khóa luận khác. Những tài liệu tham khảo đã được nhóm trích dẫn đầy đủ và chi tiết.

## Nhóm thực hiện đồ án tốt nghiệp (Ký và ghi rõ họ tên)

Đinh Thanh Tùng Trần Toàn Thắng

## TÓM TẮT LUẬN VĂN

Trong đồ án này, nhóm thực hiện đề tài muốn xây dựng một hạ tầng IoT có thể giám sát môi trường phát triển của cây trồng dựa trên công nghệ LoraWan. Hạ tầng này sẽ cung cấp đầy đủ các thiết bị thu thập dữ liệu của môi trường, thiết bị truyền nhận dữ liệu và máy chủ điện toán đám mây. Hạ tầng này được phát triển với định hướng hỗ trợ cho các công trình nghiên cứu thực vật, các hệ thống nông nghiệp tự động, các vùng trồng trọt với quy mô lớn. Các thiết bị thu thập dữ liệu của môi trường sẽ đo các chỉ số như nhiệt độ với độ ẩm của không khí và đất, nồng độ CO<sub>2</sub> trong không khí, cường độ ánh sáng của khu vực. Nhóm thấy rằng các chỉ số này là cần thiết phải theo dõi vì chúng đều có khả năng tác động đến quá trình sinh trưởng của cây trồng. Các thiết bị thu thập dữ liệu cũng được nhóm cân nhắc thiết kế hoạt động bằng pin và tận dụng ưu thế ngoài trời của nông nghiệp để áp dụng năng lượng mặt trời, giúp hệ thống có thể triển khai dễ dàng và hoạt động lâu dài. Ngoài ra, thiết bị thu thập dữ liệu có thể chuyển đổi thành một thiết bị điều khiển từ xa, sử dụng chính công nghệ LoraWan để điều khiển các thiết bị điện có vị trí đặt ở xa tầm kiểm soát cơ bản của người giám sát.

Nhóm thực hiện đề tài còn nghiên cứu và phát triển một ứng dụng MR (Mixed Reality-Thực tế hỗn hợp). Nhóm cho rằng việc người dùng chỉ cần đeo một chiếc kính, đi dạo xung quanh các cây trồng cần theo dõi, vừa xem xét tình trạng của cây vừa có thể phân tích những số liệu của các tác nhân môi trường sẽ nâng cao năng suất và khả năng phán đoán của người giám sát. Bên cạnh khả năng giám sát và điều khiển, ứng dụng còn được tích hợp Digital Twin giúp cho người dùng hiểu rõ hơn về phần cứng mà họ đang sử dụng. Trong đề tài này, nhóm sẽ trình bày về công nghệ MR, cũng như diễn đạt tầm nhìn của nhóm trong bài báo cáo về công nghệ này, nhằm chứng minh khả năng tương thích, tính ứng dụng thực tiễn cũng như tiềm lực kinh tế của công nghệ MR khi được đưa vào ứng dụng lao động sản xuất. Trước mắt ứng dụng MR này sẽ cung cấp một bảng điều khiển để người dùng có thể xem xét các chỉ số môi trường và điều khiển các thiết bị, người dùng cũng có thể xem biểu đồ chỉ số của từng tác nhân theo thời gian. Ngoài ra, ứng dụng này còn có hướng dẫn cơ bản cách lắp đặt thiết bị thu thập mà nhóm đã thiết kế, cung cấp thông tin chi tiết về các thành phần mà người dùng có thể tác động vào, đây là một tính năng mà nhóm cho rằng tương lai nó sẽ được ứng dụng rất rộng rãi trong môi trường sản xuất. Ứng dụng này sẽ được triển khai trên điện thoại, mặc dù sử dụng ứng dụng trên điện thoại không đem hoàn toàn cảm giác sử dụng một phần mềm MR như trên các thiết bị MR khác, nhưng điện thoại là thiết bị có chi phí thấp nhất trong các thiết bị có thể chạy ứng dụng MR. Nhóm hy vọng ứng dụng này sẽ đem lại một làn sóng mới trong trải nghiệm của người dùng.

# Mục lục

<b>1 TỔNG QUAN</b>	<b>1</b>
1.1 Giới thiệu . . . . .	1
1.2 Mục tiêu đề tài . . . . .	2
1.3 Giới hạn đề tài . . . . .	3
1.4 Phương pháp nghiên cứu . . . . .	4
1.5 Bố cục đồ án . . . . .	4
<b>2 CƠ SỞ LÝ THUYẾT</b>	<b>6</b>
2.1 Tổng quan về hệ thống quản lý nông nghiệp kết hợp công nghệ MR . . . . .	6
2.2 Công nghệ Mixed Reality . . . . .	6
2.3 Công nghệ Digital Twin . . . . .	7
2.4 Engine Unity . . . . .	8
2.4.1 Tổng quan về SRPs . . . . .	8
2.4.2 Mixed Reality Toolkit 2 . . . . .	10
2.4.3 Tổng quan về Vuforia Engine . . . . .	11
2.5 Chuẩn giao tiếp I2C . . . . .	11
2.6 Chuẩn giao tiếp RS485 và giao thức RS485 Modbus RTU . . . . .	12
2.7 Giao thức truyền thông MQTT . . . . .	13
2.8 Giao thức HTTP và kiến trúc REST API . . . . .	14
2.8.1 Giao thức HTTP . . . . .	14
2.8.2 Kiến trúc REST API . . . . .	15
2.9 Sóng Lora và mạng LoraWan . . . . .	15
2.9.1 Sóng Lora . . . . .	15
2.9.2 Mạng LoraWan . . . . .	16
<b>3 THIẾT KẾ VÀ THI CÔNG HỆ THỐNG</b>	<b>20</b>

3.1	Yêu cầu và đặc tính của hệ thống . . . . .	20
3.1.1	Yêu cầu của hệ thống . . . . .	20
3.1.2	Sơ đồ tổng quan . . . . .	21
3.1.3	Đặc tính kỹ thuật . . . . .	22
3.2	Đặc tả hệ thống . . . . .	22
3.2.1	Phần mềm sử dụng. . . . .	22
3.2.2	Danh sách các chức năng . . . . .	24
3.3	Một số phần cứng được sử dụng . . . . .	25
3.3.1	Máy tính nhúng Raspberry Pi 4 . . . . .	25
3.3.2	Bộ Kit Gateway Pi HAT RAK5146 . . . . .	25
3.3.3	Module điều khiển RAK3172 . . . . .	26
3.3.4	IC sạc năng lượng mặt trời CN3791 . . . . .	27
3.3.5	IC hạ áp TPS62840 . . . . .	28
3.3.6	Ví điều khiển CH32V003 . . . . .	28
3.3.7	Cảm biến CO <sub>2</sub> SCD40 . . . . .	29
3.3.8	Cảm biến ánh sáng BH1750 . . . . .	30
3.3.9	IC MAX3485EESA+T . . . . .	31
3.4	Xây dựng máy chủ và trạm trung gian . . . . .	32
3.4.1	Xây dựng máy chủ . . . . .	32
3.4.2	Xây dựng trạm trung gian . . . . .	36
3.5	Quá trình phát triển cảm biến . . . . .	38
3.5.1	Thiết kế phần cứng cảm biến . . . . .	38
3.5.2	Xây dựng mã nguồn cho cảm biến . . . . .	43
3.6	Quá trình phát triển trạm thu thập dữ liệu . . . . .	46
3.6.1	Thiết kế phần cứng trạm thu thập dữ liệu . . . . .	46
3.6.2	Xây dựng mã nguồn cho trạm thu thập dữ liệu và trạm điều khiển	53
3.7	Thiết lập môi trường phát triển ứng dụng MR . . . . .	56
3.7.1	Thiết lập trong Player Settings . . . . .	57
3.7.2	Thiết lập về XR Plugin Management . . . . .	59

3.7.3	Thiết lập MRTK hoạt động trên điện thoại thông minh . . . . .	60
3.8	Phát triển ứng dụng MR . . . . .	61
3.8.1	Tiến hành vẽ và tối ưu hoá mô hình 3D . . . . .	61
3.8.2	Thiết kế hoạt ảnh lắp ráp trong Blender . . . . .	63
3.8.3	Thiết kế shader cho material . . . . .	64
3.8.4	Phát triển chức năng thêm và xoá vị trí của trạm . . . . .	65
3.8.5	Triển khai thuật toán tìm kiếm trạm gần nhất trong phạm vi . .	66
3.8.6	Phát triển chức năng hiển thị dữ liệu. . . . .	67
3.8.7	Thiết kế điều khiển thiết bị . . . . .	69
3.8.8	Triển khai Singleton quản lý mã nguồn . . . . .	69
3.9	Thiết kế giao diện quản lý . . . . .	70
3.10	Thiết kế chức năng hiển thị hướng dẫn . . . . .	72
<b>4</b>	<b>KẾT QUẢ</b>	<b>74</b>
4.1	Kết quả triển khai hạ tầng IoT . . . . .	74
4.1.1	Kết quả xây dựng máy chủ đám mây và trạm trung gian . . . . .	74
4.1.2	Kết quả phát triển cảm biến RS485 Modbus RTU . . . . .	76
4.1.3	Kết quả phát triển trạm thu thập dữ liệu . . . . .	79
4.1.4	Kết quả triển khai trạm điều khiển . . . . .	82
4.2	Đánh giá ứng dụng MR . . . . .	83
4.3	Kết quả kiểm tra trên các thiết bị. . . . .	84
4.4	Kết quả triển khai hệ thống vào thực tế . . . . .	85
<b>5</b>	<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b>	<b>88</b>
5.1	Kết luận . . . . .	88
5.2	Hướng phát triển . . . . .	89
5.3	Quan điểm của nhóm về tiềm năng của MR . . . . .	90
<b>TÀI LIỆU THAM KHẢO</b>		<b>92</b>

# Danh sách bảng

Bảng 3.1	Danh sách các phần mềm được sử dụng để xây dựng hệ thống. . . . .	23
Bảng 3.2	Danh sách các chức năng của hệ thống . . . . .	24
Bảng 3.3	Bảng thông tin phần cứng của Raspberry Pi 4 2GB và 8GB . . . . .	25
Bảng 3.4	Bảng Gateway RAK5146 . . . . .	26
Bảng 3.5	Bảng thông tin chi tiết module RAK3172 . . . . .	27
Bảng 3.6	Bảng thông tin chi tiết IC CN3791 . . . . .	27
Bảng 3.7	Bảng thông tin chi tiết IC TPS62840 . . . . .	28
Bảng 3.8	Bảng thông tin chi tiết vi điều khiển CH32V003 . . . . .	29
Bảng 3.9	Bảng thông tin chi tiết cảm biến SCD40 . . . . .	30
Bảng 3.10	Bảng thông tin chi tiết cảm biến BH1750 . . . . .	31
Bảng 3.11	Bảng thông tin chi tiết IC MAX3485EESA+T . . . . .	31
Bảng 3.12	Bảng mã lệnh Modbus RTU cho cảm biến không khí . . . . .	45
Bảng 3.13	Bảng mã lệnh Modbus RTU cho cảm biến đất . . . . .	45
Bảng 3.14	Các gói thư viện được sử dụng trong dự án. . . . .	57
Bảng 4.1	Bảng thông tin hiệu năng khi giám sát và điều khiển. . . . .	83
Bảng 4.2	Bảng thông tin hiệu năng khi dùng chức năng lắp đặt và hướng dẫn. .	84
Bảng 4.3	Kết quả kiểm tra trên một số thiết bị thực tế . . . . .	84

# Danh sách hình vẽ

Hình 2.1	Mô hình đồ họa có thể lập trình mới [1] . . . . .	9
Hình 2.2	Cấu trúc cơ bản của một gói tin Modbus RTU . . . . .	13
Hình 2.3	Ảnh giới thiệu về kiến trúc của mạng LoraWan . . . . .	17
Hình 3.1	Mô hình hệ thống quản lý nông nghiệp kết hợp công nghệ MR . .	21
Hình 3.2	Sơ đồ tổng quan các ứng dụng chính và mối quan hệ giữa chúng .	32
Hình 3.3	Giao diện The Things Stack khi đã cài đặt xong và đăng nhập . .	33
Hình 3.4	Ảnh minh họa cách xác định vị trạm ở gần người dùng . . . . .	35
Hình 3.5	Lưu đồ thuật toán xác định trạm trong phạm vi bán kính . . . . .	36
Hình 3.6	Ảnh minh họa khi đăng ký xong một trạm trung gian . . . . .	37
Hình 3.7	Sơ đồ tổng quát của cả hai cảm biến . . . . .	38
Hình 3.8	Cảm biến nhiệt độ (bên trái) và cảm biến độ ẩm (biên phải) được sử dụng cho cảm biến đất . . . . .	39
Hình 3.9	Sơ đồ nguyên lý của cảm biến không khí . . . . .	39
Hình 3.10	Sơ đồ nguyên lý khối nguồn của cả hai cảm biến . . . . .	40
Hình 3.11	Sơ đồ nguyên lý khối chuyển đổi UART sang RS485 của cả hai cảm biến . . . . .	40
Hình 3.12	Sơ đồ nguyên lý khối cảm biến I2C . . . . .	41
Hình 3.13	Sơ đồ nguyên lý khối vi điều khiển CH32V003 . . . . .	41
Hình 3.14	Ảnh 3D mặt trên của cảm biến không khí sau khi thiết kế . . . . .	42
Hình 3.15	Ảnh 3D mặt dưới của cảm biến không khí sau khi thiết kế . . . . .	42
Hình 3.16	Ảnh 3D mặt trên của cảm biến đất sau khi thiết kế . . . . .	42
Hình 3.17	Ảnh 3D mặt dưới của cảm biến đất sau khi thiết kế . . . . .	43
Hình 3.18	Sơ đồ nguyên lý của cảm biến đất . . . . .	43
Hình 3.19	Giao diện Mounriver Studio khi đang mở một dự án . . . . .	44
Hình 3.20	Lưu đồ thuật toán lúc khởi động và từng chu kì vòng lặp của cả 2 cảm biến . . . . .	46
Hình 3.21	Sơ đồ tổng quát của cả trạm thu thập dữ liệu . . . . .	47

Hình 3.22 Sơ đồ nguyên lý tổng quát của trạm thu thập dữ liệu . . . . .	47
Hình 3.23 Sơ đồ nguyên lý bộ bảo vệ pin . . . . .	48
Hình 3.24 Sơ đồ nguyên lý bộ sạc pin bằng năng lượng mặt trời . . . . .	48
Hình 3.25 Sơ đồ nguyên lý bộ giám sát năng lượng . . . . .	49
Hình 3.26 Bộ hạ áp pin sang 3.3V . . . . .	50
Hình 3.27 Bộ tăng áp cấp nguồn cho cảm biến RS485 . . . . .	51
Hình 3.28 Sơ đồ nguyên lý bộ điều khiển RS485 . . . . .	51
Hình 3.29 Sơ đồ nguyên lý module điều khiển chính RAK3172 . . . . .	52
Hình 3.30 Ảnh 3D của trạm thu thập dữ liệu sau khi thiết kế . . . . .	52
Hình 3.31 Các cấu hình cần thiết để trạm thu thập có thể kết nối đến máy chủ	53
Hình 3.32 Lưu đồ thuật toán lúc khởi động và mỗi chu kì của trạm thu thập dữ liệu . . . . .	54
Hình 3.33 Cách cấu hình thời gian ngủ cho trạm từ trang web máy chủ . . .	55
Hình 3.34 Giao diện tạo một Project mới trong Unity . . . . .	56
Hình 3.35 Cấu hình không gian màu. . . . .	57
Hình 3.36 Các cấu hình quan trọng để xuất bản ứng dụng. . . . .	58
Hình 3.37 Giao diện của XR Plug-in Management. . . . .	59
Hình 3.38 Vị trí của tùy chọn "Add Scene with Configure." . . . . .	60
Hình 3.39 Các mô hình 3D được sử dụng trong hệ thống . . . . .	61
Hình 3.40 Lên màu và trải UV map cho mô hình. . . . .	62
Hình 3.41 Bên trái là mô hình từ Altium, bên phải là mô hình đã được tối ưu.	62
Hình 3.42 Thiết kế hoạt ảnh bằng Blender. . . . .	63
Hình 3.43 Giao diện shader Graph. . . . .	64
Hình 3.44 Lưu đồ thuật toán cho sự kiện thêm và xoá vị trí của trạm . . . . .	65
Hình 3.45 Lưu đồ nhận mã định danh trạm gần nhất từ máy chủ. . . . .	67
Hình 3.46 Lưu đồ hiển thị dữ liệu trong một vòng lặp. . . . .	67
Hình 3.47 Lưu đồ hiển thị biểu đồ dữ liệu . . . . .	68
Hình 3.48 Lưu đồ điều khiển thiết bị . . . . .	69
Hình 3.49 Giao diện hiển thị dữ liệu trạm tức thời. . . . .	70

Hình 3.50 Giao diện hiển thị lịch sử dữ liệu. . . . .	70
Hình 3.51 Giao diện thao tác lưu, xoá, điều khiển, phát hiện trạm, chuyển chế độ . . . . .	71
Hình 3.52 Lưu đồ sử lý sự kiện hướng dẫn trong một vòng lặp . . . . .	72
Hình 3.53 Hoạt ảnh hiển thị trực tiếp trên vật thể. . . . .	72
Hình 3.54 Giao diện hướng dẫn lắp đặt . . . . .	73
Hình 3.55 Giao diện miêu tả các cổng kết nối. . . . .	73
Hình 4.1 Hình ảnh trang web The Things Network của máy chủ đám mây khi được truy cập bằng tên miền . . . . .	74
Hình 4.2 Hình ảnh trang web Home Assistant của máy chủ đám mây khi được truy cập bằng tên miền . . . . .	75
Hình 4.3 Hình ảnh kiểm tra khả năng truy xuất dữ liệu cảm biến bằng Postman	75
Hình 4.4 Hình ảnh thực tế của trạm trung gian . . . . .	76
Hình 4.5 Hình ảnh thực tế mặt trên của cảm biến không khí . . . . .	77
Hình 4.6 Hình ảnh thực tế mặt dưới của cảm biến không khí . . . . .	77
Hình 4.7 Hình ảnh thực tế mặt trên của cảm biến đất . . . . .	77
Hình 4.8 Hình ảnh thực tế mặt dưới của cảm biến đất . . . . .	78
Hình 4.9 Hình ảnh giao tiếp giữa máy tính và cảm biến bằng phần mềm Hercules . . . . .	78
Hình 4.10 Ảnh thực tế của trạm sau khi đã hoàn thiện . . . . .	79
Hình 4.11 Ảnh khoanh vùng các khối phần cứng của trạm thu thập . . . . .	79
Hình 4.12 Biểu đồ ghi nhận dòng sạc năng lượng mặt trời và điện áp pin thực tế của trạm thu thập dữ liệu . . . . .	80
Hình 4.13 Kết quả đo dòng tiêu thụ của trạm khi cấp nguồn 12V cho các cảm biến . . . . .	81
Hình 4.14 Kết quả đo dòng tiêu thụ của trạm khi cấp nguồn 5V cho các cảm biến . . . . .	82
Hình 4.15 Hình ảnh thực tế của trạm điều khiển . . . . .	82
Hình 4.16 Hình ảnh trạm thu thập dữ liệu được chuẩn bị để chạy thực tế ngoài trời . . . . .	85
Hình 4.17 Ảnh một phần biểu đồ điện áp pin của trạm số một trong một tuần	86

Hình 4.18 Hình ảnh trạm thu thập dữ liệu được lắp đặt thực tế ngoài vườn . . . . . 86

Hình 4.19 Ảnh một phần biểu đồ nhiệt độ không khí do trạm đo được thực tế 87

# Danh mục các từ viết tắt

Dưới đây là danh mục các từ viết tắt và định nghĩa được sử dụng trong luận văn.

Các từ viết tắt	Định nghĩa
API	Application programming interface
AR	Augmented Reality
GPS	Global Positioning System
IC	Integrated Circuit
IDE	Integrated Development Environment
IoT	Internet of Things
MR	Mixed reality
MRTK	Mixed Reality Toolkit
SDI	Serial Debug Interface
SDK	Software Development Kit
SPI	Serial Peripheral Interface
SRPs	Scriptable Render Pipeline
SSL	Secure Sockets Layer
TLS	Transport Layer Security
UWP	Universal Windows Platform
VR	Virtual Reality
XR	Extended Reality

# Chương 1

## TỔNG QUAN

### 1.1 Giới thiệu

Trong thời đại công nghiệp 4.0 hiện nay, việc chuyển mình áp dụng những công nghệ mới vào các ngành nghề hay lĩnh vực truyền thống đang là một vấn đề trọng yếu luôn được các doanh nghiệp để tâm và đầu tư phát triển, nhằm mục tiêu nâng cao năng suất lao động cũng như cải thiện hiệu quả công việc. Vậy nên các công nghệ liên quan đến MR cũng không nằm ngoài xu hướng phát triển đó. Trong những năm gần đây, ta có thể thấy được rằng các thiết bị hiển thị XR hay các thiết bị hỗ trợ khác ngày càng trở nên rẻ và dễ tiếp cận với người dùng hơn. Từ đó mở ra cơ hội nghiên cứu và ứng dụng công nghệ này vào đời sống xã hội cũng như lao động sản xuất. Công nghệ MR cũng mở ra cho chúng ta rất nhiều hướng đi mới lạ trong các lĩnh vực về việc đào tạo kỹ sư, giám sát thiết bị...

Nhóm thực hiện khóa luận nhận thấy những điểm hạn chế trong việc đào tạo kỹ sư thông qua các phương pháp truyền thống như sử dụng các tài liệu giấy, video hướng dẫn 2D thường thiếu tính trực quan và tương tác, điều này có thể dẫn đến sự sai sót trong quá trình thực hiện công việc. Nhóm cho rằng khi ứng dụng công nghệ MR vào các công việc hướng dẫn này sẽ mang đến một trải nghiệm thực tế trên các mô hình ảo, từ đó có thể giảm thiểu chi phí đào tạo, hướng dẫn cũng như là giảm thiểu những tai nạn, hư hỏng không đáng có thể xảy ra.

IoT đã tạo ra một bước phát triển đột phá trong lĩnh vực nông nghiệp bằng cách cung cấp khả năng giám sát và điều khiển từ xa các thiết bị và hệ thống. Với IoT, người giám sát có thể thu thập dữ liệu chi tiết về môi trường và tình trạng cây trồng thông qua các cảm biến được đặt khắp nơi trên cánh đồng, chẳng hạn như cảm biến đo độ ẩm đất, nhiệt độ, ánh sáng và nồng độ CO<sub>2</sub>. Dữ liệu này được gửi liên tục đến các nền tảng đám mây, nơi nó có thể được lưu trữ, hiển thị, thậm chí có thể phân tích để đưa ra các quyết định chính xác và kịp thời về việc tưới tiêu, bón phân và bảo vệ cây trồng. Việc điều khiển các thiết bị nông nghiệp từ xa cũng trở nên dễ dàng hơn bao giờ hết, cho phép người giám sát kích hoạt hệ thống tưới tiêu, hệ thống chiếu sáng, máy móc canh tác, hoặc hệ thống thông gió chỉ bằng một cú nhấp chuột từ bất kỳ đâu có kết nối Internet.

Khi kết hợp với công nghệ MR (Mixed Reality), việc giám sát dữ liệu môi trường trở nên trực quan và tiện lợi hơn. MR cung cấp một giao diện ảo sinh động, nơi người giám sát có thể đeo kính MR và ngay lập tức nhìn thấy các thông số môi trường và tình

trạng cây trồng hiển thị trên hình ảnh thực tế của cánh đồng. Điều này không chỉ nâng cao hiệu quả giám sát mà còn giúp người giám sát phát hiện sớm các vấn đề và có biện pháp khắc phục nhanh chóng, từ đó tối ưu hóa năng suất và chất lượng sản phẩm nông nghiệp.

Bài báo cáo này sẽ đưa ra một hệ thống có thiên hướng hỗ trợ cho lĩnh vực nghiên cứu nông nghiệp, để chứng minh tính khả thi và thực tiễn của việc ứng dụng MR kết hợp với IoT vào các hoạt động giám sát và điều khiển. Đây là một ví dụ tiêu biểu cho việc đưa những công nghệ mới vào các hoạt động truyền thống nhằm giúp giảm thiểu chi phí đầu tư, tăng khả năng giám sát thiết bị.

Bài báo cáo này cũng sẽ trình bày về quy trình triển khai một hệ thống giám sát có tích hợp IoT và MR, cung cấp cái nhìn tổng quan về các phương pháp và công nghệ sử dụng, cũng như thách thức phải đối mặt trong quá trình tiến hành triển khai và thực hiện. Ngoài ra cũng sẽ đưa ra kết luận về tính khả thi khi sử dụng công nghệ MR vào các hoạt động lao động và sản xuất.

Mục tiêu cuối cùng của mô hình là tham khảo cho việc triển khai các công nghệ về MR vào hoạt động lao động sản xuất, đóng góp vào việc tạo ra các giải pháp hiệu quả và tiên bội hơn trong việc giải quyết các vấn đề liên quan. Đặc biệt bài báo cáo này có thể cung cấp một số dữ liệu quan trọng về những khó khăn, thử thách và tiềm năng của công nghệ MR mang lại.

## 1.2 Mục tiêu đề tài

Mục tiêu của đề tài là nghiên cứu kết hợp giữa MR cùng với IoT để tạo ra một sản phẩm mang tính thực tiễn có thể ứng dụng vào các hoạt động giám sát các thiết bị từ xa một cách dễ dàng và tiện lợi, cũng như là hướng dẫn thực hiện các thao tác cần thiết trên thiết bị khi sử dụng thông qua MR và Digital Twin từ đó nâng cao năng suất lao động. Trong quá trình lên ý tưởng, nhóm thực hiện đề tài đã đặt ra mục tiêu cụ thể bao gồm:

- Mô hình giám sát nghiên cứu nông nghiệp là sự kết hợp của MR và IoT với mục tiêu phục vụ cho việc nâng cao sự tiện lợi về khả năng giám sát các thiết bị ở mọi nơi mọi lúc.
- Thiết kế cảm biến có thể đo được các tác nhân môi trường xung quanh có ảnh hưởng đến cây trồng, cảm biến có thể dùng cho các hệ thống khác nhau.
- Tạo điều kiện cho việc quản lý nông trại thông minh, giảm thiểu sức lao động thủ công và tăng hiệu quả sản xuất thông qua việc tự động hóa và điều khiển từ xa các thiết bị nông nghiệp.

- Xây dựng một cơ sở hạ tầng IoT cho phép thu thập dữ liệu từ các cảm biến, thiết bị từ khoảng cách xa. Trong điều kiện có kết nối internet, cho phép người dùng có thể truy cập vào hệ thống mọi lúc mọi nơi.
- Thiết kế một ứng dụng có thể dễ dàng chuyển đổi sang nhiều thiết bị hỗ trợ sử dụng công nghệ MR khác nhau từ đó tạo tính linh hoạt cho hệ thống. Ứng dụng này sẽ được phát triển giao diện tối ưu nhất cho trải nghiệm của người dùng.
- Mô hình này còn cung cấp cho một công cụ hướng dẫn lắp đặt sử dụng thiết bị thông qua MR và Digital Twin. Điều này sẽ góp phần cho người dùng hiểu rõ hơn về phần cứng mà họ đang sử dụng, làm cho quá trình lắp đặt và sử dụng đơn giản hơn và giảm thiểu sai sót.

### 1.3 Giới hạn đề tài

Do gặp khó khăn về mặt thời gian trong việc nghiên cứu học hỏi những công nghệ mới, chi phí cũng là một vấn đề lớn vì các ứng dụng về XR đòi hỏi thiết bị MR chuyên dụng và các thiết bị đó có cấu hình mạnh mẽ, camera chất lượng cao và phần mềm chuyên dụng để xử lý dữ liệu hình ảnh và âm thanh 3D. Điều này có thể dẫn đến chi phí đầu tư cao cho hệ thống. Vậy nên với quy mô ứng dụng của đề tài nên mô hình này sẽ đặt trọng tâm giải quyết các vấn đề sau:

- Nghiên cứu, thiết kế và thi công một mô hình giám sát nông nghiệp nhỏ, sử dụng điện thoại thay cho thiết bị MR đắt đỏ.
- Nhóm thực hiện tập trung vào việc ổn định của hệ thống, dữ liệu sẽ được đo đạc cũng như hiển thị một cách nhanh chóng.
- Đề tài vẫn đang trong giai đoạn cải thiện về chức năng cũng như về trải nghiệm người dùng nên mô hình cần có người có một ít kỹ năng cũng như kinh nghiệm sử dụng thiết bị hiển thị XR mới có thể trải nghiệm một cách trọn vẹn được.
- Hệ thống hiện tại chỉ hỗ trợ các cảm biến tự thiết kế cho đề tài, chưa phát triển để hỗ trợ các cảm biến có sẵn trên thị trường.
- Máy chủ đám mây được triển khai với hệ thống điện và internet gia đình nên vẫn chưa đảm bảo được độ ổn định và khả năng đáp ứng 24/24.
- Hiện tại ứng dụng chỉ hỗ trợ những thiết bị có tích hợp ARCore và API cảm biến độ sâu.

## 1.4 Phương pháp nghiên cứu

Để đạt được mục tiêu của đề tài, nhóm thực hiện đề tài đã sử dụng một loạt các phương pháp thu thập dữ liệu, thực nghiệm, và phân tích để tổng kết kinh nghiệm:

- Phương pháp thu thập số liệu:

- Tiến hành quan sát trực tiếp phiên bản hoàn chỉnh nhất của ứng dụng để đảm bảo rằng nó đáp ứng đúng yêu cầu và đạt được chất lượng mong đợi.
- Thực hiện những cuộc khảo sát với những người thử nghiệm sản phẩm có đa dạng về độ tuổi và kinh nghiệm. Cách này có thể thu thập ý kiến đa chiều, từ đó cải thiện sản phẩm để phản ánh đúng nhu cầu của người sử dụng.

- Phương pháp thực nghiệm:

- Thiết kế và triển khai hệ thống: Thiết kế và triển khai hệ thống theo lý thuyết.
- Tiến hành thử nghiệm: Thực hiện các thử nghiệm theo phương pháp thử và sai liên tục để đảm bảo việc đưa ra giá trị hiệu suất tốt nhất.

- Phương pháp phân tích tổng kết kinh nghiệm:

- Đánh giá hiệu quả: Tiến hành đánh giá hiệu quả của từng phương pháp được sử dụng ở các phiên bản khác nhau của ứng dụng.
- Phân tích dữ liệu: Thực hiện phân tích chi tiết các số liệu thu được từ mỗi cài đặt khác nhau.
- Tổng kết kinh nghiệm: Dựa trên các kết quả được phân tích trước đó, nhóm đã đúc kết và tổng hợp những biện pháp tối ưu nhất để áp dụng vào công trình phát triển và nghiên cứu ứng dụng.

## 1.5 Bố cục đồ án

Chương 1: TỔNG QUAN. Chương này sẽ giới thiệu tổng quan về thiết kế, mục tiêu và kết quả mong đợi của hệ thống.

Chương 2: CƠ SỞ LÝ THUYẾT. Trong chương này, các cơ sở lý thuyết được áp dụng trong hệ thống sẽ được trình bày một cách tổng quan.

Chương 3: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG. Chương này sẽ phân tích các yêu cầu, đặc tính kỹ thuật, thiết kế phần cứng và phần mềm, cách kết nối hệ thống và nguyên lý hoạt động để tạo ra một mô hình có tính ứng dụng cao.

**Chương 4: KẾT QUẢ.** Sau khi hoàn thành quá trình nghiên cứu, thiết kế và thi công, chương này sẽ trình bày các kết quả khi vận hành mô hình và đưa ra những nhận xét về hệ thống đã xây dựng.

**Chương 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.** Chương này sẽ tổng kết quá trình nghiên cứu và thực hiện đề tài, đưa ra các kết luận và đề xuất hướng phát triển của đề tài trong tương lai.

# **Chương 2**

## **CƠ SỞ LÝ THUYẾT**

### **2.1 Tổng quan về hệ thống quản lý nông nghiệp kết hợp công nghệ MR**

Các hệ thống giám sát môi trường của cây trồng một cách tự động đang được triển khai và sử dụng ở rất nhiều nơi trên thế giới. Với mục đích tăng khả năng kiểm soát quá trình sinh trưởng của thực vật, thu thập được các số liệu chính xác của từng tác nhân môi trường tự nhiên. Từ đó, người giám sát sẽ có cái nhìn trực quan về sự ảnh hưởng của môi trường lên cây trồng, rút ra được các đặc tính của cây và tìm ra cách đối phó với các trường hợp xấu do môi trường tự nhiên tác động. Các dữ liệu số từ cảm biến sẽ được lưu lại trong hệ thống cơ sở dữ liệu, cho phép các dự án sử dụng trí tuệ nhân tạo có thể tận dụng để huấn luyện các mô hình phức tạp cho phép ứng dụng trong tự động hóa nông nghiệp, phát hiện sâu bệnh, đề xuất giải pháp ứng phó. Việc kết hợp hệ thống này với công nghệ MR được đề xuất với định hướng nâng tầm trải nghiệm của người dùng lên thêm một bậc, cho phép họ vừa quan sát cây trồng một cách trực tiếp và vừa có thể quan sát các số liệu của môi trường tác động lên cây trồng.

### **2.2 Công nghệ Mixed Reality**

MR (Mixed Reality) là một công nghệ cung cấp cho người dùng sử dụng các thiết bị hiển thị XR có thể vẫn nhìn thấy được thế giới xung quanh nhưng đồng thời cũng hiển thị thêm các hình ảnh, video và thông tin 3D ảo, không những thế người dùng còn có thể tương tác với những mô hình ảo đó thông qua cử chỉ tay hoặc giọng nói ... Với công nghệ VR, nó đưa bạn vào một thế giới ảo hoàn toàn, nơi người dùng có thể di chuyển, tương tác với các vật thể ở thế giới ảo, nhưng không nhìn thấy và tương tác thế giới thực xung quanh. Còn AR bổ sung các thông tin ảo vào thế giới thực. Với MR là sự kết hợp của cả AR và VR. Nó cho phép người dùng nhìn thấy các mô hình ảo xuất hiện trên thế giới thực như trong một phần mềm AR cũng như là có thể tương tác với các mô hình ảo đó như trong VR. Ngày nay công nghệ thực tế tăng cường có nhiều tiềm năng to lớn trong nhiều lĩnh vực:

- Giáo dục: MR có thể tạo ra một môi trường học tập sinh động và trực quan hơn, từ đó giúp học sinh dễ dàng tiếp thu kiến thức.
- Đào tạo: MR còn có thể sử dụng để đào tạo công nhân, nhân viên, kỹ sư, thậm chí

là hỗ trợ các buổi thực hành của sinh viên trong trường từ đó đưa ra một hướng tiếp cận mới về học tập và trao đổi kỹ năng.

- Phục vụ nâng cao năng suất công việc: Ứng dụng MR có thể giúp các công ty đào tạo và hướng dẫn nhân viên các công việc bảo trì máy móc một cách đơn giản và an toàn hơn. Các máy móc cần được kiểm tra thường xuyên và trong một số trường hợp còn phải điều chỉnh sao cho phù hợp với nhu cầu công việc tại thời điểm đó. Những điều chỉnh này hầu hết do những người có chuyên môn cao thực hiện, nhưng việc bố trí nhân lực có chuyên môn cao mọi lúc mọi nơi để xem xét tình trạng của mọi máy sẽ đòi hỏi chi phí rất cao. Bằng cách sử dụng công nghệ MR, các công nhân mặt dù có rất ít kiến thức cũng có thể thao tác theo những gì hướng dẫn hiện trực tiếp trên vật thể.

Tuy nhiên công nghệ này hiện nay vẫn còn một số hạn chế nhất định:

- Giá thành cao: Các thiết bị MR hiện nay khá đắt đỏ, khiến cho việc sử dụng chúng trở nên hạn chế.
- Yêu cầu về phần cứng: MR đòi hỏi phần cứng mạnh mẽ để có thể hoạt động hiệu quả.
- Rủi ro về sức khỏe: Việc không quen sử dụng các thiết bị MR có thể gây ra một số tình trạng xấu đến sức khoẻ, chẳng hạn như chóng mặt, buồn nôn và nhức đầu.

Các thiết bị hỗ trợ sử dụng MR trong thực tế:

- Cave: Là một căn phòng nhỏ mà trong đó người dùng được bao quanh bởi các hình chiếu xung quanh, trên và dưới họ.
- Màn hình hiển thị trên các loại kính chắn gió: HUD là một màn hình hiển thị hình ảnh trực tiếp trước mặt người xem mà không làm thay đổi môi trường quan sát của người sử dụng.
- Các loại kính XR: Là loại thiết bị được đeo trên đầu, thiết bị sử dụng một hoặc hai thấu kính quang học để chiếu hình ảnh trực tiếp trước mắt người dùng.
- Điện thoại di động.

## 2.3 Công nghệ Digital Twin

Digital Twin là một bản sao được số hoá của vật thể thực tế hoặc có thể đó là bản sao của một quy trình hoặc hệ thống. Các bản sao này người dùng có thể tương tác bình thường như các thiết bị thực tế. Các tính năng cụ thể của Digital Twin bao gồm:

- Mô Hình Số Hóa: Ứng dụng cần tạo ra một bản sao kỹ thuật số của thiết bị thực tế hoặc hệ thống đang giám sát. Bản sao này phải phản ánh cấu trúc, hành vi và bối cảnh của đối tượng thực.
- Dữ Liệu Thời Gian Thực: Ứng dụng phải liên tục cập nhật dữ liệu từ các cảm biến bên ngoài, phản ánh chính xác tình trạng hiện tại của đối tượng thực.
- Phân Tích và Mô Phỏng: Ứng dụng cần phải phân tích và mô phỏng các kích thước khác nhau dựa trên dữ liệu thu thập được. Điều này giúp dự đoán và tối ưu hóa hiệu suất của đối tượng thực.
- Tương Tác Hai Chiều: Ứng dụng cần có khả năng tương tác hai chiều với đối tượng thực tế, tức là không chỉ thu thập dữ liệu mà còn có thể gửi lệnh hoặc điều khiển đối tượng đó.

## 2.4 Engine Unity

Unity là một trong những game engine phát triển trò chơi phổ biến nhất hiện nay, đặc biệt là có hỗ trợ phát triển game, ứng dụng dành cho kính thực tế ảo cũng như thực tế tăng cường và hỗn hợp. Đây là engine (công cụ) được thiết kế với một môi trường phát triển toàn diện, Unity cung cấp đầy đủ các assets (tài nguyên), shader (đồ bong) cũng như các phương pháp render (kết xuất) tối ưu, giúp phát triển ứng dụng XR cũng như phát triển game nói chung có thể tạo ra những sản phẩm chất lượng cao. Unity nổi bật trong lĩnh vực phát triển các ứng dụng XR nhờ vào việc nó hỗ trợ hầu hết các công cụ SDK phổ biến như: Nreal, Oculus plug-in, MRTK... Mang lại sự linh hoạt và tiện lợi cho các nhà phát triển khi muốn đưa sản phẩm của mình lên nhiều nền tảng khác nhau. Bằng cách sử dụng ngôn ngữ bậc cao C#, Unity trở thành một engine thân thiện với các lập trình viên so với các engine khác. Hơn nữa Unity còn hỗ trợ cả đa nền tảng, giúp việc chuyển đổi giữa các nền tảng như Windows, macOS, IOS, Android, UWP... Trở nên dễ dàng hơn. Unity cung cấp cho lập trình viên các thuật toán mạnh mẽ dành cho việc tối ưu hóa về đồ họa cũng như các công nghệ render pipeline thích hợp tùy thuộc vào nhu cầu và mục tiêu của game hay ứng dụng. Một điểm mạnh khác của Unity là môi trường gõ lỗi mạnh mẽ, cho phép lập trình viên gõ lỗi mã nguồn cũng như gõ lỗi quá trình kết xuất hình ảnh một cách hiệu quả. Công cụ gõ lỗi này giúp phát hiện và khắc phục các lỗi nhanh chóng, đảm bảo sản phẩm cuối cùng đạt chất lượng cao nhất.

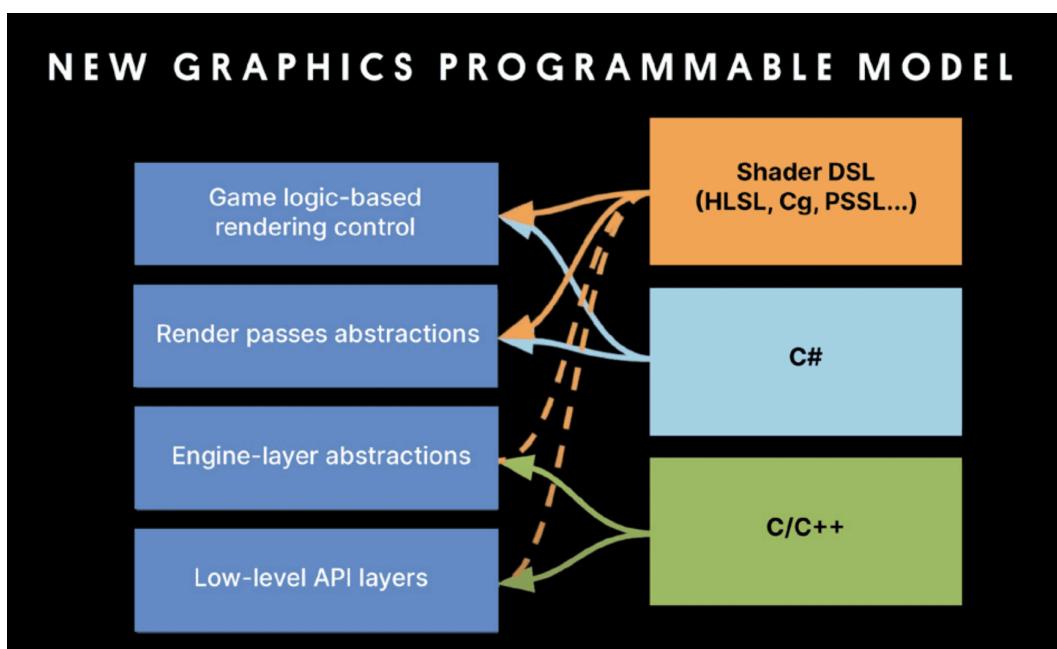
### 2.4.1 Tổng quan về SRPs

Một trong những điểm mạnh lớn nhất của Unity đó là đa nền tảng. Một ứng dụng lý tưởng nhất là nó có khả năng được triển khai trên nhiều nền tảng nhất có thể, từ các

dòng máy tính để bàn cao cấp đến các thiết bị nhỏ gọn hơn như điện thoại di động. Vậy nên, Build-in Render Pipeline được phát triển là một giải pháp chìa khóa để thực hiện ý tưởng đa nền tảng được Unity hỗ trợ. Nó hỗ trợ các tính năng kết hợp đồ họa thuận tiện để sử dụng với các pipeline Forward và pipeline Deferred [1].

Tuy nhiên, Built-in Render Pipeline cũng có những hạn chế xung quanh. Phần lớn các mã nguồn trong này được viết bằng C++ và không có cơ chế sửa đổi hay tùy biến theo mục tiêu của nhà phát triển [1].

Scriptable Render Pipelines được phát triển để hỗ trợ một quy trình làm việc đa nền tảng. Nó có thể giúp nhà phát triển có thể thay đổi linh hoạt cho số lượng lớn nhất các thiết bị phần cứng, từ hiệu năng cao đến dòng sản phẩm hiệu năng thấp. Có thể cấu hình các quy trình Render bằng C# và C++. Có thể sử dụng C# rất có ý nghĩa vì các lệnh thực thi mới sẽ không cần phải biên dịch lại sau mỗi lần thay đổi. Linh hoạt trong việc hỗ trợ phát triển các kiến trúc hỗ trợ tạo nên ứng dụng. Trong SRPs, người dùng có thể tạo mới một SRPs từ đầu hoặc sửa đổi các SRPs có sẵn như HDRP hoặc URP nhằm phù hợp với mục tiêu sử dụng.



Hình 2.1: Mô hình đồ họa có thể lập trình mới [1].

Những điểm mạnh khi sử dụng URP:

- Dễ dàng tiếp cận, vì trong URP các shader có thể dễ dàng thực hiện hơn bằng chức năng shader graph có trong URP.
- Có thể mở rộng và sử dụng tùy biến: URP cho phép người dùng tùy chỉnh khả năng hiện có của Pipeline thành các khả năng mới phù hợp với nhu cầu. Giúp nó trở thành một lựa chọn tuyệt vời cho các nhà phát triển trình độ cao.
- Cung cấp nhiều tùy chọn Render, Universal Render hỗ trợ cả các đường dẫn Forward và Deferred.
- URP có hiệu năng ngang bằng hoặc cao hơn so với Built-in Render Pipeline. Nó có khả năng xác định ánh sáng một cách hiệu quả, trong render Forward, nó đánh giá tất cả các ánh sáng trong một lần duy nhất. Có những cải tiến về CPU và GPU khi vẽ các mesh nhờ có SRP Batcher, đảm bảo các lệnh callback sẽ ít hơn từ đó cải thiện cách xử lý độ sâu. Ngoài ra URP sử dụng bộ nhớ hiệu quả hơn trên các thiết bị di động, dẫn đến việc tiêu thụ ít điện năng hơn, tuổi thọ pin lâu hơn [1].

#### 2.4.2 Mixed Reality Toolkit 2

Microsoft Mixed Reality Toolkit dành cho Unity (MRTK-Unity) là một dự án của Microsoft trong lĩnh vực phát triển thiết bị thực tế kết hợp, cung cấp một bộ công cụ đầy đủ các tính năng để hỗ trợ và thúc đẩy quá trình phát triển ứng dụng thực tế ảo đa nền tảng được tích hợp trong engine Unity. MRTK-Unity mang lại nhiều ưu điểm và tính năng quan trọng giúp cho việc phát triển các ứng dụng XR trở nên dễ dàng và nhanh chóng hơn, bao gồm:

- MRTK-Unity linh hoạt với hệ thống đầu vào có thể tương thích trên nhiều nền tảng và cung cấp khối xây dựng để tạo ra trải nghiệm tương tác trong không gian thực và giao diện người dùng trực quan.
- Cung cấp một môi trường cho phép nhà phát triển có thể theo dõi, kiểm tra ngay lập tức sự thay đổi của chương trình. Hỗ trợ quá trình phát triển và kiểm thử.
- Vận hành như một môi trường phát triển mở rộng, cho phép các nhà phát triển có thể thay đổi các thành phần lõi của chương trình để đáp ứng các nhu cầu cụ thể của dự án.

### 2.4.3 Tổng quan về Vuforia Engine

Là một SDK Dành cho các thiết bị di động hoặc các thiết bị có hỗ trợ các tính năng về thực tế tăng cường. SDK này sử dụng thị giác máy tính để phát hiện và xử lý các hình ảnh hoặc model 3D trong thế giới thực thông qua camera của các thiết bị di động. Khi đã phát hiện hình ảnh hoặc model 3D trong thế giới thực SDK này bắt đầu hiển thị đối tượng tương ứng với góc nhìn của mục tiêu như vậy ta có thể nhìn vật thể ảo như một phần của thế giới thật.

Vuforia cung cấp API bằng các ngôn ngữ như Java, C++, .NET.. thông qua phần mở rộng cho Unity Engine. Bằng cách này SDK có thể hỗ trợ phát triển các ứng dụng trên IOS, Android và UWP, đồng thời việc phát triển ứng dụng trên Unity cũng giúp cho SDK này có thể linh động trong việc thay đổi nền tảng sử dụng.

Hơn nữa, Vuforia không chỉ hỗ trợ phát triển ứng dụng AR trên các nền tảng di động mà còn mở rộng khả năng đến các thiết bị đeo như kính AR. Điều này mở ra cơ hội cho việc tạo ra các trải nghiệm AR phong phú hơn, từ các ứng dụng giải trí và trò chơi đến các giải pháp doanh nghiệp như hướng dẫn lắp ráp, bảo trì thiết bị, và các công cụ hỗ trợ đào tạo thực tế.

## 2.5 Chuẩn giao tiếp I2C

Giao thức I2C (Inter-Integrated Circuit) là một chuẩn truyền thông ngắn dùng để kết nối các linh kiện điện tử trên cùng một bảng mạch in (PCB - Printed Circuit Board) hoặc giữa các thiết bị gần nhau. Được phát triển bởi Philips, hiện nay là NXP Semiconductors, I2C được thiết kế nhằm đơn giản hóa và tối ưu hóa quá trình truyền tải dữ liệu giữa các linh kiện trong hệ thống điện tử.

I2C hoạt động theo mô hình master/slave, với một thiết bị chủ (master) điều khiển quá trình truyền tải và một hoặc nhiều thiết bị con (slave) nhận lệnh từ thiết bị chủ. Giao thức này sử dụng hai đường truyền chính: SDA (Serial Data Line) để truyền dữ liệu và SCL (Serial Clock Line) để đồng bộ hóa dữ liệu giữa các thiết bị.

Một điểm hay của I2C là khả năng kết nối nhiều thiết bị trên cùng một đường dẫn dữ liệu và đường dẫn xung đồng hồ đồng bộ hóa, giúp giảm thiểu số lượng chân kết nối cần thiết và tăng khả năng tích hợp trong các hệ thống phức tạp. Mỗi thiết bị trên bus I2C có một địa chỉ riêng biệt, giúp thiết bị chủ dễ dàng xác định và giao tiếp với từng thiết bị con cụ thể.

I2C hỗ trợ hai chế độ địa chỉ: 7-bit và 10-bit. Chế độ 7-bit phổ biến hơn và phù hợp với hầu hết các ứng dụng. Chế độ 10-bit cung cấp sự linh hoạt hơn cho các hệ thống yêu cầu số lượng thiết bị lớn hơn hoặc địa chỉ mở rộng.

Tốc độ truyền tải dữ liệu của I2C có thể điều chỉnh để phù hợp với yêu cầu của từng ứng dụng, thường được đo bằng kHz và có thể đạt đến MHz cho các ứng dụng cần tốc độ cao.

Giao thức I2C được sử dụng rộng rãi trong nhiều loại thiết bị điện tử như cảm biến, bộ điều khiển, EEPROM (Electrically Erasable Programmable Read-Only Memory), và các vi điều khiển khác. Nhờ tính linh hoạt, đơn giản và khả năng tích hợp cao, I2C trở thành một lựa chọn phổ biến trong việc kết nối các thành phần trong hệ thống điện tử hiện đại.

## 2.6 Chuẩn giao tiếp RS485 và giao thức RS485 Modbus RTU

RS485 là một chuẩn truyền thông nối tiếp được sử dụng rộng rãi trong các ứng dụng công nghiệp nhờ khả năng truyền dữ liệu ổn định qua khoảng cách xa và môi trường nhiễu. Chuẩn này được định nghĩa bởi EIA/TIA (Electronic Industries Alliance/Telecommunications Industry Association) và có tên gọi đầy đủ là EIA/TIA-485. Một trong những đặc điểm nổi bật của RS485 là khả năng hỗ trợ truyền thông đa điểm (multi-point), cho phép kết nối lên đến 32 thiết bị trên cùng một đường truyền, mở rộng khả năng ứng dụng trong các hệ thống tự động hóa và điều khiển.

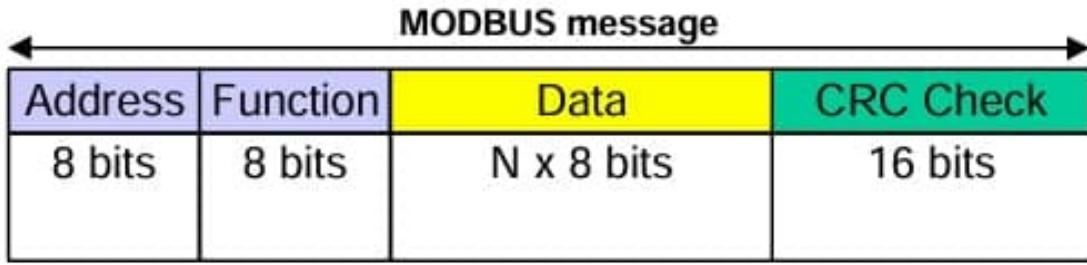
RS485 sử dụng cặp dây xoắn đôi (twisted pair) để truyền dữ liệu, giúp giảm thiểu nhiễu điện từ và cho phép truyền dữ liệu ở khoảng cách lên đến 1200 mét với tốc độ truyền tải tối đa là 10 Mbps (tốc độ truyền tải giảm khi khoảng cách tăng lên). Điều này làm cho RS485 trở thành lựa chọn lý tưởng cho các hệ thống cần kết nối nhiều thiết bị cách xa nhau trong môi trường công nghiệp.

Trong hệ thống RS485, dữ liệu được truyền theo dạng tín hiệu vi sai (differential signal), nghĩa là thông tin được mã hóa thành hai tín hiệu ngược pha nhau. Điều này giúp tăng khả năng chống nhiễu và đảm bảo tính toàn vẹn của dữ liệu ngay cả trong môi trường có nhiễu cao. Các thiết bị truyền thông RS485 thường hoạt động theo mô hình master/slave hoặc peer-to-peer, tùy thuộc vào giao thức cụ thể được sử dụng.

Một trong những giao thức phổ biến nhất sử dụng chuẩn RS485 là Modbus RTU (Remote Terminal Unit). Modbus RTU là một giao thức truyền thông serial đơn giản và mạnh mẽ, được phát triển bởi Modicon vào cuối những năm 1970. Giao thức này được thiết kế để truyền thông tin giữa các thiết bị điện tử thông qua các mạng nối tiếp như RS232, RS485 và RS422.

Modbus RTU hoạt động theo mô hình master/slave, trong đó một thiết bị master gửi yêu cầu tới các thiết bị slave và các slave này phản hồi lại master. Mỗi thông điệp Modbus RTU bao gồm một địa chỉ thiết bị (device address), một mã chức năng (function code), dữ liệu (data) và một kiểm tra lỗi (CRC - Cyclic Redundancy Check). Địa chỉ thiết

bị cho phép master xác định thiết bị nào sẽ nhận hoặc gửi dữ liệu, trong khi mã chức năng xác định hành động cần thực hiện (ví dụ: đọc dữ liệu, ghi dữ liệu, kiểm tra tình trạng thiết bị).



Hình 2.2: Cấu trúc cơ bản của một gói tin Modbus RTU

Modbus RTU là giao thức truyền thông được ưa chuộng trong các ứng dụng công nghiệp do tính đơn giản, khả năng mở rộng và tính tương thích cao. Giao thức này cho phép truyền dữ liệu hiệu quả và tin cậy giữa các thiết bị như PLC (Programmable Logic Controller), cảm biến, bộ điều khiển và hệ thống SCADA (Supervisory Control and Data Acquisition). Với khả năng truyền thông qua RS485, Modbus RTU có thể hoạt động trong các môi trường nhiễu cao và trên các khoảng cách xa, điều mà các giao thức khác khó có thể đạt được.

Nhờ vào tính linh hoạt và hiệu quả, RS485 và Modbus RTU đã trở thành tiêu chuẩn trong các hệ thống tự động hóa công nghiệp, cung cấp giải pháp truyền thông ổn định và tin cậy cho các ứng dụng yêu cầu khả năng kết nối đa điểm và truyền dữ liệu qua khoảng cách lớn.

## 2.7 Giao thức truyền thông MQTT

MQTT (Message Queuing Telemetry Transport) là một giao thức truyền thông nhẹ, được thiết kế để truyền tải dữ liệu giữa các thiết bị trong môi trường máy tính và mạng máy tính. Giao thức này được phát triển nhằm đáp ứng nhu cầu truyền thông đơn giản, hiệu quả và tiết kiệm năng lượng trong các hệ thống IoT (Internet of Things).

MQTT sử dụng mô hình kiến trúc client-server, nơi các thiết bị được phân chia thành người gửi (publisher) và người nhận (subscriber). Điều này giúp tối ưu hóa quá trình truyền tải thông tin giữa các thành phần trong hệ thống. Mỗi thiết bị MQTT có thể tham gia vào nhiều chủ đề (topic), và dữ liệu được truyền tải qua các chủ đề này.

Một trong những điểm mạnh của MQTT là tính nhẹ và linh hoạt. Giao thức này hoạt động theo mô hình publish/subscribe, cho phép các thiết bị đăng ký để nhận thông tin từ các chủ đề mà chúng quan tâm, và cũng có thể gửi dữ liệu đến các chủ đề đó. Điều này giúp giảm lưu lượng dữ liệu truyền tải, tối ưu hóa băng thông mạng. MQTT hỗ trợ

cơ chế QoS (Quality of Service) để đảm bảo chất lượng truyền tải dữ liệu. Có ba mức QoS trong MQTT: 0 (at most once), 1 (at least once), và 2 (exactly once). Mức QoS này xác định mức độ an toàn và chính xác của dữ liệu truyền tải.

MQTT được thiết kế để hoạt động trên nhiều loại mạng và giao thức truyền tải dữ liệu, bao gồm TCP/IP, WebSocket và các giao thức truyền tải không đồng bộ khác. Điều này giúp MQTT dễ dàng tích hợp vào các hệ thống hiện có. MQTT đã trở thành một tiêu chuẩn quan trọng trong lĩnh vực IoT, hỗ trợ truyền tải dữ liệu liên tục giữa hàng tỷ thiết bị. Sự đơn giản, hiệu quả và khả năng mở rộng của MQTT đã khiến nó trở thành lựa chọn phổ biến cho các ứng dụng giám sát và điều khiển trong IoT, cung cấp nền tảng ổn định và đáng tin cậy cho các hệ thống phức tạp.

## 2.8 Giao thức HTTP và kiến trúc REST API

### 2.8.1 Giao thức HTTP

HTTP (Hypertext Transfer Protocol) là một giao thức ứng dụng cho các hệ thống thông tin siêu văn bản phân tán, hợp tác và đa phương tiện. HTTP là nền tảng của mọi giao tiếp dữ liệu trên World Wide Web (WWW). Nó xác định cách các thông điệp được định dạng và truyền tải, và cách các trình duyệt và máy chủ web đáp ứng với các lệnh khác nhau.

HTTP hoạt động theo mô hình client-server, nơi một client gửi một yêu cầu HTTP tới máy chủ, và máy chủ trả về một phản hồi HTTP. Một yêu cầu HTTP bao gồm các thành phần như phương thức (GET, POST, PUT, DELETE), URL, và các header. Phản hồi HTTP bao gồm mã trạng thái (status code), header, và nội dung trả về (body).

Các phương thức HTTP thường thấy:

- GET: Yêu cầu tài nguyên từ máy chủ.
- POST: Gửi dữ liệu tới máy chủ để tạo một tài nguyên mới.
- PUT: Cập nhật một tài nguyên hiện có trên máy chủ.
- DELETE: Xóa một tài nguyên từ máy chủ

Các mã trạng thái của gói tin HTTP:

- 1xx (Informational): Yêu cầu nhận và tiếp tục xử lý.
- 2xx (Success): Yêu cầu đã được xử lý thành công.
- 3xx (Redirection): Cần thêm hành động để hoàn thành yêu cầu.

- 4xx (Client Error): Lỗi từ phía client.
- 5xx (Server Error): Lỗi từ phía máy chủ.

### 2.8.2 Kiến trúc REST API

REST (Representational State Transfer) là một phong cách kiến trúc được sử dụng để thiết kế các dịch vụ web nhẹ, có thể mở rộng và có thể tương tác dễ dàng với các ứng dụng khác. REST API là một ứng dụng của HTTP cho phép các hệ thống khác nhau giao tiếp với nhau thông qua các giao thức tiêu chuẩn.

Một số nguyên tắc khi thiết kế REST API:

- Giao diện Thông Nhất (Uniform Interface): Sử dụng các phương thức HTTP tiêu chuẩn để thực hiện các thao tác CRUD (Create, Read, Update, Delete) trên các tài nguyên.
- Trạng Thái Phiên (Stateless): Mỗi yêu cầu từ client tới máy chủ phải chứa đủ thông tin để máy chủ hiểu và xử lý yêu cầu, mà không cần sử dụng trạng thái phiên lưu trữ trên máy chủ.
- Tài Nguyên Được Định Danh Duy Nhất: Mỗi tài nguyên trên máy chủ được xác định bởi một URL duy nhất.
- Khả Năng Chia Tách Client-Server: Client và server có thể được phát triển và thay đổi một cách độc lập, miễn là giao diện API không thay đổi.

Các ưu điểm khi sử dụng REST API:

- Đơn Giản và Linh Hoạt: REST API dễ dàng thiết kế và triển khai, phù hợp với nhiều loại ứng dụng khác nhau.
- Hiệu Suất Cao: Nhờ vào tính chất stateless, REST API có thể xử lý một số lượng lớn các yêu cầu đồng thời.
- Khả Năng Mở Rộng: REST API có thể dễ dàng mở rộng và tích hợp với các dịch vụ và hệ thống khác.

## 2.9 Sóng Lora và mạng LoraWan

### 2.9.1 Sóng Lora

LoRa (Long Range) là một công nghệ truyền thông không dây được thiết kế đặc biệt để truyền dữ liệu trên khoảng cách xa với mức tiêu thụ năng lượng thấp. Được

phát triển bởi Semtech Corporation, LoRa sử dụng kỹ thuật điều chế dải tần trai (spread spectrum modulation) để cung cấp khả năng truyền tải ổn định và tin cậy ngay cả trong các môi trường có nhiễu cao. Công nghệ này hoạt động chủ yếu trong các băng tần không cần cấp phép (unlicensed bands) như ISM (Industrial, Scientific, and Medical), bao gồm các băng tần 868 MHz ở châu Âu, 915 MHz ở Bắc Mỹ và 433 MHz ở châu Á.

Một trong những ưu điểm nổi bật của LoRa là khả năng truyền dữ liệu trên khoảng cách xa, lên đến 15-20 km trong điều kiện lý tưởng ở khu vực nông thôn và vài km trong môi trường đô thị với nhiều vật cản. Đồng thời, LoRa có thể truyền tải dữ liệu với tốc độ thấp, nhưng bù lại nó tiêu thụ rất ít năng lượng, làm cho nó trở thành lựa chọn lý tưởng cho các ứng dụng Internet of Things (IoT) yêu cầu pin lâu dài và truyền dữ liệu không liên tục.

### 2.9.2 Mạng LoraWan

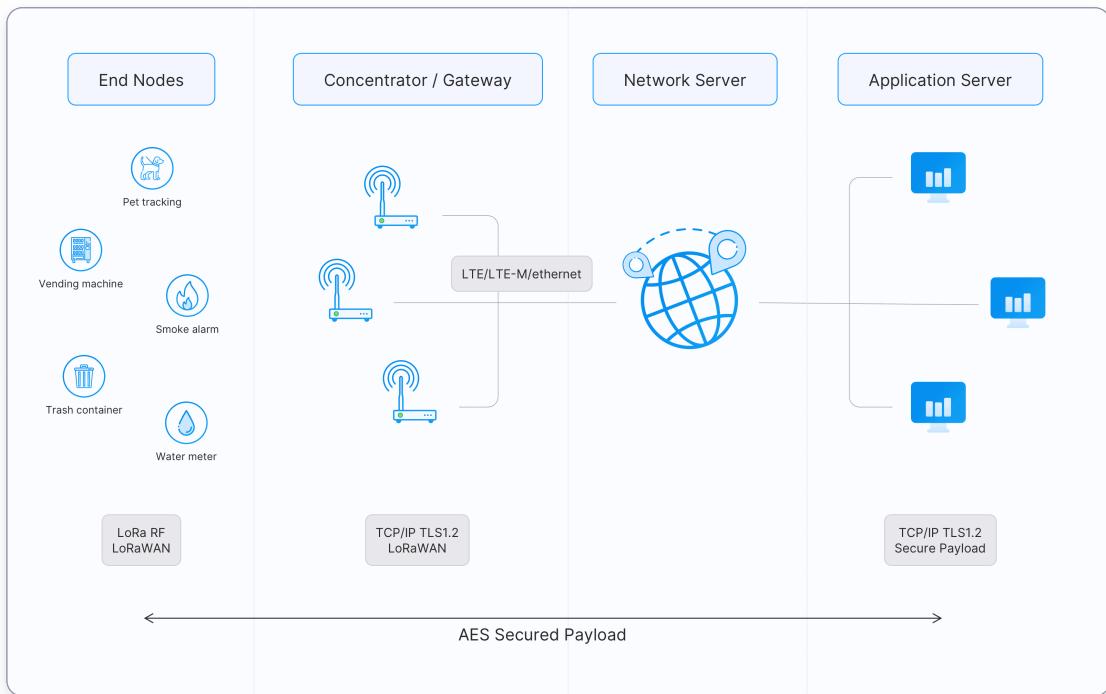
LoRaWAN (Long Range Wide Area Network) là giao thức mạng LPWAN (Low Power Wide Area Network-Mạng diện rộng công suất thấp) tiết kiệm năng lượng được phát triển cho việc kết nối các thiết bị hoạt động bằng pin với internet trong các mạng khu vực, mạng quốc gia, thậm chí là toàn cầu. Giao thức này được xác định và hỗ trợ như một tiêu chuẩn chính thức bởi Liên minh Viễn thông Quốc tế (ITU) theo tên gọi ITU-T Y.4480.

LoRaWAN đáp ứng yêu cầu của Internet of Things (IoT) bằng cách cung cấp một phương tiện giao tiếp hai chiều, đảm bảo bảo mật từ đầu đến cuối, hỗ trợ tính di động và cung cấp dịch vụ định vị. Điều này cho phép các thiết bị IoT truyền dữ liệu về cơ sở hạ tầng mạng LoRaWAN, từ đó kết nối với internet và các ứng dụng trên đám mây.

Sự phát triển và quản lý tiếp theo của giao thức LoRaWAN được thực hiện bởi LoRa Alliance (Liên minh LoRa), đây là tổ chức phi lợi nhuận, trong đó Semtech mang vai trò là thành viên sáng lập. Tổ chức này tập trung vào việc mở rộng và phát triển cộng đồng LoRaWAN, thúc đẩy việc sử dụng giao thức này trong các ứng dụng thực tế và đảm bảo tính tương thích và mở rộng của nó trên toàn cầu.

Một mạng LoRaWAN điển hình sẽ có kiến trúc bao gồm các thiết bị đầu cuối, trạm trung gian, máy chủ mạng (network server) và máy chủ ứng dụng:

- Thiết Bị Cuối (End Devices): Thiết bị cuối là những thiết bị IoT gửi dữ liệu đến cơ sở hạ tầng mạng và nhận các lệnh hoặc dữ liệu từ đó. Điều đặc biệt về thiết bị cuối là chúng có thể hoạt động trong thời gian dài với pin, giúp tiết kiệm năng lượng và mở rộng vận hành.
- Trạm trung gian (Gateway): Gateway chịu trách nhiệm thu thập dữ liệu từ các thiết



Hình 2.3: Ảnh giới thiệu về kiến trúc của mạng LoRaWAN

bị cuối và chuyển nó đến mạng LoRaWAN. Các gateway thường được đặt ở vị trí cao để đảm bảo phủ sóng rộng và tối ưu hóa việc thu thập dữ liệu từ các thiết bị.

- **Máy Chủ Mạng (Network Servers):** Máy chủ mạng quản lý toàn bộ hệ thống, bao gồm việc xử lý dữ liệu, quản lý kết nối và đảm bảo an ninh. Máy chủ này cũng chịu trách nhiệm xác thực và định tuyến dữ liệu đến các ứng dụng cuối cùng.
- **Máy chủ ứng dụng (Application Server):** Xử lý các dữ liệu nhận được từ máy chủ mạng, lưu trữ chúng và cho phép ứng dụng khác truy cập vào để sử dụng.

Một phần quan trọng trong kiến trúc của LoRaWAN là phân loại các thiết bị theo các lớp khác nhau, mỗi lớp có các đặc điểm và ứng dụng riêng. Hiểu về các lớp thiết bị này giúp tối ưu hóa việc sử dụng LoRaWAN cho các ứng dụng cụ thể. Dưới đây là chi tiết về ba lớp thiết bị chính của LoRaWAN:

- **Lớp A:** là lớp thiết bị cơ bản và tiết kiệm năng lượng nhất trong LoRaWAN. Thiết bị chỉ mở kênh truyền thông khi có dữ liệu để gửi. Thiết bị ở trạng thái ngủ phần lớn thời gian và chỉ thức dậy để gửi dữ liệu. Sau khi gửi dữ liệu, thiết bị mở hai cửa sổ nhận ngắn để nhận dữ liệu từ máy chủ nếu có. Do chu kỳ ngủ dài, việc gửi dữ liệu xuống (downlink) bị hạn chế và chỉ có thể thực hiện sau khi thiết bị gửi dữ liệu lên (uplink). Thường được sử dụng cho các thiết bị đo nhiệt độ, độ ẩm, hoặc cảm biến chất lượng không khí, các thiết bị giám sát pin, hoặc thiết bị không cần giao tiếp thường xuyên [2].

- Lớp B: cải tiến hơn lớp A bằng cách cung cấp các cửa sổ nhận định kỳ. Thiết bị đồng bộ hóa với các beacon từ gateway để mở cửa sổ nhận dữ liệu vào những thời điểm xác định. Tăng khả năng nhận dữ liệu xuống (downlink) vì thiết bị mở cửa sổ nhận theo lịch trình định kỳ, không phụ thuộc vào việc gửi dữ liệu lên (uplink). Thường được sử dụng cho các thiết bị cần phản ứng nhanh với các tín hiệu từ hệ thống trung tâm, các ứng dụng cần gửi lệnh định kỳ tới thiết bị, như đèn đường hoặc hệ thống tưới tiêu thông minh [2].
- Lớp C: có cửa sổ nhận liên tục, chỉ đóng khi thiết bị gửi dữ liệu. Điều này giúp thiết bị luôn sẵn sàng nhận dữ liệu từ máy chủ. Do luôn trong trạng thái nghe, thiết bị lớp C tiêu thụ năng lượng nhiều hơn so với lớp A và lớp B. Được sử dụng cho các thiết bị cần giao tiếp liên tục, như van điều khiển, máy móc công nghiệp [2].

Trong mạng LoRaWAN, mỗi thiết bị đầu cuối cần phải được kích hoạt trước khi có thể gửi và nhận tin nhắn. Việc kích hoạt này đảm bảo rằng thiết bị được nhận diện chính xác và có thể giao tiếp an toàn với mạng. Có hai phương thức kích hoạt chính được sử dụng: Kích hoạt qua không khí (Over-The-Air Activation - OTAA) và Kích hoạt bằng cá nhân hóa (Activation By Personalization - ABP).

OTAA là phương thức kích hoạt an toàn và được khuyến nghị nhất. Trong quy trình này, thiết bị thực hiện một quy trình tham gia với mạng. Thiết bị gửi một yêu cầu tham gia (Join-request) đến mạng, sau đó mạng sẽ gửi lại một chấp nhận tham gia (Join-accept). Trong quá trình này, địa chỉ thiết bị động (DevAddr) và các khóa bảo mật (AppSKey, NwkSKey) được tạo ra và trao đổi một cách an toàn. Các bước chính trong quy trình OTAA bao gồm:

- Thiết bị gửi Join-request với các thông tin như AppEUI, DevEUI, và DevNonce.
- Mạng xử lý yêu cầu và gửi lại Join-accept với các thông tin như AppNonce, NetID, và DevAddr.
- Thiết bị sử dụng AppKey để giải mã thông điệp Join-accept và tạo ra các khóa phiên (session keys).

Trong phương thức ABP, địa chỉ thiết bị và các khóa bảo mật được gán trực tiếp vào thiết bị từ trước. Điều này có nghĩa là các thông tin như DevAddr, NwkSKey, và AppSKey phải được lập trình vào thiết bị thủ công trước khi triển khai. Phương thức ABP ít an toàn hơn so với OTAA và không cho phép thiết bị chuyển đổi nhà cung cấp mạng mà không cần thay đổi thủ công các khóa bảo mật. Các bước chính trong ABP bao gồm:

- Gán trực tiếp DevAddr, NwkSKey, và AppSKey vào thiết bị.
- Thiết bị bắt đầu giao tiếp với mạng mà không cần quy trình tham gia bổ sung.

# Chương 3

## THIẾT KẾ VÀ THI CÔNG HỆ THỐNG

### 3.1 Yêu cầu và đặc tính của hệ thống

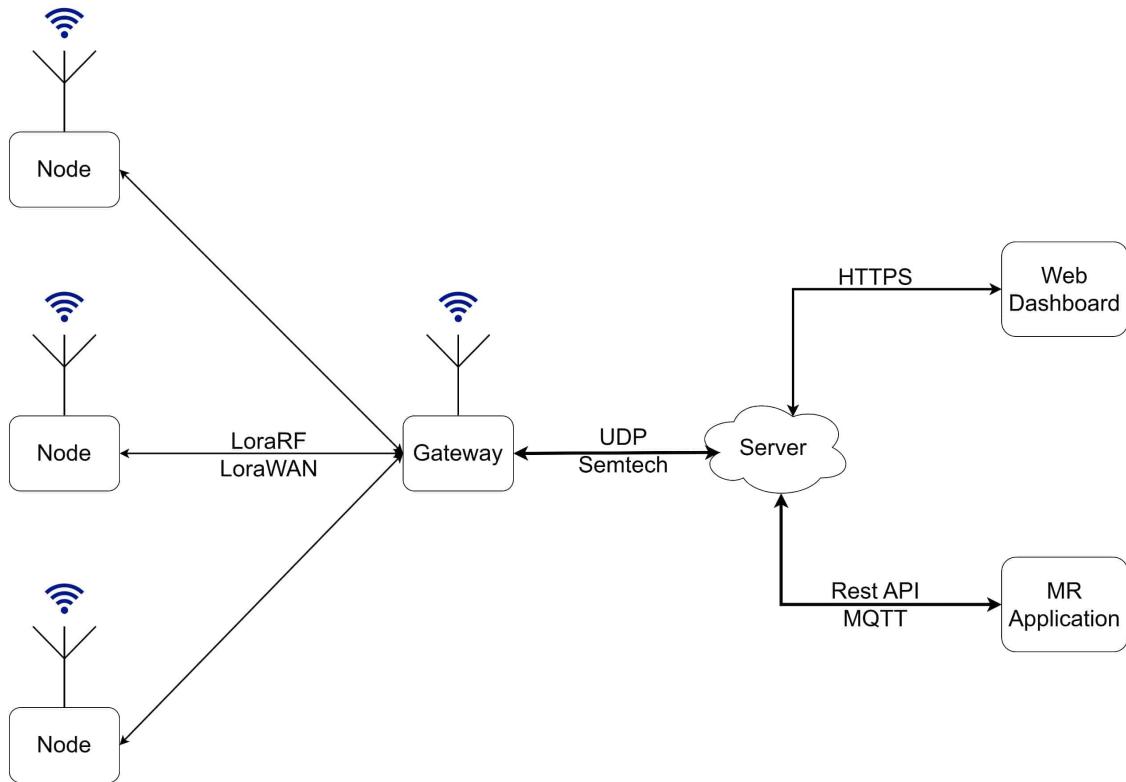
#### 3.1.1 Yêu cầu của hệ thống

Trong thời gian tìm hiểu cách giám sát quá trình phát triển của cây trồng của những người nông dân và các kỹ sư nông nghiệp, nhóm thực hiện đề tài đã đúc kết một số kinh nghiệm và nghiên cứu thiết kế một mô hình đáp ứng các yêu cầu sau:

- Thiết kế cảm biến thu thập được các dữ liệu liên quan môi trường cây trồng, cảm biến hỗ trợ chuẩn giao thức RS485 Modbus RTU. Cảm biến phải có kích thước nhỏ gọn, có thể đáp ứng cho các ứng dụng khác.
- Thiết kế một bộ trạm thu thập dữ liệu chạy bằng pin, với tiêu chí tiết kiệm năng lượng, hỗ trợ sạc bằng năng lượng mặt trời. Trạm này có thể kết nối với nhiều loại cảm biến khác nhau có sẵn trên thị trường qua chuẩn giao thức RS485 Modbus RTU.
- Máy chủ trung tâm hoạt động như một thiết bị điện toán đám mây, cho phép lưu trữ dữ liệu các cảm biến, cung cấp các phương thức cho phép các ứng dụng khác truy vấn dữ liệu, chạy các thuật toán xử lý các chức năng đặc biệt.
- Ứng dụng MR được thiết kế có thể quan sát các thông số môi trường. Hệ thống phải hoạt động ổn định không gặp phải tình trạng bị quá tải. Đảm bảo hiệu suất phản hồi nhanh chóng trong mọi tình huống sử dụng.
- Thiết kế giao diện người dùng thân thiện, trực quan dễ nắm bắt. Các yếu tố về UI được bố trí logic và dễ tiếp cận. Giúp người dùng có thể dễ dàng thao tác với hệ thống.
- Ứng dụng MR cần được thiết kế với khả năng mở rộng và tương thích cao, cho phép dễ dàng thêm các tính năng mới và tương thích với các thiết bị, công nghệ trong tương lai. Cũng như dễ dàng chuyển đổi sang một số thiết bị XR khác tùy vào mục đích sử dụng.
- Khả năng nhận diện vật thể nhanh chóng, các mô hình kỹ thuật số được hiển thị rõ ràng và chi tiết.

### 3.1.2 Sơ đồ tổng quan

Về tổng thể thì mô hình này được lên ý tưởng từ sự kết hợp của cả hai công nghệ hoạt động riêng biệt bổ trợ cho nhau như hình sau:



Hình 3.1: Mô hình hệ thống quản lý nông nghiệp kết hợp công nghệ MR

- Công nghệ IoT cung cấp khả năng đọc dữ liệu từ các cảm biến sử dụng giao thức RS485 Modbus RTU. Các dữ liệu này có thể gửi lên máy chủ xử lý dữ liệu thông qua một trạm trung gian từ một khoảng cách rất xa bằng công nghệ sóng Lora với giao thức LoraWan. Các cảm biến sẽ được lắp đặt vào các trạm thu thập dữ liệu được đặt cách trạm trung gian trong bán kính vài kilomet. Các trạm thu thập dữ liệu sẽ hoạt động theo một chu kỳ cố định. Bắt đầu chu kì, trạm này sẽ thức dậy, bật các phần cứng cần thiết, đọc toàn bộ dữ liệu cảm biến, tắt các phần cứng sau khi đọc, nén dữ liệu số lại thành một gói tin, gửi lên trạm trung gian, chờ nhận được gói tin xác nhận và đi ngủ. Trạm trung gian đóng vai trò trung chuyển gói tin từ trạm thu thập dữ liệu và gửi dữ liệu đó lên máy chủ quản lý qua internet, sau đó lại chuyển gói tin xác nhận từ máy chủ về trạm thu thập dữ liệu. Máy chủ quản lý phụ trách quản lý các trạm thu thập dữ liệu và trạm trung chuyển, nó có nhiệm vụ xác thực gói tin, giải mã dữ liệu, phản hồi gói tin xác nhận. Các dữ liệu sau khi được giải mã sẽ gửi đến các ứng dụng khác bằng giao thức MQTT để lưu trữ và trình chiếu trên các bảng điều khiển. Các ứng dụng ngoài máy chủ có thể lấy các

dữ liệu môi trường qua các giao thức HTTP hoặc MQTT.

- Công nghệ MR đóng vai trò trung tâm giữa người dùng với hệ thống IoT, thông qua ứng dụng tích hợp công nghệ MR này, người dùng có thể quan sát và quản lý dữ liệu từ cơ sở dữ liệu được lưu trữ trên hệ thống thông qua Rest API và giao thức MQTT một cách trực quan và sinh động. Giao diện MR giúp cho người dùng dễ dàng theo dõi các thông số quan trọng như độ ẩm, nhiệt độ, nồng độ CO<sub>2</sub>, cường độ ánh sáng, năng lượng còn lại... Cũng như là tình trạng phát triển của cây trồng trong môi trường thực tế, nhờ vào sự kết hợp giữa thực và ảo. Người dùng có thể dễ dàng tương tác với các nút điều khiển ảo xuất hiện trước mắt một cách vô cùng tiện lợi và nhanh chóng. Các nút điều khiển này có thể là chức năng hiển thị cũng như là điều chỉnh trạng thái của các thiết bị điện. Cuối cùng ứng dụng có cung cấp thêm các hướng dẫn chi tiết và trực quan nhằm hỗ trợ người dùng trong quá trình thực hiện các thao tác kỹ thuật như lắp ráp, cài đặt và bảo trì phần cứng. Thông qua các mô hình 3D ảo trực tiếp xuất hiện trên vật thể thật để hướng dẫn.

### 3.1.3 Đặc tính kỹ thuật

Để hệ thống được vận hành một cách ổn định cần phải đáp ứng được các đặc tính kỹ thuật sau:

- Cảm biến được thiết kế với dây điện áp hoạt động rộng, đáp ứng được nhiều trường hợp ứng dụng khác nhau.
- Trạm thu thập dữ liệu có thể kiểm soát năng lượng để đảm bảo khả năng duy trì hoạt động trong điều kiện không được sạc trong thời gian dài.
- Ứng dụng MR đảm bảo có tần số quét ổn định trên thiết bị có thể chạy XR thông dụng.
- Các tiến trình xử lý được bố trí hợp lý, tạo cảm giác thoải mái cho người sử dụng.
- Tương thích đa nền tảng, đảm bảo ứng dụng hoạt động tốt trên nhiều thiết bị khác nhau, từ điện thoại di động, máy tính bảng đến các thiết bị đeo như HoloLens, Magic Leap, ..

## 3.2 Đặc tả hệ thống

### 3.2.1 Phần mềm sử dụng.

Đồ án này đã sử dụng một số phần mềm chuyên dụng để hỗ trợ cho quá trình nghiên cứu và thực hiện. Những phần mềm dưới đây đã đóng góp quan trọng vào sự thành công

của đồ án, giúp hoàn thiện công việc một cách hiệu quả và chuyên nghiệp:

Bảng 3.1: Danh sách các phần mềm được sử dụng để xây dựng hệ thống.

<b>Stt</b>	<b>Phần mềm</b>	<b>Phiên bản</b>	<b>Loại</b>	<b>Chức năng</b>
1	Unity Engine	2022.3.14f1	Engine	Là môi trường phát triển, gõ lõi ứng dụng chính.
2	Visual Studio	2022	IDE	Môi trường lập trình chính cho ứng dụng MR.
3	Blender	4.0	App	Vẽ các model 3D.
4	AR Core	5.0.7	App	Hỗ trợ sử dụng MR trên một số dòng điện thoại.
5	Model target generator	10.22	App	Dùng cho mô hình Vuforia học model.
6	Microsoft Mixed Reality Feature Tool	1.0.2209.0	App	Thêm các gói SDK MRTK2.
7	Altium Designer	21.0.9	App	Thiết kế PCB cho các cảm biến và trạm thu thập dữ liệu .
8	MounRiver Studio	1.91	IDE	Môi trường lập trình và gõ lõi cho cảm biến
9	Arduino IDE	2.3.2	App	Môi trường lập trình cho trạm thu thập cảm biến.
10	Power Profiler	4.1.0	App	Phần mềm hỗ trợ đánh giá năng lượng tiêu hao của trạm thu thập dữ liệu.
11	Logic 2	2.4.14	App	Phần mềm phân tích logic của các chuẩn giao thức ngoại vi của vi điều khiển qua một mô đun hỗ trợ.
12	Sublime Text	4169	App	Trình soạn thảo văn bản dùng để cấu hình máy chủ.

### 3.2.2 Danh sách các chức năng

Sau đây sẽ là danh sách về những chức năng trong phần mềm và phần cứng được phát triển:

Bảng 3.2: Danh sách các chức năng của hệ thống

<b>Stt</b>	<b>Chức năng</b>	<b>Phiên bản</b>	<b>Mô tả</b>
1	Giám sát dữ liệu thông qua bảng thông tin và biểu đồ trên MR	1.0	Thông tin chi tiết về các trạm sẽ được hiển thị trên bảng, cho phép người dùng xem dữ liệu thời gian thực một cách dễ dàng. Bảng này không chỉ cung cấp dữ liệu tại thời điểm hiện tại mà còn tích hợp các biểu đồ trực quan, giúp người dùng theo dõi và phân tích các thông số đã được ghi nhận theo thời gian thực.
2	Tự động xác định trạm thu thập dữ liệu ở gần	1.0	Dữ liệu về kinh độ và vĩ độ của người dùng sẽ liên tục được cập nhật lên hệ thống, từ đó xác định được trạm ở gần nhất để hiển thị dữ liệu tương ứng, tối ưu trải nghiệm của người dùng.
3	Điều khiển thiết bị	1.0	Người dùng có thể sử dụng điều khiển các thiết bị chiếu sáng hoặc tưới tiêu từ xa thông qua ứng dụng MR hoặc trang web.
5	Cấu hình chu kỳ lấy dữ liệu	1.0	Người dùng có thể can thiệp vào thời gian ngủ của trạm thu thập dữ liệu theo từng ứng dụng khác nhau.
6	Hướng dẫn cách lắp đặt thiết bị	1.0	Ứng dụng sẽ có giao diện hướng dẫn người dùng cách kết nối các phần cứng bên ngoài để thiết bị có thể hoạt động.
7	Giới thiệu thông tin phần cứng	1.0	Ứng dụng sẽ cung cấp các thông tin chi tiết của từng thành phần mà người dùng có thể tác động lên phần cứng của trạm.

### 3.3 Một số phần cứng được sử dụng

#### 3.3.1 Máy tính nhúng Raspberry Pi 4

Raspberry Pi 4, được phát hành bởi Raspberry Pi Foundation vào tháng 6 năm 2019, là một phiên bản cải tiến với hiệu suất vượt trội và nhiều tính năng mới. Nó sở hữu bộ vi xử lý mạnh mẽ, hỗ trợ nhiều dung lượng RAM khác nhau và khả năng kết nối đa dạng. Raspberry Pi 4 trở thành lựa chọn lý tưởng cho nhiều ứng dụng từ giáo dục, nghiên cứu đến phát triển các dự án công nghệ và IoT. Bảng 3.1 sẽ cung cấp các thông tin chi tiết về phần cứng của Raspberry Pi 4.

Bảng 3.3: Bảng thông tin phần cứng của Raspberry Pi 4 2GB và 8GB

Thông số kỹ thuật	Chi tiết
Bộ vi xử lý (CPU)	Broadcom BCM2711, Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
RAM	2GB, 4GB hoặc 8GB LPDDR4-3200 SDRAM
Kết nối mạng	Gigabit Ethernet
Wi-Fi	802.11 b/g/n/ac (2.4 GHz và 5.0 GHz)
Bluetooth	Bluetooth 5.0, BLE
Cổng USB	USB 3.0, 2 x USB 2.0
Cổng video	2 x micro HDMI (hỗ trợ tối đa 4Kp60)
GPIO	40-pin GPIO header
Lưu trữ	MicroSD, hỗ trợ boot từ USB 3.0
Nguồn cấp	5V/3A DC qua USB-C connector
Kết nối camera, màn hình	2-lane MIPI CSI
Kích thước	85.6 mm x 56.5 mm x 17 mm

#### 3.3.2 Bộ Kit Gateway Pi HAT RAK5146

Bộ Kit Gateway Pi HAT RAK5146 là một giải pháp hoàn hảo cho việc triển khai mạng LoRaWAN, được thiết kế để hoạt động liền mạch với Raspberry Pi, được sản xuất bởi RAKwireless. Bộ Kit được trang bị module RAK5146, hỗ trợ các băng tần LoRa toàn cầu và tích hợp với chuẩn LoRaWAN 1.0.x/1.1.x, có phiên bản hỗ trợ tần số 920-923MHz phù hợp với quy định tần số thiết bị LPWAN không cần đăng ký trong Thông tư 08/2021/TT-BTTTT [3]. Với khả năng phủ sóng rộng và hiệu suất cao, RAK5146 cung cấp khả năng truyền tải dữ liệu ổn định và đáng tin cậy, lý tưởng cho các ứng dụng IoT như giám sát môi trường, quản lý đô thị thông minh và các giải pháp công nghiệp. Bộ kit này dễ dàng lắp đặt và cấu hình, giúp người dùng nhanh chóng triển khai hệ thống

mạng LoRaWAN của riêng mình. Các thông tin chi tiết của RAK5146 được trình bày trong bảng sau:

Bảng 3.4: Bảng Gateway RAK5146

<b>Thông Số</b>	<b>Chi Tiết</b>
<b>Model</b>	RAK5146
<b>Băng tần hỗ trợ</b>	- EU868 - US915 - AU915 - AS923 - IN865
<b>Chuẩn LoRaWAN</b>	LoRaWAN 1.0.3/1.1
<b>Chipset LoRa</b>	Semtech SX1303
<b>Số lượng kênh</b>	8 kênh đa SF (Spreading Factor)
<b>Giao tiếp</b>	SPI
<b>Điện áp hoạt động</b>	5V (qua cổng USB-C của Raspberry Pi)
<b>Công suất đầu ra</b>	Tối đa 27 dBm
<b>Độ nhạy thu</b>	Tối đa -139 dBm
<b>Kích thước</b>	65mm x 30mm
<b>Nhiệt độ hoạt động</b>	-40°C đến 85°C
<b>Tích hợp GPS</b>	Có
<b>Ăng ten</b>	Kết nối qua cổng I-PEX (u.FL)
<b>Phần mềm hỗ trợ</b>	- ChirpStack - TTN (The Things Network) - LoRaServer
<b>Ứng dụng</b>	- Giám sát môi trường - Quản lý đô thị thông minh - Giải pháp công nghiệp - Hệ thống nông nghiệp thông minh

### 3.3.3 Module điều khiển RAK3172

Module RAK3172 là một lựa chọn mạnh mẽ và hiệu quả cho các ứng dụng IoT, được thiết kế bởi RAKwireless. Sử dụng chip Semtech SX1262, RAK3172 hỗ trợ các băng tần LoRa toàn cầu và tương thích với chuẩn LoRaWAN 1.0.x/1.1.x. Với kích thước nhỏ gọn và tiêu thụ năng lượng thấp, module này lý tưởng cho các ứng dụng cảm biến, giám sát môi trường, và các thiết bị IoT di động. Với sự tương thích hoàn toàn với các phiên bản chuẩn LoRaWAN 1.0.x và 1.1.x, RAK3172 đảm bảo tính tương thích cao và khả năng tích hợp dễ dàng vào các hệ thống mạng LoRaWAN hiện có. Điều này giúp các nhà phát triển tiết kiệm thời gian và công sức trong quá trình phát triển sản phẩm, đồng thời đảm bảo hiệu suất và độ tin cậy của hệ thống.

Ngoài ra, RAK3172 được thiết kế để tiêu thụ năng lượng rất thấp, làm cho nó trở thành một giải pháp lý tưởng cho các thiết bị IoT cần hoạt động trong thời gian dài mà

không cần thay pin thường xuyên. Điều này rất quan trọng trong các ứng dụng giám sát môi trường hoặc các thiết bị IoT di động, nơi mà việc tiết kiệm năng lượng là yếu tố then chốt. Các thông tin chi tiết của RAK3172 được trình bày trong bảng sau:

Bảng 3.5: Bảng thông tin chi tiết module RAK3172

Thông Số	Chi Tiết
<b>Model</b>	RAK3172
<b>Chipset LoRa</b>	Semtech SX1262
<b>Băng tần hỗ trợ</b>	- EU868 - US915 - AU915 - AS923 - IN865 - KR920
<b>Chuẩn LoRaWAN</b>	LoRaWAN 1.0.3/1.1
<b>Giao tiếp</b>	UART, SPI, I2C
<b>Điện áp hoạt động</b>	2,0V - 3,6V
<b>Công suất đầu ra</b>	Tối đa 22 dBm
<b>Độ nhạy thu</b>	Tối đa -148 dBm
<b>Tiêu thụ năng lượng</b>	- Chế độ ngủ: < 2 µA - Chế độ chờ: < 6 mA - Chế độ truyền: < 100 mA
<b>Kích thước</b>	15mm x 15mm x 2,5mm
<b>Nhiệt độ hoạt động</b>	-40°C đến 85°C
<b>Ăng ten</b>	Kết nối qua cổng I-PEX (u.FL) hoặc pad hàn PCB

### 3.3.4 IC sạc năng lượng mặt trời CN3791

IC CN3791 là một bộ điều khiển sạc pin lithium ion hoạt động theo chế độ chuyển đổi PWM, có thể sạc pin từ tấm pin năng lượng mặt trời với chức năng theo dõi điểm công suất tối đa với ít linh kiện bên ngoài. IC CN3791 được thiết kế đặc biệt để sạc pin lithium ion với chế độ dòng điện và điện áp liên tục. Bảng dưới đây sẽ cung cấp một số thông tin cơ bản của IC này:

Bảng 3.6: Bảng thông tin chi tiết IC CN3791

Thông Số	Chi Tiết
<b>Model</b>	CN3791
<b>Điện áp đầu vào</b>	4,4V - 28V
<b>Điện áp sạc đầu ra</b>	4,2V cho pin lithium-ion
<b>Chế độ hoạt động</b>	Tự động sạc và ngắt sạc, sạc MPPT
<b>Nhiệt độ hoạt động</b>	-40°C đến 85°C
<b>Kích thước gói</b>	SSOP-10

### 3.3.5 IC hạ áp TPS62840

IC TPS62840 là một bộ điều chỉnh điện áp hạ áp (buck converter) siêu hiệu quả, được thiết kế bởi Texas Instruments. Với dòng điện tĩnh có thể thấp đến mức 60 nA, IC đặc biệt phù hợp cho các ứng dụng yêu cầu tiêu thụ năng lượng thấp, TPS62840 cung cấp dòng điện đầu ra lên đến 750 mA và có hiệu suất cao ở mức tải nhẹ, giúp kéo dài tuổi thọ pin. Với thiết kế nhỏ gọn và các tính năng bảo vệ toàn diện, IC này lý tưởng cho các thiết bị điện tử cầm tay, cảm biến không dây và các ứng dụng IoT. Các thông số cơ bản của IC này sẽ được liệt kê dưới đây:

Bảng 3.7: Bảng thông tin chi tiết IC TPS62840

Thông Số	Chi Tiết
<b>Model</b>	TPS62840
<b>Điện áp đầu vào</b>	1,8V - 6,5V
<b>Điện áp đầu ra</b>	1,8V - 3,3V (có thể điều chỉnh)
<b>Dòng đầu ra tối đa</b>	750 mA
<b>Dòng tĩnh</b>	60 nA
<b>Chế độ hoạt động</b>	Chế độ tiết kiệm năng lượng (Power-Save Mode)
<b>Nhiệt độ hoạt động</b>	-40°C đến 125°C
<b>Kích thước gói</b>	SON-8
<b>Ứng dụng</b>	- Thiết bị đeo - Cảm biến không dây - Các thiết bị IoT - Điện thoại di động - Hệ thống năng lượng mặt trời

### 3.3.6 Vi điều khiển CH32V003

Vi điều khiển CH32V003 là một sản phẩm của WCH (Nanjing Qinheng Microelectronics), được thiết kế với hiệu suất cao và chi phí thấp, chỉ 3.700 VND cho mỗi chiếc. Được phát triển dựa trên lõi QingKe RISC-V2A, vi điều khiển này hỗ trợ tần số chính của hệ thống lên đến 48 MHz, đảm bảo độ hiệu quả cho các ứng dụng nhúng.

Khả năng giao tiếp đa dạng của CH32V003 là một trong những điểm mạnh của nó, hỗ trợ nhiều giao thức khác nhau như UART, SPI, và I2C. Điều này cho phép vi điều khiển dễ dàng kết nối và tương tác với nhiều loại cảm biến và thiết bị ngoại vi, mở rộng khả năng ứng dụng của nó trong nhiều lĩnh vực khác nhau.

Một yếu tố quan trọng khác của CH32V003 là mức tiêu thụ năng lượng thấp, làm cho nó trở thành một lựa chọn lý tưởng cho các ứng dụng yêu cầu tiết kiệm năng lượng. Điều này đặc biệt quan trọng đối với các thiết bị điện tử nhỏ gọn, có chức năng cố định hoặc các bộ chuyển đổi giao thức, nơi mà việc duy trì hoạt động lâu dài mà không cần

thay pin thường xuyên là yếu tố then chốt.

Nhờ vào những đặc điểm nổi bật này, vi điều khiển CH32V003 mang đến một giải pháp hiệu quả, tiết kiệm và linh hoạt cho các nhà phát triển và kỹ sư trong việc thiết kế các sản phẩm điện tử hiện đại và thông minh. Bảng dưới đây sẽ trình bày các đặc điểm nổi bật của CH32V003:

Bảng 3.8: Bảng thông tin chi tiết vi điều khiển CH32V003

Thông Số	Chi Tiết
<b>Model</b>	CH32V003
<b>Kiến trúc</b>	RISC-V
<b>Tốc độ xung nhịp</b>	Lên đến 48 MHz
<b>Bộ nhớ flash</b>	16 KB
<b>RAM</b>	2 KB
<b>Giao tiếp</b>	UART, I2C, SPI
<b>DMA</b>	1 bộ, 7 kênh
<b>ADC</b>	1 bộ, 10-bit
<b>Số chân I/O</b>	Lên đến 18 chân GPIO
<b>Điện áp hoạt động</b>	2,7V - 5,5V
<b>Tiêu thụ năng lượng</b>	- Chế độ hoạt động: < 6 mA @ 48 MHz - Chế độ chờ: < 10 µA
<b>Nhiệt độ hoạt động</b>	-40°C đến 85°C
<b>Kích thước gói</b>	SOP, TSSOP hoặc QFN

### 3.3.7 Cảm biến CO<sub>2</sub> SCD40

Cảm biến SCD40 là một loại cảm biến khí CO<sub>2</sub> nhỏ gọn và có hiệu suất cao, được phát triển bởi công ty Sensirion. Thiết bị này được thiết kế dựa trên nguyên lý cảm biến NDIR photoacoustic và tích hợp công nghệ PASens và CMOSens đã được cấp bằng sáng chế của Sensirion. Nhờ những công nghệ tiên tiến này, SCD40 có khả năng cung cấp các phép đo nồng độ CO<sub>2</sub> trong không khí với độ chính xác cao.

Kích thước nhỏ gọn của cảm biến SCD40 là một ưu điểm nổi bật, giúp nó dễ dàng tích hợp vào nhiều thiết bị và hệ thống khác nhau. Hơn nữa, cảm biến còn được trang bị các tính năng hiệu chỉnh tự động, đảm bảo hoạt động ổn định và đáng tin cậy trong thời gian dài. Vì vậy, SCD40 là lựa chọn lý tưởng cho nhiều ứng dụng khác nhau, bao gồm các hệ thống thông gió, quản lý chất lượng không khí trong nhà, và các thiết bị IoT. Các ứng dụng này không chỉ giúp cải thiện chất lượng không khí mà còn góp phần nâng cao

sức khỏe và sự thoải mái cho người sử dụng.

Bảng 3.9: Bảng thông tin chi tiết cảm biến SCD40

Thông Số	Chi Tiết
<b>Model</b>	SCD40
<b>Công nghệ đo</b>	Quang phổ kế hồng ngoại (Photoacoustic Sensing)
<b>Phạm vi đo CO<sub>2</sub></b>	400 - 2000 ppm
<b>Độ chính xác</b>	$\pm(50 \text{ ppm} + 5\% \text{ giá trị đo})$
<b>Phạm vi đo nhiệt độ</b>	- 10°C – 60°C
<b>Độ chính xác</b>	$\pm 1,5 \text{ }^{\circ}\text{C}$
<b>Phạm vi đo độ ẩm</b>	0 %RH – 100 %RH
<b>Độ chính xác</b>	$\pm 9 \% \text{ RH}$
<b>Giao tiếp</b>	I2C
<b>Điện áp hoạt động</b>	2,4V - 5,5V
<b>Kích thước</b>	10,1 mm x 10,1 mm x 6,5 mm
<b>Nhiệt độ hoạt động</b>	-10°C đến 60°C
<b>Tính năng đặc biệt</b>	<ul style="list-style-type: none"> <li>- Hiệu chỉnh tự động (Automatic Baseline Correction)</li> <li>- Bù nhiệt độ và độ ẩm</li> </ul>
<b>Ứng dụng</b>	<ul style="list-style-type: none"> <li>- Hệ thống thông gió - Quản lý chất lượng không khí trong nhà - Các thiết bị IoT - Thiết bị điều khiển HVAC</li> </ul>

### 3.3.8 Cảm biến ánh sáng BH1750

Cảm biến BH1750, phát triển bởi Rohm Semiconductor, là một thiết bị đo cường độ ánh sáng môi trường và chuyển đổi kết quả thành tín hiệu kỹ thuật số, dễ dàng tích hợp với vi điều khiển qua giao tiếp I2C. Với khả năng đo lường chính xác và tiêu thụ năng lượng thấp, BH1750 được sử dụng rộng rãi trong nhiều ứng dụng như hệ thống đèn tự động, thiết bị di động, và các hệ thống chiếu sáng thông minh. Cảm biến này giúp điều chỉnh độ sáng, cải thiện trải nghiệm người dùng và tiết kiệm năng lượng đáng kể. Ngoài ra, BH1750 còn có thiết kế nhỏ gọn và tính năng hiệu quả, làm cho nó trở thành lựa chọn lý tưởng cho các dự án IoT và các thiết bị điện tử tiên tiến khác. Khả năng đo ánh sáng trong dải rộng và độ nhạy cao cũng là điểm mạnh của BH1750, giúp nó đáp ứng tốt các yêu cầu khắt khe trong nhiều ứng dụng công nghiệp và tiêu dùng.

Bảng 3.10: Bảng thông tin chi tiết cảm biến BH1750

Thông Số	Chi Tiết
<b>Model</b>	BH1750
<b>Phạm vi đo</b>	0 - 65.535 lux
<b>Độ chính xác</b>	$\pm 20\%$
<b>Giao tiếp</b>	I2C
<b>Điện áp hoạt động</b>	2,4V - 3,6V
<b>Tiêu thụ năng lượng</b>	- Chế độ hoạt động: 0,19 mA - Chế độ chờ: 0,001 mA
<b>Độ phân giải</b>	1 lux
<b>Thời gian đo</b>	- Chế độ đo cao: 120 ms - Chế độ đo thấp: 16 ms
<b>Nhiệt độ hoạt động</b>	-40°C đến 85°C
<b>Kích thước gói</b>	6-pin SOP (2,8mm x 2,9mm)
<b>Ứng dụng</b>	- Thiết bị di động - Đèn tự động - Hệ thống chiếu sáng thông minh - Thiết bị gia dụng

### 3.3.9 IC MAX3485EESA+T

IC MAX3485EESA+T là bộ chuyển đổi tín hiệu RS-485/RS-422 do Maxim Integrated phát triển, hỗ trợ truyền dữ liệu lên đến 10 Mbps. Với khả năng truyền thông hai chiều trên một cặp dây đơn, IC này lý tưởng cho các ứng dụng công nghiệp và hệ thống tự động hóa, đảm bảo truyền dữ liệu nhanh chóng, ổn định và chính xác.

Bảng 3.11: Bảng thông tin chi tiết IC MAX3485EESA+T

Thông Số	Chi Tiết
<b>Model</b>	MAX3485EESA+T
<b>Giao tiếp</b>	RS-485/RS-422
<b>Tốc độ truyền dữ liệu</b>	Lên đến 10 Mbps
<b>Điện áp hoạt động</b>	3,3V
<b>Bảo vệ</b>	Bảo vệ quá áp, chống nhiễu ESD
<b>Chế độ tiết kiệm năng lượng</b>	Chế độ tắt với dòng tiêu thụ 1 nA
<b>Nhiệt độ hoạt động</b>	-40°C đến 85°C
<b>Kích thước gói</b>	8-pin SOIC
<b>Ứng dụng</b>	Hệ thống tự động hóa công nghiệp, Giao tiếp dữ liệu khoảng cách xa, Mạng cảm biến

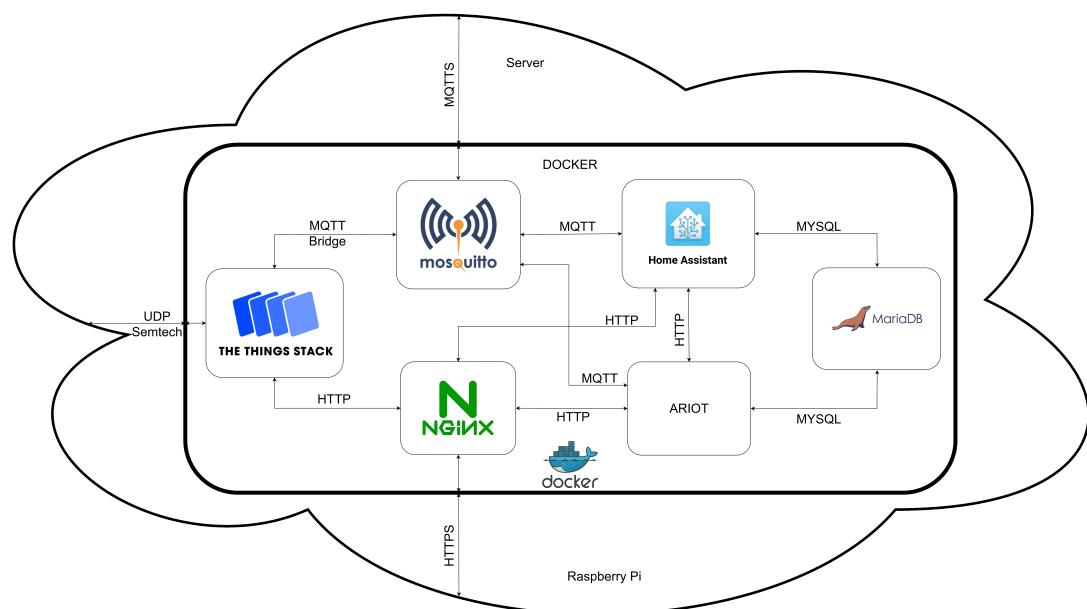
## 3.4 Xây dựng máy chủ và trạm trung gian

### 3.4.1 Xây dựng máy chủ

Nhóm thực hiện đề tài đã quyết định dựng lên một máy chủ điện toán đám mây, máy chủ này vừa đóng vai trò là máy chủ mạng vừa là máy chủ ứng dụng, cho phép các hạ tầng khác trong đồ án này có thể truy cập từ mọi nơi. Phần cứng được sử dụng làm máy chủ là máy tính nhúng Raspberry Pi 4 8GB, chiếc máy này đang chạy hệ điều hành Raspberry Pi OS (64-bit, Bookworm). Tất cả các ứng dụng cần thiết sẽ được triển khai và quản lý bằng Docker và Docker Compose.

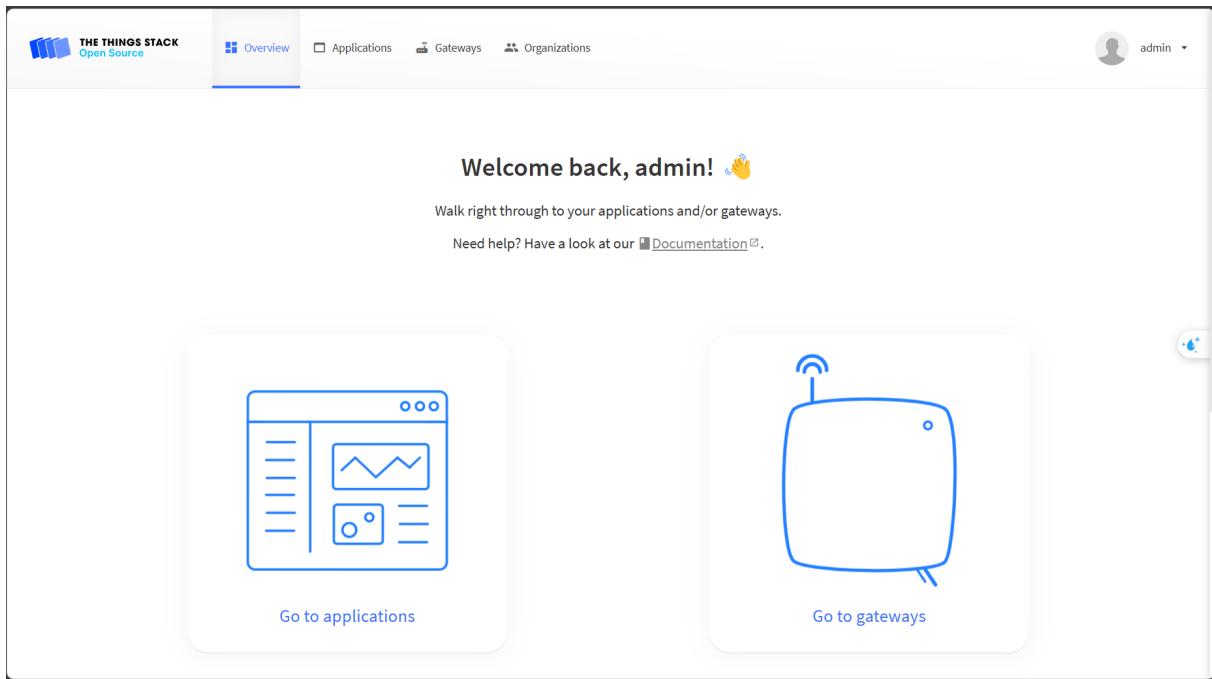
Docker là một nền tảng mã nguồn mở giúp tự động hóa quá trình triển khai ứng dụng trong các vùng chứa (containers). Vùng chứa Docker là một môi trường nhẹ, độc lập, bao gồm tất cả các thành phần cần thiết để chạy ứng dụng, bao gồm mã nguồn, thư viện, và cấu hình. Docker cho phép nhà phát triển xây dựng, kiểm tra, và triển khai ứng dụng một cách nhất quán và hiệu quả trên nhiều môi trường khác nhau.

Docker Compose là một công cụ cho phép định nghĩa và quản lý nhiều vùng chứa Docker như một dịch vụ duy nhất. Docker Compose giúp đơn giản hóa việc quản lý các ứng dụng phức tạp bằng cách cung cấp một cách dễ dàng để cấu hình và khởi động toàn bộ môi trường phát triển chỉ bằng một lệnh duy nhất. Đây là các ứng dụng chính được triển khai cho đồ án này: Postgres, Redis, Stack, Home Assistant, MariaDB, Mosquitto Broker, Swag, ARIOT. Mối quan hệ giữa chúng được trình bày như hình 3.2.



Hình 3.2: Sơ đồ tổng quan các ứng dụng chính và mối quan hệ giữa chúng

Postgres, Redis, Stack là 3 ứng dụng được triển khai riêng cho The Things Stack, đây chính là 1 trong các nền tảng máy chủ mạng của hạ tầng LoRaWAN đã được lựa chọn. The Things Stack là một nền tảng máy chủ mạng LoRaWAN, được phát triển bởi The Things Industries. Nó cung cấp các công cụ và dịch vụ để triển khai, quản lý và vận hành các mạng LoRaWAN, từ các dự án nhỏ đến quy mô doanh nghiệp. The Things Stack hỗ trợ nhiều tính năng như quản lý thiết bị, cấu hình mạng, bảo mật và tích hợp với các dịch vụ đám mây khác. Nó cũng tương thích với nhiều thiết bị và trạm trung gian LoRaWAN, giúp đơn giản hóa việc triển khai và quản lý mạng IoT. Việc triển khai The Things Stack đã được hướng dẫn đầy đủ tại trang web chính chủ [4]. Sau khi đã cài đặt hoàn tất, ta sẽ có một giao diện web như sau:



Hình 3.3: Giao diện The Things Stack khi đã cài đặt xong và đăng nhập

Home Assistant (HASS) là một nền tảng mã nguồn mở mạnh mẽ dành cho tự động hóa, cho phép kiểm soát và tích hợp nhiều thiết bị và dịch vụ khác nhau trong hệ thống của mình, đây chính là một trong những máy chủ ứng dụng cho hệ thống này. Được thiết kế để chạy trên các thiết bị như Raspberry Pi, HASS cung cấp giao diện người dùng trực quan và khả năng tùy biến cao, giúp tạo ra các kịch bản tự động hóa phức tạp. Với cộng đồng hỗ trợ đông đảo và liên tục cập nhật, HASS là lựa chọn lý tưởng cho những ai muốn tối ưu hóa và tự động hóa hệ thống.

Nhóm thực hiện đề tài muốn sử dụng HASS để biểu diễn số liệu cảm biến trên bảng điều khiển trên web (web dashboard) và lưu trữ dữ liệu đó vào cơ sở dữ liệu. Ngoài ra, HASS còn cung cấp một số API cho phép ứng dụng khác (điển hình là ứng dụng MR) có thể truy xuất dữ liệu cảm biến.

Về cơ sở dữ liệu dùng để lưu trữ số liệu của toàn bộ cảm biến, nhóm quyết định sử dụng MariaDB. Đây là một hệ quản trị cơ sở dữ liệu mã nguồn mở, xuất phát từ MySQL, tương thích với cao với HASS. MariaDB hỗ trợ nhiều tính năng tiên tiến như kiểm tra toàn vẹn dữ liệu, bảo mật mạnh mẽ, và các công cụ sao lưu phục hồi. Hệ thống này cũng tương thích với nhiều công cụ và thư viện phổ biến, giúp dễ dàng tích hợp vào các ứng dụng hiện có trong đồ án này.

Ứng dụng tiếp theo được triển khai là Mosquitto Broker. Đây là một dịch vụ mã nguồn mở, tuân thủ giao thức MQTT (Message Queuing Telemetry Transport), giúp quản lý và truyền tải thông điệp giữa các thiết bị IoT một cách hiệu quả. Mosquitto Broker đóng vai trò là trung gian, nhận và phát các thông điệp từ các cảm biến và thiết bị đầu cuối, đảm bảo dữ liệu được truyền tải một cách nhanh chóng và đáng tin cậy. Các ứng dụng như HASS, The Things Stack, ArIoT, ứng dụng MR đều cần sử dụng MQTT để kết nối với nhau. Bản thân The Things Stack đã có sẵn một dịch vụ MQTT cho phép ứng dụng bên ngoài kết nối, tuy nhiên họ lại giới hạn quyền truy cập vào các chủ đề (topic), làm cho các ứng dụng khác không thể kết nối với nhau bằng dịch vụ MQTT này. Vì thế Mosquitto Broker được chọn để vừa làm dịch vụ cung cấp kết nối MQTT chính cho các ứng dụng khác và vừa làm cầu nối (MQTT Bridge) cho The Things Stack.

Để có thể công khai máy chủ lên hệ thống Internet một cách an toàn và thuận tiện, nhóm thực hiện đề tài đã sử dụng ứng dụng Swag (Secure Web Application Gateway). Swag cung cấp một giải pháp toàn diện để quản lý chứng chỉ SSL/TLS và cấu hình proxy ngược (reverse proxy), giúp bảo vệ máy chủ trước các mối đe dọa từ bên ngoài và đảm bảo an toàn cho dữ liệu truyền tải:

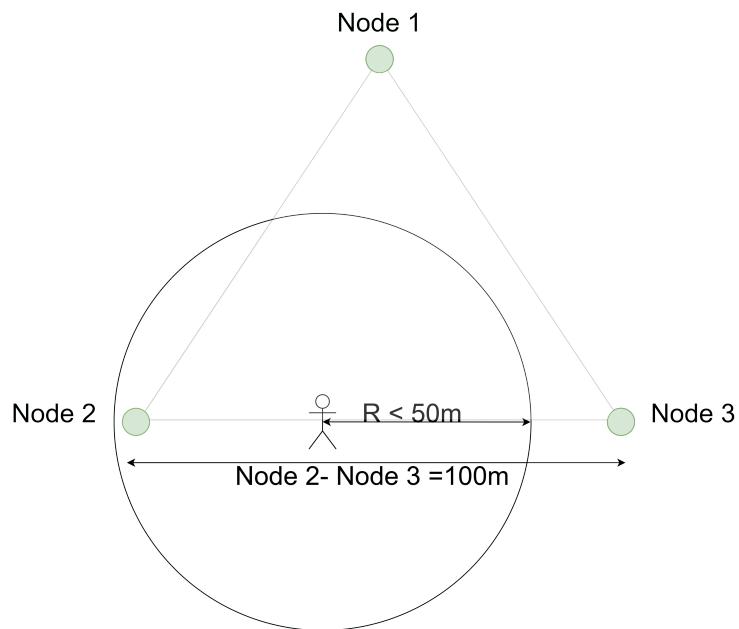
- Swag cho phép cấu hình để quản lý chứng chỉ SSL từ Let's Encrypt. Chứng chỉ này có thể áp dụng cho nhiều ứng dụng khác nhau, điển hình là HTTPS và MQTTS.
- Swag có khả năng thiết lập reverse proxy bằng Nginx để điều hướng các truy cập bên ngoài internet đến các dịch vụ nội bộ. Chỉ với một tên miền chính, nhóm có thể tạo ra các tên miền con khác nhau cho các dịch vụ khác nhau, giúp cho các ứng dụng khác dễ dàng kết nối chính xác đến dịch vụ cần thiết.

Ứng dụng cuối cùng được triển khai là ArIoT, đây là một ứng dụng do nhóm thực hiện đề tài tự lập trình bằng ngôn ngữ Python với mục đích xử lý một số tác vụ cho ứng dụng MR. Tác vụ đầu tiên là chạy backend cung cấp một bộ RestAPI để ứng dụng MR lấy bộ dữ liệu cảm biến theo thời gian và thực hiện quản lý tọa độ các trạm thu thập dữ liệu.

Ban đầu ứng dụng MR lấy dữ liệu từ REST API do HASS cung cấp, tuy nhiên có một vấn đề gấp phải là toàn bộ dữ liệu thời gian được lưu trên cơ sở dữ liệu đều là múi

giờ UTC, nhưng múi giờ của nước ta là UTC+7, điều này làm cho ứng dụng MR có khả năng đổi mới với việc xử lý lượng lớn dữ liệu thời gian. Nhóm thực hiện đề tài cho rằng việc này không tối ưu, cho nên đã quyết định cho máy chủ tự xử lý vấn đề này trước khi gửi dữ liệu về thiết bị MR. Việc lưu trữ tọa độ các trạm thu thập dữ liệu cũng do backend này phụ trách, ứng dụng MR sẽ thực hiện đánh dấu tọa độ của trạm hoặc xóa đánh dấu thông qua các API mà ứng dụng này cung cấp.

Tác vụ thứ 2 của ứng dụng ArIoT là xác định trạm thu thập dữ liệu nào đang nằm trong vùng bán kính của người dùng ứng dụng MR, giá trị bán kính này có thể thay đổi phụ thuộc vào cách bố trí hệ thống và không thể có hơn một trạm trong một vùng bán kính. Tác vụ này sẽ giúp cho ứng dụng MR có thể xác định được người dùng đang ở khu vực nào, từ đó dữ liệu của các cảm biến sẽ tự thay đổi để phù hợp với vị trí người dùng.



Hình 3.4: Ảnh minh họa cách xác định vị trạm ở gần người dùng

Dựa vào ảnh minh họa 3.4, giả định có 3 trạm được đặt thành một hình tam giác cân, mỗi trạm cách nhau 100 m, khi bán kính quét được đặt dưới 50 thì chắc chắn sẽ không có 2 trạm nào có thể cùng xuất hiện trong phạm vi 50 m đó. Tương tự với các trường hợp khác, chỉ cần tìm khoảng cách 2 trạm gần nhất, điều chỉnh bán kính quét lại dưới một nửa khoảng cách đó, hệ thống có thể hoạt động bình thường.

Để có thể xác định được trạm thu thập có nằm trong bán kính hay không, nhóm thực hiện đề tài sử dụng công thức Haversine để tính khoảng cách giữa người dùng và các trạm dữ liệu. Haversine là một phương pháp trong toán học và địa lý để tính khoảng cách giữa hai điểm trên bề mặt của một hình cầu, dựa trên vĩ độ và kinh độ của hai điểm

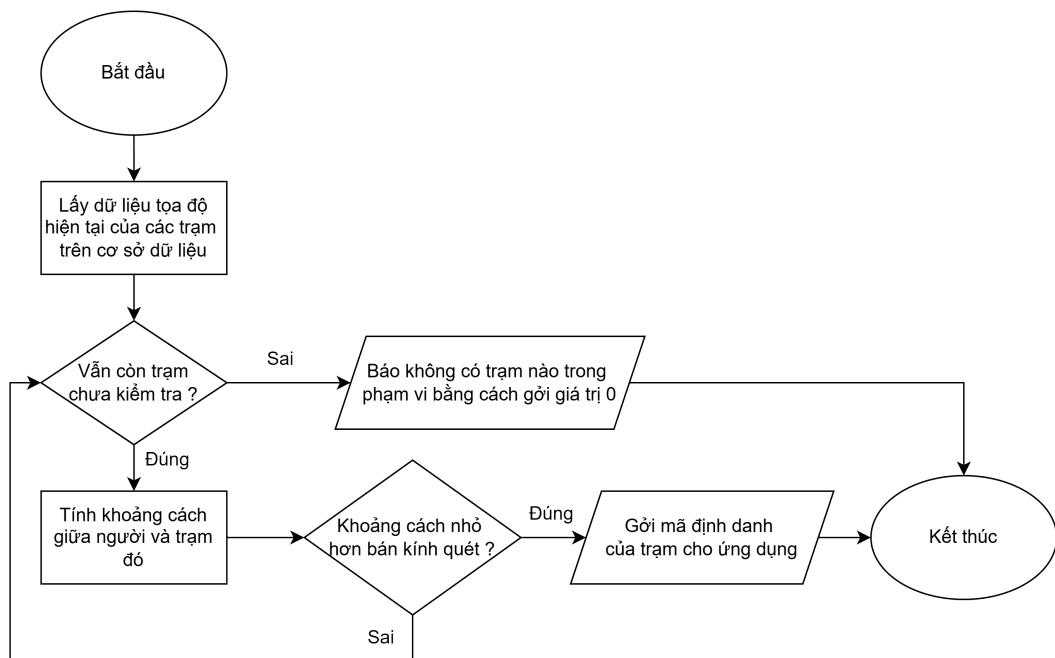
đó [5]. Công thức Haversine tính khoảng cách  $d$  giữa hai điểm trên bề mặt hình cầu:

$$d = 2R \cdot \arcsin \left( \sqrt{\sin^2 \left( \frac{\varphi_2 - \varphi_1}{2} \right) + \cos(\varphi_1) \cdot \cos(\varphi_2) \cdot \sin^2 \left( \frac{\lambda_2 - \lambda_1}{2} \right)} \right), \quad (3.1)$$

trong đó:

- $\varphi_1, \varphi_2$ : vĩ độ của điểm 1 và điểm 2 (đơn vị: radian).
- $\lambda_1, \lambda_2$ : kinh độ của điểm 1 và điểm 2 (đơn vị: radian).
- $R$ : bán kính của Trái Đất (khoảng 6,371 km).

Dựa vào công thức Haversine này, nhóm thực hiện đề tài đã xác định trạm thu thập theo lưu đồ thuật toán sau:



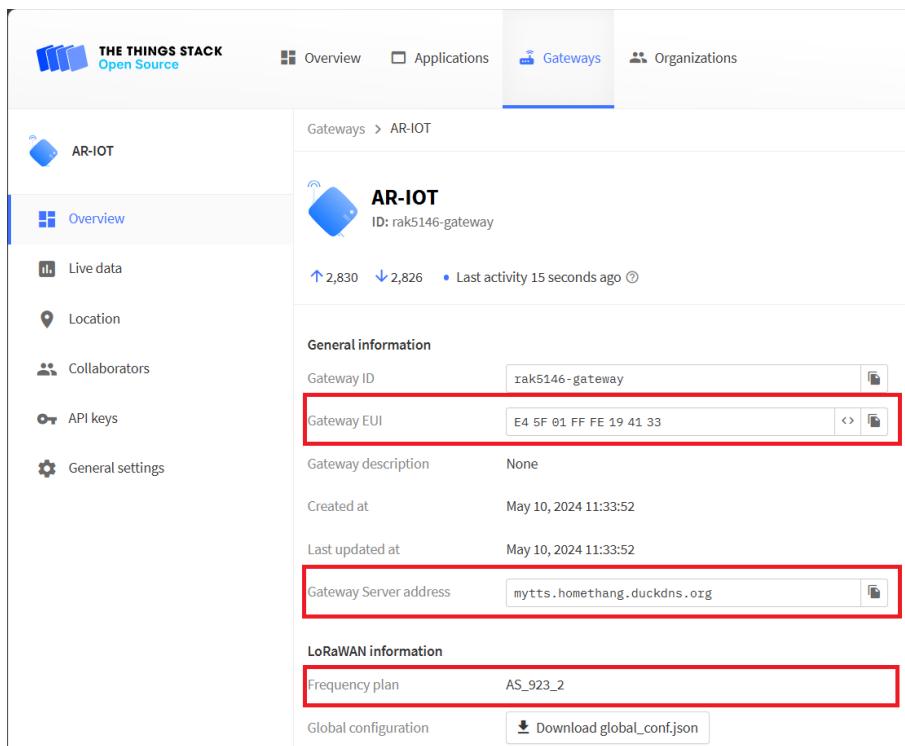
Hình 3.5: Lưu đồ thuật toán xác định trạm trong phạm vi bán kính

### 3.4.2 Xây dựng trạm trung gian

Sau khi máy chủ đã được chuẩn bị hoàn tất, nhóm sẽ dựng một trạm trung gian (gateway). Trạm trung gian trong hạ tầng LoRaWan có nhiệm vụ luân chuyển dữ liệu giữa trạm thu thập dữ liệu và máy chủ. Theo quy định của LoRa Alliance, một trạm trung gian đủ tiêu chuẩn phải hỗ trợ 8 kênh trỏ lên. Việc này giúp tối ưu hóa khả năng xử lý dữ liệu, tăng cường hiệu suất và độ tin cậy của hệ thống, đồng thời hỗ trợ nhiều

thiết bị và tần số khác nhau trong cùng một thời điểm. Điều này cho phép các mạng LoRaWAN hoạt động hiệu quả và đáp ứng tốt các yêu cầu về truyền tải dữ liệu trong các ứng dụng IoT. Chính vì thế, module RAK5146 đã được sử dụng để xây dựng trạm trung gian.

Ngoài việc bộ kit RAK5146 có hỗ trợ đầy đủ chức năng phần cứng của một trạm trung gian thì phần mềm cài đặt của nó cũng được hãng RAKwireless cung cấp đầy đủ, cài đặt đơn giản. Tất cả hướng dẫn cài đặt đều được trình bày đầy đủ tại kho lưu trữ trên Github của họ [6]. Trong lúc chờ đợi cài đặt xong các ứng dụng cần thiết cho trạm trung gian, nhóm thực hiện đề tài sẽ khởi tạo một trạm trung gian trên The Things Stack.



Hình 3.6: Ảnh minh họa khi đăng ký xong một trạm trung gian

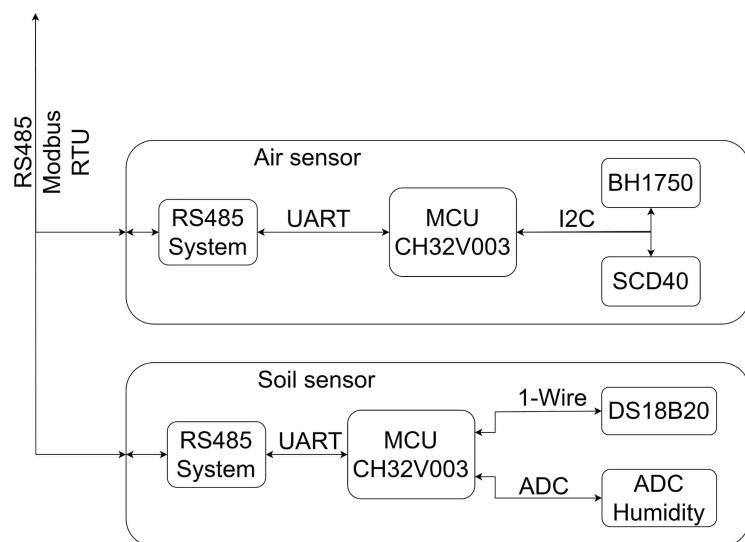
Các vùng đánh dấu trên hình 3.6 là các vùng cần chú ý sau khi đăng ký một trạm trung gian. Vì ở nước ta cho phép sử dụng các thiết bị LPWAN có tần số thuộc 920-923 MHz, cho nên Frequency Plan (kế hoạch tần số) phải được cấu hình là AS923-2 dựa theo chuẩn của LoRa Alliance [7]. Hai vùng Gateway EUI và Gateway Server address sẽ được dùng cho tệp cấu hình **global\_conf.json** ở trong đường dẫn **/opt/ttn-gateway/packet\_forwarder/lora\_pkt\_fwd** trên trạm trung gian khi đã cài đặt xong. Các cấu hình này mang ý nghĩa trạm này sẽ kết nối máy chủ **mytts.homethang.duckdns.org** với mã định danh là **E45F01FFFE194133**, trạm này sẽ hoạt động ở dải tần số AS923-2. Khi mọi thứ được cấu hình hoàn tất, kết nối đến máy chủ thành công, trạm trung gian này chỉ cần cắm nguồn là có thể sử dụng.

## 3.5 Quá trình phát triển cảm biến

### 3.5.1 Thiết kế phần cứng cảm biến

Dựa vào yêu cầu ban đầu do nhóm thực hiện đề tài đặt ra, nhóm đã thiết kế 2 cảm biến dùng để thu thập dữ liệu từ môi trường tự nhiên. Một cảm biến đo các chỉ số trên không khí như nồng độ khí CO<sub>2</sub>, cường độ ánh sáng, nhiệt độ không khí, độ ẩm. Cảm biến còn lại đo nhiệt độ và độ ẩm của đất.

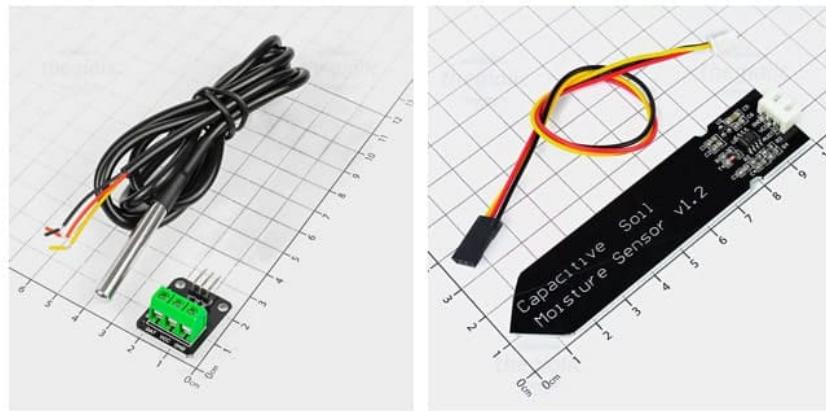
Vì tính đặc thù của hệ thống, nhóm thực hiện đề tài quyết định dùng giao thức RS485 Modbus RTU cho cảm biến để có thể đặt cảm biến dưới mặt đất mà vẫn có thể kết nối với trạm thu thập dữ liệu được đặt ở trên cao. Để có thể đọc dữ liệu của cảm biến qua cổng RS485, nhóm đã chọn một vi điều khiển làm thiết bị trung gian xử lý dữ liệu, đó là vi điều khiển WCH CH32V003, với kích thước rất gọn, giá thành rất rẻ và có đủ các chuẩn giao tiếp. Nhóm thực hiện đề tài đánh giá rằng vi điều khiển này rất thích hợp cho các ứng dụng thiết kế cảm biến RS485 Modbus RTU này. Hình sau đây sẽ thể hiện sơ đồ tổng quát của các cảm biến nhóm dự định thiết kế:



Hình 3.7: Sơ đồ tổng quát của cả hai cảm biến

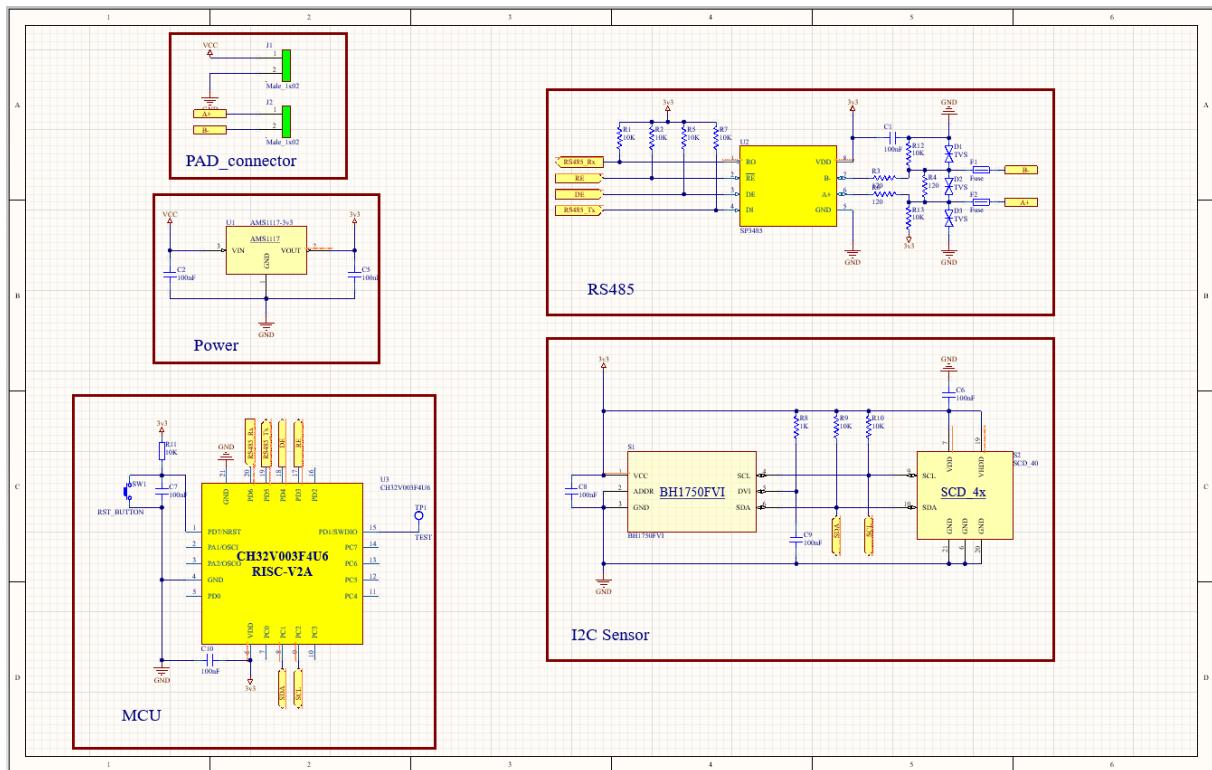
Với cảm biến đo các chỉ số không khí, nhóm dùng cảm biến SCD40 để đo nhiệt độ, độ ẩm và nồng độ CO<sub>2</sub> của không khí. Ngoài ra nhóm còn sử dụng thêm cảm biến BH1750 để đo được cường độ ánh sáng của môi trường. Cả 2 cảm biến này đều dùng giao thức I2C để giao tiếp cho nên nhiệm vụ của CH32V003 là đọc dữ liệu qua đường I2C và chờ sự kiện từ RS485 để gửi dữ liệu đi. Tương tự với cảm biến đất, nhiệt độ của đất được đo bởi cảm biến DS18B20 như hình 3.8, vì cảm biến này có thiết kế chống nước, độ bền cao, thích hợp cảm sâu vào đất. Độ ẩm của đất thì được đo bằng một cảm biến điện dung được thiết kế có khả năng chống ăn mòn tốt, có đầu ra là ADC cho vi

điều khiển có thể tính toán độ ẩm.



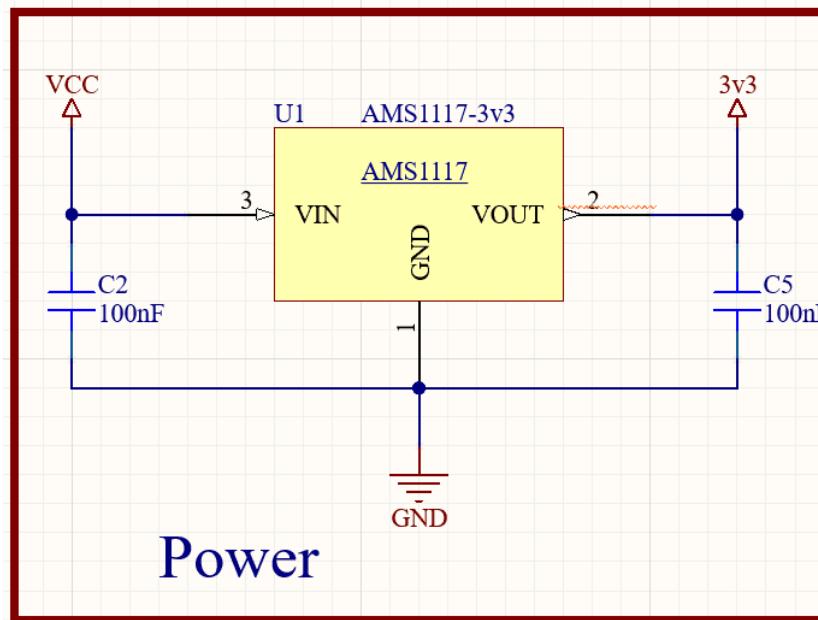
Hình 3.8: Cảm biến nhiệt độ (bên trái) và cảm biến độ ẩm (biên phải) được sử dụng cho cảm biến đất

Sau khi đã chọn xong các thành phần cho cảm biến, nhóm thực hiện để tài đã bắt tay vào thiết kế sơ đồ nguyên lý và mạch in. Sử dụng phần mềm Altium để thiết kế các sơ đồ nguyên lý và mạch in cho hệ thống này. Đây là một phần mềm trả phí chuyên dụng để thiết kế mạch, nó cung cấp đầy đủ công cụ hỗ trợ cho một kỹ sư có thể thiết kế mạch điện tử từ đơn giản đến phức tạp. Dưới đây là ảnh sơ đồ nguyên lý của cảm biến không khí:



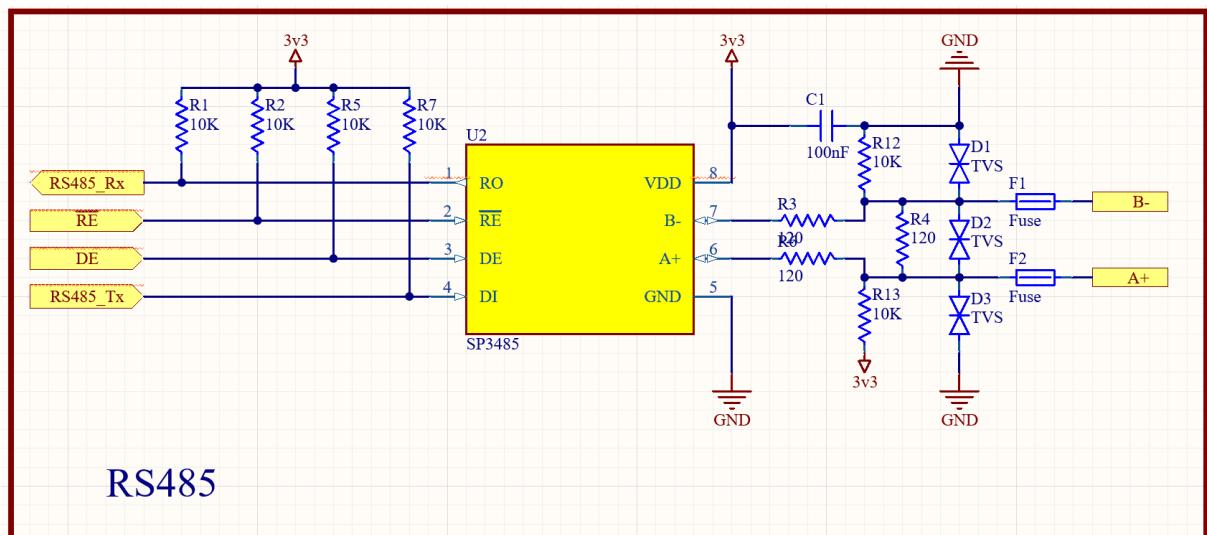
Hình 3.9: Sơ đồ nguyên lý của cảm biến không khí

Khối nguồn của cảm biến được nhóm thực hiện đề tài thiết kế sử dụng IC ổn áp AMS1117-3V3 và hai tụ lọc nguồn cơ bản ở hai đầu để cấp nguồn cho toàn bộ mạch như hình 3.10. IC này có thể chịu được điện áp lên đến 15V, phù hợp cho nhiều hệ thống khác nhau.



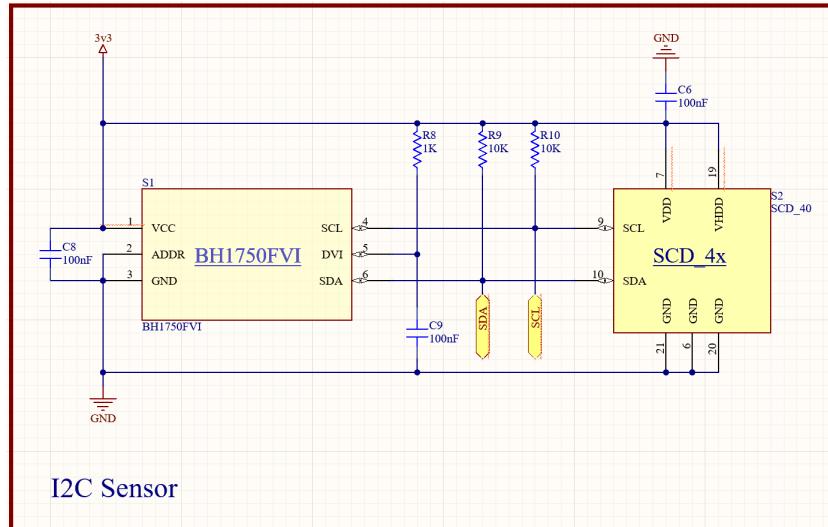
Hình 3.10: Sơ đồ nguyên lý khối nguồn của cả hai cảm biến

Nhóm thực hiện đề tài thiết kế khôi chuyển đổi giao thức UART sang chuẩn RS485 sử dụng IC SP3485, một IC chuyên chuyển đổi tín hiệu TTL/CMOS sang tín hiệu vi sai hoạt động với điện áp 3,3V. Trên hai đường A+ và B- của IC nhóm thực hiện đề tài còn lắp thêm hai cầu chì tự phục hồi để bảo vệ quá dòng và các TVS Diode để bảo vệ mạch khỏi các xung nhiễu đến từ môi trường như sơ đồ nguyên lý sau:



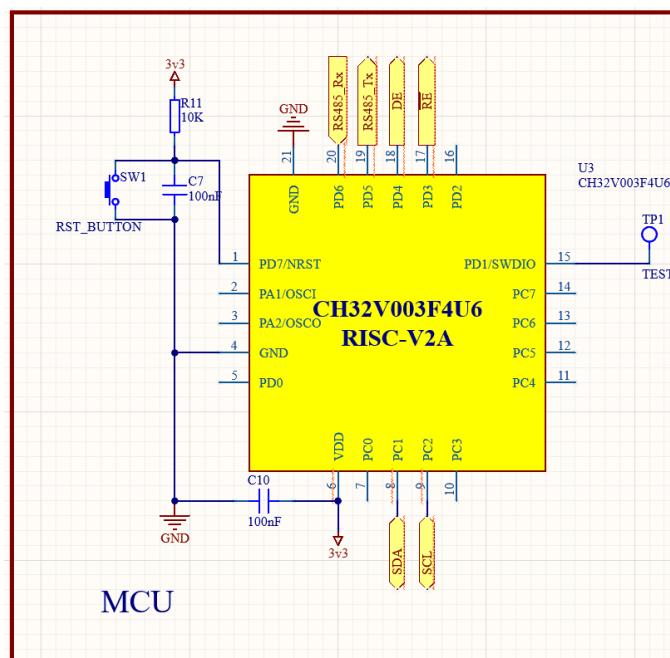
Hình 3.11: Sơ đồ nguyên lý khôi chuyển đổi UART sang RS485 của cả hai cảm biến

Khối cảm biến của cảm biến không khí được nhóm thiết kế chung đường dữ liệu I2C, có đầy đủ các tụ lọc nguồn, điện trở điều khiển địa chỉ và điện trở kéo lên cho đường I2C theo sơ đồ nguyên lý dưới đây:



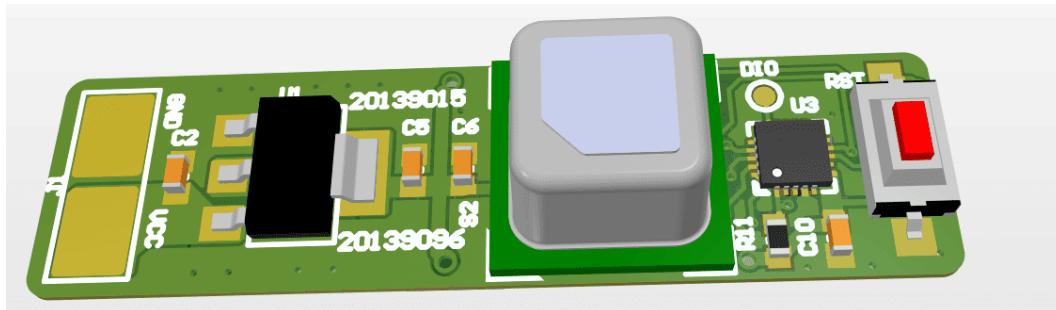
Hình 3.12: Sơ đồ nguyên lý khôi cảm biến I2C

Cuối cùng là khôi vi điều khiển trung tâm, nơi điều khiển việc đọc dữ liệu cảm biến và giao tiếp với thiết bị khác qua giao thức UART đến từ IC SP3485. Vi điều khiển CH32V003 được chọn với kiểu đóng gói F4U6 có kích thước siêu gọn, chỉ cần thiết kế thêm bộ nút bấm cho khởi động lại, tụ lọc nguồn, một điểm chì hàn cho phép nạp mã nguồn và gỡ lỗi qua giao thức SDI.

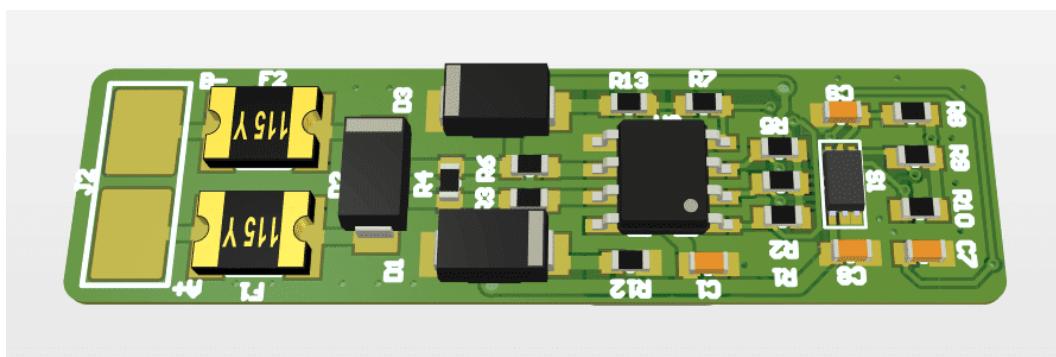


Hình 3.13: Sơ đồ nguyên lý khôi vi điều khiển CH32V003

Sau khi đã hoàn thành sơ đồ nguyên lý, nhóm sẽ bắt tay vào thiết kế mạch điện. Các linh kiện được bố trí trên cả 2 mặt của mạch để tối ưu kích thước hết mức có thể. Linh kiện được bố trí hợp lý để dây điện có thể đi thật tối ưu, hạn chế các nhiễu và suy hao trên dây điện. Kết quả phần cứng thiết kế dưới dạng 3D như các hình 3.14 và 3.15.

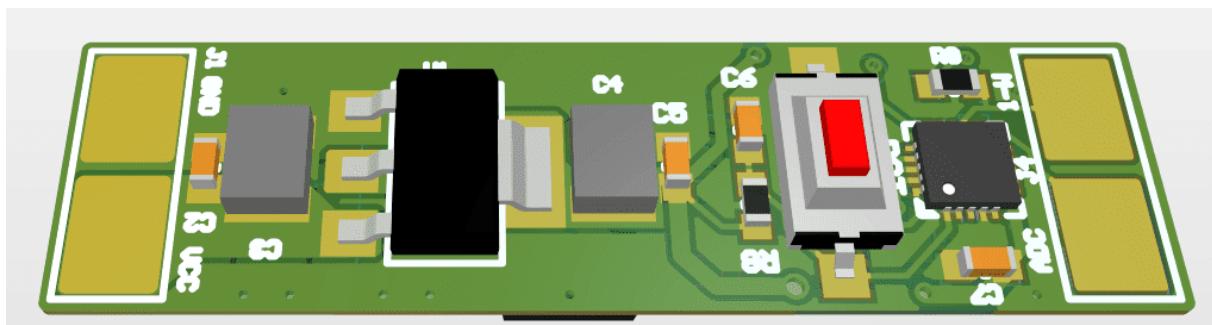


Hình 3.14: Ảnh 3D mặt trên của cảm biến không khí sau khi thiết kế

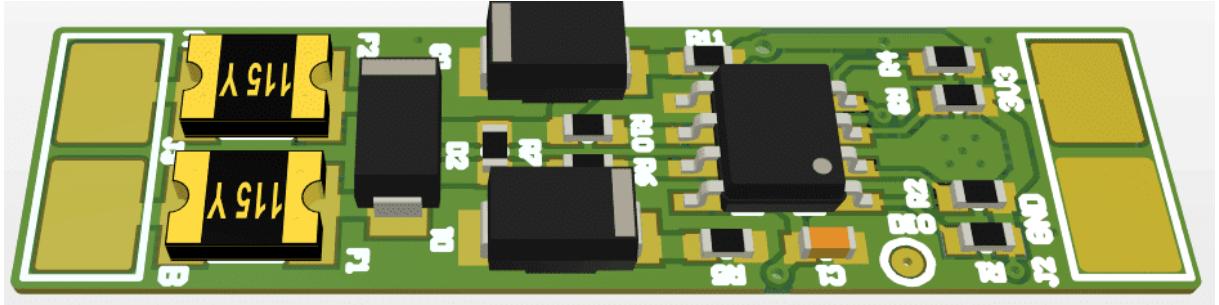


Hình 3.15: Ảnh 3D mặt dưới của cảm biến không khí sau khi thiết kế

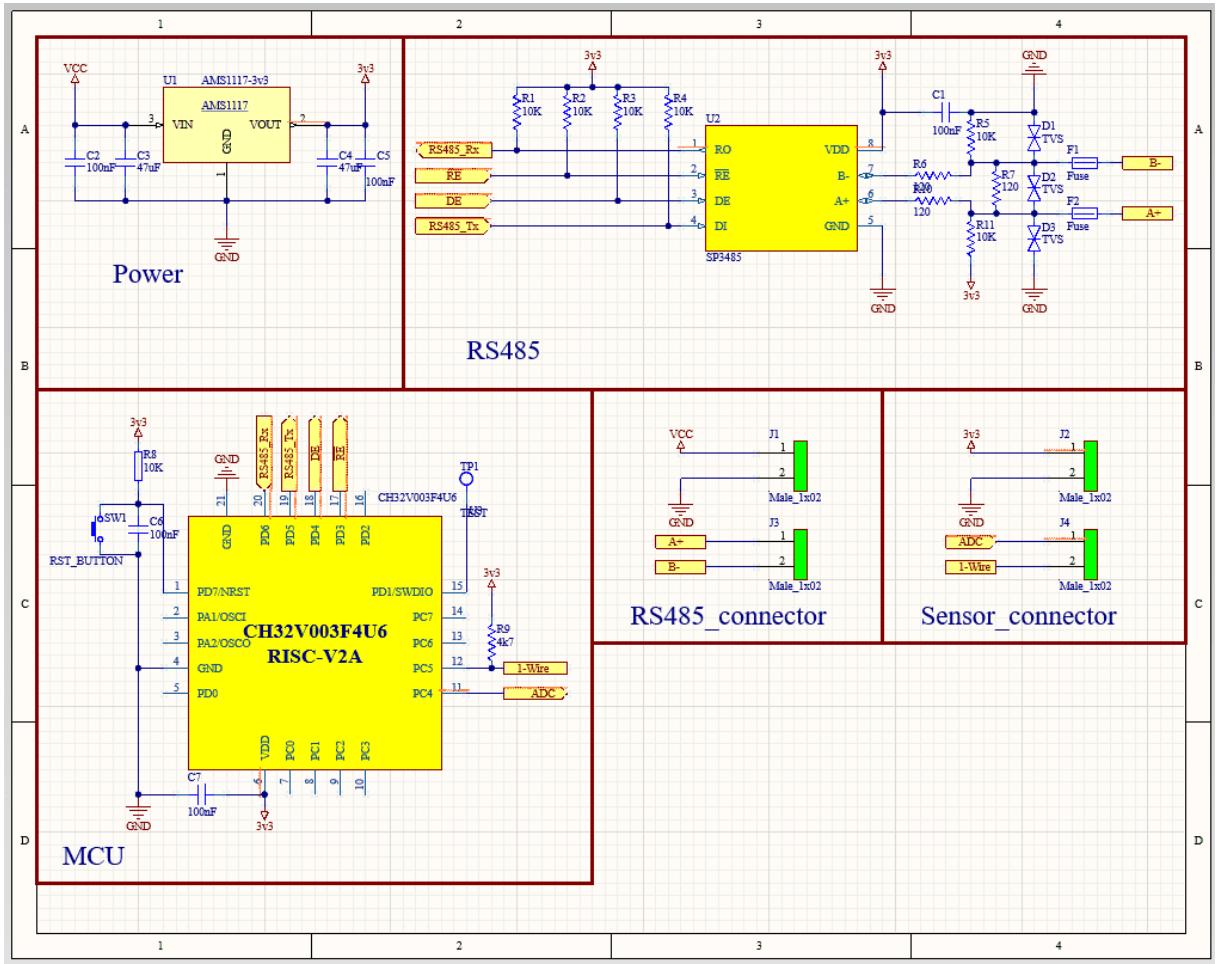
Tương tự với cảm biến không khí, cảm biến đất cũng có các khối vi điều khiển, khối nguồn, khối RS485. Nhưng các cảm biến đo chỉ số đất được gắn rời với PCB vì thiết kế chống ẩm của chúng. Cho nên mạch cảm biến đất sẽ đơn giản hơn. Các hình 3.16, 3.17 và 3.18 lần lượt là các ảnh 3D của thiết kế phần cứng và sơ đồ nguyên lý của cảm biến đất.



Hình 3.16: Ảnh 3D mặt trên của cảm biến đất sau khi thiết kế



Hình 3.17: Ảnh 3D mặt dưới của cảm biến đất sau khi thiết kế

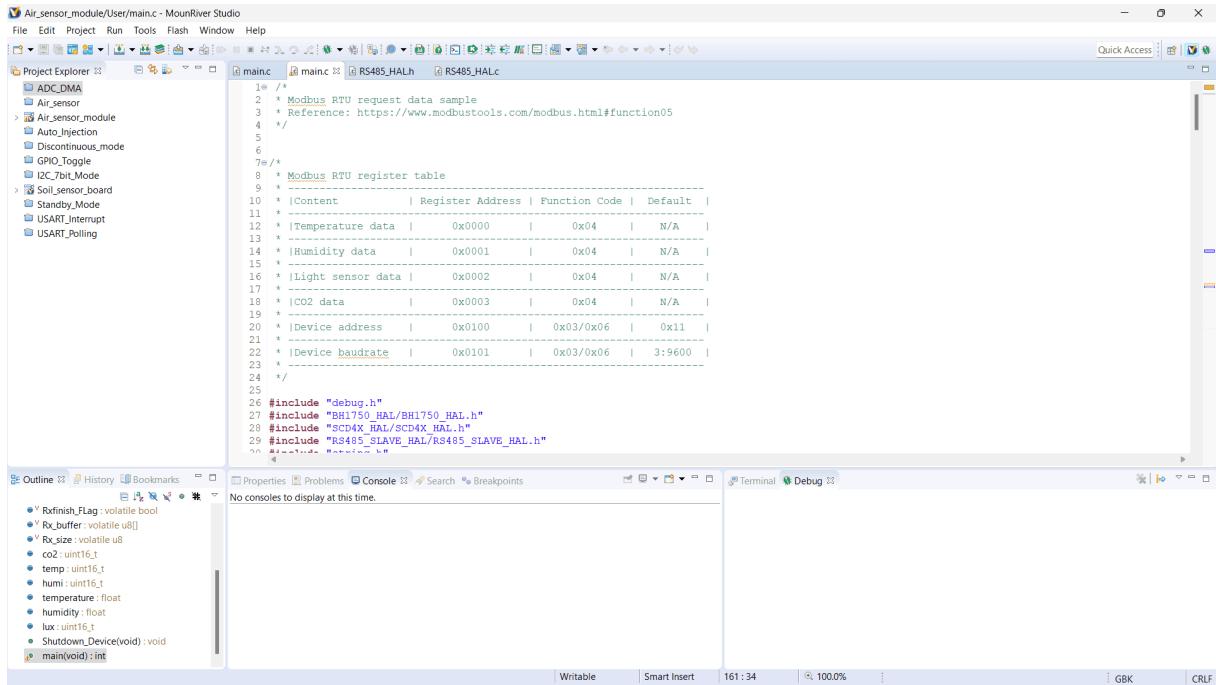


Hình 3.18: Sơ đồ nguyên lý của cảm biến đất

### 3.5.2 Xây dựng mã nguồn cho cảm biến

Khi đã thiết kế xong phần cứng, nhóm thực hiện đề tài sẽ đặt thi công PCB, thời gian thi công khoảng 10 ngày. Trong thời gian này, nhóm sẽ bắt tay vào viết mã nguồn cho cảm biến. Mã nguồn này viết cho vi điều khiển CH32V003, cho nên nhóm đã dùng IDE MounRiver Studio do chính hãng WCH cung cấp. MounRiver Studio là một Môi trường Phát triển Tích hợp (IDE) được thiết kế đặc biệt cho việc lập trình các vi điều

khiển RISC-V. Nền tảng này dựa trên Eclipse, tương tự như các IDE phổ biến khác như STM32CubeIDE. IDE này hỗ trợ nhiều tính năng cần thiết cho việc phát triển, biên dịch và gỡ lỗi các ứng dụng dựa trên RISC-V.



Hình 3.19: Giao diện Mounriver Studio khi đang mở một dự án

Trong quá trình nghiên cứu lập trình cho vi điều khiển này, nhóm đã gặp một số khó khăn đến từ SDK (Bộ công cụ phát triển phần mềm) do hãng cung cấp. Bộ thư viện trong SDK, hãng chỉ cung cấp các phương thức cơ bản tương tác với những thanh ghi. Trong bộ mã nguồn mẫu do hãng cung cấp, các chứng năng I/O và các giao thức chỉ được lập trình chạy thử với led hoặc giao tiếp với chính vi điều khiển CH32V003 khác, không hoạt động với các cảm biến. Đây lại là một vi điều khiển khá mới, cho nên số lượng mã nguồn được công khai trên mạng cũng không nhiều, cộng đồng hỗ trợ cũng ít cho nên nhóm đã mất khá nhiều thời gian cho việc lập trình, chạy thử và gỡ lỗi cho các chuẩn giao tiếp. Sau khi đã đảm bảo các chuẩn giao tiếp đã hoạt động bình thường nhóm mới bắt đầu xây dựng chương trình chính. Quá trình nhóm thực hiện để tài phát triển mã nguồn cho cảm biến đã trải qua các bước sau:

- Bước 1: Đọc hướng dẫn sử dụng (reference manual), nghiên cứu cách vi điều khiển xử lý các giao thức.
- Bước 2: Nghiên cứu SDK, mã nguồn mẫu do hãng cung cấp và một số mã nguồn được công khai trên mạng có liên quan.
- Bước 3: Tạo dự án mới trong IDE, nạp một số mã nguồn mẫu để chạy thử module.

- Bước 4: Viết bộ thư viện HAL (Hardware Abstraction Layer) cho các giao thức UART, I2C, 1-wire (vì SDK do hãng cung cấp chỉ hỗ trợ tương tác với các thanh ghi, không có thư viện sẵn cho các chuẩn giao thức), chạy thử và gỡ lỗi các thư viện HAL đã viết
- Bước 5: Viết thư viện giao tiếp, xử lý dữ liệu của cảm biến và IC RS485, chạy thử từng thiết bị và gỡ lỗi.
- Bước 6: Thiết kế một bảng mã lệnh RS485 Modbus RTU cho từng cảm biến (bảng 3.4 và 3.5).
- Bước 7: Lập trình chương trình chính theo bảng mã lệnh đã thiết kế.
- Bước 8: Chạy thử toàn bộ các lệnh, gỡ lỗi, tối ưu chương trình.

Bảng 3.12: Bảng mã lệnh Modbus RTU cho cảm biến không khí

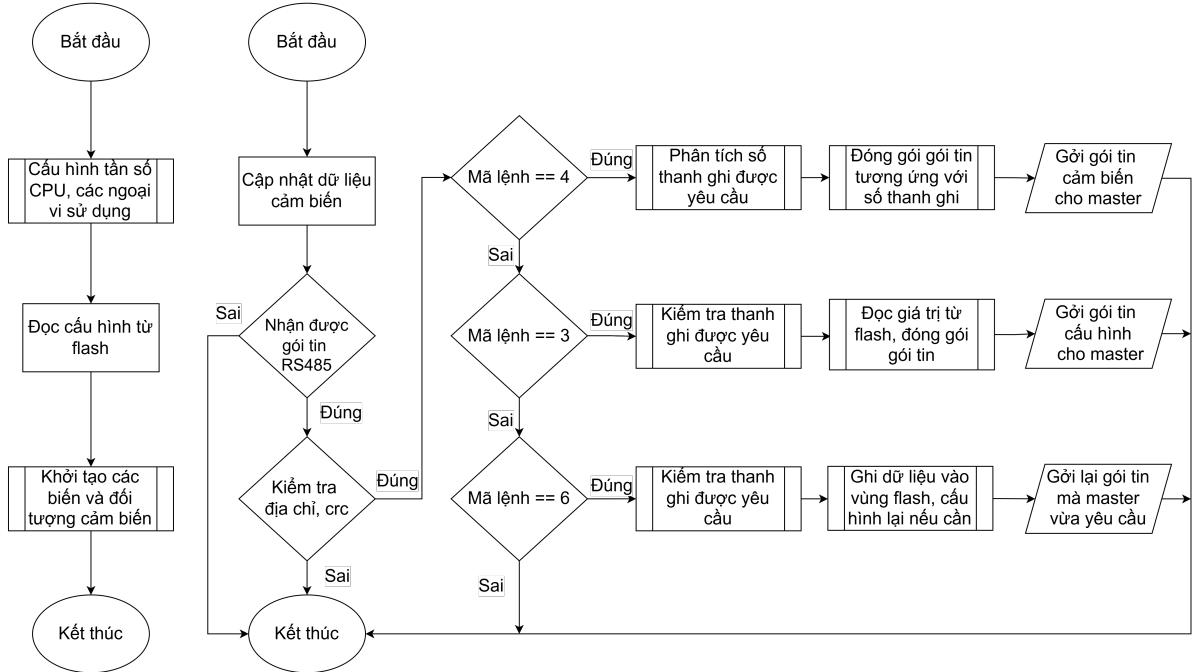
Nội dung	Địa chỉ thanh ghi	Mã lệnh	Giá trị ban đầu
Nhiệt độ	0x0000	0x04	N/A
Độ ẩm	0x0001	0x04	N/A
Cường độ ánh sáng	0x0002	0x04	N/A
Nồng độ CO <sub>2</sub>	0x0003	0x04	N/A
Địa chỉ thiết bị	0x0100	0x03/0x06	0x12
Baudrate của thiết bị	0x0101	0x03/0x06	3:9600

Bảng 3.13: Bảng mã lệnh Modbus RTU cho cảm biến đất

Nội dung	Địa chỉ thanh ghi	Mã lệnh	Giá trị ban đầu
Nhiệt độ	0x0000	0x04	N/A
Độ ẩm	0x0001	0x04	N/A
Địa chỉ thiết bị	0x0100	0x03/0x06	0x11
Baudrate của thiết bị	0x0101	0x03/0x06	3:9600

Chương trình chính của cả hai cảm biến sẽ có cách thức hoạt động tương tự nhau. Khi thiết bị được khởi động, vi điều khiển sẽ thực hiện các cấu hình cơ bản về tần số hoạt động và các ngoại vi, sau đó sẽ đọc các giá trị cấu hình địa chỉ và baudrate được lưu trước đó trong flash tại vùng nhớ tùy chọn cho người dùng (User Option Bytes). Với lần đầu nạp chương trình, vùng nhớ đó chưa được ghi giá trị vào thì sẽ ghi giá trị mặc định được lưu trong code. Tiếp theo, tạo ra các biến cần thiết và cấu hình các cảm biến để chúng hoạt động và bắt đầu vòng lặp vô hạn. Bắt đầu vòng lặp, thiết bị sẽ cập nhật

dữ liệu cảm biến, sau đó kiểm tra xem cờ nhận được gói tin từ RS485 có bật hay không, nếu không thì sẽ bắt đầu vòng lặp mới, nếu có thì sẽ kiểm tra tiếp địa chỉ và crc của gói tin, nếu đúng cả hai thì sẽ thực hiện tiếp yêu cầu, nếu sai thì lại bỏ qua gói tin và tiếp tục vòng lặp mới. Sau đây là lưu đồ thuật toán tổng quát cho cả hai cảm biến:



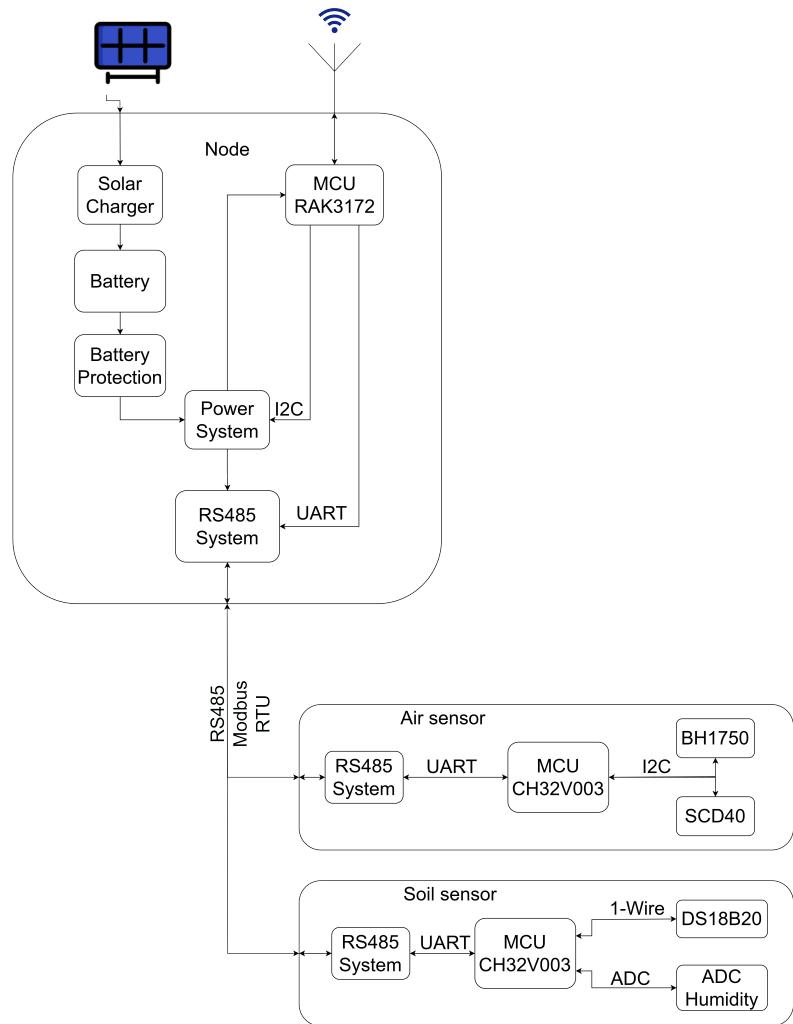
Hình 3.20: Lưu đồ thuật toán lúc khởi động và từng chu kì vòng lặp của cả 2 cảm biến

## 3.6 Quá trình phát triển trạm thu thập dữ liệu

### 3.6.1 Thiết kế phần cứng trạm thu thập dữ liệu

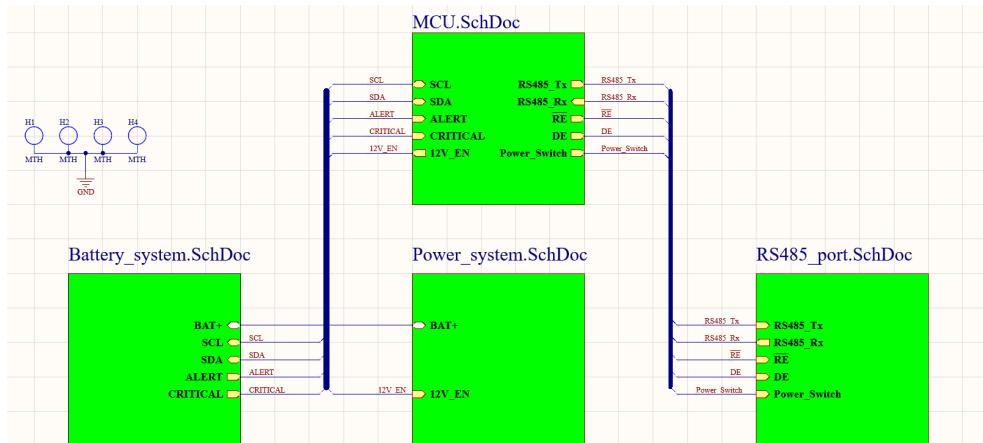
Sau khi hoàn thiện các cảm biến, nhóm đã bắt đầu thiết kế trạm thu thập dữ liệu. Theo như yêu cầu ban đầu đã đặt ra, trạm thu thập dữ liệu này sẽ hoạt động bằng pin, có thể sạc bằng năng lượng mặt trời thông qua một IC quản lý sạc MPPT, giúp đảm bảo hiệu suất sạc và tuổi thọ tấm pin năng lượng mặt trời. Năng lượng tiêu thụ của trạm phải được kiểm soát thật tốt, đảm bảo duy trì hoạt động trong trường hợp không có đủ ánh sáng để sạc trong thời gian dài.

Để kiểm soát tốt năng lượng tiêu thụ của trạm thu thập dữ liệu, toàn bộ các IC trên mạch đều được cân nhắc về năng lượng tiêu thụ, khả năng điều chỉnh chế độ hoạt động, giá thành và cả nguồn hàng cung cấp. Đảm bảo cân bằng chi phí và hiệu suất trong tầm năng lực của nhóm. Sơ đồ tổng quan của hệ thống do nhóm thực hiện đề tài lên ý tưởng được trình bày như hình 3.21.



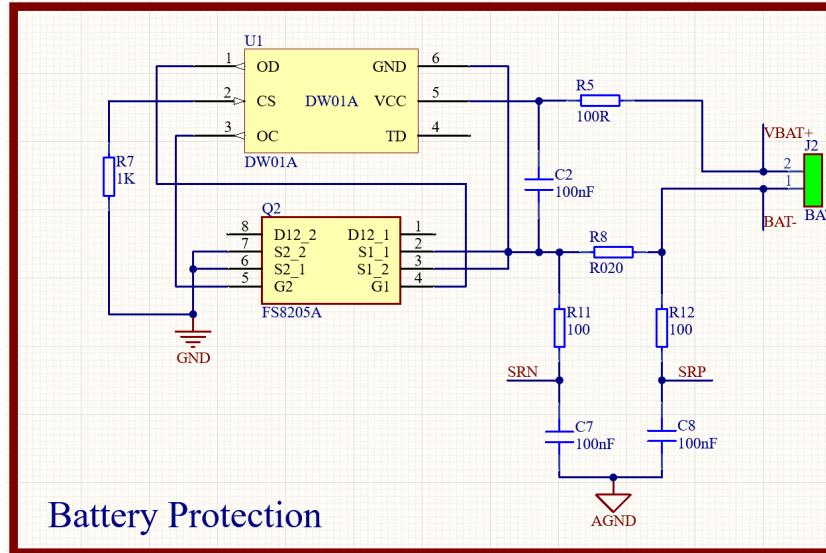
Hình 3.21: Sơ đồ tổng quát của cả trạm thu thập dữ liệu

Trạm thu thập dữ liệu này cũng được nhóm thực hiện đề tài thiết kế bằng phần mềm Altium. Nhóm chia sơ đồ nguyên lý thành 4 khối chính: Khối pin, khối nguồn, khối RS485 và khối điều khiển chính. Sơ đồ nguyên lý tổng quát như sau:



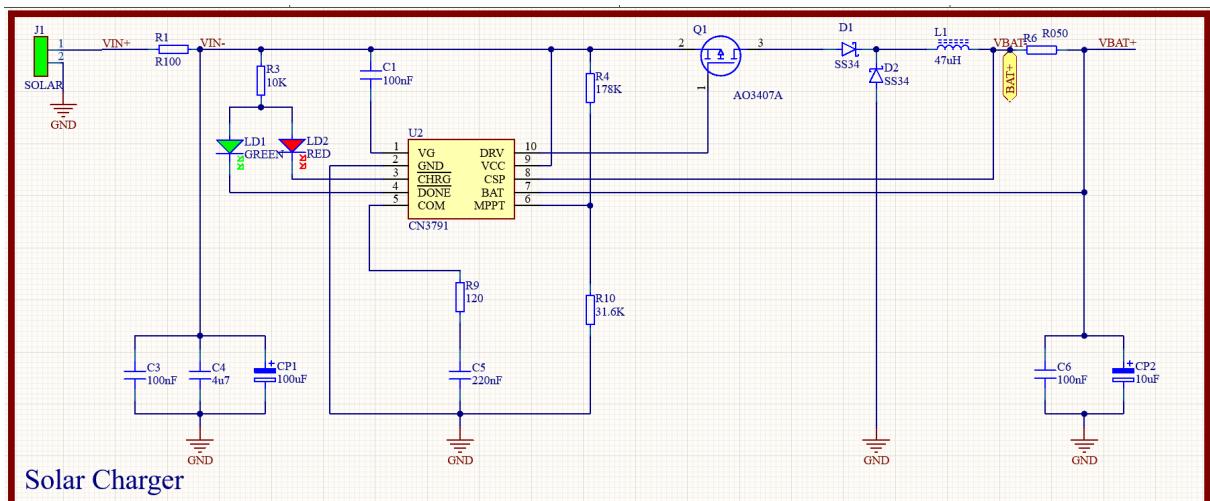
Hình 3.22: Sơ đồ nguyên lý tổng quát của trạm thu thập dữ liệu

Khối pin được nhóm thiết kế để bảo vệ pin, sạc pin và giám sát năng lượng hoạt động của pin và bộ sạc. Bộ bảo vệ pin sẽ có nguyên lý như hình sau:



Hình 3.23: Sơ đồ nguyên lý bộ bảo vệ pin

Bộ bảo vệ pin sử dụng IC DW01A và mosfet 2 chiều FS8250A. Đây là 2 linh kiện thường thấy nhất trong các bộ bảo vệ pin thông dụng. IC DW01A có nhiệm vụ bảo vệ quá trình sạc và xả của pin, không cho phép sạc pin quá ngưỡng áp 4,3V và xả pin khi thấp hơn ngưỡng áp 2,5V, chống xả quá dòng khi xảy ra ngắn mạch, giúp đảm bảo tuổi thọ pin và chống cháy nổ. IC này hoạt động theo cơ chế kiểm tra điện áp và dòng điện trên pin, dựa vào tình trạng hiện tại mà điều khiển mosfet 2 chiều để kiểm soát dòng sạc vào và xả ra của pin.



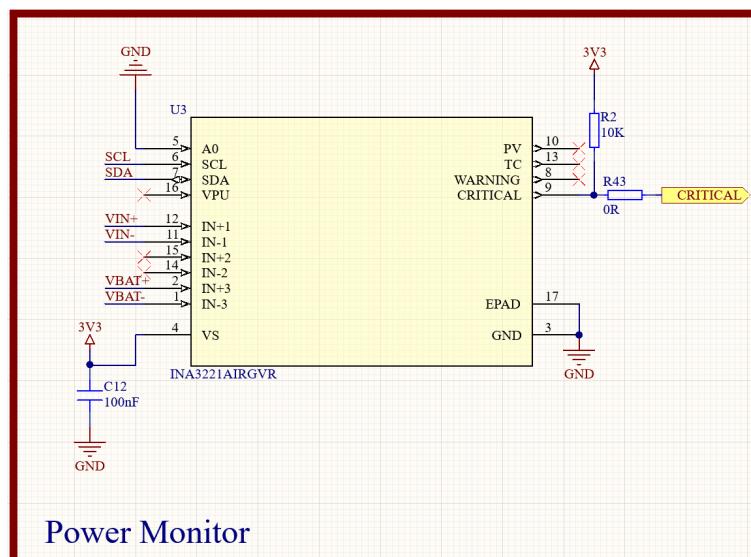
Hình 3.24: Sơ đồ nguyên lý bộ sạc pin bằng năng lượng mặt trời

Hình 3.24 là sơ đồ nguyên lý của bộ sạc pin năng lượng mặt trời sử dụng IC CN3791 để kiểm soát quá trình sạc pin. Đây là một IC sạc năng lượng mặt trời hỗ trợ MPPT

(Maximum Power Point Tracking), là một kỹ thuật được sử dụng trong các hệ thống năng lượng mặt trời để tối ưu hóa lượng điện năng mà tấm pin mặt trời có thể cung cấp. Các tấm pin mặt trời có một điểm hoạt động tối ưu, gọi là điểm công suất tối đa, tại đó sản lượng điện là cao nhất. Tuy nhiên, điểm này thay đổi liên tục do sự biến đổi của ánh sáng mặt trời, nhiệt độ, và các yếu tố môi trường khác. Chức năng của MPPT là liên tục theo dõi và điều chỉnh các thông số của hệ thống để giữ cho tấm pin mặt trời hoạt động ở điểm công suất tối đa. Điều này giúp hệ thống năng lượng mặt trời khai thác được nhiều năng lượng nhất có thể, ngay cả khi điều kiện môi trường thay đổi. So với sạc thông thường, việc sử dụng sạc MPPT sẽ mang lại các ưu điểm sau:

- Tối ưu hiệu suất năng lượng: MPPT giúp tấm pin mặt trời hoạt động ở điểm công suất tối đa, đảm bảo thu năng lượng tối đa trong các điều kiện ánh sáng khác nhau.
- Tăng hiệu suất sạc pin: MPPT tối ưu hóa đầu vào từ tấm pin, tăng tốc độ sạc và hiệu quả, kéo dài tuổi thọ pin.
- Hoạt động tốt trong điều kiện thay đổi: MPPT tự điều chỉnh theo biến đổi ánh sáng, duy trì hiệu suất ổn định.
- Bảo vệ hệ thống: MPPT bảo vệ pin và các thành phần khác khỏi điều kiện sạc không lý tưởng, giảm nguy cơ hư hỏng và bảo vệ đầu tư.

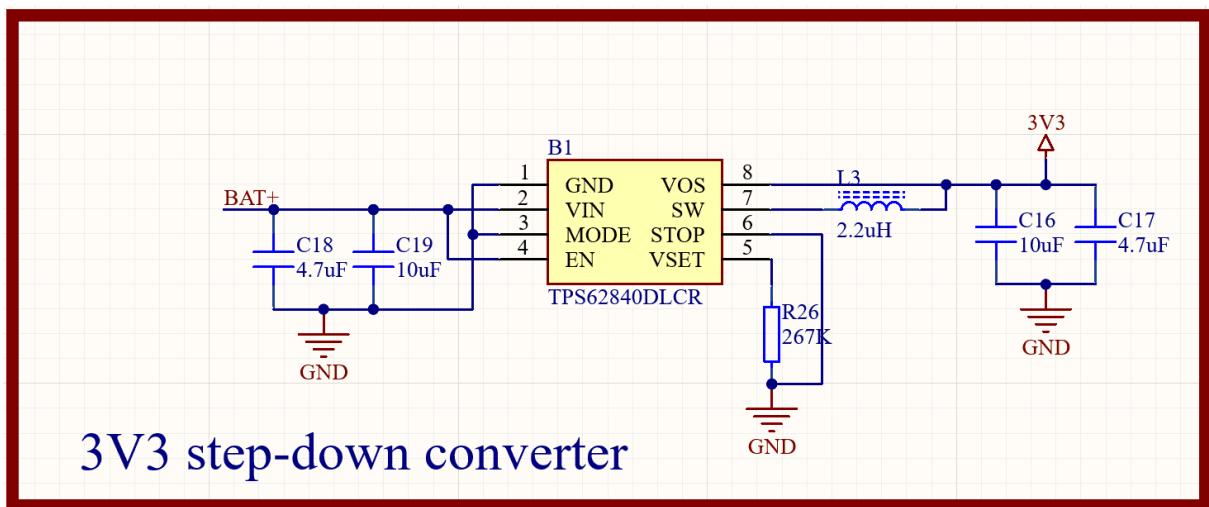
Vì những ưu điểm trên, nhóm cho rằng IC CN3791 rất phù hợp với thiết kế của trạm này. Trên sơ đồ nguyên lý tại hình 3.24, diode D1 là một diode schottky SS34, được nhóm thêm vào với mục đích chặn dòng điện từ pin đổ vào CN3791 khi nó không hoạt động, giảm năng lượng suy hao không cần thiết.



Hình 3.25: Sơ đồ nguyên lý bộ giám sát năng lượng

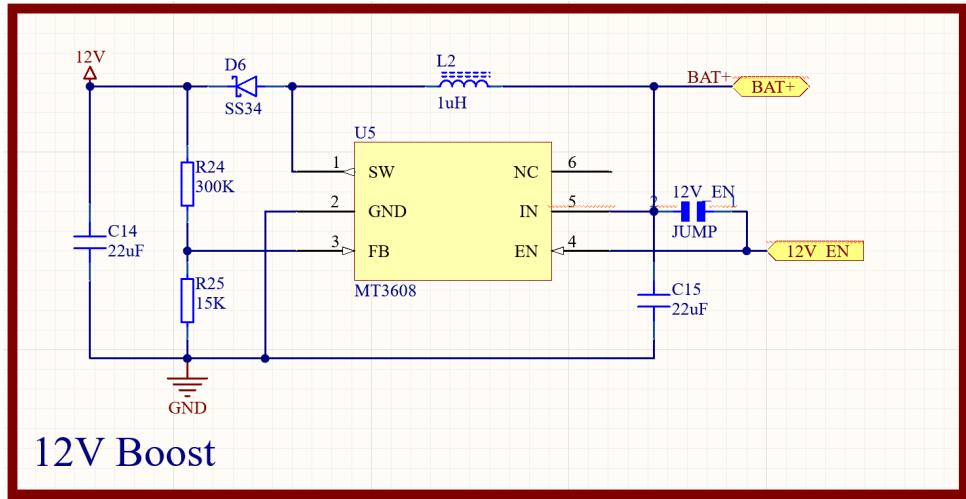
Ngoài ra, trên khối pin còn có một IC INA3221 chuyên giám sát năng lượng được thiết kế như hình 3.25, IC này phụ trách đo điện áp và dòng điện tức thời của pin nguồn và pin năng lượng mặt trời ở một thời điểm cố định khi trạm thu thập này thức dậy thực hiện một chu kì mới. Dựa vào tình trạng nguồn điện tương đối này, người dùng có thể can thiệp vào cơ chế tự điều chỉnh chu kì hoạt động của trạm, giúp chủ động tối ưu thời gian duy trì hoạt động.

Bộ giám sát nguồn điện này cùng với module điều khiển chính được cấp nguồn hoạt động bởi IC ổn áp TPS62840 với dòng tĩnh cực thấp (60nA). TPS62840 là một bộ hạ áp hiệu suất cao, được thiết kế để cung cấp nguồn điện ổn định và hiệu quả cho các thiết bị có yêu cầu năng lượng thấp. Theo hình 3.26, chân MODE của TPS62840 được kéo xuống GND để IC tự chuyển chế độ qua lại giữa PWM (công suất cao) và PFM (tiết kiệm năng lượng) dựa vào dòng tải đầu ra, giúp giảm năng lượng suy hao khi trạm vào trạng thái ngủ. Sơ đồ nguyên lý của TPS62840 được thiết kế như sau:



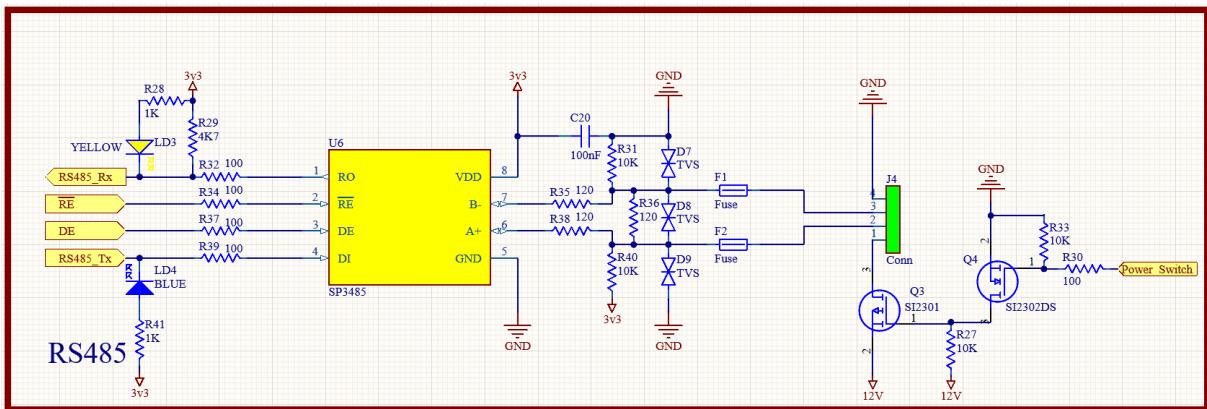
Hình 3.26: Bộ hạ áp pin sang 3.3V

Vì trạm này được thiết kế để có thể truyền thật xa qua sóng Lora, cho nên nó sẽ được đặt ở trên cao hết mức có thể. Vì vậy nhóm đã quyết định sử dụng chuẩn RS485 làm giao thức chính để trạm có thể đọc dữ liệu cảm biến được đặt từ dưới đất như phần trước đã đề cập. Nguồn cấp cho các cảm biến RS485 sẽ do IC MT3608 tăng điện áp từ pin phụ trách, điện áp đầu ra có thể điều chỉnh từ 5V đến 28V bằng cách thay đổi điện trở R24 và R25 trên hình 3.27, tùy thuộc vào loại cảm biến sử dụng và trường hợp ứng dụng. Chân EN của IC sẽ do vi điều khiển chính kiểm soát, chỉ khi nào trạm thức dậy thì IC mới thực hiện tăng áp, khi đọc xong cảm biến thì sẽ kéo chân EN về mức thấp để IC ngưng hoạt động, giúp giảm năng lượng tiêu thụ của trạm, kéo dài thời gian hoạt động.



Hình 3.27: Bộ tăng áp cấp nguồn cho cảm biến RS485

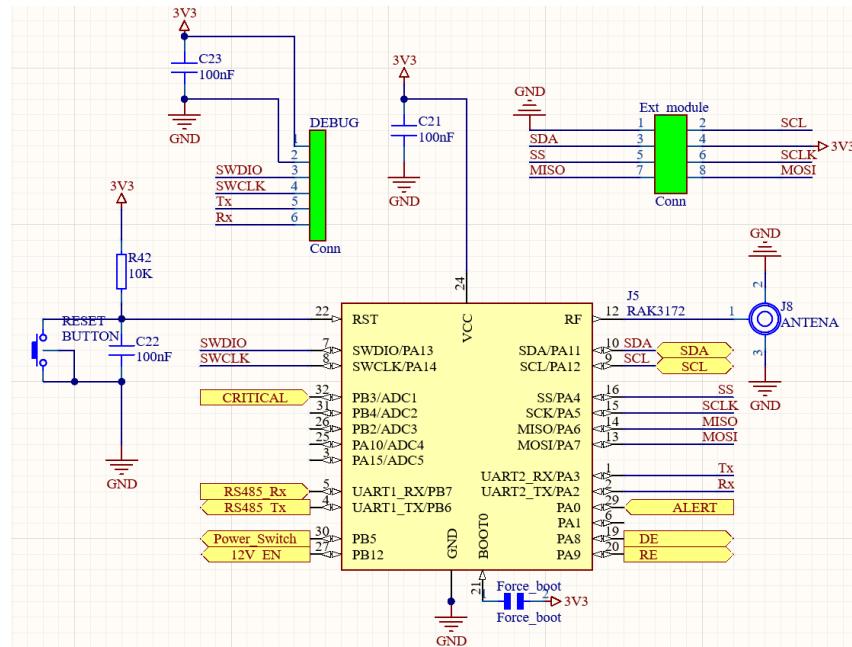
Khối điều khiển RS485 của trạm có thiết kế tương tự như các cảm biến. Tuy nhiên ở cổng kết nối đầu ra, nơi đặt nguồn cấp và hai đường A+ B- của RS485, nhóm có thiết kế thêm 2 mosfet kênh N và kênh P với mục đích ngắt nguồn ra từ bộ tăng áp của MT3608, ngắt điện hoàn toàn các cảm biến để tiết kiệm năng lượng.



Hình 3.28: Sơ đồ nguyên lý bộ điều khiển RS485

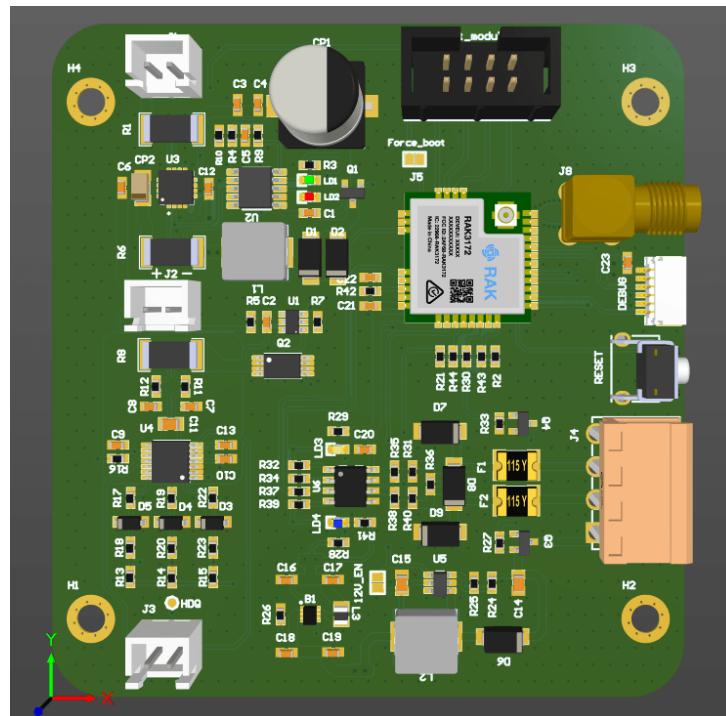
Module điều khiển RAK3172 được nhóm chọn làm vi điều khiển chính cho trạm thu thập dữ liệu này. RAK3172 là một module Lora được thiết kế trên nền tảng vi điều khiển STM32, tích hợp các tính năng mạnh mẽ như giao tiếp Lora với khoảng cách truyền xa, tiêu thụ điện năng thấp và khả năng hoạt động ổn định trong môi trường khắc nghiệt. Với khả năng truyền dữ liệu không dây hiệu quả, RAK3172 cho phép trạm thu thập và truyền tải dữ liệu cảm biến tới trung tâm điều khiển một cách liên tục và đáng tin cậy. Trong khối điều khiển chính này, nhóm thực hiện đề tài đã thêm đầy đủ các cổng nạp mã nguồn và gỡ lỗi, nút Reset, tụ lọc nguồn, cổng anten mở rộng SMA. Ngoài ra, nhóm có thêm một cổng IDC 8 chân để cho phép thiết kế thêm các module rời cho trạm, hỗ trợ giao tiếp qua các chuẩn I2C và SPI, cho phép mở rộng các tính năng khác. Trong

đồ án này, nhóm thực hiện đề tài sẽ sử dụng cổng ICD này để kết nối với một module điều khiển rơ-le, tạo thành một trạm điều khiển các thiết bị có trạng thái là bật và tắt, điển hình là đèn hoặc máy bơm.



Hình 3.29: Sơ đồ nguyên lý module điều khiển chính RAK3172

Sau khi đã hoàn thành sơ đồ nguyên lý, nhóm đã bắt tay vào thiết kế PCB. Đây là hình ảnh 3D của PCB sau khi thiết kế hoàn tất:

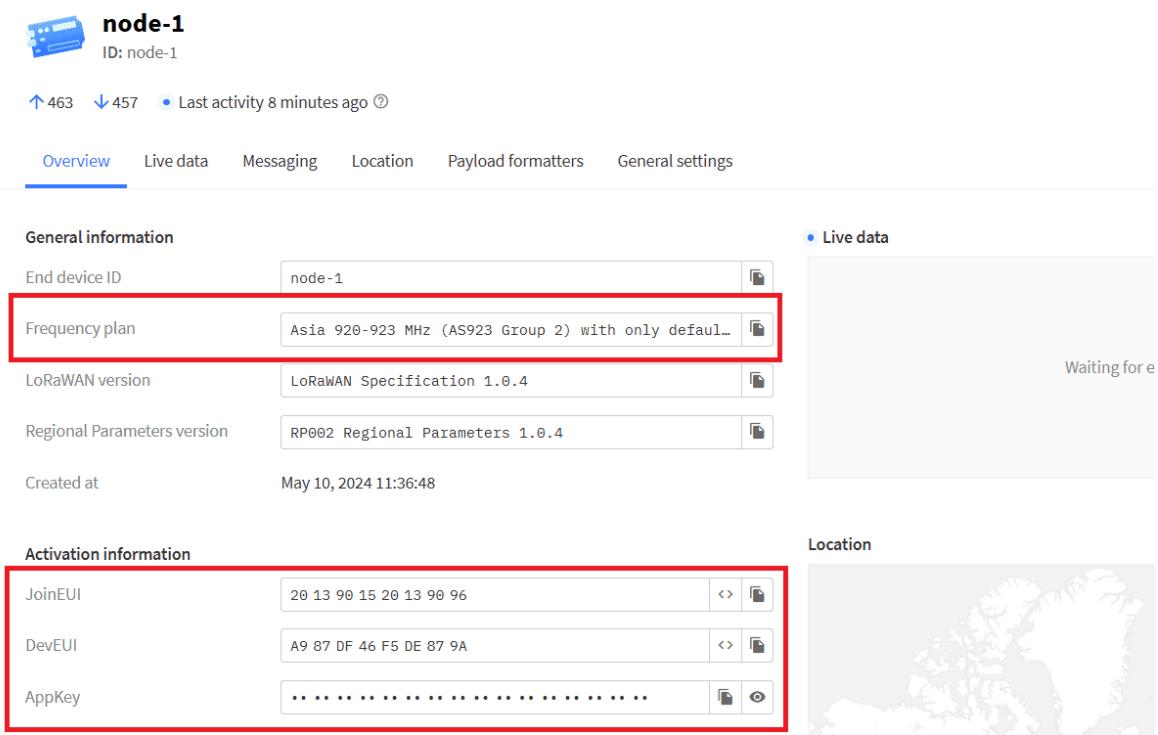


Hình 3.30: Ảnh 3D của trạm thu thập dữ liệu sau khi thiết kế

### 3.6.2 Xây dựng mã nguồn cho trạm thu thập dữ liệu và trạm điều khiển

Một trong các lí do cân nhắc lựa chọn module RAK3172 làm vi điều khiển chính cho trạm thu thập dữ liệu này là vì hãng RAKwireless có cung cấp sẵn một bộ SDK mạnh mẽ có tên là RUI3 (RAKwireless Unified Interface V3), cho phép người dùng lập trình bằng Arduino IDE hoặc Visual Studio, cấu hình qua các lệnh AT. Trong đồ án này, nhóm đã sử dụng Arduino IDE để lập trình cho RAK3172. Toàn bộ hướng dẫn cài đặt và sử dụng đều được hãng RAKwireless hướng dẫn trong tài liệu trên web [?].

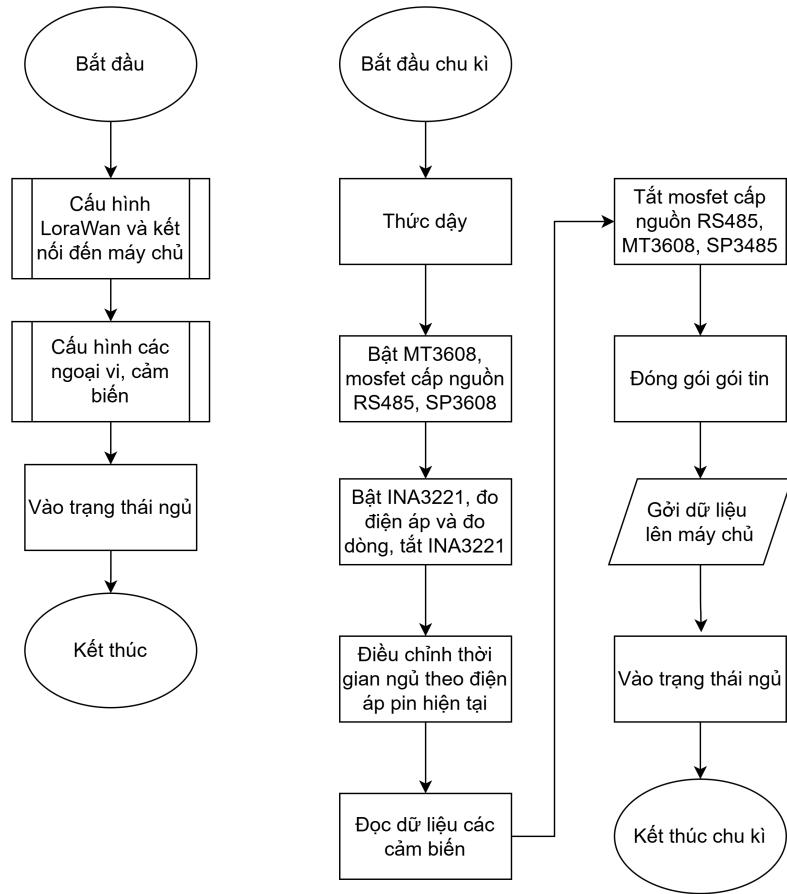
Sau khi đã cài đặt RUI3, nhóm đã bắt đầu nghiên cứu cách triển khai LoraWan trên RAK3172. Các trạm thu thập dữ liệu sẽ được cấu hình chạy ở lớp A, phương thức kích hoạt OTAA. Điều cần làm đầu tiên là tạo một thiết bị đầu cuối trên máy chủ The Things Stack, cấu hình tần số, JoinEUI, DevEUI và AppKey để trạm có thể kết nối lên máy chủ. Các cấu hình quan trọng được khoanh vùng màu đỏ như hình sau:



Hình 3.31: Các cấu hình cần thiết để trạm thu thập có thể kết nối đến máy chủ

Thực hiện sao chép các khóa kích hoạt (JoinEUI, DevEUI và AppKey) vào mã nguồn, nạp chương trình và chạy thử. Khi đã kết nối thành công lên máy chủ, việc tiếp theo cần làm là lập trình đọc dữ liệu của các cảm biến, đóng gói dữ liệu và tối ưu hóa chương trình.

Trạm thu thập này hoạt động theo lưu đồ thuật toán sau:



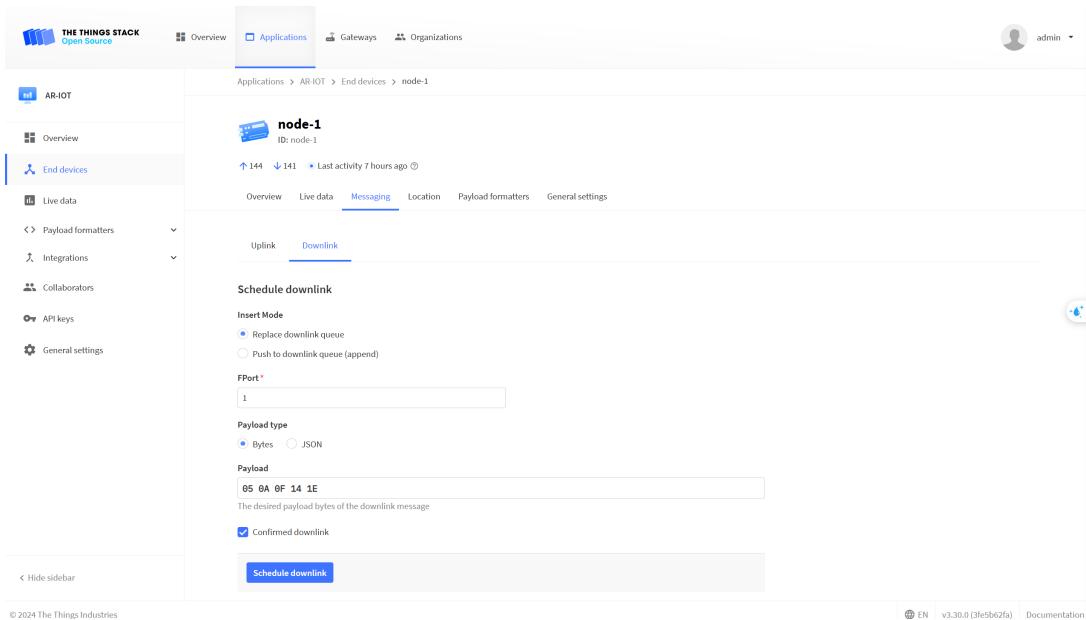
Hình 3.32: Lưu đồ thuật toán lúc khởi động và mỗi chu kì của trạm thu thập dữ liệu

Khi trạm vừa được cấp nguồn, RAK3172 sẽ gửi gói tin kích hoạt OTAA lên máy chủ dựa theo các khóa đã được lập trình sẵn. Cho đến khi nào kích hoạt hoàn tất, RAK3172 đã nhận được gói tin xác nhận đã cho phép tham gia vào mạng LoRaWan thì RAK3172 sẽ thực hiện cấu hình các chân IO điều khiển, cấu hình UART cho RS485, I2C cho INA3221. Các cấu hình ban đầu này đều đặt toàn bộ vào trạng thái tắt, sau đó RAK3172 sẽ đi vào chu kỳ ngủ đầu tiên.

Điều đầu tiên khi RAK3172 thức dậy phải làm là bật IC MT3608 để bắt đầu tăng điện áp cấp cho cảm biến RS485, chờ một ít thời gian để điện áp trở nên ổn định, sau đó bật mosfet cấp nguồn cho cảm biến và bật IC SP3485 để khởi RS485 sẵn sàng hoạt động. Vì cảm biến CO<sub>2</sub> cần 5 giây cho mỗi lần cập nhật dữ liệu mới, cho nên trong lúc chờ cảm biến không khí có dữ liệu, trạm sẽ bật IC INA3221 lên, đo điện áp và dòng điện của các khối pin nguồn và pin năng lượng mặt trời trước. Sau khi đo xong các giá trị điện áp và dòng điện, RAK3172 sẽ tắt INA3221 ngay lập tức, sau đó điều chỉnh thời gian ngủ của trạm.

Thời gian ngủ được điều chỉnh dựa vào một mảng có năm phần tử được lập trình

sẵn, năm phần tử là thời gian ngủ của trạm dựa vào năm điều kiện khác nhau, điều kiện là hiện tại điện áp pin đang ở trong ngưỡng nào và dòng sạc của tấm pin năng lượng mặt trời có đủ cao hay không. Mảng quyết định thời gian ngủ này có thể được can thiệp bằng cách điều khiển máy chủ gửi gói tin downlink cho trạm, gói tin này chứa 5 byte tăng dần, mỗi byte tương ứng với từng phần tử của mảng, tùy thuộc vào môi trường triển khai mà điều chỉnh mảng này một cách hợp lý. Ví dụ để cấu hình mảng này với các thời gian ngủ lần lượt là 5, 10, 15, 20, 30 phút cho trạm có id là node-1, cách đơn giản nhất để gửi gói tin downlink cho trạm là dùng trực tiếp giao diện web của máy chủ như hình sau:



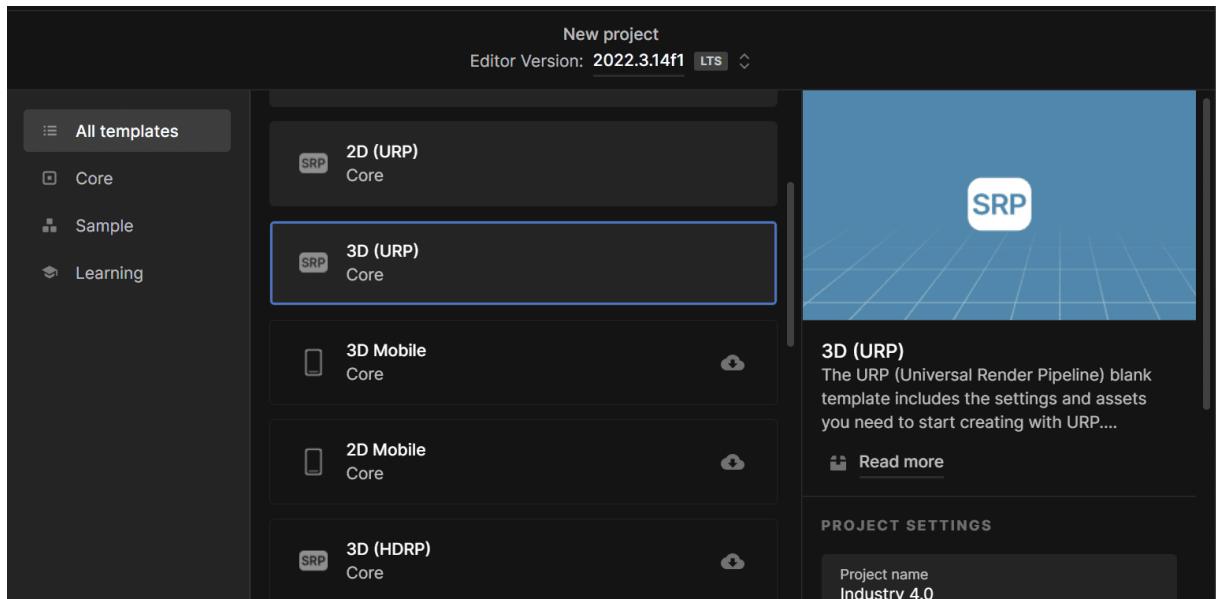
Hình 3.33: Cách cấu hình thời gian ngủ cho trạm từ trang web máy chủ

Sau khi đã điều chỉnh thời gian ngủ của trạm, RAK3172 sẽ tiếp tục chờ cho đến khi đủ 5 giây, sau đó sẽ bắt đầu đọc dữ liệu của các cảm biến không khí và cảm biến đất bằng giao thức Modbus RTU. Khi đã đọc xong toàn bộ cảm biến thì ngay lập tức tắt mosfet cấp nguồn cho cảm biến, sau đó tắt MT3608 và SP3485. Toàn bộ dữ liệu đã đo được (kể cả điện áp và dòng điện) được lưu vào một mảng byte theo một thứ tự cố định, mảng byte sẽ được gửi lên máy chủ. Sau khi đã gửi gói tin uplink xong thì RAK3172 lại trở về trạng thái ngủ.

Đối với phát triển mã nguồn của trạm điều khiển, nhóm thực hiện đề tài sẽ lập trình cho trạm này hoạt động ở lớp C, lớp này chính là lớp sẽ liên tục nghe gói tin downlink từ máy chủ để điều khiển các rơ-le. Việc điều khiển trạm này khá tương tự việc cấu hình thời gian ngủ của trạm thu thập dữ liệu, người dùng chỉ việc gửi 2 byte, byte đầu là thứ tự rơ-le, byte thứ hai là trạng thái tắt/mở (00/01). Trạm này cũng sẽ gửi gói tin uplink mỗi 30 giây để đồng bộ trạng thái rơ-le lên hệ thống.

### 3.7 Thiết lập môi trường phát triển ứng dụng MR

Tạo một Project mới với SRPs được sử dụng là URP. Trong hình 3.34 là giao diện khi khởi tạo một dự án mới trong Unity, giao diện này bao gồm tên phiên bản Unity sử dụng, Render pipeline sẽ được dùng trong dự án và tên của dự án.



Hình 3.34: Giao diện tạo một Project mới trong Unity

Sau đó cài đặt SDK MRTK2 từ ứng dụng Microsoft Mixed Reality Feature Tool. SDK sẽ bao gồm:

- Mixed Reality Toolkit Extension: Đây là một tập hợp các công cụ và tiện ích bổ sung được thiết kế để mở rộng và cải thiện khả năng của MRTK.
- Mixed Reality Toolkit Standard Assets: là một tập hợp các tài sản tiêu chuẩn được cung cấp trong MRTK để hỗ trợ trong việc xây dựng các ứng dụng thực tế hỗn hợp (Mixed Reality) một cách dễ dàng và nhanh chóng hơn.
- Mixed Reality OpenXR Pulgin: Là một phần mở rộng của OpenXR, được thiết kế để hỗ trợ phát triển các ứng dụng thực tế hỗn hợp (Mixed Reality) trên nhiều thiết bị và nền tảng khác nhau.

Ngoài những thư viện mặc định được Unity cung cấp, nhóm phát triển đã cài đặt và tích hợp thêm một số gói thư viện quan trọng để đáp ứng nhu cầu cụ thể của dự án. Các thư viện này bao gồm: AR Foundation, Google ARCore XR Plugin, Newtonsoft Json, Universal Render Pipeline, Model target generator, M2MqttUnity-master. Mỗi thư viện được lựa chọn trong dự án đều có những tính năng và ứng dụng cụ thể, nhằm đáp ứng các yêu cầu kỹ thuật.

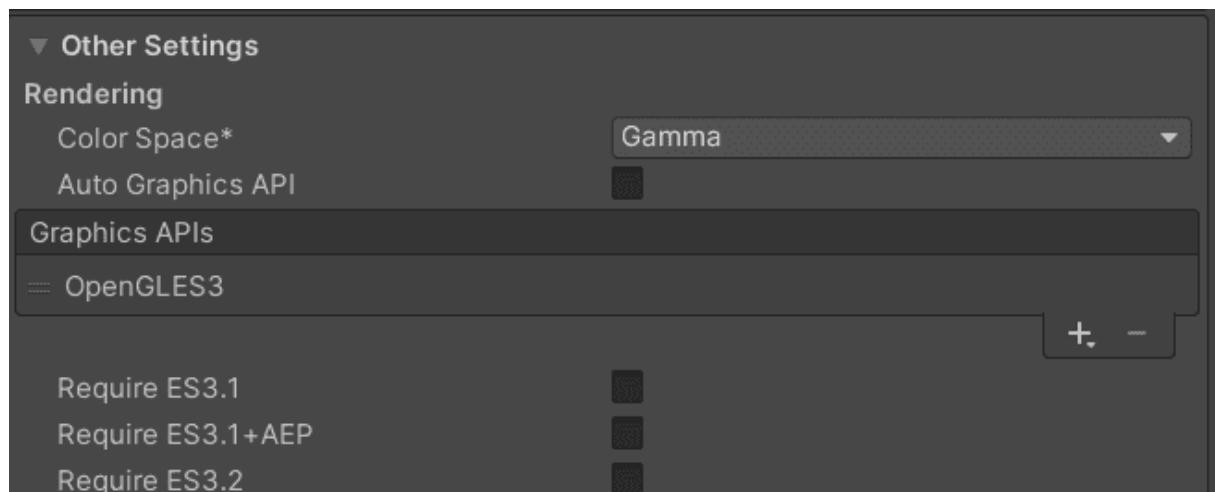
Các thông tin chi tiết về phiên bản, chức năng sử dụng từng thư viện sẽ được trình bày trong bảng dưới:

Bảng 3.14: Các gói thư viện được sử dụng trong dự án.

<b>Số</b>	<b>Gói thư viện</b>	<b>Phiên bản</b>	<b>Chức năng</b>
1	AR Foundation	5.0.7	Cung cấp một bộ API duy nhất để tạo ra các ứng dụng AR mà có thể chạy trên các thiết bị hỗ trợ ARKit của Apple và ARCore của Google.
2	Google ARCore XR Plugin	5.0.7	Cho phép sử dụng ARCore Google cho dự án Unity.
3	Newtonsoft Json	3.2.1	Chuyển json thành các định dạng khác.
4	Universal Render Pipeline	14.0.8	Hỗ trợ sử dụng URP.
5	Model target generator	10.22	Dùng cho mô hình Vuforia học model.
6	M2MqttUnity-master	1.0	Tích hợp MQTT vào Unity.

### 3.7.1 Thiết lập trong Player Settings

Đầu tiên nhóm thực hiện đề tài sẽ cấu hình Rendering, sử dụng không gian màu Gamma như trong hình 3.35 vì Không gian màu này lưu trữ giá trị màu sắc theo đường cong gamma, mô phỏng cách thức mà mắt người nhìn thấy ánh sáng.



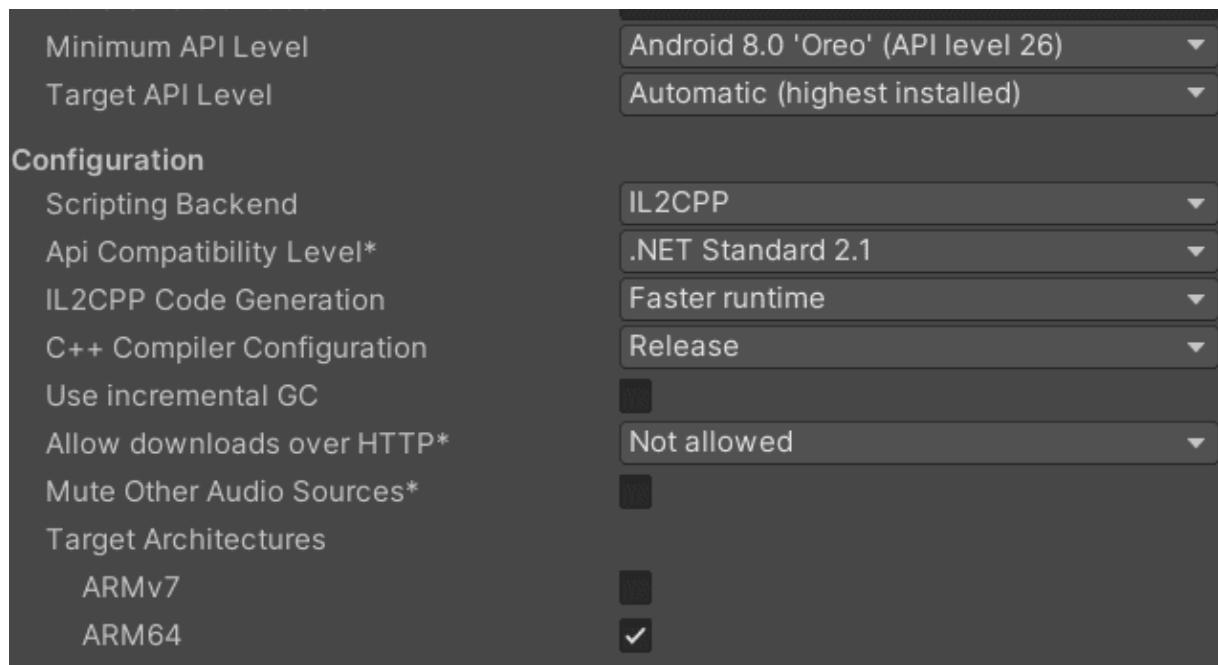
Hình 3.35: Cấu hình không gian màu.

So với việc sử dụng Linear Space thì Gamma sẽ đem lại nhiều lợi ích hơn trong chương trình như:

- Màn hình điện thoại thường hiển thị hình ảnh theo đường cong gamma. Việc sử dụng không gian màu Gamma sẽ đảm bảo rằng hình ảnh được hiển thị chính xác trên màn hình.
- Linear Space có thể dẫn đến hiện tượng "banding" (sọc màu) trên màn hình điện thoại, đặc biệt là khi hiển thị các gradient màu.
- Gamma có thể giúp cải thiện hiệu suất trên một số thiết bị di động.

API đồ họa được nhóm sử dụng trong dự án là OpenGL ES3 có thể thấy qua hình 3.35. Đây là một phiên bản đặc biệt cho các thiết bị di động và nhúng như điện thoại di động, máy tính bảng, và các hệ thống nhúng khác. Đây là một API đồ họa nhẹ nhưng hiệu quả, cung cấp các công cụ và tính năng để tạo ra các ứng dụng đồ họa 2D và 3D. API này là yêu cầu phổ biến khi sử dụng các ứng dụng XR trên các thiết bị tích hợp ARCore.

Các thiết lập để xuất bản chương trình cho Unity Engine được trình bày tại hình sau:



Hình 3.36: Các cấu hình quan trọng để xuất bản ứng dụng.

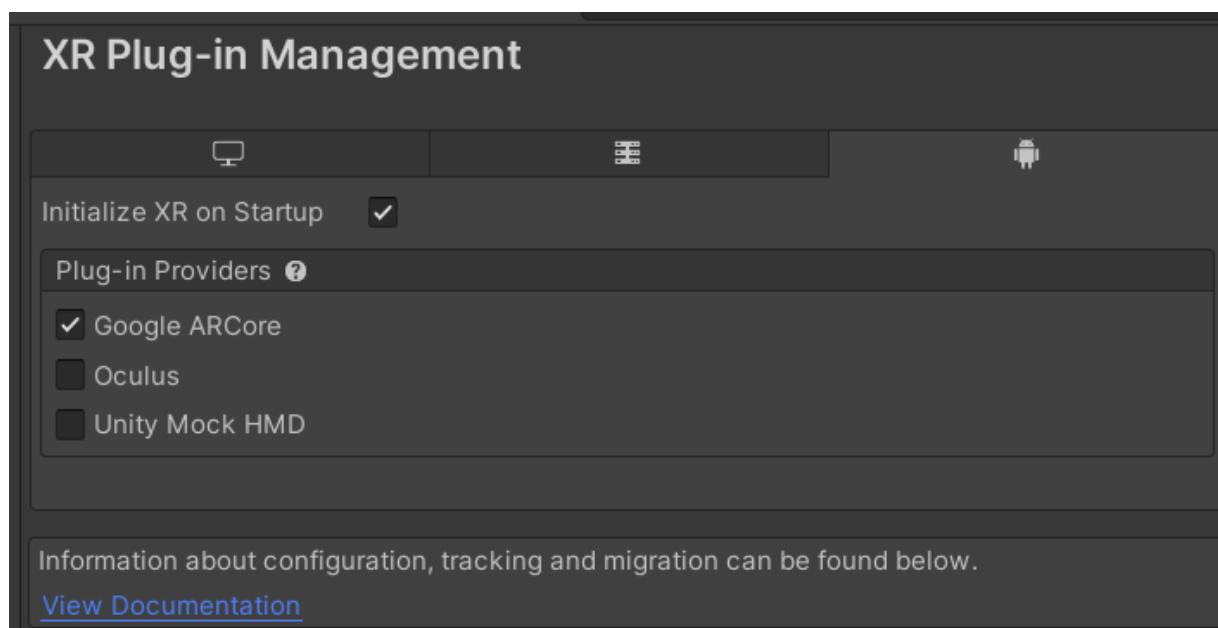
Trong phần Scripting backend nhóm chọn sử dụng IL2CPP thay vì sử dụng Mono, để nhằm chuyển đổi mã C# gốc trong Unity thành mã C++, sau đó từ C++ sẽ được biên dịch thành các tệp thực thi cho nền tảng sử dụng ứng dụng.

Do yêu cầu về tính ổn định và tương thích với Playstore, nhóm đã quyết định lựa chọn phiên bản Android 8.0 là phiên bản Android tối thiểu để ứng dụng có thể hoạt động. Lý do cho phiên bản Android 8.0 được lựa chọn là:

- Hỗ trợ ARCore: Phiên bản Android thấp nhất mà ARCore có thể hoạt động là Android 7.0. Tuy nhiên, để đảm bảo trải nghiệm tốt nhất cho người dùng và giảm thiểu các lỗi tiềm ẩn, nhóm đã chọn Android 8.0 là phiên bản tối thiểu.
- Yêu cầu 64-bit của Playstore: Kể từ năm 2019, Google Play yêu cầu tất cả các ứng dụng mới phải hỗ trợ kiến trúc 64-bit. Việc lựa chọn ARM64 đáp ứng yêu cầu này và đảm bảo khả năng tương thích với Playstore.

Với những cân nhắc trên, Android 8.0 và ARM64 là lựa chọn phù hợp nhất cho dự án, mang lại sự ổn định, hiệu suất và khả năng tương thích tốt nhất cho ứng dụng.

### 3.7.2 Thiết lập về XR Plugin Management



Hình 3.37: Giao diện của XR Plug-in Management.

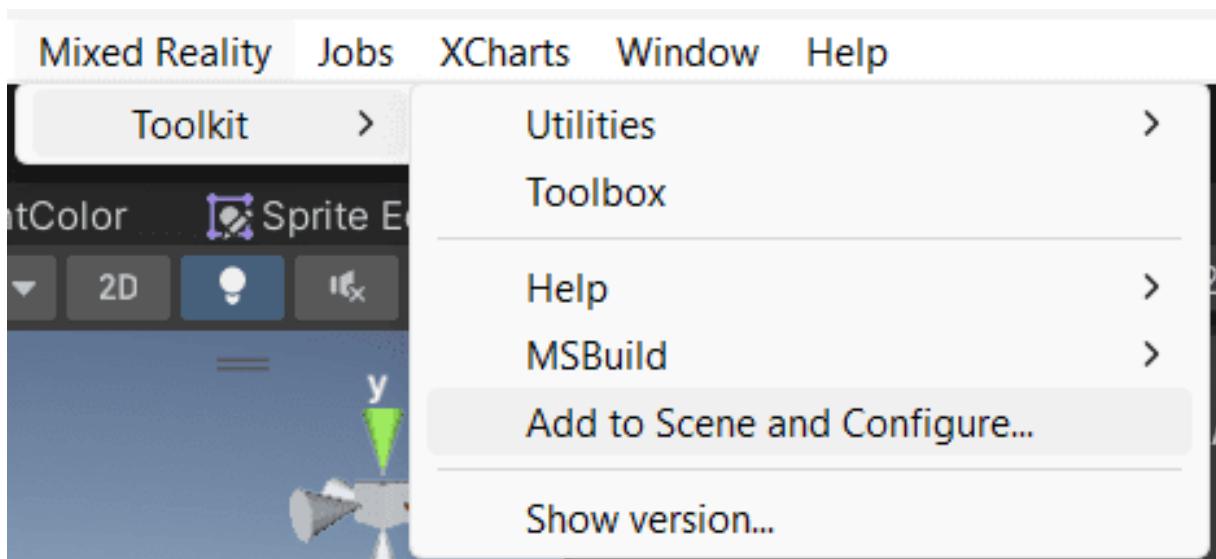
XR Plugin Management là một công cụ quản lý có trong Unity có thể được thấy rõ ở hình 3.37, là một thành phần cốt lõi giúp quản lý các plugin XR. Nó cung cấp một giao diện trực quan và dễ sử dụng, cho phép các nhà phát triển dễ dàng thêm, cấu hình, và quản lý các plugin XR cần thiết cho dự án. Công cụ này hỗ trợ các plugin cho nhiều nền tảng XR khác nhau, bao gồm ARCore (Android), ARKit (iOS), Windows Mixed Reality, và nhiều nền tảng khác. Tại hình 3.37 cho thấy dự án đang sử dụng plugin

Google ARCore nhằm để sử dụng các ứng dụng XR trên các thiết bị Android được tích hợp ARCore.

### 3.7.3 Thiết lập MRTK hoạt động trên điện thoại thông minh

Mixed Reality Toolkit (MRTK) cung cấp nhiều công cụ và tính năng để hỗ trợ phát triển các ứng dụng thực tế hỗn hợp (MR) trong Unity. Một trong những tính năng hữu ích của MRTK là "Add Scene with Configure". Tính năng này giúp nhóm thiết lập và cấu hình một cảnh (scene) mới với các cài đặt MRTK mặc định, tiết kiệm thời gian và đảm bảo rằng cảnh của chương trình được cấu hình đúng cách để sử dụng các khả năng của MRTK.

Nhóm thực hiện đề tài sử dụng "Add Scene with Configure" ở hình 3.38 để thiết lập tự động các cấu hình cần thiết. Khi sử dụng MRTK sẽ tự động thêm các thành phần cần thiết vào cảnh của chương trình, chẳng hạn như Mixed Reality Toolkit và các cấu hình của nó, bao gồm việc thiết lập các cài đặt đầu vào, quản lý cảnh, hệ thống tương tác, và các cấu hình khác liên quan đến MRTK. "Add Scene with Configure" đảm bảo rằng các thành phần này được cấu hình và sẵn sàng để hệ thống sử dụng mà không cần phải thiết lập lại từ đầu.



Hình 3.38: Vị trí của tuỳ chọn "Add Scene with Configure."

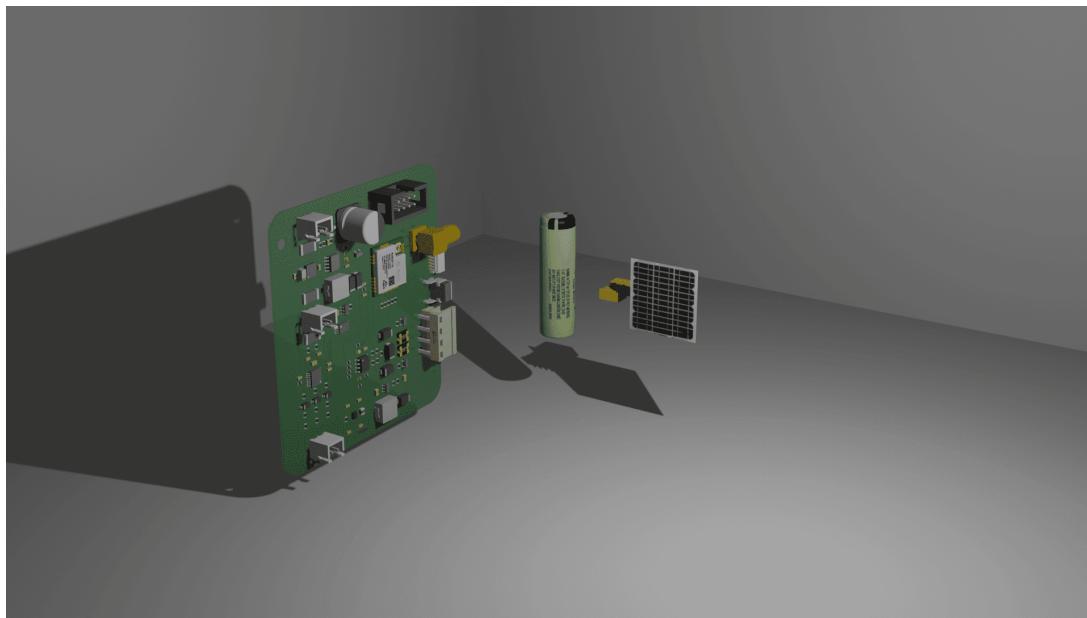
Trong thành phần MixedRealityToolkit, nhóm đã thiết lập để ứng dụng sử dụng camera Unity AR Foundation. Đây là một tính năng được tích hợp trong AR Foundation. AR Foundation là một framework của Unity cho phép phát triển ứng dụng AR trên nhiều nền tảng khác nhau (ARKit của Apple và ARCore của Google). Khi sử dụng AR Foundation Camera Settings trong MRTK, nhóm đang kết hợp các tính năng của AR Foundation với khả năng của MRTK để tạo ra các ứng dụng MR. Cụ thể, nó giúp cấu

hình camera để phù hợp với các yêu cầu của AR Foundation trong khi vẫn duy trì các khả năng MR có trong MRTK. Hoạt động đúng với các yêu cầu của AR Foundation là bao gồm việc thiết lập chiều cao camera, hướng camera, và các thuộc tính khác liên quan đến AR. Cũng như là thiết lập cho phép các ứng dụng có thể hoạt động trên các thiết bị hỗ trợ ARKit của Apple và ARCore của Google.

## 3.8 Phát triển ứng dụng MR

### 3.8.1 Tiến hành vẽ và tối ưu hóa mô hình 3D

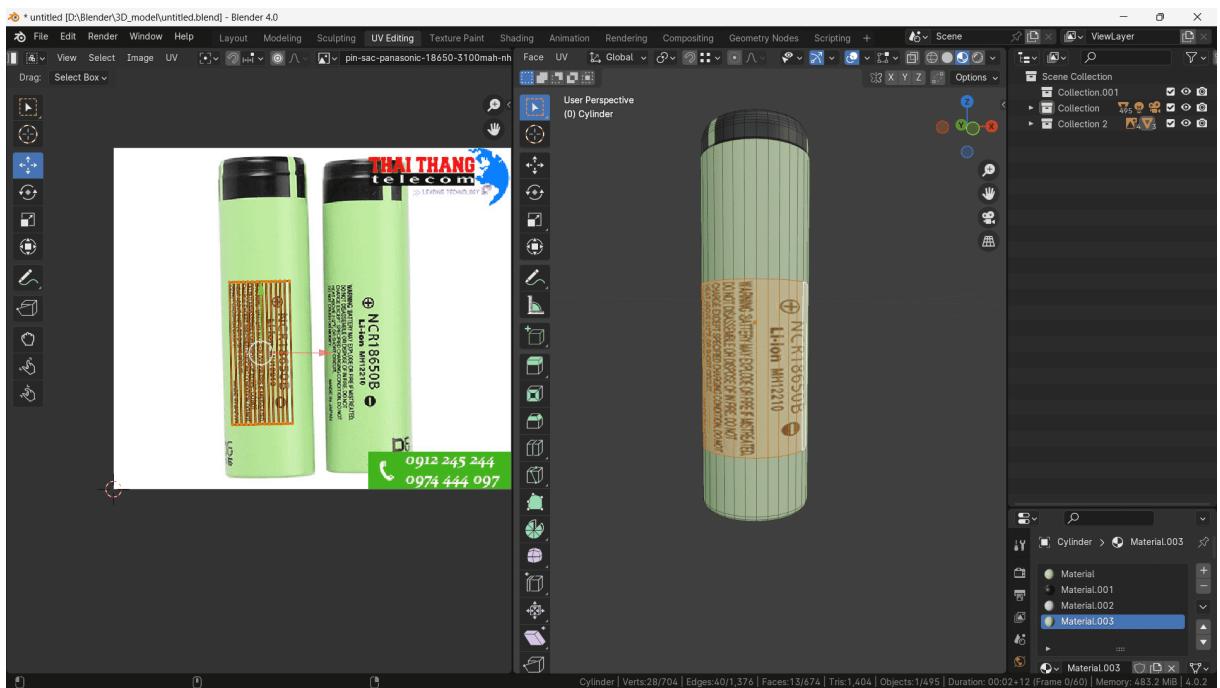
Nhóm thực hiện đề tài đã sử dụng phần mềm Blender để tạo ra các mô hình 3D chi tiết. Các mô hình này bao gồm các thành phần quan trọng như pin, chân cắm, và tấm năng lượng mặt trời.Ngoài việc sử dụng phần mềm Blender để tạo ra các mô hình 3D chi tiết, nhóm còn tích hợp một mô hình mạch điện của trạm từ phần mềm vẽ mạch điện Altium. Sau khi hoàn thiện các mô hình trong Blender, nhóm đã tiến hành tối ưu hóa số lượng đa giác (polygons) để đảm bảo hiệu suất khi đưa vào Unity. Chi tiết các mô hình được nhóm sử dụng trong bài báo cáo nằm ở hình 3.39.



Hình 3.39: Các mô hình 3D được sử dụng trong hệ thống .

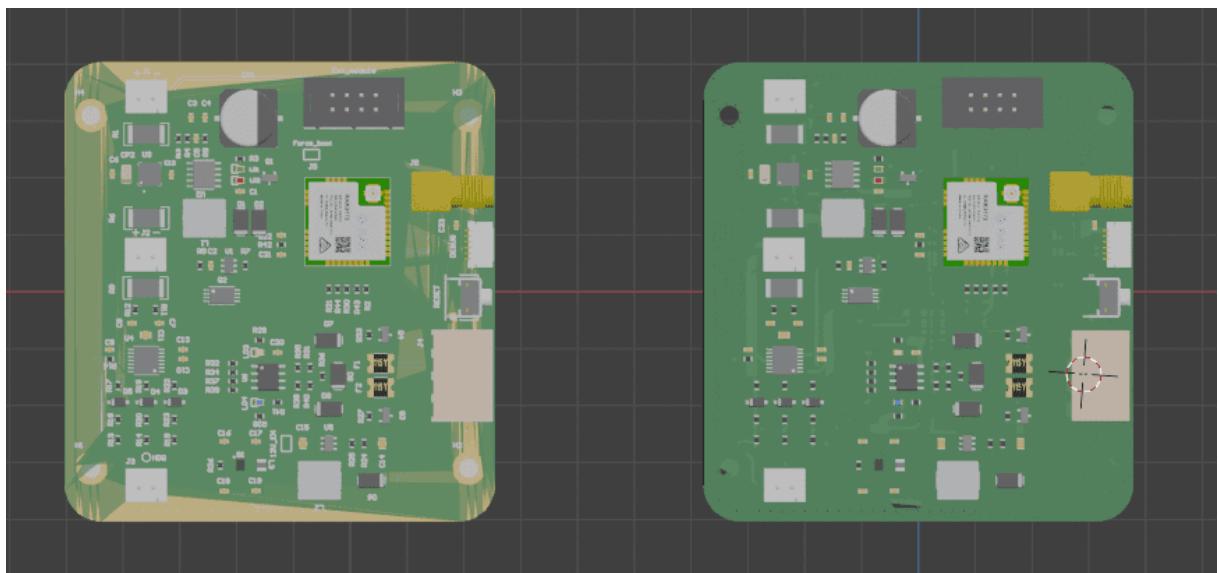
Sau đây là mô tả chi tiết về các bước tiến hành vẽ và hoàn thiện một mô hình 3D. Đầu tiên, nhóm sẽ bắt đầu bằng việc tạo dựng lưới cơ bản từ một khối hình đơn giản như khối hộp. Sau khi hoàn thành việc dựng lưới cơ bản, nhóm sẽ tiến hành vẽ màu và thêm các chi tiết bề mặt để mô hình trông sống động và thực tế hơn. Hoạt động trắc UV được diễn tả ở hình 3.40.

Như đã trình bày ở trên, ngoài việc sử dụng Blender để dựng các mô hình 3D chi



Hình 3.40: Lên màu và trải UV map cho mô hình.

tiết, nhóm còn sử dụng phần mềm Altium để thiết kế mạch điện. Tuy nhiên, mô hình từ Altium không thể đưa trực tiếp vào Unity vì số lượng đa giác quá cao, điều này sẽ làm ảnh hưởng đến hiệu năng tổng thể của chương trình. Vì vậy, trước khi nhập vào Unity, nhóm đã tiến hành tối ưu hóa các đa giác trên mô hình, giảm chúng xuống mức thấp nhất có thể mà vẫn giữ được hình dạng gần như lúc ban đầu.

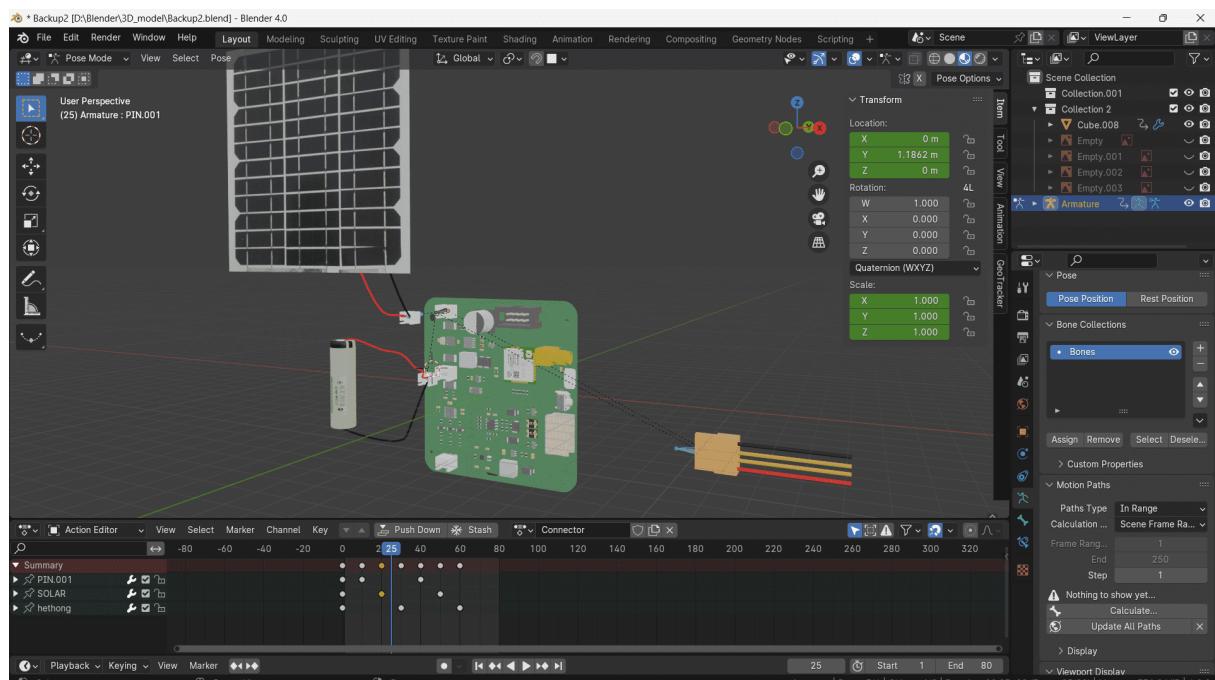


Hình 3.41: Bên trái là mô hình từ Altium, bên phải là mô hình đã được tối ưu.

Qua hình 3.41 thì mô hình phía bên phải là mô hình được đưa trực tiếp từ Altium với số lượng đa giác lên tới 517,200. Đây là một con số rất lớn, đặc biệt khi so sánh với

tiêu chuẩn tối ưu để chạy ổn định trên các thiết bị di động phổ thông mà nhóm đã đúc kết được qua kinh nghiệm thực hiện các dự án khác nhau, con số tối ưu đó vào khoảng 100,000 đa giác trên mỗi khung hình. Điều này cho thấy rằng, nếu sử dụng mô hình này trực tiếp, hiệu năng của ứng dụng sẽ bị ảnh hưởng nghiêm trọng, dẫn đến trải nghiệm người dùng kém hoặc nghiệm trọng hơn là ứng dụng hoàn toàn không thể sử dụng được. Mô hình bên trái là mô hình sau khi được tối ưu, với chỉ 33,961 đa giác, vẫn giữ được hình dạng và chi tiết gần như ban đầu nhưng lại tiêu thụ ít tài nguyên hơn nhiều.

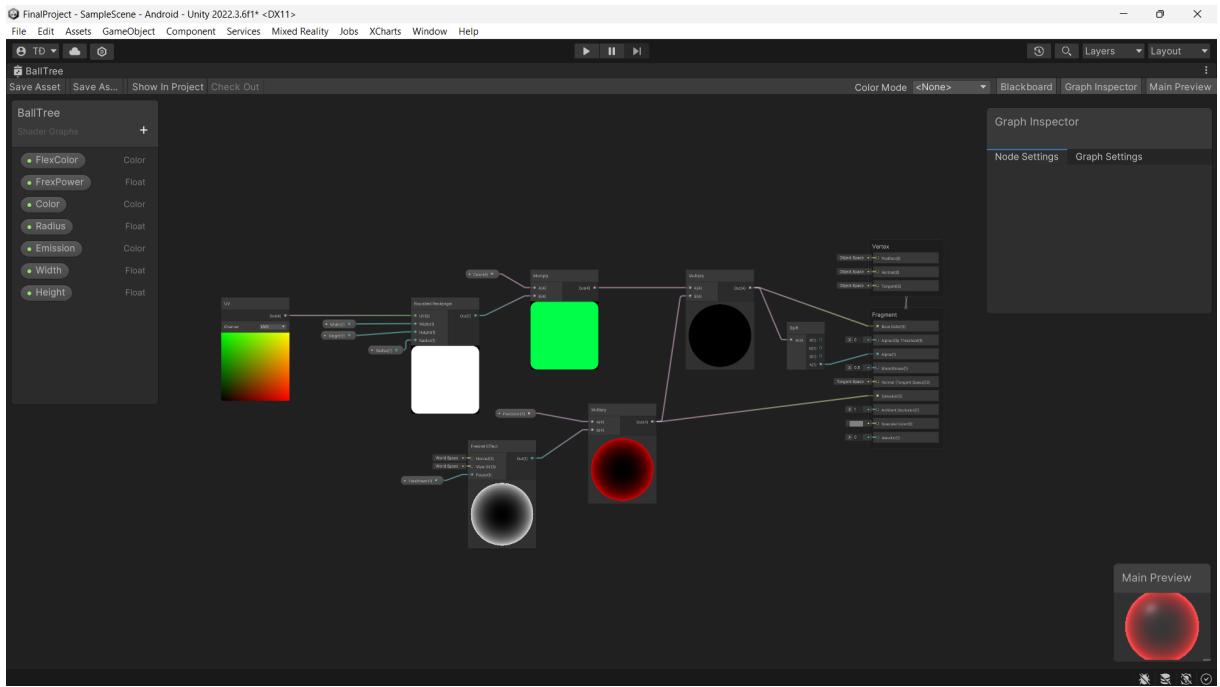
### 3.8.2 Thiết kế hoạt ảnh lắp ráp trong Blender



Hình 3.42: Thiết kế hoạt ảnh bằng Blender.

Hình ảnh trên diễn tả các quá trình để tạo ra một hoạt ảnh trong Blender và tích hợp nó vào Unity. Chi tiết các bước thực hiện như sau. Đầu tiên, tạo một bộ khung xương để điều khiển mô hình. Bộ khung xương này được gọi là "rig", bao gồm nhiều xương liên kết với nhau. Mỗi xương sẽ điều khiển một phần cụ thể của lưỡi của mô hình. Việc sử dụng khung xương giúp tạo ra các chuyển động tự nhiên và linh hoạt cho mô hình, dễ dàng điều chỉnh và kiểm soát. Sau khi tạo xong bộ khung xương, bước tiếp theo là liên kết các xương với lưỡi mô hình. Điều này đảm bảo khi xương di chuyển, các phần tương ứng của mô hình cũng di chuyển theo. Các hoạt ảnh được dựng trên một đoạn duy nhất để có thể dễ dàng đưa vào Unity. Quá trình này bao gồm việc thiết kế các chuyển động của mô hình bằng cách di chuyển các xương ở các vị trí khác nhau. Cuối cùng, xuất mô hình và hoạt ảnh dưới định dạng Blend từ Blender. Nhập tệp Blend vào Unity. Unity sẽ tự động nhận diện và tích hợp các hoạt ảnh, lưỡi và khung xương từ tệp này.

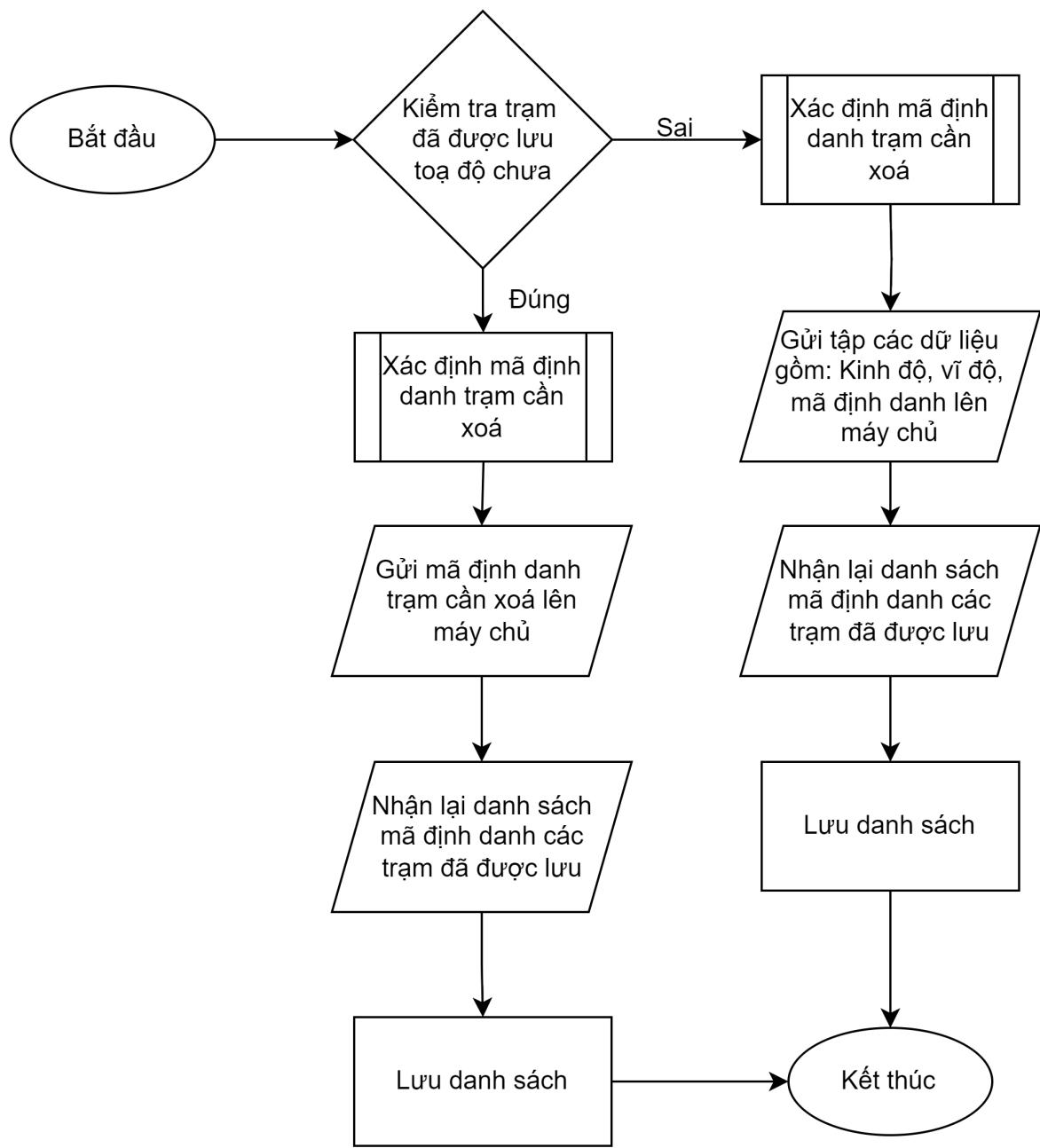
### 3.8.3 Thiết kế shader cho material



Hình 3.43: Giao diện shader Graph.

Hình trên minh họa quá trình xây dựng một shader bằng cách sử dụng các nút (Node) được cung cấp bởi URP. Quá trình này được thực hiện theo các bước sau: Ở bước đầu tiên, shader sẽ lấy tọa độ UV của bề mặt vật thể thông qua node UV. Tọa độ UV là hệ tọa độ 2D dùng để ánh xạ texture lên bề mặt của mô hình 3D. Tọa độ UV này sau đó được đưa vào node Rounded Rectangle, giúp tạo ra hình chữ nhật với các góc bo tròn. Node này sử dụng các giá trị chiều rộng (Width), chiều cao (Height), và bán kính bo tròn (Radius) để xác định hình dạng và kích thước của hình chữ nhật. Node Rounded Rectangle được kết hợp với node Color để thêm màu sắc cơ bản cho hình chữ nhật. Màu sắc này sau đó được nhân với kết quả của node Rounded Rectangle thông qua node Multiply. Điều này giúp tạo ra hình ảnh màu sắc đầu tiên của hình chữ nhật bo tròn. Bên cạnh đó hiệu ứng quang học được tạo ra bằng cách sử dụng node Fresnel Effect, giúp tạo hiệu ứng quang học cho hình chữ nhật. Node này sử dụng các thông số về hướng nhìn và hướng bề mặt để tạo ra hiệu ứng ánh sáng mượt mà. Cuối cùng, các kênh màu RGBA của hình ảnh được tách ra thông qua node Split. Việc tách kênh giúp điều chỉnh các thông số màu sắc và alpha một cách riêng biệt, đảm bảo rằng shader có thể hiển thị đúng màu sắc và độ trong suốt cần thiết. Kết quả cuối cùng của quá trình này được đưa vào node Fragment để kết hợp và hiển thị shader. Shader này sau đó được áp dụng vào material của vật thể, giúp hiển thị các hiệu ứng màu sắc và ánh sáng phức tạp trên bề mặt của vật thể.

### 3.8.4 Phát triển chức năng thêm và xoá vị trí của trạm



Hình 3.44: Lưu đồ thuật toán cho sự kiện thêm và xoá vị trí của trạm

Quá trình thêm và xoá vị trí của các trạm trong hệ thống hoạt động như sau: Ban đầu, chương trình sẽ hiển thị cho người dùng một danh sách các nút được đánh số từ một đến chín, tương ứng với các trạm trong hệ thống. Mỗi nút sẽ đại diện cho một trạm, và những trạm nào đã được lưu tọa độ vị trí sẽ có nút tương ứng phát sáng, tạo điều kiện dễ dàng cho người dùng nhận diện.

Khi người dùng nhấn vào một nút đang phát sáng, điều này có nghĩa là họ muốn xoá trạm tương ứng với nút đó. Ngay lập tức, dữ liệu về mã định danh của trạm sẽ được

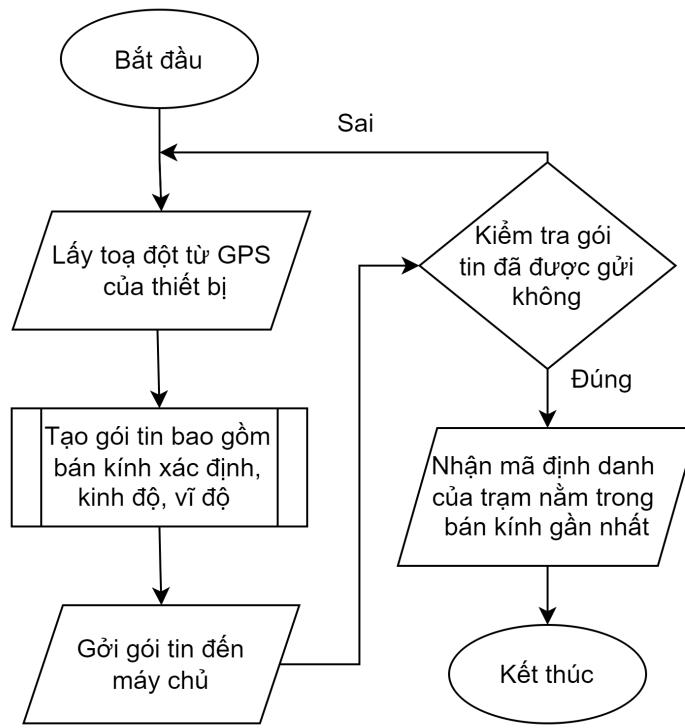
gửi tới máy chủ. máy chủ sẽ thực hiện hành động xóa trạm khỏi cơ sở dữ liệu.

Ngược lại, khi người dùng nhấn vào một nút không phát sáng, chương trình hiểu rằng người dùng muốn thêm một trạm mới tại vị trí này. Lúc này chương trình sẽ lấy tọa độ GPS từ người dùng, chương trình sẽ gửi thông tin về tọa độ vị trí và số thứ tự của trạm thu thập dữ liệu lên máy chủ, máy chủ sẽ tiếp nhận và lưu trạm mới vào cơ sở dữ liệu.

Sau khi hoàn thành một trong hai thao tác thêm hoặc xóa trạm, máy chủ sẽ gửi lại cho người dùng thông tin cập nhật về các trạm đã được lưu trữ trong hệ thống. Người dùng có thể thấy sự thay đổi này ngay lập tức trên giao diện của chương trình, giúp họ dễ dàng quản lý và điều chỉnh vị trí các trạm một cách hiệu quả. Có thể thấy quá trình hoạt động của chương trình qua lưu đồ hình 3.44.

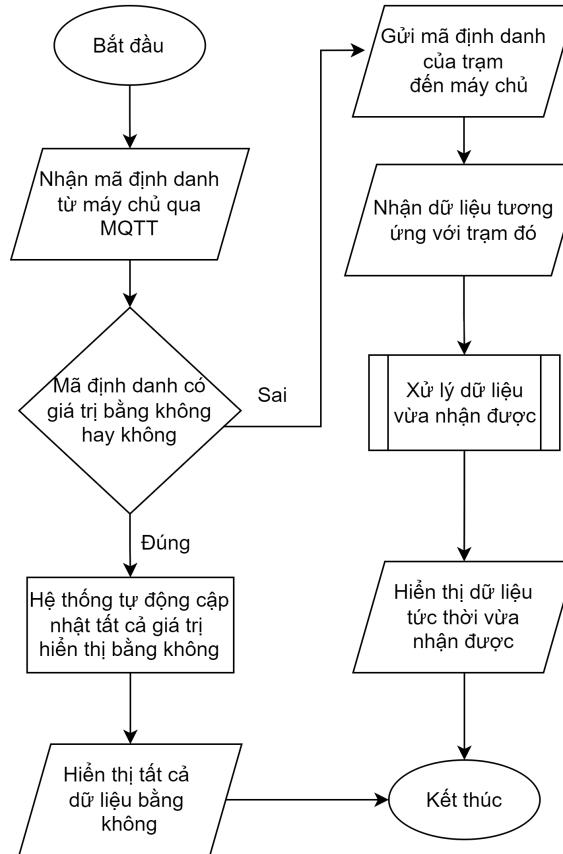
### 3.8.5 Triển khai thuật toán tìm kiếm trạm gần nhất trong phạm vi

Quá trình xác định trạm gần nhất được thực hiện theo các bước chi tiết như sau. Ngay khi chương trình bắt đầu, tọa độ của người dùng sẽ được liên tục cập nhật từ hệ thống GPS của thiết bị khoảng một giây một lần. Quá trình này đảm bảo rằng vị trí hiện tại của người dùng luôn được theo dõi một cách chính xác và thời gian thực. Tọa độ vừa được cập nhật sẽ được đóng gói cùng với bán kính quét để tạo thành một gói dữ liệu, bán kính quét nhằm xác định phạm vi trong đó hệ thống sẽ tìm kiếm các trạm gần nhất. Gói dữ liệu này sẽ được gửi lên máy chủ thông qua giao thức MQTT. MQTT được chọn vì nó giúp giảm thiểu lượng dữ liệu gửi đi so với REST API. Các tin nhắn MQTT có kích thước nhỏ và yêu cầu ít dữ liệu hơn để thiết lập và duy trì kết nối so với HTTP. Điều này rất quan trọng để đảm bảo hiệu suất và tiết kiệm băng thông, đặc biệt trong các ứng dụng cần cập nhật liên tục và nhanh chóng. Dữ liệu về tọa độ và bán kính quét được gửi lên máy chủ mỗi giây một lần. Điều này đảm bảo rằng máy chủ luôn có thông tin vị trí mới nhất của thiết bị, giúp phát hiện trạm gần nhất một cách nhanh chóng và chính xác. Máy chủ, sau khi nhận được gói dữ liệu, sẽ tiến hành xử lý để xác định trạm gần nhất dựa trên tọa độ và bán kính quét được cung cấp. Khi máy chủ xác định được trạm gần nhất, nó sẽ ngay lập tức gửi mã định danh của trạm đó về cho chương trình để kích hoạt các hành động tiếp theo. Tất cả quá trình này được thể hiện ở lưu đồ hình 3.45.



Hình 3.45: Lưu đồ nhận mã định danh trạm gần nhất từ máy chủ.

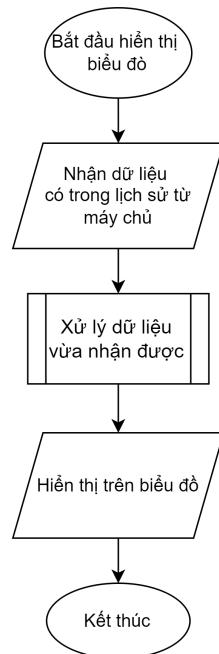
### 3.8.6 Phát triển chức năng hiển thị dữ liệu.



Hình 3.46: Lưu đồ hiển thị dữ liệu trong một vòng lặp.

Sau khi nhận được mã định danh trạm từ máy chủ, chương trình sẽ tiến hành kiểm tra mã định danh này. Nếu mã định danh bằng không, điều đó có nghĩa là không có trạm nào ở gần vị trí hiện tại. Trong trường hợp này, chương trình sẽ hiển thị tất cả các thông số bằng không. Nếu mã định danh đã nhận thay đổi so với mã định danh trước đó, chương trình sẽ gửi mã định danh mới này lên máy chủ để yêu cầu dữ liệu chi tiết của trạm tương ứng. Khi dữ liệu từ máy chủ được nhận về, chương trình sẽ bắt đầu quá trình xử lý và hiển thị dữ liệu này trên giao diện người dùng tương ứng. Dữ liệu có thể bao gồm thông tin về các cảm biến, thông số kỹ thuật, và trạng thái hoạt động của trạm. Trong lưu đồ ở hình 3.46 trình bày rõ các thao tác của chương trình khi hiển thị dữ liệu lên giao diện.

Trong hình 3.47 sẽ trình bày sự kiện hiển thị lịch sử các giá trị đã đo được từ các thiết bị có trong trạm, chương trình cung cấp một giao diện đồ họa tiện lợi. Người dùng chỉ cần thao tác ấn vào biểu tượng tương ứng với loại dữ liệu muốn xem.

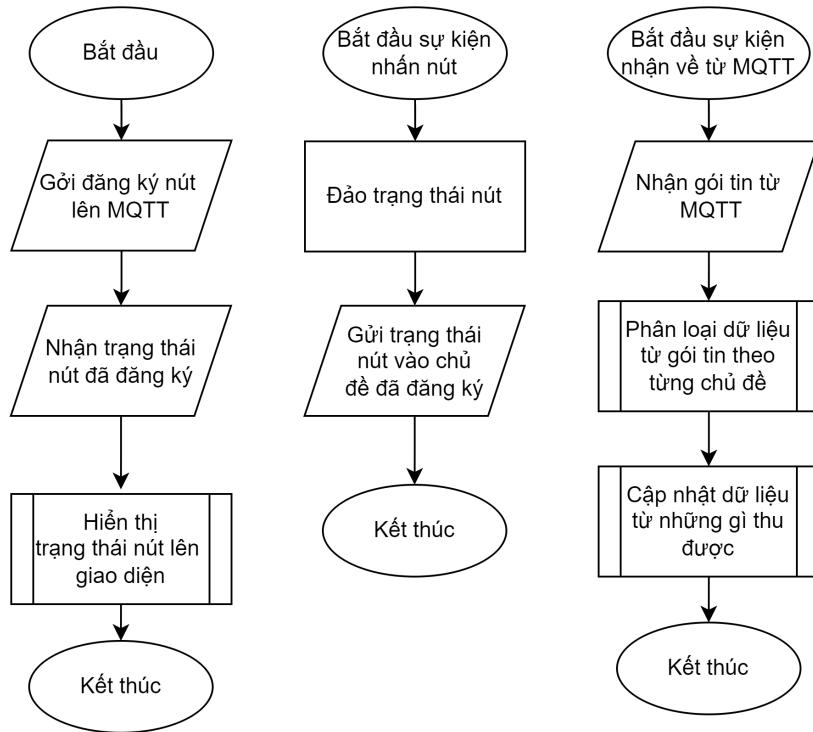


Hình 3.47: Lưu đồ hiển thị biểu đồ dữ liệu

Ngay lập tức, một giao diện biểu đồ sẽ mở ra, hiển thị các dữ liệu đo đạc của trạm theo thời gian. Giao diện biểu đồ này cho phép người dùng tùy chỉnh để quan sát dữ liệu trong các khoảng thời gian khác nhau, từ hiện tại đến quá khứ. Nhờ đó, người dùng có thể dễ dàng theo dõi và phân tích dữ liệu theo nhu cầu cụ thể của mình, giúp cải thiện hiệu quả quản lý và giám sát trạm. Quá trình này không chỉ đảm bảo rằng người dùng luôn có thông tin cập nhật và chính xác về trạm gần nhất mà còn cung cấp một công cụ mạnh mẽ để phân tích dữ liệu lịch sử, từ đó hỗ trợ các quyết định quản lý và vận hành hiệu quả hơn.

### 3.8.7 Thiết kế điều khiển thiết bị

Khi ứng dụng được khởi động, hệ thống ngay lập tức kết nối với máy chủ MQTT và đăng ký vào các chủ đề (topics) liên quan đến các nút điều khiển thiết bị. Sau khi đăng ký thành công, hệ thống bắt đầu nhận dữ liệu từ các chủ đề này, cập nhật giao diện người dùng theo trạng thái hiện tại của các nút điều khiển. Khi người dùng tương tác và nhấn vào một nút điều khiển, trạng thái của nút sẽ thay đổi. Hệ thống sẽ ghi nhận thay đổi này và gửi thông tin cập nhật lên máy chủ MQTT, đảm bảo rằng trạng thái mới của nút được phản ánh trên toàn hệ thống và mọi thiết bị liên quan. Quy trình này giúp đảm bảo rằng mọi thao tác của người dùng đều được cập nhật tức thì, đồng thời giữ cho giao diện người dùng luôn hiển thị chính xác trạng thái hiện tại của các thiết bị, mang lại trải nghiệm mượt mà và nhất quán. Tất cả các quá trình được diễn tả qua ở các lưu đồ sau:



Hình 3.48: Lưu đồ điều khiển thiết bị

### 3.8.8 Triển khai Singleton quản lý mã nguồn

Mẫu thiết kế (Pattern) là các giải pháp tổng quát, có thể tái sử dụng cho các vấn đề thường gặp trong thiết kế phần mềm. Các mẫu thiết kế này mô tả cách sắp xếp và tổ chức các đối tượng, lớp và cách chúng tương tác với nhau để giải quyết một vấn đề cụ thể.

Nhóm sử dụng mẫu thiết kế Singleton nhằm quản lý tài nguyên hệ thống. Singleton

tạo ra một đối tượng duy nhất từ một lớp để tránh khởi tạo nhiều đối tượng, tiết kiệm bộ nhớ và tăng hiệu năng cho chương trình. Tạo ra một đối tượng chứa các thông tin được chia sẻ, cho phép dễ dàng truy cập từ bất kỳ lớp và hàm nào của chương trình.

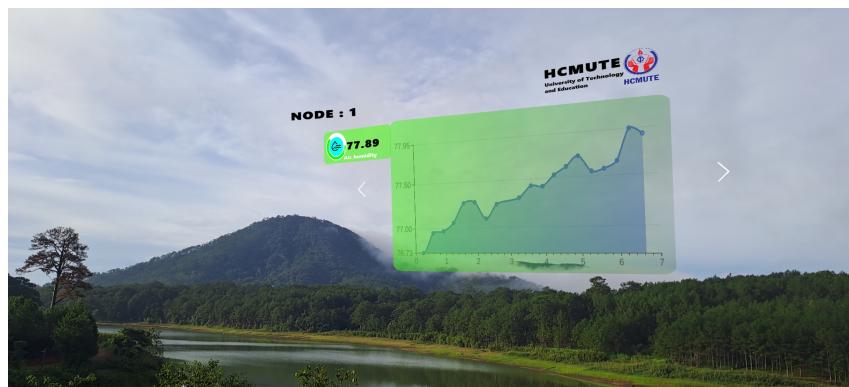
### 3.9 Thiết kế giao diện quản lý

Trong đề tài này nhóm sử dụng Unity để tạo nên các giao diện người dùng với mục đích chính là quan sát dữ liệu.



Hình 3.49: Giao diện hiển thị dữ liệu trạm tức thời.

Tại màn hình 3.49, chương trình sẽ hiển thị các thông tin như: Trạm ở gần nhất được chương trình phát hiện, các thông số của trạm tại thời điểm mà trạm được chương trình phát hiện. Các thông số bao gồm: Nhiệt độ không khí, độ ẩm không khí, dòng xả của bin, điện áp của bin, điện áp năng lượng mặt trời, dòng sạc của năng lượng mặt trời, mức CO<sub>2</sub> có trong không khí, cường độ ánh sáng, độ ẩm đất, nhiệt độ đất.



Hình 3.50: Giao diện hiển thị lịch sử dữ liệu.

Sau khi người dùng nhấn vào biểu tượng tương ứng với các thông số mà người dùng muốn xem như hình 3.50. Giao diện biểu đồ được thiết kế để hiển thị dữ liệu cảm biến một cách rõ ràng và trực quan. Các thông số được biểu diễn dưới dạng biểu đồ số liệu với hai trục chính: Trục X Đại diện cho thời gian, cho phép người dùng theo dõi các

thay đổi và xu hướng của dữ liệu cảm biến theo từng khoảnh khắc, trục Y Đại diện cho giá trị của cảm biến tại các thời điểm cụ thể, giúp người dùng dễ dàng quan sát mức độ biến đổi của các thông số môi trường như nhiệt độ, độ ẩm, mức CO<sub>2</sub> và nhiều thông số khác. Người dùng có thể tương tác với biểu đồ sử dụng các phím điều hướng của chương trình để di chuyển đồ thị, cho phép người dùng dễ dàng di chuyển qua các khoảng thời gian khác nhau và tập trung vào những khoảng thời gian cụ thể mà họ quan tâm.

Trên giao diện được hiển thị ở hình 3.51, người dùng được cung cấp một công cụ để quản lý chức năng phát hiện trạm gần nhất. Các chức năng và thông tin hiển thị được thiết kế nhằm tối ưu hóa trải nghiệm người dùng, giúp họ dễ dàng thao tác và theo dõi các trạm. Người dùng có thể điều chỉnh bán kính phát hiện trạm gần nhất thông qua các phím chức năng: " > ": Mỗi lần nhấn phím này sẽ tăng chu vi phát hiện thêm một đơn vị, cho phép mở rộng phạm vi tìm kiếm từng bước nhỏ, " » ": Mỗi lần nhấn phím này sẽ tăng bán kính phát hiện thêm năm đơn vị, giúp nhanh chóng mở rộng phạm vi tìm kiếm khi cần thiết và ngược lại. Các phím chức năng này giúp người dùng linh hoạt trong việc điều chỉnh phạm vi phát hiện tùy theo nhu cầu cụ thể, từ việc tìm kiếm trong phạm vi gần đến việc mở rộng tìm kiếm ra xa hơn.

Giao diện cung cấp một bảng danh sách các trạm có thể được lưu trữ trên máy chủ. Bảng danh sách này không chỉ liệt kê các trạm mà còn cung cấp thông tin về trạng thái lưu tọa độ của mỗi trạm. Các Trạm đã lưu tọa độ sẽ hiển thị màu xanh lá, nghĩa là tọa độ của chúng đã được lưu thành công trên hệ thống. Giao diện được hiển thị như hình:



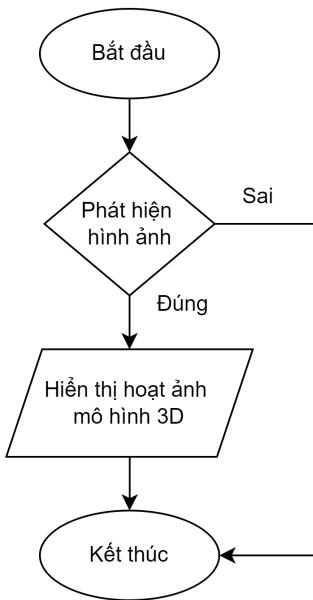
Hình 3.51: Giao diện thao tác lưu, xoá, điều khiển, phát hiện trạm, chuyển chế độ

Khi người dùng chọn những trạm xanh lá hệ thống sẽ tự động hiểu là người dùng muốn xoá trạm đó. Còn khi người dùng ấn vào các nút màu xanh dương thì hệ thống sẽ hiểu là người dùng đang muốn đánh dấu một trạm mới tại vị trí đó.

Giao diện tại hình 3.51 không chỉ đơn thuần là một màn hình hiển thị, thêm và xoá các trạm, mà còn tích hợp nhiều chức năng quan trọng khác trong chương trình. Một là khả năng điều khiển hệ thống bơm nước và ánh sáng, giúp người dùng dễ dàng quản lý và điều chỉnh các thiết bị này theo nhu cầu thực tế. Bên cạnh đó, giao diện còn cung cấp

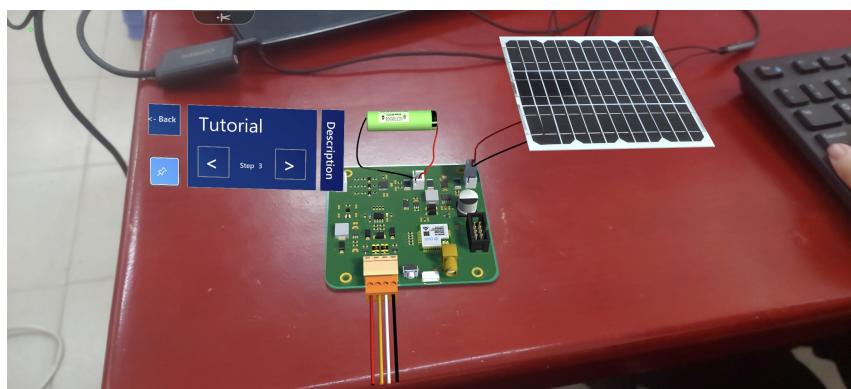
một nút nhấn tiện lợi cho phép người dùng chuyển sang chế độ hiển thị hướng dẫn. Tính năng này giúp người dùng nhanh chóng truy cập vào các hướng dẫn cần thiết, tối ưu hóa trải nghiệm sử dụng và đảm bảo rằng mọi thao tác đều được thực hiện một cách chính xác và hiệu quả.

### 3.10 Thiết kế chức năng hiển thị hướng dẫn



Hình 3.52: Lưu đồ sử lý sự kiện hướng dẫn trong một vòng lặp

Trong lưu đồ trên, hệ thống sẽ sử dụng Vuforia để kiểm tra xem đã phát hiện đối tượng ảnh hoặc vật thể chưa. Khi quá trình kiểm tra bắt đầu, hệ thống sẽ liên tục quét và nhận diện các đối tượng. Nếu phát hiện, hệ thống sẽ tiến hành hiển thị hoạt ảnh hướng dẫn lặp đặt, lặp đi lặp lại để người dùng có thể dễ dàng theo dõi. Nếu không phát hiện đối tượng nào, quá trình sẽ kết thúc mà không hiển thị bất kỳ nội dung nào.



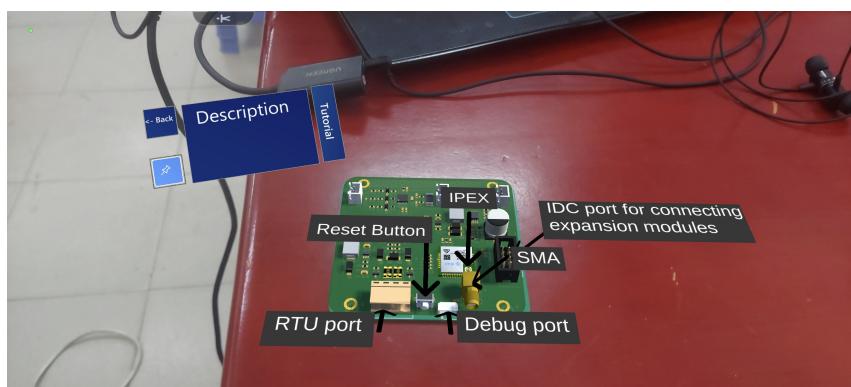
Hình 3.53: Hoạt ảnh hiển thị trực tiếp trên vật thể.

Trong hình 3.53, có thể thấy hoạt ảnh đang được hiển thị trực tiếp trên một vật thể thực tế. Điều này mang lại rất nhiều lợi ích trong quá trình hướng dẫn lắp đặt. Hoạt ảnh trực quan giúp người dùng dễ dàng nắm bắt các bước lắp đặt một cách rõ ràng và chi tiết hơn. Thay vì phải đọc các hướng dẫn văn bản phức tạp và dễ gây nhầm lẫn, người dùng có thể nhìn thấy từng bước cụ thể được minh họa ngay trên chính vật thể mà họ đang làm việc. Phương pháp này không chỉ tiết kiệm thời gian mà còn giảm thiểu sai sót, nâng cao hiệu quả và chất lượng công việc.



Hình 3.54: Giao diện hướng dẫn lắp đặt

Ứng dụng này còn tích hợp một tính năng đặc biệt cho phép người dùng hiển thị và quản lý các cổng kết nối mở rộng cho hệ thống, người dùng chỉ cần thay đổi chế độ từ "Hướng dẫn" ở hình 3.54 sang "Mô tả" ở hình 3.55 để kích hoạt chức năng này. Các cổng này dễ dàng thêm, tháo hoặc cấu hình theo nhu cầu sử dụng. Chức năng này mang lại sự tiện lợi cũng như tăng cường khả năng tùy biến của hệ thống, đảm bảo ứng dụng có thể đáp ứng tốt hơn các yêu cầu đa dạng của người dùng.



Hình 3.55: Giao diện miêu tả các cổng kết nối.

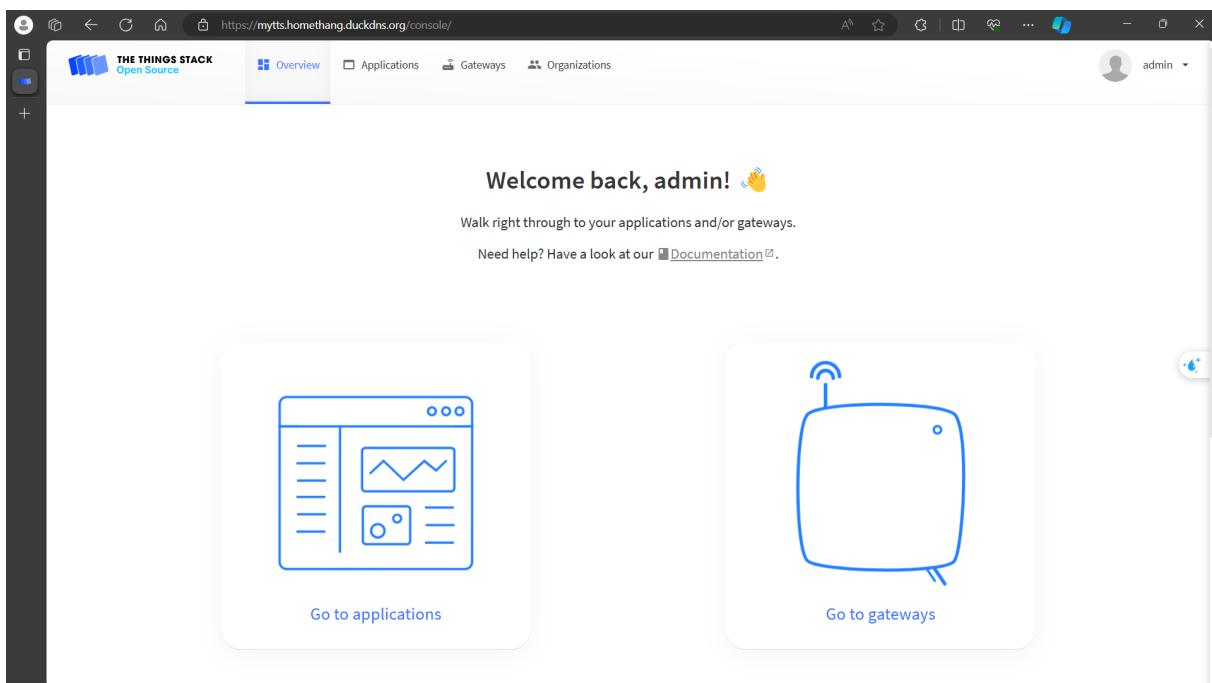
# Chương 4

## KẾT QUẢ

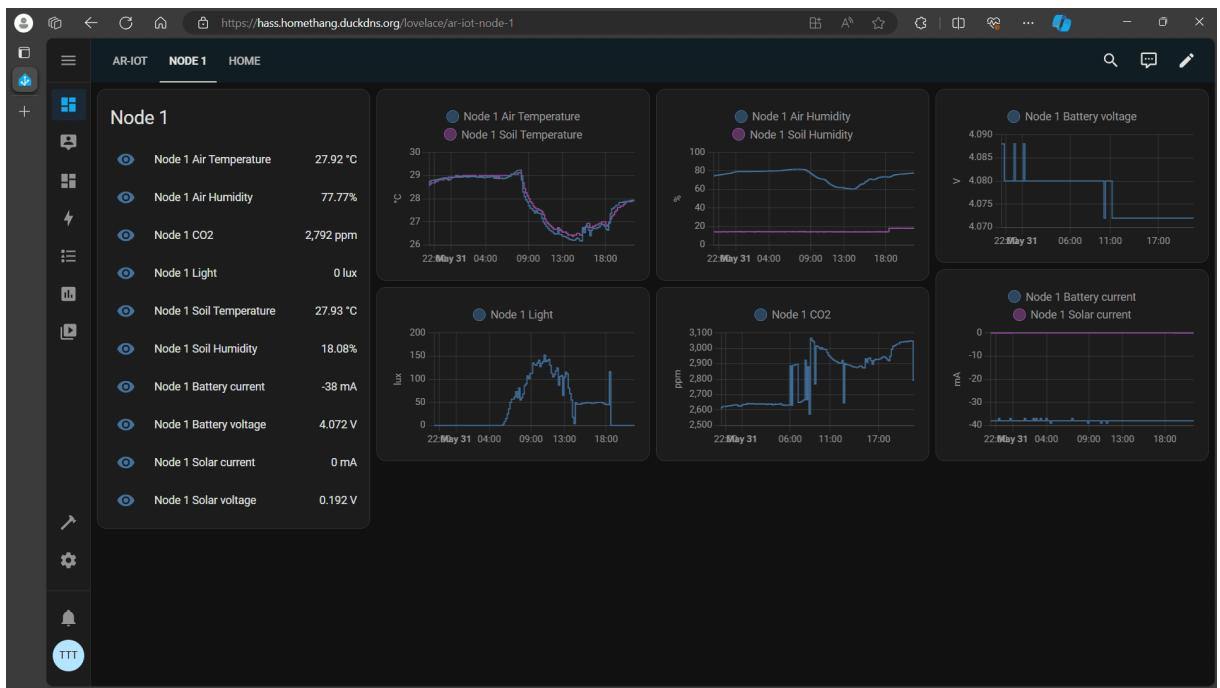
### 4.1 Kết quả triển khai hạ tầng IoT

#### 4.1.1 Kết quả xây dựng máy chủ đám mây và trạm trung gian

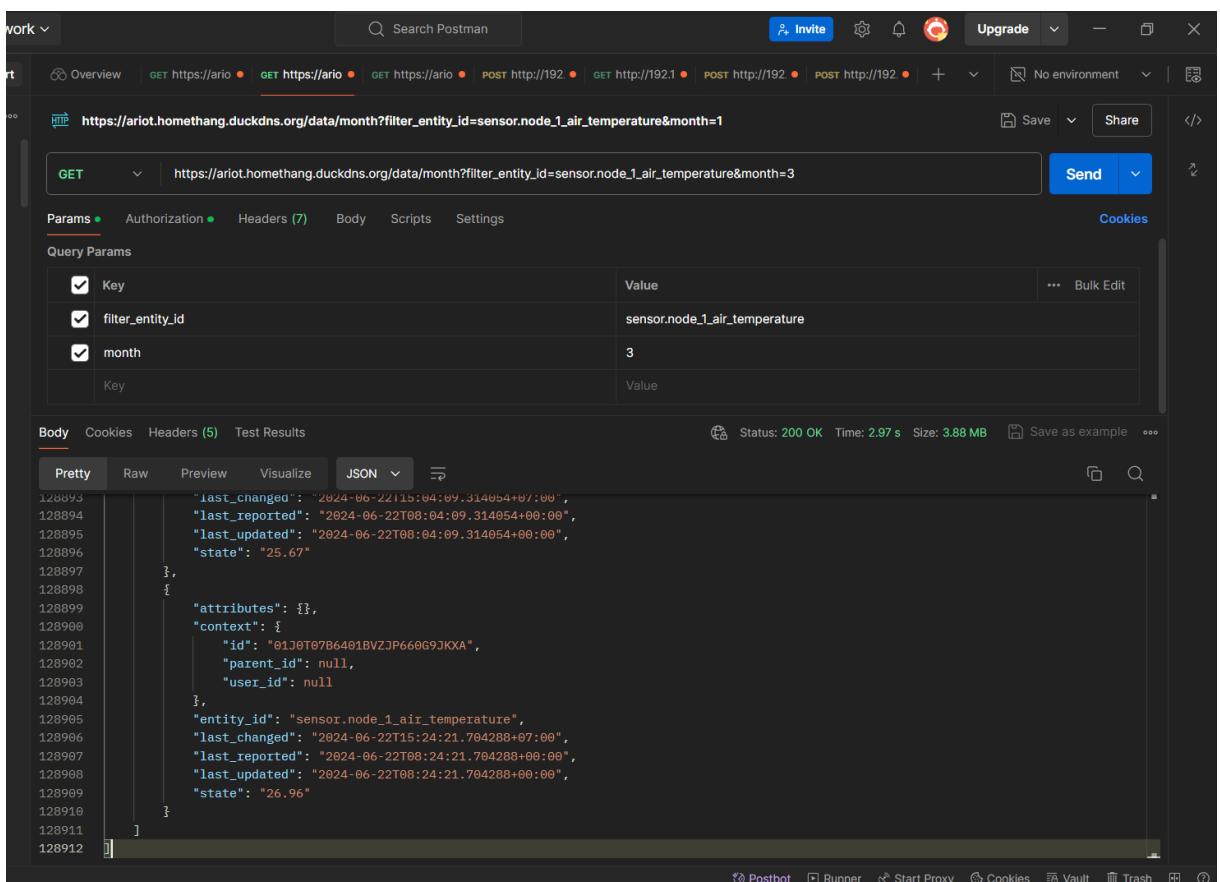
Sau khi hoàn tất quá trình cài đặt và cấu hình, máy chủ đã đi vào hoạt động ổn định. Toàn bộ chức năng của các ứng dụng trên máy chủ đã được kiểm tra kỹ lưỡng trong mạng nội bộ để đảm bảo chúng hoạt động đúng như mong đợi. Các bước kiểm tra bao gồm việc xác minh tất cả các dịch vụ chạy chính xác, không có lỗi phát sinh và các ứng dụng có thể giao tiếp với nhau một cách mượt mà. Khi cấu hình tầng mạng hoàn tất, các dịch vụ và giao thức cần thiết của máy chủ đã có thể truy cập từ các vùng khác ngoài mạng nội bộ. Điều này bao gồm việc mở các cổng mạng, cấu hình firewall và thiết lập các quy tắc truy cập để đảm bảo bảo mật. Trong trường hợp máy chủ mất nguồn đột ngột và có nguồn lại, các ứng dụng sẽ tự khởi động và hoạt động bình thường. Hình 4.1 và 4.2 lần lượt là giao diện web của máy chủ mạng The Things Network và Home Assistant khi truy cập qua tên miền của máy chủ đám mây:



Hình 4.1: Hình ảnh trang web The Things Network của máy chủ đám mây khi được truy cập bằng tên miền



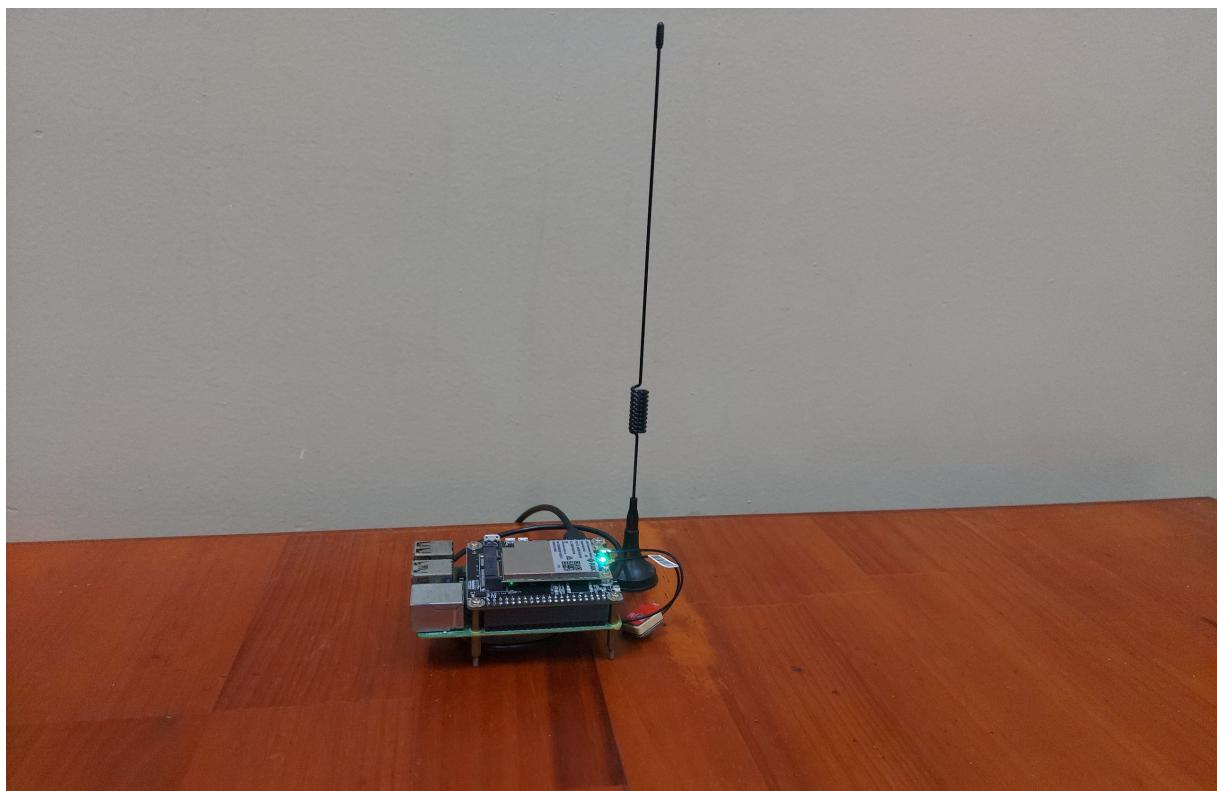
Hình 4.2: Hình ảnh trang web Home Assistant của máy chủ đám mây khi được truy cập bằng tên miền



Hình 4.3: Hình ảnh kiểm tra khả năng truy xuất dữ liệu cảm biến bằng Postman

Khi kiểm thử khả năng truy xuất dữ liệu lịch sử của cảm biến từ máy chủ, nhóm đã thực hiện lấy dữ liệu qua REST API do ứng dụng ARIOT trên máy chủ cung cấp, dữ liệu được lấy là nhiệt độ không khí của trạm số 1 trong thời gian 3 tháng gần nhất. Kết quả là đã lấy được một gói tin dữ liệu khoảng 3.88 MB trong 2.97 giây kể từ lúc yêu cầu (hình 4.3), gói tin là một danh sách cách json chứa khoảng 10000 mẫu dữ liệu nhiệt độ không khí của trạm số 1 đã được ghi lại trong suốt quá trình chạy thử (chu kì lấy mẫu trong khoảng từ 1-15 phút). Điều này cho thấy máy chủ có khả năng đáp ứng các yêu cầu truy xuất dữ liệu từ các ứng dụng bên ngoài (điển hình là ứng dụng MR của nhóm) để trực quan hóa dữ liệu hoặc phục vụ cho các ứng dụng phân tích dữ liệu, ứng dụng trí tuệ nhân tạo,...với tốc độ được đảm bảo.

Trạm trung gian cũng đã cấu hình hoàn tất và hoạt động ổn định, sau đây là ảnh thực tế của trạm trung gian:



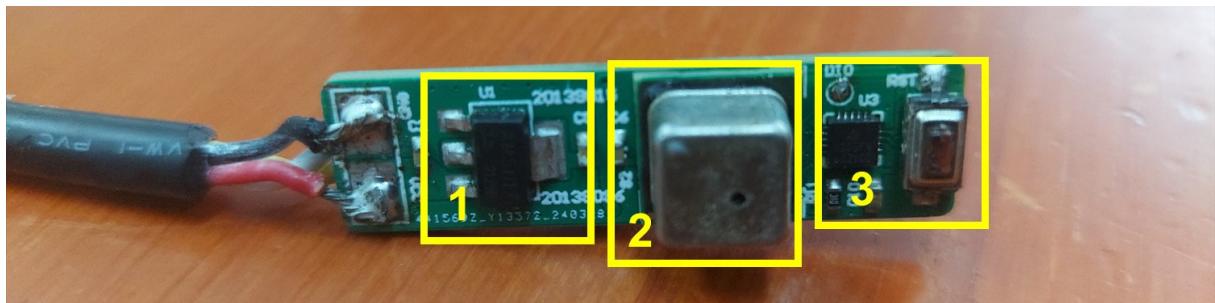
Hình 4.4: Hình ảnh thực tế của trạm trung gian

#### 4.1.2 Kết quả phát triển cảm biến RS485 Modbus RTU

Sau khi hoàn thành thiết kế phần cứng và nạp mã nguồn, cả hai cảm biến Modbus RTU đã được kiểm thử kỹ lưỡng bằng cách kết nối với nhiều loại bộ giao tiếp RS485 khác nhau, bao gồm các mạch tích hợp, các mô đun chuyển đổi RS485 sang UART và USB. Các chức năng của cảm biến được kiểm tra cả bằng vi điều khiển và máy tính. Kết quả kiểm thử cho thấy các chức năng hoạt động ổn định. Khi cảm biến được kết nối với

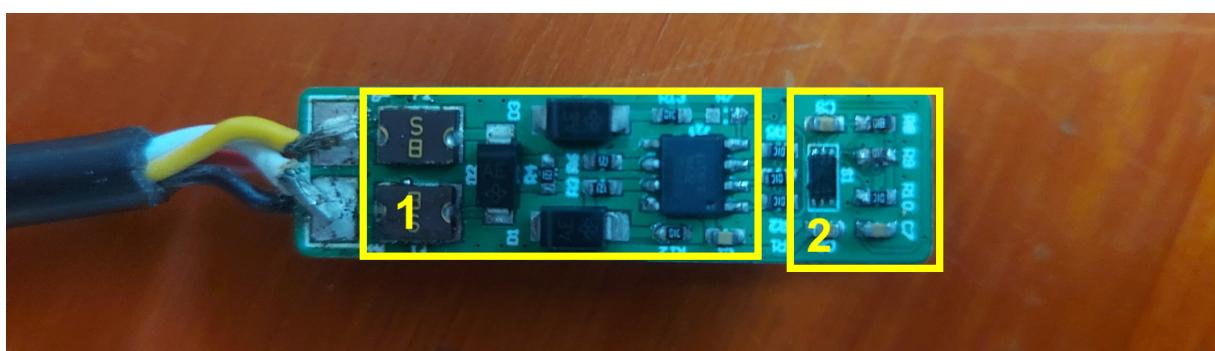
trạm thu thập dữ liệu, chúng hoạt động bình thường mà không gặp bất kỳ lỗi nào. Sau đây là một số hình ảnh phần cứng thực tế sau khi đã thiết kế:

- Mặt trên của cảm biến không khí được thiết kế bao gồm khối nguồn (1), cảm biến SCD40 (2) và khối điều khiển CH32V003 (3).



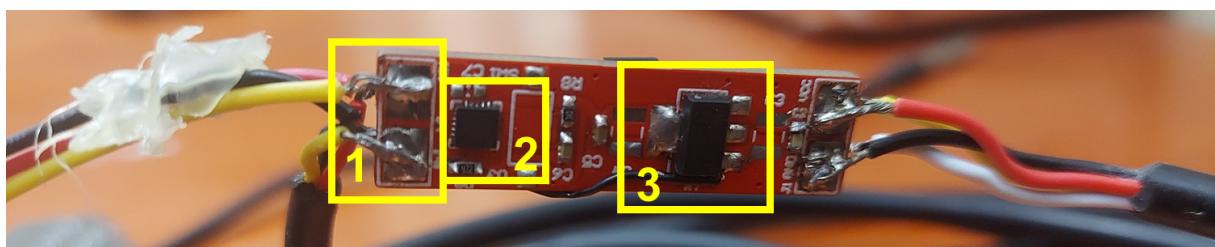
Hình 4.5: Hình ảnh thực tế mặt trên của cảm biến không khí

- Mặt dưới của cảm biến không khí bao gồm khối RS485 (1) và cảm biến ánh sáng BH1750 (2).



Hình 4.6: Hình ảnh thực tế mặt dưới của cảm biến không khí

- Mặt trên của cảm biến đất bao gồm chân để hàn cảm biến DS1820B và cảm biến độ ẩm đất (1), vi điều khiển CH32V003 (2), khối nguồn (3).



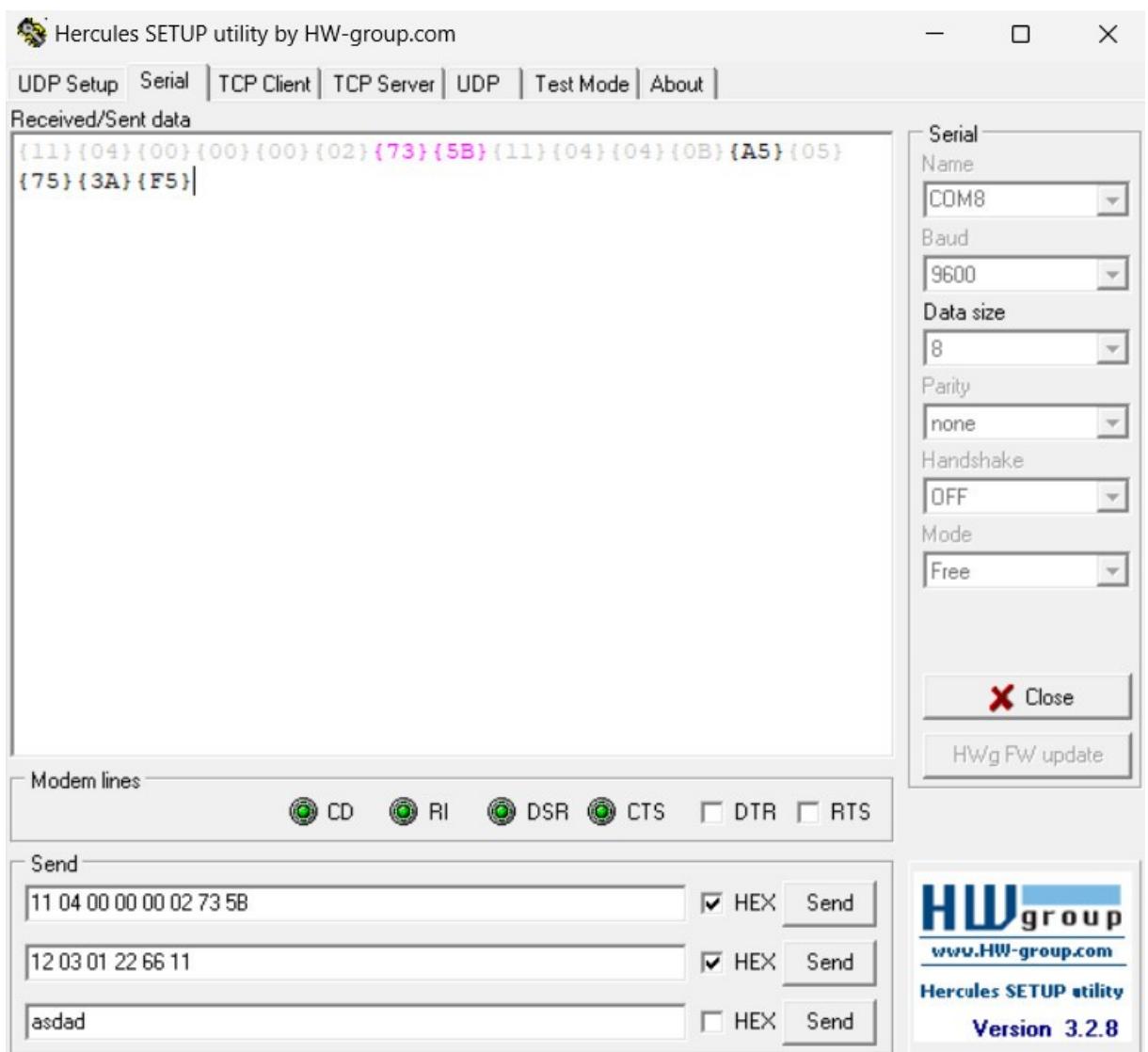
Hình 4.7: Hình ảnh thực tế mặt trên của cảm biến đất

- Mặt dưới của cảm biến đất chỉ có khía cắm RS485 được dàn trải đều trên mặt.



Hình 4.8: Hình ảnh thực tế mặt dưới của cảm biến đất

Để kiểm thử khả năng kết nối với các thiết bị khác, nhóm sử dụng bộ chuyển đổi USB sang RS485 để kết nối cảm biến với máy tính và dùng phần mềm Hercules để giao tiếp với cảm biến như hình 4.9. Chuỗi hex "11 04 00 00 00 02 73 5B" là chuỗi lệnh đọc dữ liệu nhiệt độ và độ ẩm của cảm biến đất, chuỗi hex "11 04 04 0B A5 05 75 3A F5" là chuỗi trả về từ cảm biến có dữ liệu nhiệt độ là 0x0BA5, tương đương số 2931, quy ra nhiệt độ là 29,31 °C. Tương tự với độ ẩm là 0x0575, tương đương 13,97%.



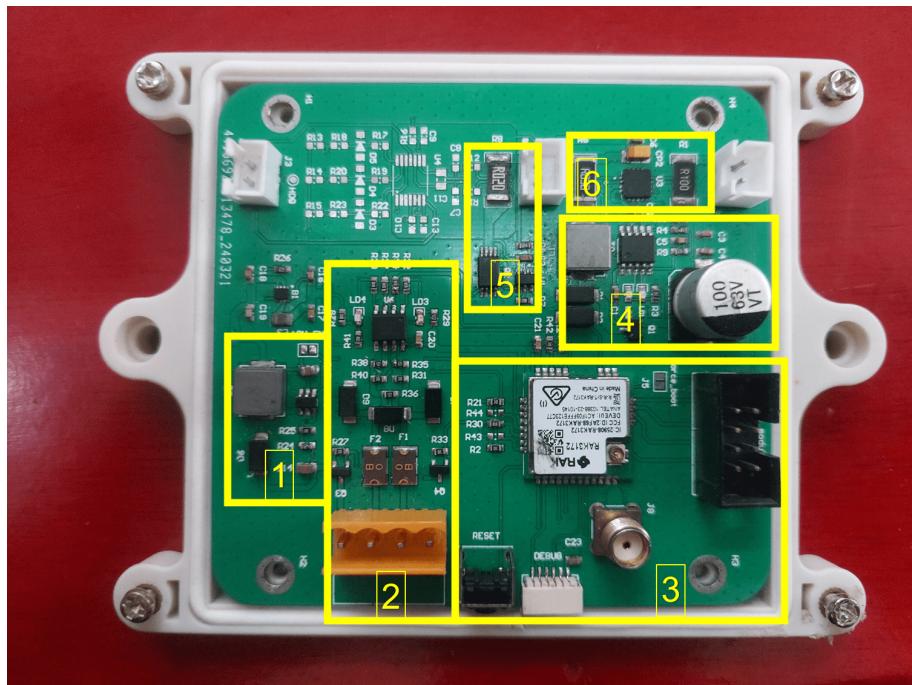
Hình 4.9: Hình ảnh giao tiếp giữa máy tính và cảm biến bằng phần mềm Hercules

#### 4.1.3 Kết quả phát triển trạm thu thập dữ liệu

Trạm thu thập dữ liệu sau khi đã hoàn thiện đã hoạt động ổn định, toàn bộ các khối được đề ra trước đó đều được kiểm tra và hoạt động bình thường. Sau đây là hình ảnh thực tế của phần cứng của trạm:



Hình 4.10: Ảnh thực tế của trạm sau khi đã hoàn thiện



Hình 4.11: Ảnh khoanh vùng các khối phần cứng của trạm thu thập

Theo hình 4.11 các khối lần lượt được đánh dấu là:

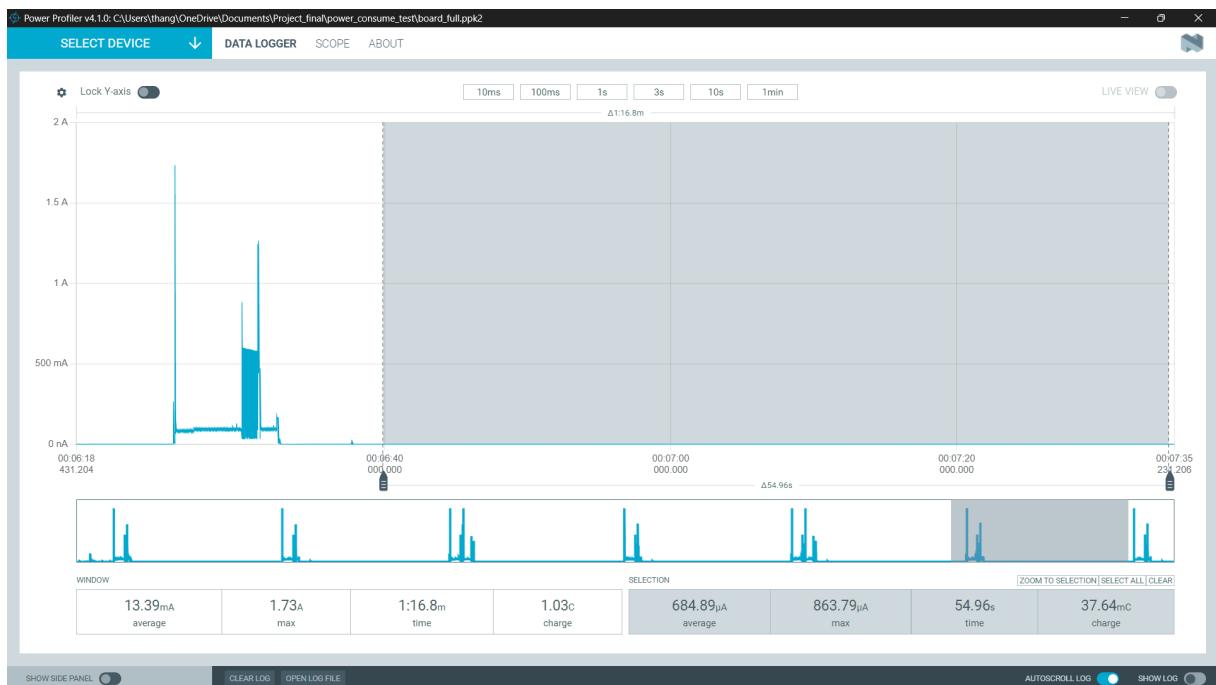
- 1: Khối tăng áp IC MT3608 để cấp nguồn cho cảm biến.
- 2: Khối kết nối RS485 bao gồm IC SP3485 và 2 mosfet cấp nguồn cho cảm biến.
- 3: Khối điều khiển chính RAK3172, cổng IDC cho module mở rộng, cổng anten SMA, nút khởi động lại và cổng gõ lõi.
- 4: Khối sạc năng lượng mặt trời IC CN3791.
- 5: Khối bảo vệ pin.
- 6: Khối giám sát năng lượng pin và năng lượng mặt trời.

Cổng RS485 đã được thử kết nối với hai cảm biến có sẵn trên thị trường như cảm biến SHT20 Modbus RTU và PTA9B01, sau đó kết nối thêm các cảm biến tự thiết kế ở trên, kết quả tất cả đều hoạt động. Bộ sạc năng lượng mặt trời có thể sạc pin với công suất tối đa ghi nhận được là 1 W khi điện áp tấm pin năng lượng mặt trời tối đa là 8 V theo như hình 4.10, trong điều kiện điệp áp pin luôn trên 4V, thời tiết không có mưa, trạm được đặt trên cao tiếp xúc trực tiếp với ánh nắng.



Hình 4.12: Biểu đồ ghi nhận dòng sạc năng lượng mặt trời và điện áp pin thực tế của trạm thu thập dữ liệu

Năng lượng tiêu thụ cũng là một vấn đề cần kiểm tra của các loại thiết bị dùng pin, vì thế nhóm thực hiện đã dùng bộ Power Profiler Kit II và phần mềm Power Profiler của hãng Nordic Semiconductor để kiểm tra dòng xả của pin khi trạm hoạt động với điều kiện không sạc trong hai trường hợp cấp nguồn cho cảm biến nhóm tự thiết kế là 5V và 12V, thời gian ngủ của trạm là 1 phút.

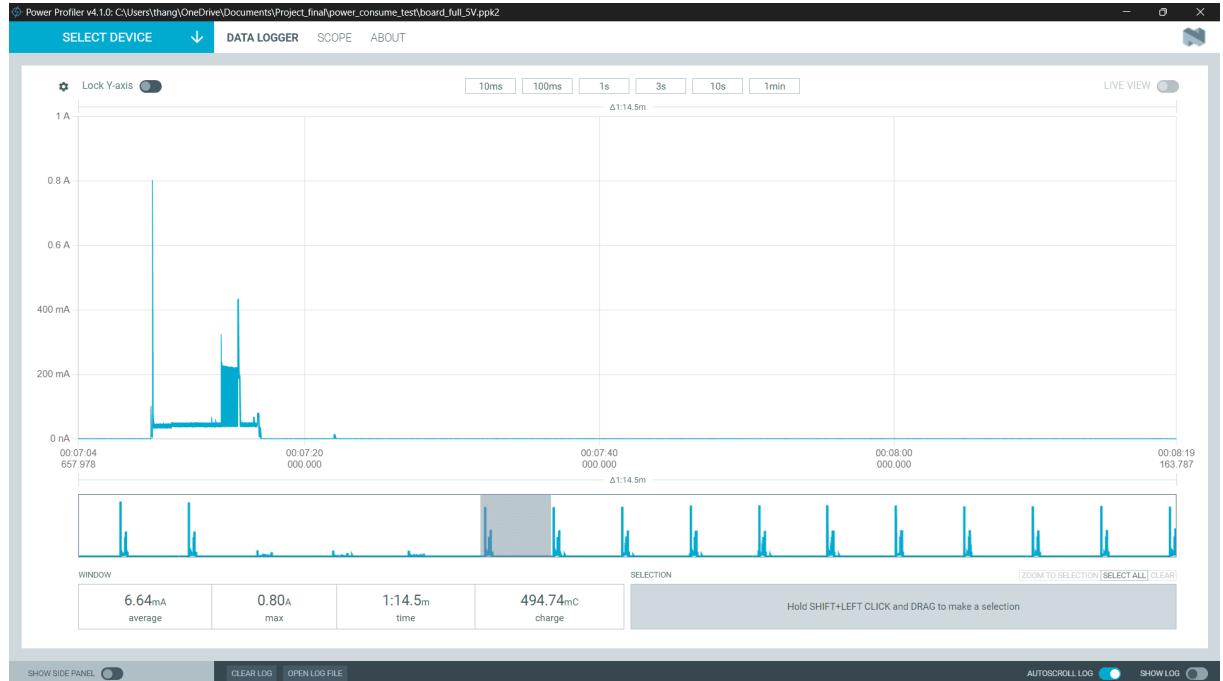


Hình 4.13: Kết quả đo dòng tiêu thụ của trạm khi cấp nguồn 12V cho các cảm biến

Theo hình 4.13, trong trường hợp áp cấp là 12V, dòng tiêu tụ trung bình cho mỗi chu kì hoạt động (điều kiện ngủ khoảng 60 giây) của trạm khoảng 14 mA (độ dài của mỗi chu kì khoảng 72 giây), dòng xả đỉnh lên đến 1,8 A khi bật nguồn cho cảm biến RS485, dòng tiêu thụ khi trạm ở trạng thái ngủ khoảng 700  $\mu$ A, dòng trung bình của trạm khi thức là 113 mA (kéo dài khoảng 8 giây). Với các thông số này, nhóm ước tính dòng tiêu thụ trung bình trong một chu kì hoạt động của trạm khi triển khai thực tế (trong điều kiện thời gian ngủ khoảng 10 phút và điện áp cấp là 12V) là khoảng 2 mA, trong trường hợp pin cấp nguồn cho trạm có dung lượng khoảng 1000 mAh thì trạm có thể hoạt động liên tục không cần sạc ít nhất 20 ngày.

Hình 4.14 là số liệu đo dòng tiêu thụ khi điện áp cấp cho cảm biến là 5V, dòng tiêu tụ trung bình cho mỗi chu kì hoạt động (điều kiện ngủ khoảng 60 giây) của trạm là khoảng 7 mA (độ dài của mỗi chu kì khoảng 72 giây), dòng xả đỉnh khoảng 0,8 A, dòng tiêu thụ khi trạm ở trạng thái ngủ cũng khoảng 700  $\mu$ A, dòng trung bình của trạm khi thức là 55 mA (kéo dài khoảng 8 giây). Dựa vào các số liệu này, nhóm ước tính dòng tiêu thụ trung bình trong một chu kì hoạt động của trạm khi triển khai thực tế (cũng trong điều kiện thời gian ngủ khoảng 10 phút và điện áp chỉ 5V) là khoảng 1,5 mA,

trong trường hợp pin cấp nguồn cho trạm có dung lượng khoảng 1000 mAh thì trạm có thể hoạt động liên tục không cần sạc ít nhất 30 ngày.



Hình 4.14: Kết quả đo dòng tiêu thụ của trạm khi cấp nguồn 5V cho các cảm biến

#### 4.1.4 Kết quả triển khai trạm điều khiển



Hình 4.15: Hình ảnh thực tế của trạm điều khiển

Trạm điều khiển sau khi kiểm thử đã hoạt động khá tốt, các chức năng điều khiển và đồng bộ trạng thái đều hoạt động, người dùng có thể điều khiển thiết bị qua web, ứng dụng MR và thậm chí là bằng giọng nói qua Google Assistant. Tuy nhiên, nhóm nhận thấy trạm điều khiển này vẫn gặp vấn đề về độ ổn định khi điều khiển. Cụ thể là sau một khoảng thời gian dài chạy thử, điều chỉnh, trạm vẫn gặp một vấn đề là đôi lúc có độ trễ cao khi nhận gói tin downlink điều khiển từ trạm trung gian, các trường hợp bị trễ này sẽ xuất hiện khi hệ thống vừa khởi động hoặc vô tình thực hiện hành động điều khiển một cách liên tục. Phần lớn thời gian còn lại, trạm điều khiển hoạt động rất ổn định trong điều kiện không thực hiện điều khiển quá nhiều.

## 4.2 Đánh giá ứng dụng MR

Ứng dụng MR trong mô hình hoạt động khi được chạy mô phỏng trên máy tính đã đo được các số liệu chi tiết ở bảng 4.1.

Bảng 4.1: Bảng thông tin hiệu năng khi giám sát và điều khiển.

<b>Thông số</b>	<b>Chi tiết</b>
<b>Batches</b>	79
<b>Saving by batching</b>	8
<b>Triangle Count</b>	620 - 1400
<b>Verts</b>	2500
<b>FPS</b>	30
<b>SetPass Calls</b>	50 - 82

Theo như số liệu mà bản 4.1 cung cấp, nhóm dự đoán ứng dụng có thể được sử dụng một cách ổn định trên mọi thiết bị di động có hệ điều hành android 8.0 trở lên. Với số lượng đa giác (Triangle) và số lượng lệnh gọi vẽ (Draw call) ở mức thấp so với tiêu chuẩn của một ứng dụng trên thiết bị di động.

Với số lượng đa giác và lệnh gọi vẽ được tối ưu hóa, ứng dụng có thể duy trì tốc độ khung hình ổn định ở mức xấp xỉ 30Hz. Điều này đảm bảo rằng người dùng sẽ trải nghiệm được sự mượt mà và liền mạch trong quá trình sử dụng ứng dụng, bất kể trên thiết bị nào. Tốc độ khung hình ổn định là yếu tố quan trọng trong việc mang lại trải nghiệm người dùng tốt, đặc biệt là đối với các ứng dụng có nhiều tương tác và hiệu ứng hình ảnh phức tạp.

Giao diện chương trình được thiết kế trực quan và đầy đủ, mang lại cảm giác thoải mái cho người dùng. Các yếu tố giao diện được sắp xếp hợp lý, giúp người dùng dễ dàng tiếp cận và hiểu rõ chức năng của ứng dụng mà không gặp khó khăn.

Ngoài ra, ứng dụng còn cung cấp một không gian điều khiển tách biệt so với màn hình quan sát. Điều này có nghĩa là người dùng có thể thực hiện các thao tác điều khiển mà không bị cản trở bởi các phần tử hiển thị khác. Thiết kế này giúp tối ưu hóa trải nghiệm người dùng, đặc biệt khi họ cần thực hiện nhiều tác vụ cùng một lúc mà không bị phân tâm.

Thông qua bảng 4.2 có thể thấy rằng khi sử dụng chức năng hiển thị hoạt ảnh hướng dẫn, hệ thống vẫn duy trì số lượng đa giác ở mức không đáng kể. Điều này giúp đảm bảo hiệu năng luôn giữ ở mức ổn định, không gây quá tải cho phần cứng và vẫn mang lại trải nghiệm người dùng mượt mà ở tốc độ khung hình mức 30hz.

Bảng 4.2: Bảng thông tin hiệu năng khi dùng chức năng lấp đặt và hướng dẫn.

<b>Thông số</b>	<b>Chi tiết</b>
<b>Batches</b>	60
<b>Saving by batching</b>	0
<b>Triangle Count</b>	90800- 100200
<b>Verts</b>	234400
<b>FPS</b>	30
<b>SetPass Calls</b>	5 - 12

### 4.3 Kết quả kiểm tra trên các thiết bị.

Bảng 4.3: Kết quả kiểm tra trên một số thiết bị thực tế

<b>Stt</b>	<b>Thiết bị</b>	<b>Android</b>	<b>FPS của ứng dụng</b>	<b>Hoạt động</b>
1	Xiaomi Redmi Note 9	12	30fps	Có thể sử dụng ứng dụng, nhưng màu bị sai.
2	Google pixel 5A	13	30fp.	Hình ảnh hiển thị đẹp ứng dụng sử dụng tốt.
3	Xiaomi Mi 11	11	0fps	Không sử dụng được ứng dụng.
4	Samsung Galaxy A31	11	30fps	Có thể sử dụng ứng dụng .
5	Samsung S20 ULTRA	11	30fps	Có thể sử dụng ứng dụng.
6	Samsung Galaxy A31	12	30fps	Có thể sử dụng ứng dụng.

Thông qua bảng kết quả 4.3 thì nhóm thấy rằng thực tế có khá nhiều thiết bị di động có thể chạy ứng dụng MR này qua plugin (ứng dụng bổ trợ) ARCore một cách hiệu quả. Bên cạnh đó cũng có một số thiết bị gặp vấn đề không tương thích với ứng dụng bổ trợ này, gây ra một số lỗi như ảnh màu camera bị âm bản và các đối tượng ảo không thể di chuyển trong không gian. Đối chiếu với bài viết về các thiết bị mà Google có hỗ trợ ARCore [8], nhóm nhận thấy kết quả thực tế này tương đồng với thông tin mà họ cung cấp.

#### 4.4 Kết quả triển khai hệ thống vào thực tế

Sau khi tất cả đã hoàn thiện, nhóm thực hiện đề tài đã tìm cách triển khai toàn bộ hệ thống này ra ngoài thực tế trong một tuần. Máy chủ được đặt cố định một chỗ, trạm trung gian được đặt trong nhà, anten Lora được kéo dài lên trên nóc nhà bằng một sợi dây dài 9m, tổng chiều cao anten so với mặt đất là khoảng 8m. Trạm thu thập dữ liệu được lắp đặt vào một chiếc hộp chống nước, các cảm biến được bọc bảo vệ đầy đủ như hình 4.16, trạm sẽ được lắp đặt trên một chiếc thang cao 6m như hình 4.18. Kết quả thu được trạm hoạt động rất ổn định, không có dấu hiệu lỗi thiết bị hoặc dữ liệu, dữ liệu môi trường liên mạch như hình 4.19. Việc tiêu hao năng lượng của pin cũng đã được kiểm tra và kết quả như hình 4.17 cho thấy trạm thu thập dữ liệu số một chạy trong 6 ngày (30/5 - 4/6) thì điện áp pin giảm từ 4,09 V xuống 4,02 V, trung bình mỗi ngày giảm 0,01 V.



Hình 4.16: Hình ảnh trạm thu thập dữ liệu được chuẩn bị để chạy thực tế ngoài trời



Hình 4.17: Ảnh một phần biểu đồ điện áp pin của trạm số một trong một tuần



Hình 4.18: Hình ảnh trạm thu thập dữ liệu được lắp đặt thực tế ngoài vườn



Hình 4.19: Ảnh một phần biểu đồ nhiệt độ không khí do trạm đo được thực tế

Phần mềm MR cũng được chạy thử bằng cách dùng điện thoại đánh dấu các địa điểm thực tế, sau đó cầm điện thoại đi xung quanh các vùng đã đánh dấu. Thực tế cho thấy, các chức năng hiển thị và cấu hình hoạt động rất ổn định, kể cả việc thay đổi hiển thị thông tin khi di chuyển qua lại phạm vi các trạm cũng rất nhanh. Tuy nhiên sau nhiều lần thử, nhóm thực hiện đề tài nhận ra việc dùng GPS sẽ gặp vấn đề sai số khoảng 5-25m tùy vào môi trường xung quanh, ảnh hưởng thời tiết, công nghệ sử dụng. Thậm chí vì ứng dụng MR sử dụng khá nhiều tài nguyên gây ra hiện tượng nóng thiết bị khi sử dụng lâu, điều này cũng ảnh hưởng đến độ chính xác của GPS. Cho nên việc triển khai các trạm thu thập ở khoảng cách dưới 50 m sẽ khiến chức năng của ứng dụng MR hoạt động không ổn định.Thêm nhiều lần thử nữa được triển khai và nhóm thực hiện đề tài nhận thấy rằng nếu các trạm đặt cách xa khoảng 80m trở lên thì ứng dụng MR có thể hoạt động ổn định, vẫn có sai số nhưng ảnh hưởng không đáng kể đến trải nghiệm của người dùng.

# Chương 5

## KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1 Kết luận

Trong quá trình nghiên cứu, nhóm thực hiện đề tài đã tích lũy được một số kinh nghiệm trong việc xây dựng một hạ tầng IoT cũng như phát triển ứng dụng MR. Sau khi đánh giá lại sơ bộ kết quả, nhóm đã đạt được những yêu cầu sau:

- Cảm biến đã đáp ứng đầy đủ các tiêu chí ban đầu đặt ra, có thể kết nối với thiết bị RS485 khác để sử dụng.
- Trạm thu thập dữ liệu đã được thiết kế thành công, việc đọc dữ liệu cảm biến đã hoạt động ổn định, năng lượng tiêu thụ cũng được kiểm soát đủ tốt để ứng phó với các điều kiện không thể sạc bằng năng lượng mặt trời trong khoảng thời gian dài.
- Trạm điều khiển cũng đã hoạt động và có thể kiểm soát hoạt động của các thiết bị điện. Tuy nhiên vẫn còn vấn đề về độ ổn định.
- Máy chủ đám mây hoạt động rất ổn định, đáp ứng kịp thời, không có lỗi phát sinh trong quá trình sử dụng.
- Ứng dụng MR đã có thể kết nối với hạ tầng IoT và hoạt động một cách ổn định trên điện thoại, các chức năng hoạt động tốt và đáp ứng đủ nhanh. Ứng dụng đã mang lại một trải nghiệm hoàn toàn mới mẻ.

Bên cạnh các thành tựu đạt được ở sản phẩm, nhóm thực hiện đề tài cũng nhận ra một vài vấn đề cần cải thiện:

- Các cảm biến được thiết kế đủ chức năng nhưng chưa được tối ưu về năng lượng sử dụng và độ chính xác của các giá trị đo được.
- Trạm thu thập dữ liệu vẫn cần điều chỉnh phần cứng để tối ưu năng lượng khi ngủ.
- Ứng dụng MR hoạt động còn một số tình trạng giật lag nhẹ khi di chuyển nhanh. Ngoài ra giao diện thiết kế trong một số trường hợp không thật sự đem lại cảm giác thoải mái cho người sử dụng.
- Ứng dụng MR sử dụng camera cho các hoạt động của nó nên tiêu tốn nhiều năng lượng. Từ đó khiến cho ứng dụng không thể sử dụng trong thời gian quá dài.

- Giải pháp GPS không thể đáp ứng cho các hệ thống trạm được triển khai với mật độ cao, các môi trường có quy mô nhỏ hoặc kín như phòng nghiên cứu.

## 5.2 Hướng phát triển

Để hoàn thiện và phát triển hơn nữa đề tài này thì nhóm đã đề ra những hướng đi trong tương lai như sau:

- Nhóm thực hiện đề tài đánh giá rằng vi điều khiển CH32V003 là một vi điều khiển rất thích hợp cho các ứng dụng đặc biệt. Vì thế sẽ tiếp tục nghiên cứu ứng dụng CH32V003 vào các cảm biến cũng như ứng dụng khác.
- Cải thiện cả hai cảm biến mà nhóm đã thiết kế, tối ưu năng lượng sử dụng và nâng cao độ chính xác.
- Nghiên cứu lược bỏ bớt các linh kiện thụ động không cần thiết trên trạm thu thập dữ liệu, điều chỉnh các khồi và nâng cấp các IC để tối ưu năng lượng tiêu hao.
- Nâng cấp toàn bộ hạ tầng IoT, phát triển hướng đến mục tiêu dễ dàng triển khai, quản lý và mở rộng hệ thống. Điển hình là nâng cấp mã nguồn của trạm thu thập dữ liệu, cho phép đọc dữ liệu của bất kỳ cảm biến nào được kết nối. Phát triển ứng dụng có thể thay thế HASS, mở rộng thêm chức năng quản lý các trạm thu thập dữ liệu, lưu trữ thông tin cách đọc dữ liệu của cảm biến Modbus RTU, điều khiển các trạm đọc đúng cảm biến tương ứng của chúng.
- Cải tiến giao diện người dùng của ứng dụng MR để tạo ra trải nghiệm sử dụng mượt mà, thân thiện và trực quan hơn. Bao gồm việc sử dụng thiết kế phản hồi nhanh, bố cục hợp lý và các yếu tố đồ họa hiện đại để người dùng có thể dễ dàng thao tác và nắm bắt thông tin. Tối ưu hóa các chức năng điều khiển để người dùng có thể dễ dàng truy cập các tính năng chính chỉ với giọng nói.
- Kết nối và tích hợp các dịch vụ thời tiết và dữ liệu vệ tinh để cung cấp thông tin chi tiết và cập nhật liên tục về điều kiện môi trường xung quanh. Điều này giúp người dùng có thể dự báo và lên kế hoạch hoạt động nông nghiệp hiệu quả hơn.
- Tăng cường bảo mật và an toàn thông tin, đảm bảo người dùng không bị rủi ro bảo mật khi sử dụng ứng dụng. Thêm chức năng đăng nhập vào tài khoản người dùng, hiện tại sản phẩm không có chức năng này là vì nhóm không tiềm được giải pháp đăng nhập nào cho người dùng thoải mái nhất.
- Phát triển và tối ưu hóa ứng dụng MR chạy trên nhiều thiết bị khác nhau như HoloLens, Magic Leap, Meta Quest 3,... Điều này giúp mở rộng phạm vi sử dụng và mang lại trải nghiệm thực tế tăng cường (MR) tốt hơn cho người dùng.

- Nâng cấp chức năng hướng dẫn trực tiếp trên vật thể thực tế. Cho phép phát hiện vật thể một cách chính xác và nhanh chóng hơn. Cũng như tăng cường tối ưu hoá hệ thống này để có thể thêm nhiều mô hình đào tạo, hướng dẫn mới.
- Tăng cường khả năng nâng cấp và cập nhật cảm biến cần hiển thị.

### 5.3 Quan điểm của nhóm về tiềm năng của MR

Qua mô hình của nhóm phát triển thì có thể thấy được rằng MR có một tiềm năng phát triển vô cùng lớn. Ứng dụng của nhóm đã chứng minh được tính thuận tiện khi quan sát dữ liệu từ một thiết bị MR. Thay vì phải sử dụng một thiết bị cầm tay hoặc một màn hình gắn trực tiếp để hiển thị giá trị tại từng vị trí muốn đi đến kiểm tra, người giám sát chỉ cần đeo một thiết bị MR, khi di chuyển đến đâu, các thông tin sẽ linh động hiển thị và thay đổi ngay trước mắt mà không cần thực hiện thêm các thao tác phức tạp khác, điểm mạnh của các thiết bị kính MR so với khi sử dụng điện thoại là người dùng có hai tay tự do không cầm nắm bất kỳ thứ gì từ đó giúp thao tác của người dùng dễ dàng và thuận tiện hơn, khả năng quan sát của người dùng cũng sẽ được cải thiện hơn. Mặc dù công nghệ IoT giúp người dùng có thể đứng ở bất cứ đâu để quan sát dữ liệu. Nhưng nếu kết hợp với MR thì sẽ cho phép người dùng trực tiếp nhìn thấy dữ liệu và thông số cần thiết ngay trên thực thể mà họ đang quan sát hoặc làm việc, làm cho trải nghiệm giám sát trở nên trực quan, sinh động hơn.

Ứng dụng của nhóm còn mang đến một tiện ích đột phá của công nghệ MR, đó là khả năng hiển thị các hướng dẫn trực tiếp trên vật thể thật. Việc sử dụng MR để hiển thị các hướng dẫn trực tiếp trên vật thể thật thay vì sử dụng các hướng dẫn truyền thống mang lại nhiều lợi ích vượt trội. Trước tiên, MR cho phép hiển thị các hướng dẫn ngay trên đối tượng thực tế, giúp người dùng dễ dàng nắm bắt các bước cần thực hiện. So với việc đọc các hướng dẫn từ sách vở hoặc màn hình, cách tiếp cận này trực quan và sinh động hơn rất nhiều. Người dùng có thể nhìn thấy chính xác những gì họ cần làm ngay trên vật thể, thay vì phải tưởng tượng hoặc cố gắng liên kết giữa hướng dẫn và thực tế. Ứng dụng MR giúp giảm thiểu sai sót và nhầm lẫn trong quá trình thao tác. Khi các hướng dẫn ảo được hiển thị trực tiếp trên vật thể thật, người dùng có thể theo dõi và thực hiện theo một cách chính xác hơn. Điều này đặc biệt quan trọng đối với những quy trình phức tạp hoặc yêu cầu độ chính xác cao, nơi mà một sai sót nhỏ có thể dẫn đến hậu quả nghiêm trọng. Việc không phải tra cứu thông tin từ các nguồn khác nhau và có thể nhìn thấy hướng dẫn ngay lập tức trên vật thể giúp tiết kiệm thời gian đáng kể. Người dùng không cần phải dừng lại để đọc hướng dẫn, mà có thể thực hiện các bước ngay lập tức theo chỉ dẫn trên MR. Điều này giúp tăng năng suất làm việc, do người dùng có thể làm việc nhanh chóng và hiệu quả hơn, không mất thời gian tìm kiếm hoặc hiểu các hướng

dẫn từ tài liệu. Đồng thời, việc người dùng chỉ cần đeo một thiết bị MR và có thể mang theo bên mình khi di chuyển đến các vị trí khác nhau sẽ tiện lợi hơn nhiều so với việc phải mang theo các tài liệu hướng dẫn hoặc phải quay lại nơi đặt màn hình để xem hướng dẫn.

Ngoài ra, MR còn mang lại một trải nghiệm người dùng hoàn toàn mới mẻ và hấp dẫn hơn so với các phương pháp truyền thống. Người dùng có thể cảm thấy hứng thú hơn khi sử dụng công nghệ mới và hiện đại, điều này có thể cải thiện sự hài lòng và động lực làm việc. Tính tương tác cao của MR cho phép người dùng không chỉ xem mà còn tương tác với các hướng dẫn, bao gồm việc thay đổi góc nhìn, phóng to, thu nhỏ hoặc thực hiện các thao tác trực tiếp trên vật thể, giúp họ hiểu rõ hơn và thực hiện các bước một cách chính xác.

Tuy nhiên, bên cạnh những ưu điểm nổi bật, công nghệ MR hiện tại vẫn còn tồn tại một số nhược điểm cần được xem xét và khắc phục. Một trong những nhược điểm đáng kể của MR là hiện tại chi phí đầu tư ban đầu rất cao. Các thiết bị MR tiên tiến và phần mềm hỗ trợ không hề rẻ, điều này có thể làm tăng chi phí triển khai và duy trì hệ thống, đặc biệt đối với các doanh nghiệp nhỏ hoặc những người dùng cá nhân. Việc đầu tư vào công nghệ MR đòi hỏi một ngân sách đáng kể, không chỉ cho thiết bị mà còn cho việc đào tạo và bảo trì. Nhưng ngày nay với sự phát triển của công nghệ và sự ra mắt của Meta Quest thì các thiết bị có thể tiếp cận MR ngày càng trở nên rẻ và dễ tiếp cận hơn với người dùng.

MR còn gặp phải hạn chế về mặt hiệu suất và khả năng tương thích. Hiện tại, các thiết bị MR vẫn còn bị giới hạn về thời lượng pin và hiệu năng xử lý. Người dùng có thể gặp phải các vấn đề như thiết bị nóng lên, thời gian sử dụng ngắn và hiệu suất không ổn định khi phải xử lý các tác vụ phức tạp.Thêm vào đó, không phải tất cả các thiết bị di động đều tương thích với MR, điều này giới hạn phạm vi ứng dụng của công nghệ.

Cuối cùng, vấn đề về sự thoải mái và sức khỏe của người dùng cũng là một điều cần lưu ý. Việc đeo thiết bị MR trong thời gian dài có thể gây ra sự khó chịu, mệt mỏi hoặc thậm chí đau đầu cho người dùng. Các thiết bị hiện tại vẫn chưa đạt được sự thoải mái tối đa và có thể ảnh hưởng đến hiệu suất làm việc của người dùng trong thời gian dài.

Những nhược điểm này cho thấy rằng, mặc dù công nghệ MR có tiềm năng lớn và mang lại nhiều lợi ích, nhưng vẫn cần có những cải tiến và hoàn thiện để vượt qua các thách thức hiện tại, từ đó mang lại trải nghiệm tốt nhất cho người dùng và mở rộng phạm vi ứng dụng trong thực tế.

# TÀI LIỆU THAM KHẢO

- [1] N. Lever, *Introduction to the Universal Render Pipeline for advanced Unity creators*, 2023, accessed: 2024-05-21. [Online]. Available: <https://unity.com/resources/introduction-universal-render-pipeline-for-advanced-unity-creators>
- [2] The Things Network, “Lorawan classes,” 2024, accessed: 2024-05-27. [Online]. Available: <https://www.thethingsnetwork.org/docs/lorawan/classes/#:~:text=Therefore%2C%20Class%20C%20devices%20can%20receive%20downlink%20messages,on%20the%20cut-off%20valve%20of%20a%20water%20meter.>
- [3] B. T. tin và Truyền thông, *Thông tư số 08/2021/TT-BTTT*, 2021, accessed: 2024-05-27. [Online]. Available: <https://vanban.chinhphu.vn/default.aspx?pageid=27160&docid=204286>
- [4] The Things Industries, “The things stack: Running with docker,” 2024, accessed: 2024-05-28. [Online]. Available: <https://www.thethingsindustries.com/docs/the-things-stack/host/docker/>
- [5] Wikipedia contributors, “Haversine formula,” 2024, accessed: 2024-05-27. [Online]. Available: [https://en.wikipedia.org/wiki/Haversine\\_formula](https://en.wikipedia.org/wiki/Haversine_formula)
- [6] RAKWireless, “Rak common for gateway,” 2024, accessed: 2024-05-27. [Online]. Available: [https://github.com/RAKWireless/rak\\_common\\_for\\_gateway](https://github.com/RAKWireless/rak_common_for_gateway)
- [7] LoRa Alliance Technical Committee Regional Parameters Workgroup, *LoRaWAN Regional Parameters RP002-1.0.4*, LoRa Alliance, 2022, <https://resources.lora-alliance.org/home/rp002-1-0-4-regional-parameters>.
- [8] Google Developers. (2024) Supported devices for arcore. Accessed: 2024-05-30. [Online]. Available: [https://developers.google.com/ar/devices?hl=vi#google\\_play](https://developers.google.com/ar/devices?hl=vi#google_play)