

University of Science



Final Project

VẬT LÝ CHO CÔNG NGHỆ THÔNG TIN

Giảng viên lý thuyết Cao Xuân Nam

Giảng viên thực hành Đặng Hoài Thương

Sinh viên Lê Hoàng Sang 21127158

Vũ Đình Chương 21127236

Nguyễn Quốc Huy 21127511

Mục lục

1 LỜI NÓI ĐẦU	3
2 TỔNG QUAN	3
2.1 Tên đề tài	3
2.2 Lý do chọn đề tài	3
2.3 Vấn đề đặt ra	4
2.4 Đặc điểm và lợi ích	5
2.5 Mục tiêu và ý nghĩa	5
3 CẤU TẠO VÀ THIẾT KẾ	6
3.1 Danh sách thiết bị cần dùng	6
4 NGUYÊN LÝ VÀ CHI TIẾT	8
4.1 Nguyên lý làm việc	8
4.2 Chi tiết hoạt động	9
4.3 Các thông số	10
5 KẾ HOẠCH CHI TIẾT	
6 PHÁC THẢO CHI TIẾT BẰNG MÔ HÌNH 3D	13
6.1 Tổng quan mô hình	13
6.2 Cấu trúc chi tiết	15
7 VẬN HÀNH VÀ XỬ LÝ	21
7.1 Cài đặt, xử lý và xuất thông tin giả lập cho người dùng	21
7.2 Điều khiển thiết bị giả lập phần cứng	23
7.3 Tính toán số liệu và tiến trình	24
7.4 Lập trình kết nối Wifi, MQTT	26
7.5 Nhận thông tin từ Website	27
7.6 Lấy thời gian thực trên mạch	27
7.7 Đẩy thông tin lên website (dạng JSON)	28
7.8 Gửi thông báo qua điện thoại	29
7.9 Gửi thông tin từ mạch lên Cloud	30
7.10 Xử lý tính logic của mạch và các tình huống khẩn cấp	32

8 Điều khiển và thông tin từ Website	34
8.1 Hình ảnh giao diện website	34
8.2 Hình ảnh cấu trúc nodes	37
8.3 Mô tả quá trình hoạt động của website	38
8.3.1 Luồng nhận dữ liệu:	38
8.3.2 Điều khiển thiết bị:	38
8.3.3 Đọc dữ liệu từ Cloud ThingSpeak:	38
9 HƯỚNG PHÁT TRIỂN THÊM	39
10 CÁC TÀI LIỆU THAM THẢO	39

1 LỜI NÓI ĐẦU

- Đây là dự án phục vụ cho môn Vật lý cho Công nghệ thông tin (Internet of things). Trong dự án có áp dụng tất cả những kiến thức cũng như các thiết bị đã được dạy bởi môn học. Ngoài ra, nhằm đáp ứng cho các yêu cầu khách quan trong đời sống, nhóm cũng đưa vào một số thiết bị mới nhưng quen thuộc với cuộc sống. Dự án hướng đến mục đích bảo vệ và phục hồi tài nguyên nên năng lượng cần cung cấp cho máy là năng lượng sạch lấy từ thiên nhiên.
- Các thành viên trong nhóm gồm có:

MSSV	Họ tên	Vai trò
21127158	Lê Hoàng Sang	Node-red, các xử lí trên website
21127236	Vũ Đình Chương	Lập trình mạch điện, gửi thông báo
21127511	Nguyễn Quốc Huy	Vẽ 3D, lập trình mạch điện

- Quy mô của dự án hiện đang ở mức giả lập, đến từ khoa Công nghệ thông tin, Trường Đại học Khoa học Tự nhiên.

2 TỔNG QUAN

2.1 Tên đề tài

THÙNG XỬ LÍ RÁC THÔNG MINH

2.2 Lý do chọn đề tài

- Rác thải sinh hoạt hàng ngày của mỗi gia đình phần lớn là rác hữu cơ. Nếu có thể tận dụng nguồn rác này để làm phân bón thì có thể giảm tải rất lớn cho việc xử lí rác cho các nhà máy và góp phần đáng kể cho việc bảo vệ môi trường. Chính vì vậy, chúng em nghiên cứu chế tạo một sản phẩm là “Thùng xử lí rác thông minh”.
- Sản phẩm sử dụng men vi sinh kết hợp với ứng dụng năng lượng mặt trời trong quá trình xử lí rác do đó tiết kiệm được tài nguyên, tạo ra phân hữu cơ, giảm ô nhiễm môi trường, mang lại lợi ích cho gia đình và cộng đồng.

2.3 Vấn đề đặt ra

- Vì sao lại sử dụng năng lượng mặt trời?

- Đây là nguồn năng lượng tái tạo không bị cạn kiệt, có thể đáp ứng nhu cầu sử dụng của tất cả mọi người một cách liên tục.
- Có thể sử dụng ở bất cứ đâu có ánh nắng mặt trời chiếu sáng, vào ban đêm có thể dùng pin dự trữ từ buổi sáng.
- Năng lượng mặt trời là nguồn năng lượng sạch có sẵn trong tự nhiên và không gây ô nhiễm môi trường.
- Do thùng rác thiết kế có sử dụng điện và chỉ có điện mặt trời mới có thể đáp ứng tính linh động của việc đặt vị trí thùng rác ở các nơi khác nhau.
- Hiệu quả sử dụng cao với chi phí đầu vào thấp.

- Vì sao lại là thùng ủ phân bón?

- Do tính tiện lợi, tiết kiệm thời gian, chi phí và công sức.
- Đảm bảo an toàn và hợp vệ sinh.
- Sử dụng nguồn năng lượng sạch.
- Phân bón được ủ bằng thùng rác ủ phân chuyên dụng sẽ tạo ra sản phẩm là chất mùn ổn định thu được từ quá trình phân hủy các chất hữu cơ, không chứa các mầm bệnh, không lôi kéo các côn trùng (do là quy trình khép kín), được lưu trữ an toàn.
- Có lợi cho sự phát triển của cây trồng: cung cấp chất dinh dưỡng cho cây.
- Tăng đồ phì nhiêu cho đất.
- Nếu mô hình ủ rác hữu cơ thành phân bón được nhân rộng, việc phân loại và sử dụng men vi sinh phân hủy rác sẽ giúp giảm thiểu đáng kể nguồn gây ô nhiễm môi trường; đồng thời, góp phần xây dựng nền nông nghiệp tiết kiệm, an toàn và bền vững.

- Áp dụng vào môn học như thế nào?

- Vận dụng kiến thức lập trình IoT, mạch điện Arduino và các thiết bị điện tử liên quan.
- Tạo ra một dự án thiết thực, tính ứng dụng và giúp ích cho cộng đồng cao.

2.4 Đặc điểm và lợi ích

- Có thể kiểm tra tiến độ và trạng thái của thùng rác qua website.
- Vì dùng bằng năng lượng mặt trời nên tiết kiệm chi phí điện năng tiêu thụ cho hộ gia đình.
- Linh động việc đặt ở các vị trí khác nhau có ánh sáng mặt trời.
- Thời gian sử dụng lâu dài, không phải tốn sức vận hành.
- Sản phẩm giúp rác thải sinh hoạt (rác thải hữu cơ) tại nhà nên có được một nguồn phân bón liên tục cung cấp cho cây trồng. Đồng thời giảm lượng rác thải ra môi trường bên ngoài và giảm chi phí xử lý rác thải của nhà nước.
- Tiết kiệm được chi phí mua phân bón cho cây cảnh, trồng trọt.

2.5 Mục tiêu và ý nghĩa

- Tạo ra một dự án có tính ứng dụng cao.
- Góp phần vào lĩnh vực công nghệ thân thiện với môi trường.
- Đưa ra một giải pháp tiện dụng, linh hoạt, tiết kiệm cho các hộ gia đình.
- Sử dụng các linh kiện giá rẻ, dễ tiếp cận đối với các gia đình phổ thông.

3 CẤU TẠO VÀ THIẾT KẾ

3.1 Danh sách thiết bị cần dùng

STT	Tên thiết bị	Số lượng	Đơn giá	Thành tiền	Đường dẫn
1	Mạch arduino R3	1	199000	199000	Mạch arduino R3
2	Module ESP8266 ESP-01	1	45000	45000	Module ESP8266 ESP-01
3	Động cơ DC 12V giảm tốc	1	99000	99000	Động cơ DC 12V giảm tốc
4	Cảm biến nhiệt độ, độ ẩm DHT11	1	35000	35000	Cảm biến nhiệt độ, độ ẩm DHT11
5	Đèn dây tóc 12V	1	25000	25000	Đèn dây tóc 12V
6	Động cơ servo MG996R nhông và bánh răng	4	66000	264000	Động cơ servo MG996R nhông và bánh răng
7	Tấm pin năng lượng mặt trời 12V - 50W, diện tích 60x80 cm	1	500000	500000	Tấm pin năng lượng mặt trời
8	Dụng cụ trộn	1	50000	50000	-
9	Module Relay 5V	1	20000	20000	Module Relay 5V
10	LED LCD 1602	1	35000	35000	LED LCD 1602
11	Men ủ EMUNIV	1	33000	33000	Men ủ EMUNIV
12	Ác quy 12V	1	400000	400000	Ác quy 12V
13	Button	4	10000	40000	Button
14	Buzzer	1	24000	24000	Buzzer
15	Dây cắm	40	5600	224000	Dây cắm
16	Hộc đựng	3	50000	150000	Hộc đựng
17	Máy xay	1	220000	220000	Máy xay
18	TỔNG	-	-	1550000	-

- **Chế phẩm vi sinh sử dụng trong quá trình xử lý rác:**

Hiện nay có nhiều chế phẩm sinh học được dùng trong quá trình ủ rác, cây xanh,... cũng như xử lý nước thải thành phân hữu cơ. Nhóm chúng em chọn chế phẩm vi sinh Emuniv trong xử lý rác thải vì dễ tìm, phạm vi áp dụng rộng và giá thành phù hợp.



- **Bộ phận xay rác hữu cơ:**

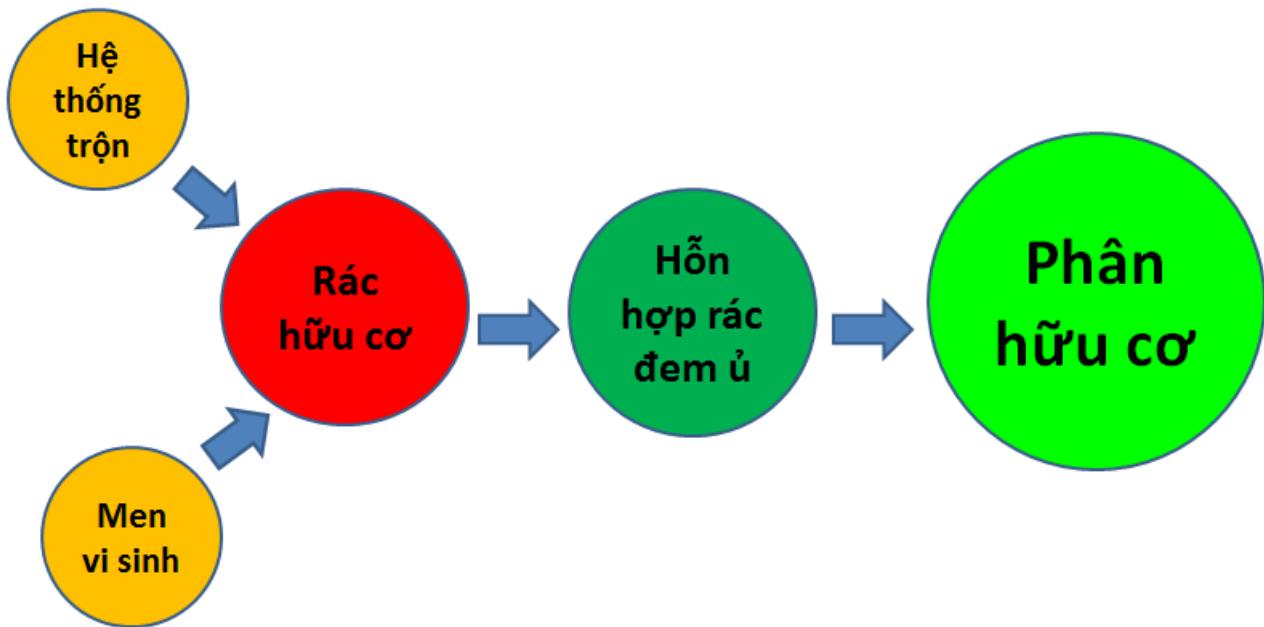
Thành phần chính là một máy xay các loại rác hữu cơ, được đấu nối cơ khí và vận hành nhờ một động cơ DC 12V giảm tốc (60 vòng/phút).



4 NGUYÊN LÝ VÀ CHI TIẾT

4.1 Nguyên lý làm việc

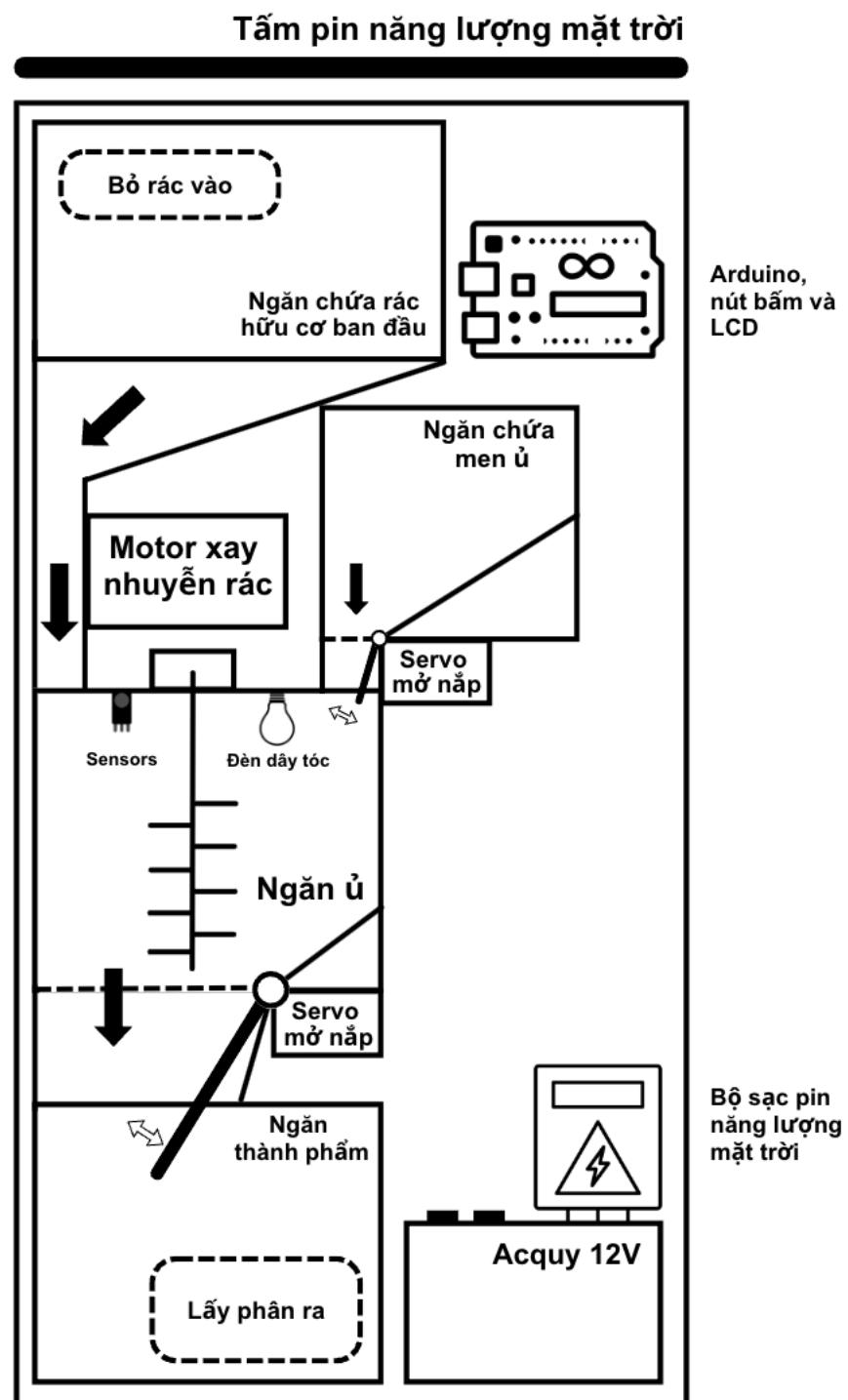
Nguyên lý chung của máy có thể hiểu đơn giản như sau:



- Bước 1: Trộn hỗn hợp rác hữu cơ và lượng men vi sinh đã tính toán trước bằng hệ thống trộn để ra hỗn hợp giữa rác và men.
- Bước 2: Ủ hỗn hợp rác hữu cơ và men trong một khoảng thời gian.
- Bước 3: Sau khi Ủ hỗn hợp rác hữu cơ và men, sau một khoảng thời gian sẽ hình thành phân hữu cơ là thành phẩm. Quá trình này có thể điều khiển từ xa thông qua website.

4.2 Chi tiết hoạt động

Sơ đồ nguyên lý



- Thùng xử lý sẽ nhận năng lượng điện từ tấm pin năng lượng mặt trời, lượng điện năng được chuyển hóa sẽ được lưu trữ vào acquy.
- Người sử dụng đầu tiên cần bấm nút nguồn để khởi động.
- Người dùng được yêu cầu chọn các chế độ tùy theo khối lượng rác khác nhau (khối lượng rác khác nhau ảnh hưởng đến khối lượng men vi sinh cần thiết). Hoặc điều khiển từ xa thông qua website (Start, chọn chế độ khôi lượng/ Pause).
- Sau đó, người dùng bấm nút Start để bắt đầu quy trình.
- Người dùng có thể kích hoạt từ xa thông qua website Node-red bằng việc chọn thông số khôi lượng và Start/Pause thiết bị. Ngoài ra, trong quá trình hoạt động, các thông số như nhiệt độ, độ ẩm, các thông báo, thông báo khẩn cấp luôn được gửi lên website và hiển thị để người dùng theo dõi.
- Rác được bỏ vào ngăn đầu tiên là ngăn chờ, từ mức khôi lượng người dùng chọn, ta ước lượng được lượng vi sinh cần thiết để ủ.
- Sau đó, rác sẽ được đẩy qua ngăn ủ bên dưới bởi bộ phận xay nhuyễn.
- Thời gian ủ có thể từ 5 đến 7 ngày để hoàn toàn được trở thành phân hữu cơ, trong khi ủ, có motor trộn để phân được ủ đều.
- Trong trường hợp thùng đang có rác hữu cơ đang trong quá trình ủ, phần ngăn chờ bên trên còn có khả năng chứa, rác vẫn được đẩy vào ngăn chờ và quá trình sẽ lặp lại khi người dùng lấy phân cũ ra và ấn Start.
- Nếu phân ủ bên dưới đã đến mốc thời gian hoàn thành, thùng sẽ báo trạng thái cho người dùng.
- Sau khi ủ thành phân hữu cơ, phân sẽ được đẩy qua ngăn dưới cùng - ngăn thành phẩm, đây là ngăn rời nên người dùng có thể dễ dàng thu nhận phân bón.
- Người dùng có thể xem lại lịch sử các phiên hoạt động của thùng rác thông qua tab History của website, thông tin này được lưu trên cloud ThingSpeak.

4.3 Các thông số

- Khi rác vừa được bỏ vào và chạy, máy sẽ xác nhận bằng âm thanh từ Buzzer.
- Trong quá trình ủ, LCD sẽ báo trạng thái hoàn thành của thùng, gồm: phần trăm ủ hoàn thành (dựa trên nhiệt độ, ánh sáng, độ ẩm và khối lượng rác người dùng nhập vào), thời gian ước tính còn lại.

- Khi người dùng bấm nút chuyển chế độ hiển thị, trên LCD sẽ đổi thông số hiển thị như thời gian, nhiệt độ hay độ ẩm trong thùng ủ.
- Ngoài ra, hệ thống còn hỗ trợ người dùng dừng tiến trình, Pause ở cả trên thiết bị và website.
- Tất cả các thông số về độ ẩm, thời gian, phần trăm đều sẽ được nêu cụ thể qua website.

5 KẾ HOẠCH CHI TIẾT

STT	Bắt đầu	Kết thúc	Công việc	Thực hiện	Ghi chú
1	20/7/2023	30/7/2023	Thiết kế mạch điện trên wokwi	Đinh Chương	
2	8/8/2023	10/8/2023	Lấy thời gian thực trên mạch	Đinh Chương	
3	15/8/2023	17/8/2023	Đẩy thông tin lên website (dạng JSON)	Đinh Chương	
4	20/7/2023	21/8/2023	Xử lý logic, các tình huống khẩn cấp	Đinh Chương	
5	11/8/2023	15/8/2023	Gửi thông báo qua điện thoại	Đinh Chương	
6	11/8/2023	15/8/2023	Gửi thông tin từ mạch lên cloud	Đinh Chương	
7	17/8/2023	23/8/2023	Kiểm thử, chạy website	Đinh Chương	
8	17/7/2023	23/8/2023	Báo cáo chi tiết	Đinh Chương	
9	24/7/2023	30/7/2023	Cài đặt và xử lý và hiển thị thông tin lên LCD cho người dùng	Quốc Huy	
10	30/7/2023	15/8/2023	Phác thảo chi tiết mô hình 3D (tổng thể và từng phần)	Quốc Huy	
11	30/7/2023	7/8/2023	Lập trình kết nối Wifi	Quốc Huy	
12	30/7/2023	15/8/2023	Nhận thông tin từ website	Quốc Huy	
13	24/7/2023	15/8/2023	Điều khiển các thiết bị theo logic hệ thống (Servo, Lcd, Relay)	Quốc Huy	
14	24/7/2023	7/8/2023	Tính toán lượng pin, lượng men và phần trăm hoàn thành	Quốc Huy	
15	17/8/2023	23/8/2023	Kiểm thử, chạy website	Quốc Huy	
16	17/8/2023	23/8/2023	Báo cáo chi tiết	Quốc Huy	
17	24/7/2023	10/8/2023	Thiết kế toàn bộ website (login, trang thông số, login, điều khiển, thông báo...)	Hoàng Sang	
18	24/7/2023	10/8/2023	Gửi thông tin từ web về mạch (start + weight/stop)	Hoàng Sang	
19	15/8/2023	20/8/2023	Query thông tin từ Cloud và hiển thị dạng table, để hiển thị lịch sử	Hoàng Sang	
20	15/8/2023	20/8/2023	Chức năng xóa lịch sử	Hoàng Sang	
21	15/8/2023	23/8/2023	Kiểm thử, chạy thuật toán từ mạch kết hợp với website	Hoàng Sang	
22	22/8/2023	23/8/2023	Quay demo	Hoàng Sang	Video demo
23	17/8/2023	23/8/2023	Báo cáo chi tiết	Hoàng Sang	
24	10/8/2023	15/8/2023	Tìm hiểu về JSON và cách làm việc với JSON bằng Node-red, JS và C++	Hoàng Sang	Ngoài yêu cầu
25	10/8/2023	15/8/2023	Tạo bộ api từ mạch gửi lên website	Hoàng Sang	Ngoài yêu cầu

6 PHÁC THẢO CHI TIẾT BẰNG MÔ HÌNH 3D

Mô hình được phác thảo trên nền tảng Autodesk Fusion 360. Gồm đầy đủ các chi tiết cũng như các thiết bị cơ bản cần có trong mô hình và liên quan đến môn học.

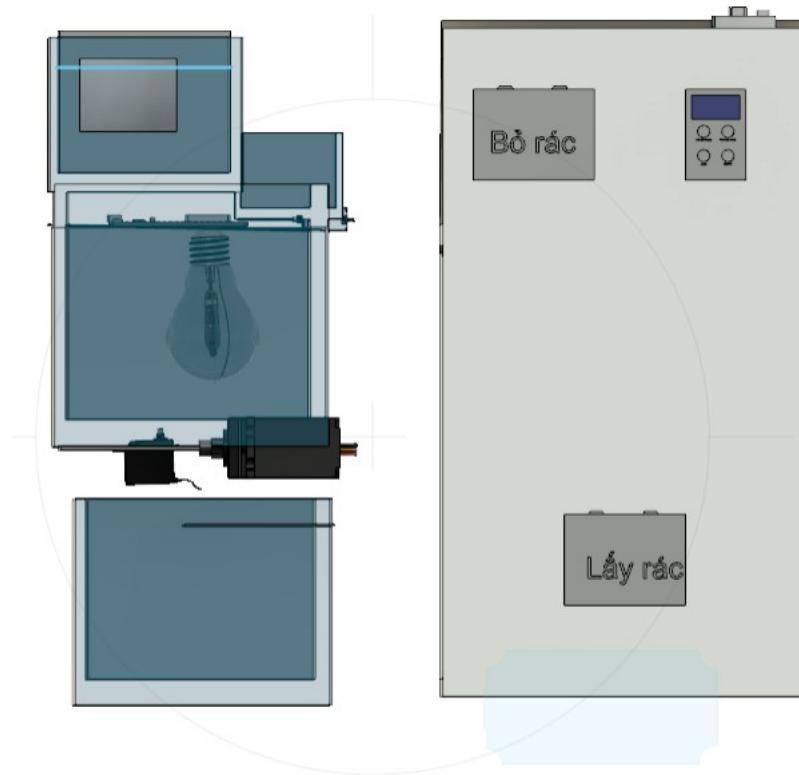
6.1 Tổng quan mô hình

- Mô hình được phác thảo để dễ dàng hình dung với phần vỏ và phần cấu trúc bên trong. Có hình như sau:



- Như hình, mô hình được xây dựng bởi 4 ngăn, gồm: ngăn men, ngăn xay (hay ngăn chứa rác), ngăn ủ, và ngăn thành phẩm.

- Mô hình được phác thảo theo vật liệu trong suốt để có thể thấy được các chi tiết bên trong. Như hình, ta có thể thấy bóng đèn, mạch điện, servo, và một số mảnh vật liệu khác. Ngoài ra, phần vỏ còn cho thấy ta có pin năng lượng mặt trời.

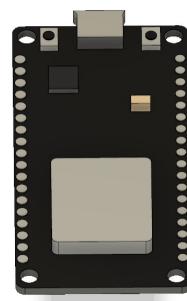


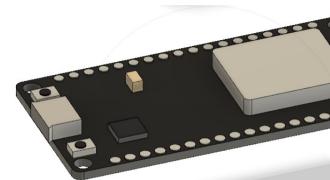
- Đây là phác thảo sau lưng mô hình, gồm các nắp đổ rác, ngăn lấy rác, các nút bấm.
- Các chi tiết chính của mô hình đều được ghi cụ thể trên từng bộ phận.



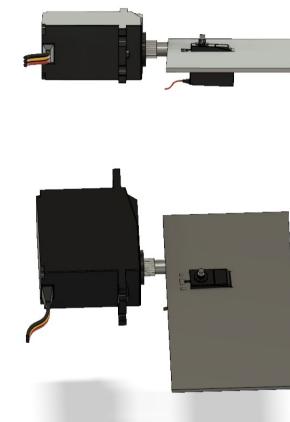
6.2 Cấu trúc chi tiết

- Trước tiên, ta sẽ tiến hành liệt kê các chi tiết và linh kiện cần thiết cho mô hình.
 - Đầu tiên là mạch ESP:





– Bộ phận tiếp theo là Servo:



– Bộ phận tiếp theo là Bóng đèn:



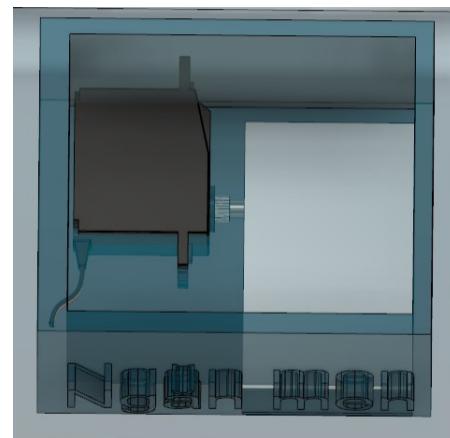
– Bộ phận tiếp theo là Máy xay rác:



Máy xay rác được lấy ý tưởng từ mô hình máy xay thịt, cấu trúc thực tế cũng sẽ giống cách vận hành này, gồm một lối vào nghiền rác và thải ra vụn rác nhỏ qua các lỗ thoát. Trục máy xoay được gắn với 1 motor để trộn thay vì phải xay bằng tay.

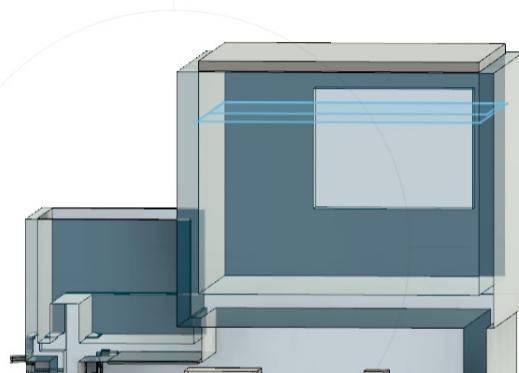
- Và sau đây chúng ta sẽ đến chi tiết các ngăn trong thiết bị. Ngăn đầu tiên là ngăn men, ngăn men có một servo đóng vai trò như một cần gạt đóng mở tự động để nhả số lượng men cần thiết tùy theo khối lượng rác đầu vào:



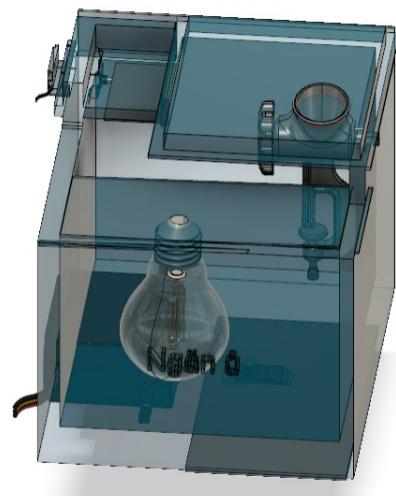


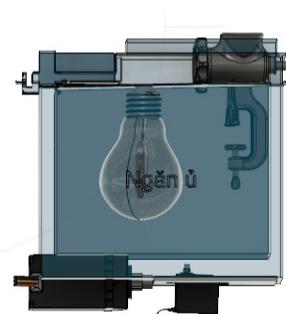
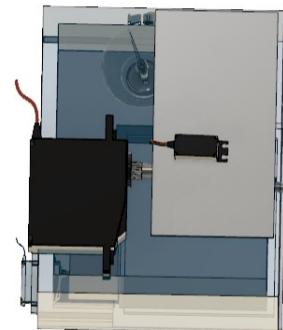
- Ngay bên cạnh ngăn men là ngăn chứa rác (hay ngăn xay). Ngăn này có một máy xay rác dùng để vừa dự trữ rác, vừa xay nhuyễn rác để đẩy rác xuống ngăn ủ, 2 hình bên dưới là ngăn men và ngăn chứa rác.





- Ngăn chính của thiết bị chính là ngăn ủ phân. Ngăn này có các thiết bị như bóng đèn dây tóc dùng để sưởi ấm và tạo nhiệt độ ủ phân, có servo trộn ngay bên dưới đáy, có servo đóng mở nắp như một cần gạt tự động đến ngăn thành phẩm sau khi phân đã ủ xong. Các hình bên dưới là ngăn ủ từ nhiều góc độ chụp khác nhau.





- Ngăn cuối cùng là ngăn ủ, nhiệm vụ đơn giản là chứa thành phẩm sau khi đã ủ xong cho người dùng lấy. Ngoài ra, ngăn ủ còn giữ khô và kín phân.

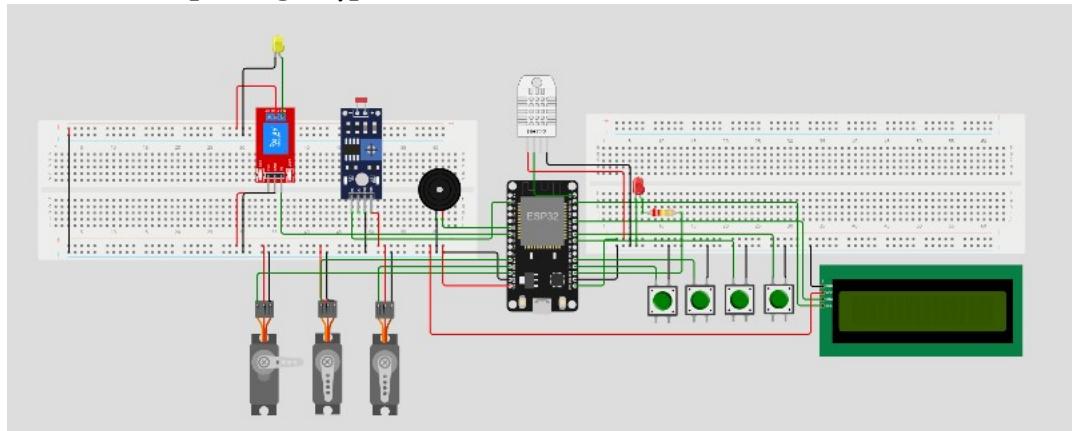




7 VẬN HÀNH VÀ XỬ LÝ

Thuật toán được triển khai, cài đặt và được kết hợp trên các nền tảng Wokwi cùng với Node-red, ThingSpeak và IFTTT. Đây là tổng quan mô hình thùng ủ phân tự động trên mạch.

Video demo sản phẩm giả lập

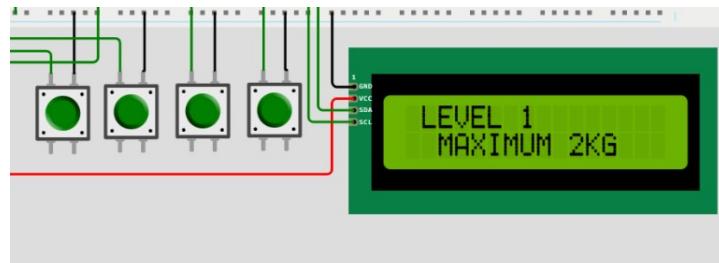


7.1 Cài đặt, xử lý và xuất thông tin giả lập cho người dùng

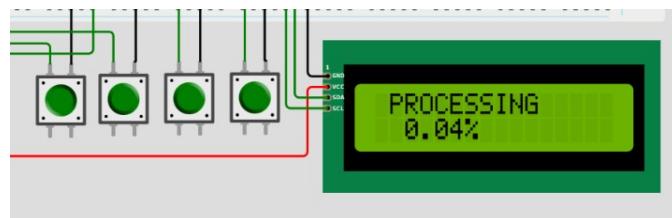
- Để người dùng nắm được các thông tin xử lý rác của thiết bị, ta cần truyền các thông số đó lên màn hình LCD. Các thông số đó bao gồm: Tiến độ hoàn thành, Thông số nhiệt độ, Thông số độ ẩm, Chế độ rác và Lượng pin của thiết bị.
- Để có thể hiện đầy đủ và tường minh các thông tin, ta sẽ áp dụng các hàm của LCD như **LCD.init()**, **LCD.backlight()** để tạo độ sáng cho màn hình cũng như là điều

chỉnh cursor để các dòng được thể hiện đúng trên màn hình.

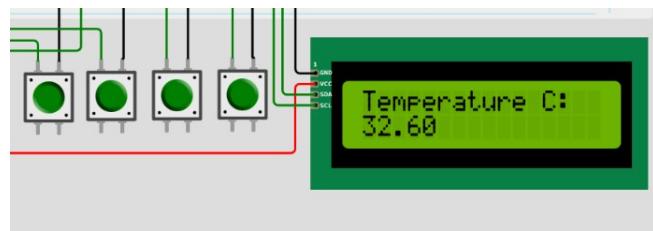
- Đầu tiên ta sẽ có trạng thái khi LCD hiển thị các mode cho người dùng lựa chọn, có 3 mode là tối đa 2kg, 5kg, 10kg. Nội dung hiển thị như sau:

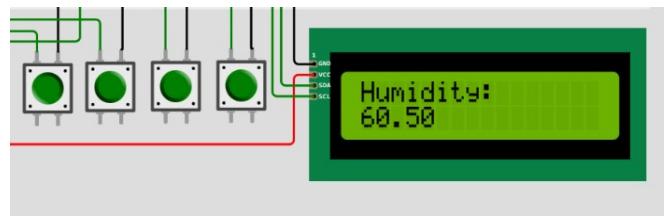


- Tiếp theo, khi thiết bị đang hoạt động, người dùng muốn xem đã được bao nhiêu phần trăm thì mode tiến độ sẽ được hỗ trợ trên màn hình. Nội dung cho biết đã hoàn thành được bao nhiêu phần trăm trên tiến độ dự đoán, nội dung như hình sau:

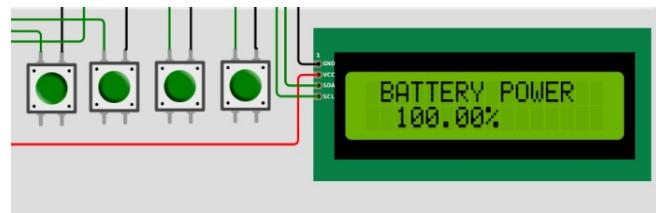


- Các thông số về nhiệt độ, độ ẩm cũng vô cùng quan trọng, nó cho biết liệu nhiệt độ có đủ ẩm và phân bón đã sẵn sàng nhờ độ khô hay chưa. Vì lẽ đó nên màn hình cũng sẽ hiển thị song song 2 thông tin nhiệt độ và độ ẩm. Nội dung chi tiết như các hình sau:



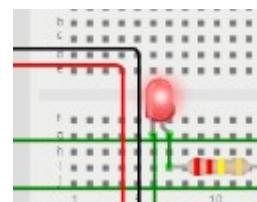


- Thông số cuối cùng và cũng không kém phần quan trọng, vì máy được vận hành nhờ nguồn năng lượng mặt trời, nên khi không có ánh sáng, lượng pin của thiết bị sẽ giảm. Chính vì điều đó, người dùng cần theo dõi lượng pin đang có của thùng rác:

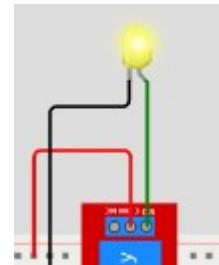


7.2 Điều khiển thiết bị giả lập phần cứng

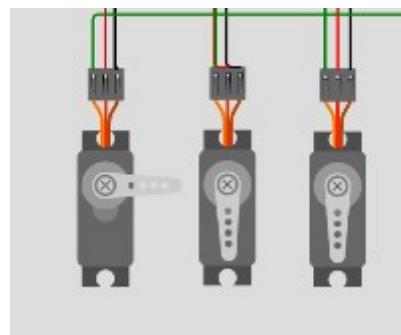
- Để thiết bị có thể chạy đúng theo những gì người dùng yêu cầu, ta cần lập trình sao cho chúng đáp ứng với nhu cầu đó nhưng vẫn nắm được các thông số hoạt động cũng như các bước logic cơ bản của thiết bị mà không làm nghịch lý quá trình diễn ra sự ủ.
- Đầu tiên, ta cần lập trình sao cho khi người dùng nhấn vào nút chạy sau khi chọn chế độ bỏ rác thì đèn led sẽ sáng. Đèn led này sáng đồng nghĩa với việc máy đang trong quá trình thực hiện đến khi hoàn thành xong.



- Cùng lúc này, bước tiếp theo người dùng sẽ tiến hành bỏ rác vào thùng, quá trình này dẫn đến việc rác cần được xay nhuyễn. Vì vậy sau khi bật đèn led, ta sẽ kích hoạt máy xay thông qua relay, cũng như bật đèn sưởi để số lượng rác xay nhuyễn được ủ ngay lập tức.



- Người dùng lựa chọn chế độ rác thì hệ thống sẽ tính toán số lượng men cần thiết cho số lượng rác tương ứng đó. Tính toán xong, hệ thống sẽ nhả men ra tùy theo kích thước rác tương ứng, quy trình nhả men này được thực hiện qua một servo đóng vai trò như một nắp đóng mở tự động. Khi lượng rác nhiều, thì servo mở sẽ lâu hơn và ngược lại.
- Trong khi rác đang ú trong ngăn, sẽ có một servo đóng vai trò như một máy trộn tự động giúp đảo phân bón cho tươi và đều, làm tăng quá trình ú.
- Và sau khi ú rác xong, một servo thứ 3 sẽ đóng vai trò như một nắp đóng mở tự động đẩy phân từ ngăn ú xuống ngăn thành phẩm. Và như vậy, chúng ta sẽ có 3 servo với 3 vai trò khác nhau trong việc vận hành xử lý rác thành phân bón.



- Tính theo thứ tự từ trái sang phải Servo lần lượt sẽ là: Servo trộn rác, Servo nhả men, Servo ngăn thành phẩm.

7.3 Tính toán số liệu và tiến trình

- Các thông số của thiết bị cần được tính toán gồm có:
 - Lượng pin

- Tiến độ hoàn thành
 - Lượng men được nhả
- Đầu tiên là lượng pin, để tính lượng pin, thiết bị được hoạt động theo nguyên lý khi có ánh sáng đủ mạnh, pin mặt trời sẽ tiếp nhận và sạc cho ác quy.
 - Ta đo được độ mạnh yếu, giá trị của ánh sáng dựa vào quang trở, khi quang trở có giá trị cao, ánh sáng sẽ yếu dần. Từ đó ta nhận biết được lượng pin của thiết bị trong tình trạng thiếu ánh sáng.
 - Cụ thể, ta sẽ lấy ngưỡng quang trở 3000 để nhận biết sự mạnh yếu của ánh sáng, khi quang trở vượt quá 3000, pin sẽ giảm, ngược lại khi quang trở thấp, ánh sáng đủ mạnh để tấm pin mặt trời hấp thụ, pin sẽ tăng.
 - Ta sẽ dựa vào thời gian quang trở vượt ngưỡng để tính toán sự tiêu hao pin. Cứ mỗi milisecond sẽ chia cho 50000, như vậy sự tiêu hao pin sẽ được giảm theo mỗi milisecond. Trung bình cứ 1 giây khi không có ánh sáng, thiết bị sẽ bị mất khoảng 0,01 đến 0,02 % pin
 - Bên cạnh đó, khi ánh sáng đủ mạnh, pin cũng sẽ phục hồi theo một khoảng tính toán mỗi milisecond sẽ chia cho 30000, và như vậy trung bình cứ 1 giây khi ánh sáng đủ mạnh, pin sẽ lại tăng thêm 0,02 đến 0,03 %.
 - Khi pin đạt mức 100 %, thiết bị sẽ không cập nhật thêm lượng pin kể cả khi ánh sáng đủ mạnh.
 - Thông số tiếp theo, đó là tiến độ hoàn thành. Tiến độ hoàn thành được tính dựa trên sự lựa chọn các mức rác đầu vào từ người dùng.
 - Tiến độ được tính tổng quát như sau:

$$\text{Thời gian ước tính} \div \text{Thời gian đã qua kể từ lúc bắt đầu}$$
 - Thời gian bắt đầu được lấy khi người dùng nhấn nút Start trên thiết bị hoặc trên website nhờ hàm **millis()**.
 - Cứ mỗi vòng lặp đã qua, một biến **nowTime** sẽ tự động lấy thời gian hiện. Từ đó ta tính được **Thời gian đã qua kể từ lúc bắt đầu**.
 - Vì trong môi trường giả lập, thời gian ước tính là một con số không quá lớn nhưng cũng tuân thủ logic về sự ủ phân theo các khối lượng rác nhất định. Cụ thể, ở mức rác 1 (tối đa 2 kg), ta có thời gian ước tính là 10000. Mức rác 2 (tối đa 5 kg), thời gian ước tính là 20000. Và cuối cùng mức rác 3 (tối đa 10 kg), thời gian ước tính là 50000.
 - Từ đó ta sẽ tính toán được thời gian hoàn thành của sản phẩm phân bón và hiển thị cho người dùng thông qua màn hình LCD và website.

- Thông số cuối cùng đó là lượng men được nhả, cũng giống như thông số về thời gian tiên độ, lượng men cũng được tính tùy thuộc vào mức rác tương ứng mà người dùng lựa chọn.
 - Do không tìm được các thiết bị đo lường khôi lượng giả lập, nên ta sẽ quy định sự vận hành của các Servo theo kích thước tương ứng.
 - Với mức rác 1 (tối đa 2 kg), Servo sẽ tiến hành mở cần gạt ở mức 90 độ cho men rơi xuống với delay là 400. Đây là khoảng thời gian đủ để một mức men nhất định rơi xuống và phù hợp với mức rác.
 - Với mức rác 2 (tối đa 5 kg), Servo sẽ tiến hành mở cần gạt ở mức 90 độ cho men rơi xuống với delay là 600.
 - Với mức rác 3 (tối đa 10 kg), Servo sẽ tiến hành mở cần gạt ở mức 90 độ cho men rơi xuống với delay là 800.

7.4 Lập trình kết nối Wifi, MQTT

- Để có thể kết nối được với website, gửi và nhận thông tin từ website, ta cần một giao thức kết nối để đảm bảo hỗ trợ đầy đủ cho các thiết bị vật lý.
- MQTT (Message Queuing Telemetry Transport) là giao thức truyền thông điệp (message) theo mô hình publish/subscribe (cung cấp/ thuê bao), được sử dụng cho các thiết bị IoT với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong mạng lưới không ổn định. Nó dựa trên một Broker (tạm dịch là “Máy chủ môi giới”) “nhẹ” (khá ít xử lý) và được thiết kế có tính mở (tức là không đặc trưng cho ứng dụng cụ thể nào), đơn giản và dễ cài đặt.
- Thiết bị kết nối vào mạng WIFI được gọi là station (trạm). Việc kết nối vào mạng Wifi được hỗ trợ bởi một access point (AP), một AP có chức năng như một hub nhưng dùng cho nhiều station. Một access point thông thường được kết nối vào một mạng dây để phát WIFI (tức là chuyển từ mạng dây sang WIFI). Do đó access point luôn được tích hợp vào router. Mỗi access point được nhận biết bằng một SSID (Service Set IDentifier), SSID cũng là tên của mạng hiển thị khi ta kết nối vào WIFI.
- Thư viện **ESP8266WiFi.h** có hỗ trợ các câu lệnh để module thực hiện việc kết nối vào WIFI (làm chức năng của station).
- Để có thể kết nối Wifi cũng như sử dụng thông qua giao thức MQTT, ta cần cài đặt các thư viện sau:

```

1 #include <WiFiClient.h>
2 #include <ESP8266WiFi.h>
3 #include <PubSubClient.h>

```

- Sau đó, ta tiến hành cài đặt các hàm cần thiết như **wifiConnect()**, **mqttConnect**, **sendRequest**, **callBack** để có thể kết nối giữa Wokwi và Node-red.

7.5 Nhận thông tin từ Website

- Hàm **callBack()** được xây dựng để hỗ trợ nhận thông tin từ Website về Wokwi.
- Cụ thể, website sẽ trả về 1 số dạng chuỗi tương ứng với từng chức năng.
 - Nếu số đó là 0 - 1 - 2, đó chính là 3 mức rác của người dùng lựa chọn trên website, tương ứng với ít rác, rác vừa, nhiều rác và cũng là tín hiệu Start.
 - Nếu đó là số 4, nghĩa là người dùng đang yêu cầu Pause hệ thống trên website và ta cần xử lý dừng trong đoạn mã lập trình.

7.6 Lấy thời gian thực trên mạch

- Thiết lập một đối tượng **NTPClient** có khả năng liên kết với máy chủ NTP tại "**vn.pool.ntp.org**", sẵn sàng để cập nhật thời gian.
- timeClient.forceUpdate()**: Đây là một phương thức gọi để cập nhật dữ liệu thời gian từ máy chủ NTP thông qua các thư viện sau:

```

1 #include <NTPClient.h>
2 #include <WiFiUdp.h>

```

- strftime(startDate, sizeof(startDate), "%Y-%m-%d", timeinfo)**: Đoạn mã này sử dụng hàm **strftime** để định dạng ngày hiện tại thành chuỗi có định dạng "Năm/Tháng/Ngày" và lưu vào biến **startDate**.
- int daysToAdd = DateExpect + mode**: Biến **daysToAdd** là tổng của hai biến **date-Expect** và **mode**.
- timeinfo->tm_mday += daysToAdd**: Thêm số ngày từ biến **daysToAdd** vào ngày hiện tại.

- **time_t newTimestamp = mktime(timeinfo):** Chuyển đổi cấu trúc struct tm đã được thay đổi thành một timestamp mới bằng cách sử dụng hàm **mkmktime**.
- Sau đó, cập nhật **timeinfo** để thể hiện ngày mới tính toán và sử dụng hàm **strftime** để định dạng ngày kết thúc, sau đó in ra ngày kết thúc.

7.7 Đẩy thông tin lên website (dạng JSON)

- Sử dụng mqtt request để đẩy thông tin lên node-red thông qua mqttClient.publish() và thư viện.

```
1 #include <ArduinoJson.h>
```

- Trong mã nguồn sử dụng hàm datapush() nhiệm vụ chính của hàm này là đóng gói các thông tin liên quan trong quá trình xử lý vào một đối tượng JSON và sau đó gửi đối tượng JSON này thông qua giao thức MQTT.
- **StaticJsonDocument<200> jsonDoc:** Đây là việc tạo một đối tượng JSON tĩnh (không cần phải cấp phát động) với dung lượng tối đa là 200 bytes. Đối tượng JSON này sẽ được sử dụng để đóng gói thông tin.

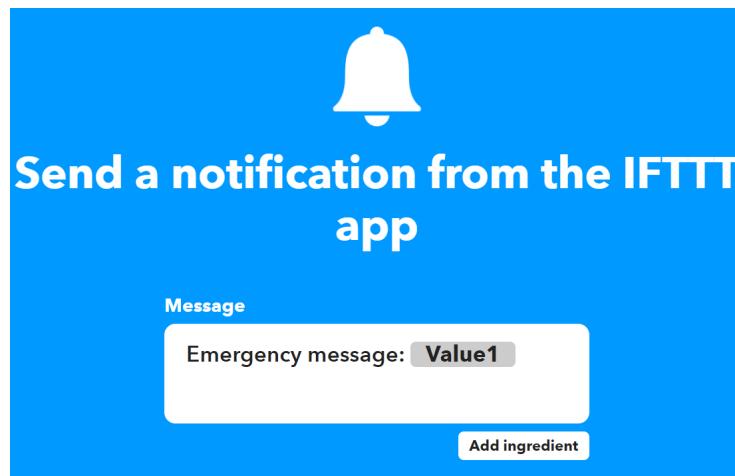
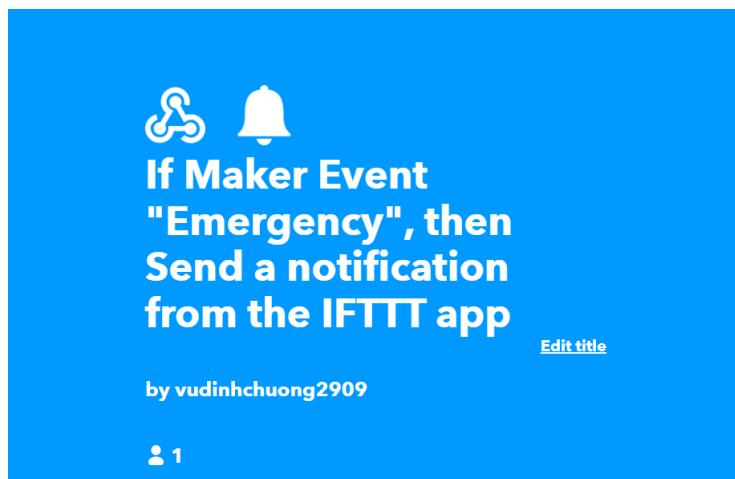
- Đóng gói JSON gồm các thuộc tính:

```
1 var jsonDoc = {
2     "temperature": 25, // Nhiệt độ trong thung lũng
3     "humidity": 60, // Độ ẩm trong khu vực
4     "start_time": "2023-08-1", // Ngày bắt đầu
5     "completion_time": "2023-08-18", // Ngày dự đoán kết thúc
6     "processing": 55, // Ước lượng mục tiêu hoàn thành
7     "message": "Processing is underway.", // Tin nhắn
8     "emergency": "Nothing.", // Tin nhắn trường hợp khẩn cấp
9     "battery": 80 // Phản trạm pin còn lại
10    };
```

- **serializeJson(jsonDoc, jsonString):** Dòng này chuyển đổi đối tượng JSON (jsonDoc) thành một chuỗi JSON (jsonString) bằng cách sử dụng hàm serializeJson.
- **mqttClient.publish("smartbin/data", jsonString.c_str()):** Cuối cùng, đoạn mã này sử dụng đối tượng MQTT (mqttClient) để gửi chuỗi JSON (jsonString) thông qua kênh MQTT có tên "smartbin/data".

7.8 Gửi thông báo qua điện thoại

- Gửi thông báo khẩn cấp qua điện thoại thông qua app **ifttt**
- Sử dụng chức năng **Webhook** và **Notification** để gửi tín hiệu http request cho ifttt và sau đó Notification thông báo về nhưng trường hợp khẩn cấp về cho điện thoại



- Sau khi đã lấy được đường dẫn http. Trong mã nguồn sử dụng http request phương pháp GET để chuyển dữ liệu.
- Connect **port = 80** với **host = "maker.ifttt.com"** và **request = "/trigger/emergency/with/key/bjIGHAgSupsGTVdiaLuzjnpRgh9Pz3_-DibMK00io52?value1 = Low_Battery"** với trường hợp khẩn cấp là Low Battery.

- Khi trường hợp khẩn cấp xảy ra sẽ gửi về cho người dùng với thông báo **Emergency message: Value1** với **Value1 = Low_Battery**.



7.9 Gửi thông tin từ mạch lên Cloud

- Tương tự như việc gửi thông báo qua điện thoại, ta dùng HTTP request gửi thông tin lên Cloud-Thinkspeak.

The screenshot shows the Thingspeak API Keys page for a channel with ID 2243631. It includes sections for Write API Key (with key XQB5QQ0J11FHSTXQ) and Read API Keys (with key BUL6510LI3D2VL30), along with API Requests and API Keys Settings.

Write API Key

Key: XQB5QQ0J11FHSTXQ

Generate New Write API Key

Read API Keys

Key: BUL6510LI3D2VL30

Note:

Save Note Delete API Key

Add New Read API Key

Help

API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.

API Keys Settings

- Write API Key: Use this key to write data to a channel. If you feel your key has been compromised, click Generate New Write API Key.
- Read API Keys: Use this key to allow other people to view your private channel feeds and charts. Click Generate New Read API Key to generate an additional read key for the channel.
- Note: Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.

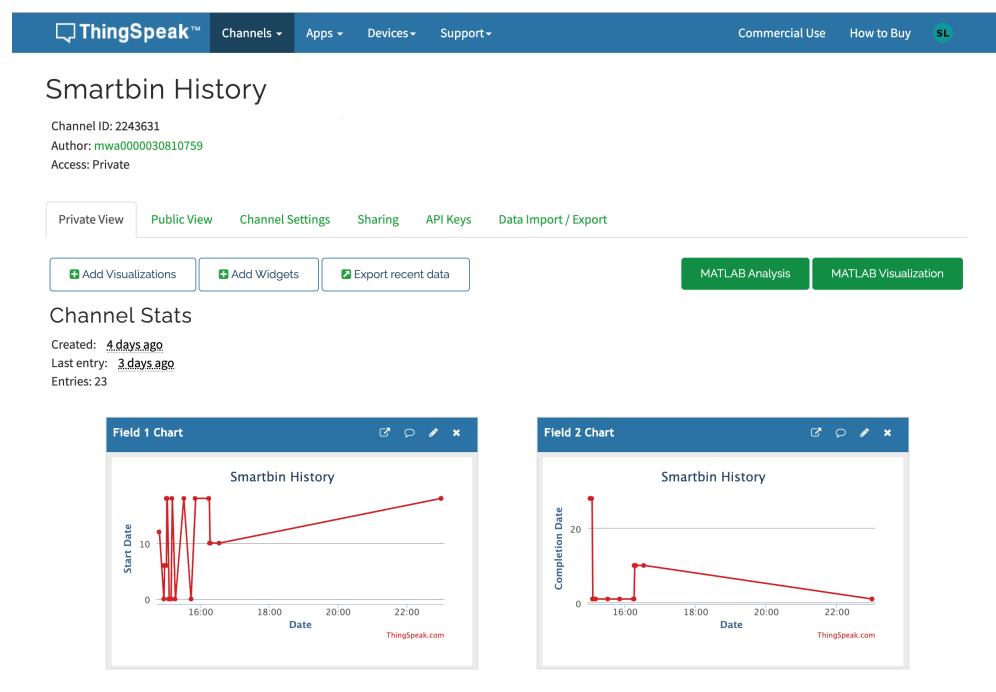
API Requests

Write a Channel Feed
GET https://api.thingspeak.com/update?api_key=XQB5QQ0J11FHSTXQ

Read a Channel Feed
GET https://api.thingspeak.com/channels/2243631/feeds.json?api_key=BUL6510LI3D2VL30

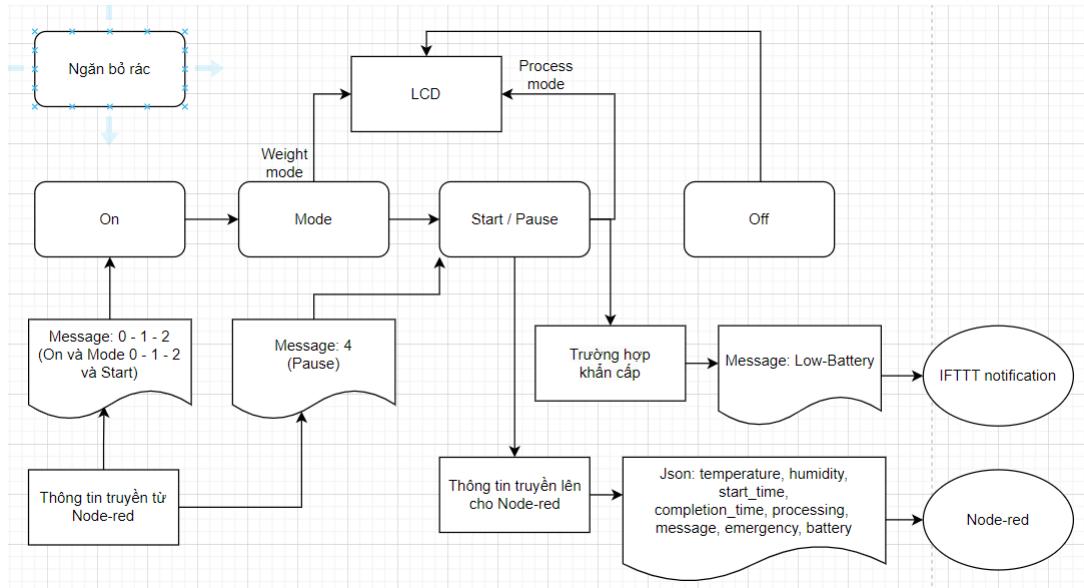
Read a Channel Field
GET https://api.thingspeak.com/channels/2243631/fields/1.json?api_key=BUL6510LI3D2VL30

- **host = "api.thingspeak.com"** và với **request = "/update?api_key=XQB5QQ0J11FHSTXQ&field1=&field2=&field3=&field4= "** với tương ứng các giá trị của các field là ngày bắt đầu, ngày kết thúc và tin nhắn khẩn cấp khi máy hoạt động.

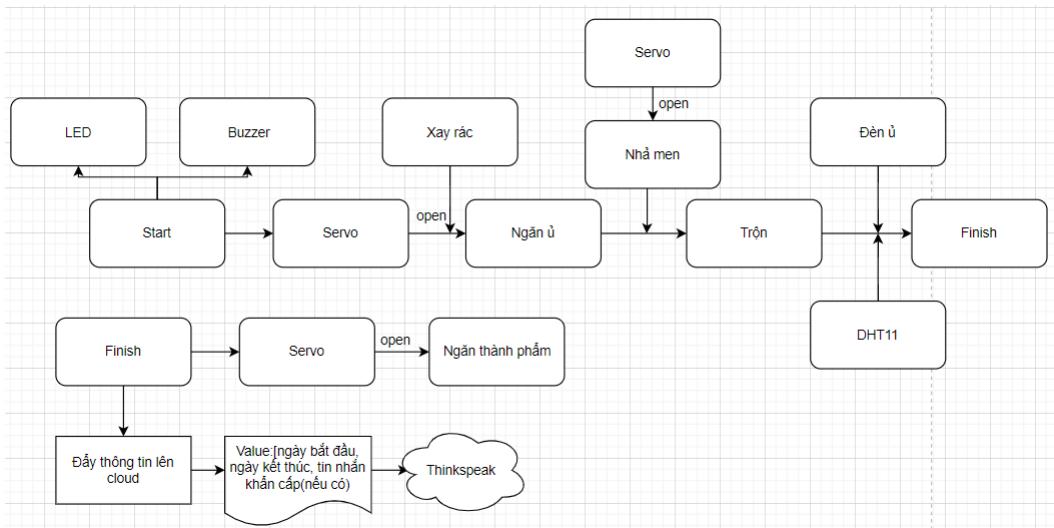


7.10 Xử lý tính logic của mạch và các tình huống khẩn cấp

- Dưới đây là sơ đồ truyền và nhận dữ liệu giữa các đối tượng trong hệ thống IoT



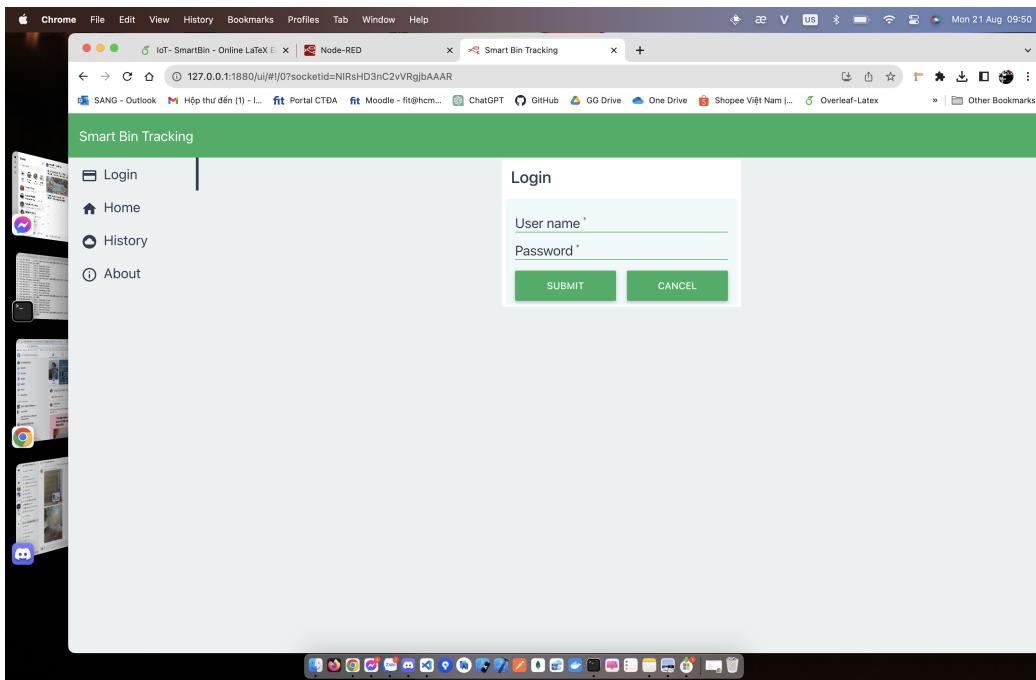
- Sử nút On để khởi động máy, chọn Mode để thể chọn mức độ nặng của rác bỏ vào để máy xả bộ men phù hợp. Sau đó nhấn nút Start để khởi động.
 - Khi bắt đầu khởi động máy sẽ lấy dữ liệu sau đó gửi lên cho node-red đồng thời cũng thể hiện lên trên LCD. Khi pin yếu Trường hợp khẩn cấp sẽ gửi thông báo về điện thoại thông qua app IFTTT.
 - Có thể nhấn nút Mode để chuyển sang các thông báo khác như Nhiệt độ và độ ẩm, Pin, Phần trăm xử lý.
 - Có thể nhấn lại nút Start để Pause thì sẽ hiển thị **PAUSE** lên LCD đến khi nhấn lại nút Start lần nữa.
 - Có thể thực hiện bật mở máy thông qua website Node-red bằng cách chọn Weight và Start trên web sau đó máy sẽ mở lên và hoạt động như trên. Có thể sử dụng Pause trên web để tạm dừng máy.
 - Bấm nút Off để tắt máy và lấy sản phẩm ở ngăn thành phẩm



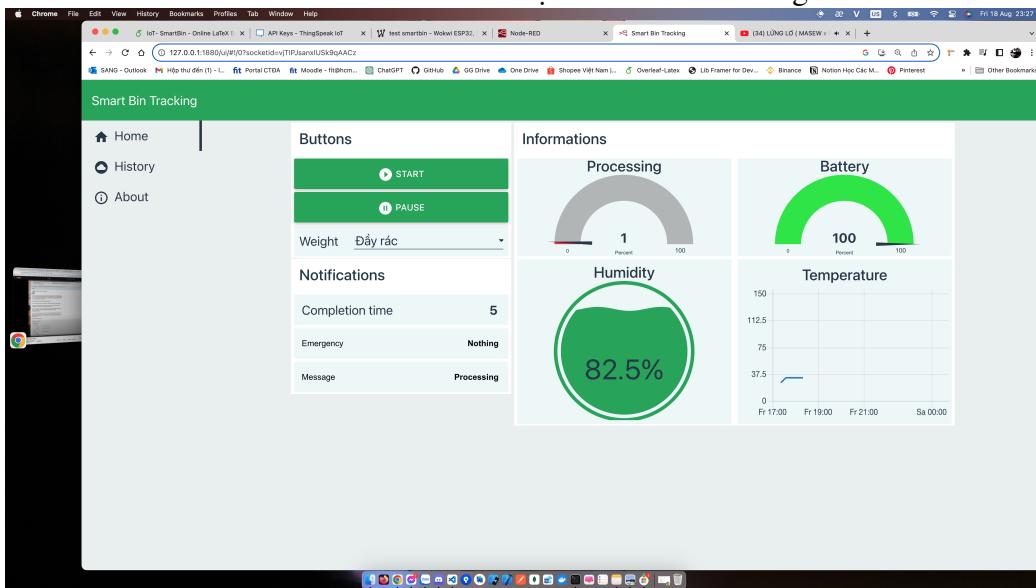
- Sau khi máy bắt đầu chạy thì LED sáng và Buzzer kêu lên để báo hiệu bắt đầu quá trình Ủ. Men sẽ được nhả xuống đồng thời máy xay sẽ xay nhuyễn rác ra. Sau đó sẽ trộn hỗn hợp rác với men và nhiệt độ từ đèn Ủ để dần dần thành phân bón. Cảm biến nhiệt độ và độ ẩm sẽ thông báo nhiệt độ và độ ẩm trong thùng Ủ.
- Khi hoàn thành thì phân bón sẽ trả ra ở ngăn thành phẩm, LED sẽ tắt và Buzzer sẽ báo hiệu tín hiệu kết thúc. Đồng thời sẽ gửi thông tin ngày bắt đầu, ngày kết thúc, tín hiệu khẩn cấp (nếu có) vào Cloud-Thinkspeak để lưu lại.

8 Điều khiển và thông tin từ Website

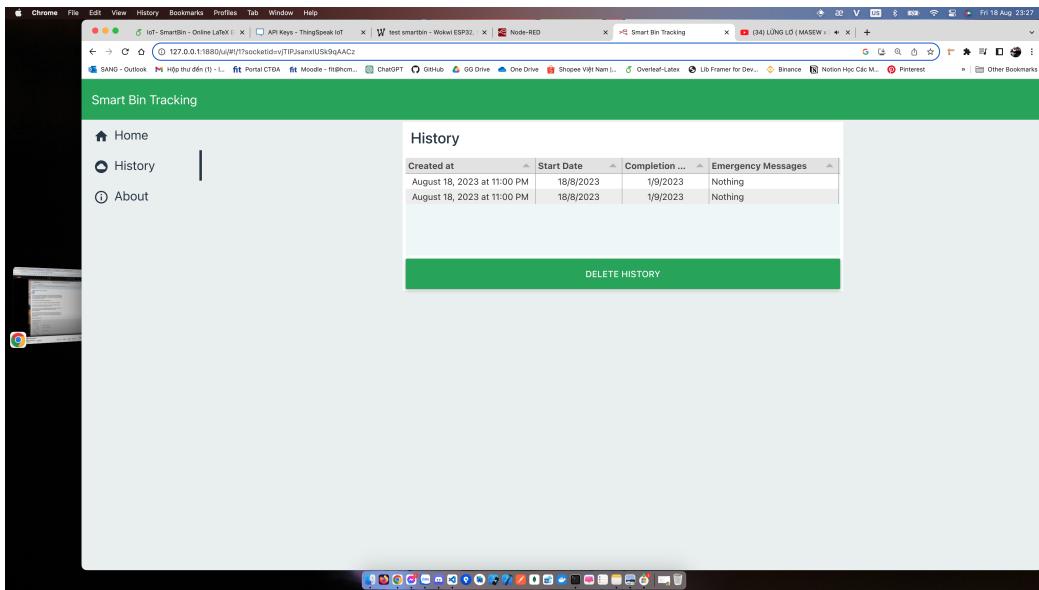
8.1 Hình ảnh giao diện website



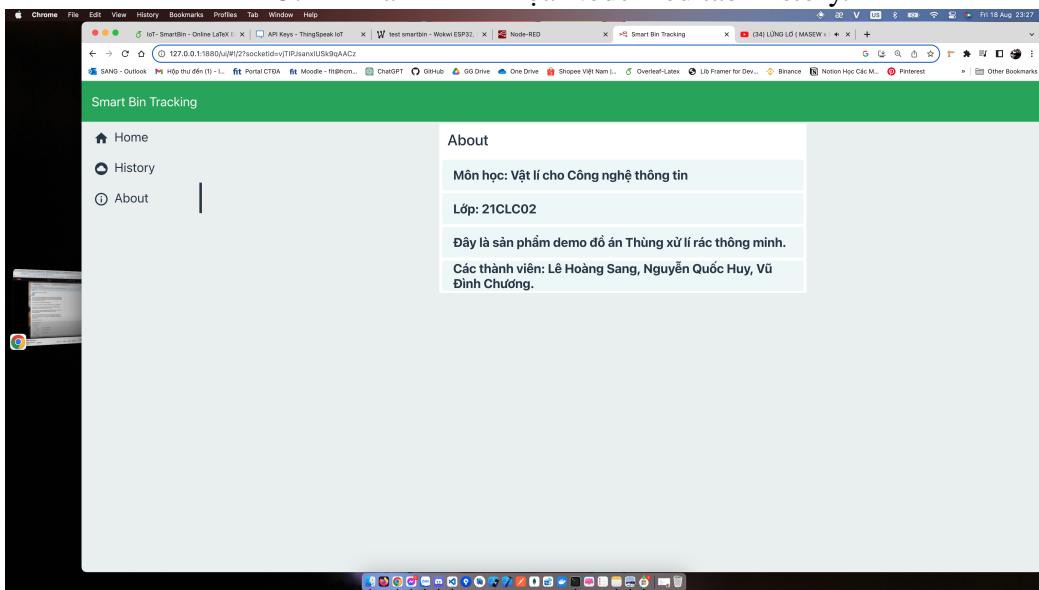
Hình 1: Hình ảnh minh họa Node-Red tab Login.



Hình 2: Hình ảnh minh họa Node-Red tab Home.

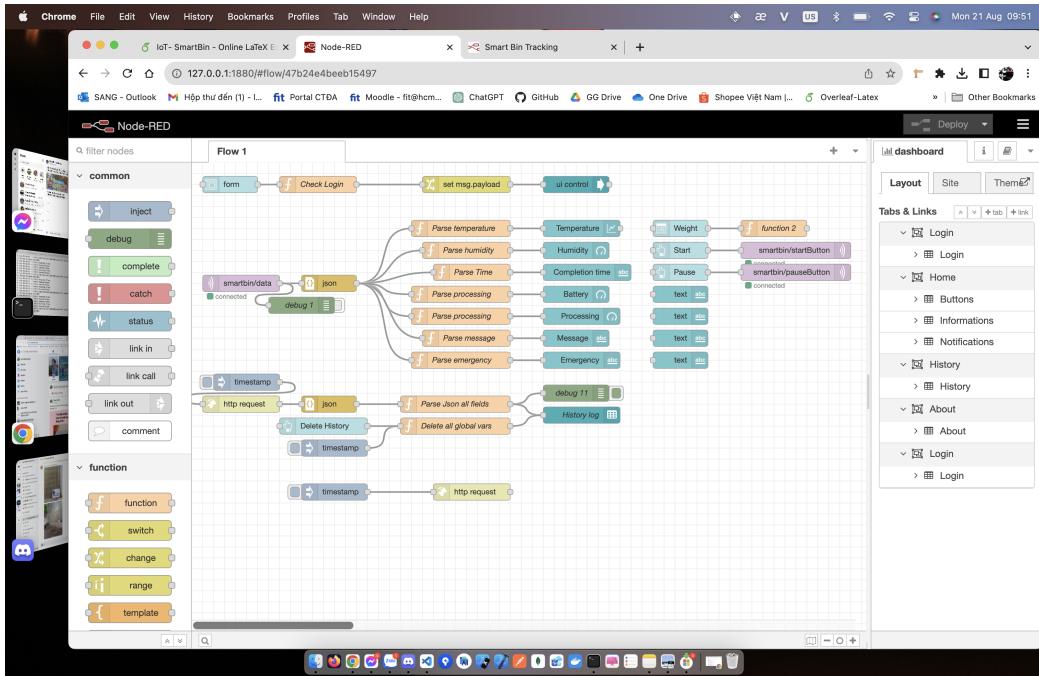


Hình 3: Hình ảnh minh họa Node-Red tab History.



Hình 4: Hình ảnh minh họa Node-Red tab About.

8.2 Hình ảnh cấu trúc nodes



Hình 5: Hình ảnh minh họa Node-Red.

8.3 Mô tả quá trình hoạt động của website

8.3.1 Luồng nhận dữ liệu:

- Đầu tiên, website sẽ nhận một JSON từ mạch điện gửi lên, gồm các thuộc tính:

```

1  var jsonDoc = {
2      "temperature": 25,
3      "humidity": 60,
4      "start_time": "2023-08-1",
5      "completion_time": "2023-08-18",
6      "processing": 55,
7      "message": "Processing is underway.",
8      "emergency": "Nothing.",
9      "battery": 80
10 } ;

```

- Các thông tin trên sẽ được lấy ra và hiển thị trên website thông qua các node ở tab Home, giúp người sử dụng có thể theo dõi từ xa các thông số (được thể hiện trực quan như hình bên trên) của quá trình ủ tại nhà.

8.3.2 Điều khiển thiết bị:

- Website cung cấp cho người dùng các nút Start/ Pause và chọn chế độ khói lượng rác bỏ vào.
- Cụ thể hơn, khi người dùng muốn kích hoạt thùng rác từ xa thì chỉ cần chọn chế độ khói lượng rác (ít, vừa, đầy) và ấn Start.
- Ngược lại, người dùng có thể ấn Pause để tạm dừng quá trình ủ rác (trong trường hợp có sự cố).
- Khi người dùng ấn nút Start, website sẽ gửi về cho mạch điện một trong ba số: 0, 1, 2 đại diện cho việc khởi động với các khói lượng ít, vừa, đầy tương ứng.
- Khi người dùng ấn Pause, website sẽ gửi về mạch điện số 4, báo hiệu tín hiệu dừng.

8.3.3 Đọc dữ liệu từ Cloud ThingSpeak:

- Mạch điện sẽ gửi thông tin về thời gian, các sự số khẩn cấp của quá trình ủ lên Cloud. Thông qua đó, website sẽ lấy các dữ liệu đó và hiển thị cho người dùng về thông tin các quy trình ủ đã hoàn thành trước đây.

- Khi các dòng thông tin quá nhiều, người dùng có thể ấn nút Xóa lịch sử để xóa bỏ các dòng hiển thị. Các dòng thông tin này được hiển thị bằng node Table, từ thư viện Table-UI.
- Node Timestamp sẽ lặp và gọi lại dòng dữ liệu mới nhất sau vài ngày hoặc một tuần để cập nhật dữ liệu mới nhất từ Cloud.
- Cách lấy dữ liệu từ Cloud như sau: node sẽ gửi http request lên Cloud sau đó nhận được một data gồm các thuộc tính created_at, start_time, completion_time, emergency. Các thuộc tính sẽ được đảm bảo được đóng gói thành JSON. Tiếp theo, hàm Parse Json all fields sẽ thực hiện push JSON mới vừa nhận vào list các JSON đã được lưu trong Flow trước đó (nếu có) để tạo thành một mảng các thông tin. List dữ liệu này sẽ được hiển thị trên tab History.
- Để thực hiện việc xóa danh sách hiển thị lịch sử, ta chỉ việc gán lại mảng logs trong Flow là [] (mảng rỗng).

9 HƯỚNG PHÁT TRIỂN THÊM

Đây là những hướng đi có thể áp dụng được trong phiên bản cải tiến thêm của máy mà nhóm đã đúc kết được:

- Cần phát triển máy hoạt động sao cho công suất lớn hơn đáp ứng nhu cầu phong phú của nhiều người.
- Phát triển nhiều hơn các chức năng, chế độ trên cả website lẫn thiết bị.
- Nghiên cứu chu kỳ hoạt động của từng loại rác khác nhau để tìm điều kiện tối ưu.
- Cần phân tích thành phần hóa học sản phẩm hoặc kiểm soát đầu vào để sản phẩm được sử dụng hiệu quả.

10 CÁC TÀI LIỆU THAM THẢO

- Uploading sensor data to ThingSpeak with Node-RED
- Working with JSON Data And JavaScript Objects in Node-Red
- Hướng dẫn cài đặt Node-Red & Làm quen với Node-red
- Hướng dẫn lập trình Arduino kết nối wifi

- Giới thiệu chế phẩm vi sinh Emuniv

Hết!